A Vacuum Tracking Method for Oscillating Stars

Tsakiris Nikos

Master Thesis
for the
Postgraduate Diploma Specialization
in Computational Physics

Aristotle University of Thessaloniki
Physics Department

March 2007

2

A Vacuum Tracking Method for Oscillating Stars

Tsakiris Nikos

Master Thesis
for the
Postgraduate Diploma Specialization
in Computational Physics

Supervisor

Stergioulas Nikolaos
Assistant Professor
AUTH

Aristotle University of Thessaloniki
Physics Department
Section of Astronomy, Astrophysics and Mechanics

March 2007

Acknowledgements

This project is my master thesis. I feel grateful that Professor Nikolaos Stergioulas was my supervisor, since without his help it would have been impossible for me to overcome many difficulties. I would like to thank Professor Stergioulas for the sincere interest he demonstrated throughout my thesis and for his help. I would also like to thank Professor Jose Antonio Font for his advice. Finally I thank my two good friends, Vasili Tziampazli and Pascali Korosoglou, for their companionship and their will to help me.

Preface

The main purpose of this essay is to present results concerning a new method for tracking the vacuum-boundary and which can be extremely helpful, when static and rotating stars are studied. The vacuum-boundary tracking method is combined with a high-order reconstruction algorithm, such as PPM. It can be easily introduced in a numerical code and therefore in the present essay it was used to simulate static and rotating stars in Newtonian gravity, which were surrounded by a vacuum region. Up to now, most of the simulations assumed the existence of an artificial atmosphere around the stars which were dynamically evolved. The results of the simulations, using this vacuum-boundary tracking algorithm are presented in the next chapters and they appear promising.

Contents

Chapter 1

Fluid dynamics

The study of fluid dynamics is a scientific branch which has developed a great deal the past few years, constantly solving already posed problems, but also giving rise to new ones. The outcome of this research is obviously extremely important to researchers that have different scientific interests. In fact, the behavior of fluids plays a crucial role in a vast area of physical phenomena, from engineering to astrophysics. However, it must be noted that the theory of fluid dynamics is mainly based on partial differential equations, which in many cases can not be solved in an analytic manner. Therefore, most of the results concerning the presence of fluids are obtained using computational methods.
The domain of our interest is astrophysics and specifically the study of static, uniformly rotating and differentially rotating stars. These stars are highly idealized as infinite cylinders, consisting of a conducting gas in hydrostatic equilibrium. Moreover, it has to be noted that these stars are studied in the frame of Newtonian Gravity.
The main purpose of this chapter is to assemble the equations that govern the behavior of these idealized stars. This can be done using the Euler equations for ideal gases, taking of course into account the effects of gravity. The Euler equations must be deduced in a cylindrical coordinate frame. The next step will be to assemble the equations of Newtonian magneto-hydrodynamics in cylindrical coordinates, since the presence of magnetic fields can play a crucial role in the evolution of the model star.
In the first part of this chapter polytropic models are presented, which will be used as the initial data for the simulations. These models are cylindrical and take into account the effects of rotation. The next part consists of the extraction of the system of equations that govern the behavior of a cylindrical Newtonian rotating ideal gas. Finally, the equations of a cylindrical Newtonian ideal MHD gas are presented. Both in the simple and the MHD case the equations are in conservative form. All these equations will serve as the theoretical background for the numerical simulations.

1.1 Polytropic models

During most of its existence a star is in a state that evolves relatively slowly and so it is logical to assume that it is close to hydrostatic and thermodynamic equilibrium. This assumption is the basis for constructing equilibrium polytropes. Two different cases are examined, namely static polytropic models and rotating ones. Cylindrical coordinates are adopted.

1.1.1 Static polytropes

The equation of hydrostatic equilibrium balances the pressure gradient and the gravitational force, so for non-rotating stars it is

$$\frac{1}{\rho}\nabla p = -\nabla\Phi \tag{1.1}$$

where p is the pressure, given by the polytropic equation of state

$$p = K\rho^{\Gamma} \tag{1.2}$$

where K is a constant and $\Gamma = 1 + \frac{1}{n}$, where n is the polytropic index. Equation (1.2) defines the relation between the pressure and the mass density ρ. Moreover, Φ in equation (1.1) is the Newtonian gravitational potential, given by Poisson's equation

$$\nabla^2\Phi = 4\pi G\rho \tag{1.3}$$

where G is the gravitational constant. In order to create a static polytrope, equations (1.1), (1.2) and (1.3), must be combined in a proper manner. The first step is to calculate the gradient of the Newtonian gravitational potential, namely $\nabla\Phi = \frac{d\Phi}{dr}$ (in cylindrical coordinates r, ϕ, z), according to the following procedure

$$\nabla^2\Phi = 4\pi G\rho \quad \Rightarrow \quad \frac{1}{r}\frac{d}{dr}\left(r\frac{d\Phi}{dr}\right) = 4\pi G\rho \quad \Rightarrow$$

$$\frac{d\Phi}{dr} = \frac{4\pi G}{r}\int_0^r \rho r dr \tag{1.4}$$

Substituting equation (1.4) to (1.1), the following equation is obtained

$$\frac{d}{dr}\left(\frac{r}{\rho}\frac{dp}{dr}\right) = -4\pi G\rho r \tag{1.5}$$

Substituting the equation of state (1.2) in (1.5) yields

$$\frac{d}{dr}\left(\frac{r}{\rho}\frac{d}{dr}(K\rho^{\Gamma})\right) = -4\pi G\rho r \tag{1.6}$$

If the following nondimensional quantities

$$\rho = \rho_c \theta^n \tag{1.7}$$

$$r = a\xi \tag{1.8}$$

$$a = \left[\frac{(n+1)K\rho_c^{\frac{1}{n}-1}}{4\pi G} \right]^{\frac{1}{2}} \tag{1.9}$$

are introduced and substituted in equation (1.6) then one obtains

$$\frac{1}{\xi}\frac{d}{d\xi}\left(\xi\frac{d\theta}{d\xi}\right) = -\theta^n \tag{1.10}$$

This is the Lane-Emden equation in cylindrical coordinates. It must be noted that in (1.7) ρ_c stands for central density. A more general form of this equation is

$$\frac{1}{\xi^{a-1}}\frac{d}{d\xi}\left(\xi^{a-1}\frac{d\theta}{d\xi}\right) = -\theta^n \tag{1.11}$$

where a is a parameter and according to its value the coordinate system is defined. If $a = 1$ then planar geometry is chosen, if $a = 2$ then it is cylindrical geometry and if $a = 3$ it is spherical. In general, equation (1.10) has to be solved numerically. Nevertheless in some cases analytic solutions do exist for specific values of the polytropic index n. The radius ξ_1 of the star is the point at which $\theta = 0$.
The boundary conditions for the Lane-Emden equation can be derived through physical assumptions. The density at the center of the star is known to be equal to ρ_c, so $\theta = 1$. Moreover, at the center, $\frac{d\rho}{dr} = 0$, which yields $\frac{d\theta}{d\xi} = 0$.

1.1.2 Rotating polytropes

In the case of rotating polytropes several modifications have to take place, especially in the hydrostatic equilibrium equation. The effects of centrifugal acceleration must be taken into account, so it is

$$\frac{1}{\rho}\frac{dp}{dr} = -\frac{d\Phi}{dr} + \Omega^2(r)r \tag{1.12}$$

where Ω(r) is the angular velocity. In order to derive the Lane-Emden equation for rotating cylindrical polytropes, the same procedure as previously, for static polytropes, must be followed. Substituting (1.4) in (1.12) yields

$$\frac{r}{\rho}\frac{dp}{dr} = -4\pi G\int_0^r \rho r dr + \Omega^2(r)r^2$$

Differentiating with respect to r yields

$$\frac{d}{dr}\left(\frac{r}{\rho}\frac{dp}{dr}\right) = -4\pi G\rho r + 2\Omega^2 r + r^2\frac{d\Omega^2}{dr} \tag{1.13}$$

By replacing p with the polytropic equation of state and using the nondimensional variables along with

$$\zeta = \frac{\Omega^2}{\pi G\rho_c} \tag{1.14}$$

the result is the Lane-Emden equation for cylindrical rotating polytropes

$$\frac{d^2\theta}{d\xi^2} + \frac{1}{\xi}\frac{d\theta}{d\xi} + \theta^n - \frac{\zeta}{2} - \frac{\xi}{4}\frac{d\zeta}{d\xi} = 0 \tag{1.15}$$

Equation (1.15) describes the case of a differentially rotating polytrope, since ζ can depend on ξ. In the case of a uniformly rotating polytrope equation (1.15) reduces simply to

$$\frac{d^2\theta}{d\xi^2} + \frac{1}{\xi}\frac{d\theta}{d\xi} + \theta^n - \frac{\zeta}{2} = 0 \tag{1.16}$$

The boundary conditions at the center are as for the non-rotating case. A particular choice for the angular velocity profile Ω, made by Cook, Shapiro, Stephens (2003), is

$$\Omega(r) = \frac{\Omega_0}{2}\left[1 + \cos\left(\frac{\pi r^2}{R^2}\right)\right] \tag{1.17}$$

and its nondimensional form is

$$\zeta(\xi) = \frac{\zeta_0}{4}\left[1 + \cos\left(\frac{\pi\xi^2}{\xi_1^2}\right)\right] \tag{1.18}$$

where $\zeta_0 = \frac{\Omega_0^2}{\pi G\rho_c}$. Equilibrium solutions exist, as long as rotation does not cause mass-shedding. Another requirement is that the density decreases monotonically inside the polytrope. Due to the different symmetry assumptions, models constructed with the same EOS do not have the same compressibility when spherical or cylindrical polytropes are used. Comparing compressibilities, the range of $0.5 < n < 1.0$ for spherical nonrotating models corresponds to the range of $0.70 < n < 1.49$ for cylindrical models. Also, while spherical nonrotating models are radially unstable for $n > 3$ and have an infinite radius for $n > 5$, cylindrical nonrotating polytropes are stable and are not known to have an infinite extend for any range of polytropic indices n.

1.2 Hydrodynamic equations

The hydrodynamic equations describe the conservation of quantities such as the density, the momentum and the energy. In fact they are called conservation laws and their covariant form is [1]

1. conservation of mass (continuity equation)

$$\rho_{,t} + \big(\rho u^{\alpha}\big)_{;\alpha} = 0 \Leftrightarrow$$

$$\rho_{,t} + \frac{1}{\sqrt{g}}\big(\sqrt{g}\rho u^{\alpha}\big)_{,\alpha} = 0 \tag{1.19}$$

2. conservation of momentum (equation of motion)

$$\big(\rho u^{\alpha}\big)_{,t} + \big(\rho u^{\alpha} u^{\beta}\big)_{;\beta} - \tau^{\alpha\beta}{}_{;\beta} = \rho f^{\alpha} \tag{1.20}$$

   with

$$\tau_{\alpha\beta} = -p\vec{I} + \vec{\Pi}$$

   where $\overleftrightarrow{I}$ is the unit tensor, $\overleftrightarrow{\Pi}$ is the viscous stress tensor and u^{α}, f^{α} are the velocity and acceleration coordinate vectors.

3. conservation of energy

$$E_{,t} + \big(Eu^{\alpha} - u_{\beta}\tau^{\alpha\beta} + Q^{\alpha}\big)_{;\alpha} = \rho\big(u_{\alpha} f^{\alpha}\big) + \rho q \tag{1.21}$$

   where $\vec{Q}$ is the energy flow vector defined as the energy that flows into or out of the material per unit area per unit time and q is the rate per unit mass and per unit time that describes the heat that is added to each material particle.

The term f^{a} that appears in the conservation laws describes acceleration due to body forces that are applied to the fluid. Moreover, g is determinant of the metric that is used. In cylindrical coordinates, $x^{a} = \big(x^{1}, x^{2}, x^{3}\big) = \big(r, \phi, z\big)$, the metric is

$$g_{\alpha\beta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.22}$$

The conservative formulation is preferable because it offers significant advantages when compared to the primitive one. In fact, formulations based on variables other than the conservative ones are inaccurate at shock waves, giving wrong shock positions. In order to achieve the best possible results it is important to work with the conservative variables, namely the density, the momentum and the total energy.

[1] Greek letters α, β, ... are used for denoting abstract spatial coordinate indices of vectors.

For an ideal fluid (Q, q, $\tau^{\alpha\beta}$ vanish) the components of the system of equations in cylindrical coordinates (a non-orthonormal basis), are

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial r}\big(\rho u^r\big) = -\frac{1}{r}\rho u^r \tag{1.23}$$

$$\frac{\partial}{\partial t}\big(\rho u^r\big) + \frac{\partial}{\partial r}\Big[\rho\big(u^r\big)^2 + p\Big] = -\frac{1}{r}\rho\big(u^r\big)^2 + r\rho\big(u^\phi\big)^2 + \rho f^r \tag{1.24}$$

$$\frac{\partial}{\partial t}\big(\rho u^\phi\big) + \frac{\partial}{\partial r}\big(\rho u^r u^\phi\big) = -\frac{3}{r}\rho u^r u^\phi \tag{1.25}$$

$$\frac{\partial E}{\partial t} + \frac{\partial}{\partial r}\big[u^r\big(E+p\big)\big] = -\frac{1}{r}u^r\big(E+p\big) + \rho u^r f^r \tag{1.26}$$

where

$$E = \frac{1}{2}\rho\left((u^r)^2 + (ru^\phi)^2\right) + \rho\epsilon \tag{1.27}$$

The system of equations (1.23)-(1.26) is not yet complete, since two terms that appear in them, namely the pressure p and the body force f^r, have not yet been sufficiently specified. Two more equations need to be added to the ones already presented. The first one is the equation of state. During evolution a Γ-law EOS is assumed

$$p = \big(\Gamma - 1\big)\rho\epsilon \tag{1.28}$$

where ϵ is the specific internal energy. Equation (1.28) takes into account the effects of heating due to shocks.
The acceleration due to body forces, f^r, is in fact the Newtonian gravitational force, which is equal to

$$f^r = -\frac{d\Phi}{dr}$$

By solving Poisson's equation, the value of the gravitational potential Φ can be determined at every point which in turn yields the acceleration of gravity.
If in the term $\rho\epsilon$ is substituted to (1.27), according to the equation of state (1.28), then

$$E = \frac{1}{2}\rho\left((u^r)^2 + (ru^\phi)^2\right) + \frac{p}{\Gamma - 1} \tag{1.29}$$

Equations (1.23)-(1.26) can be written in a more compact form as

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_r = \mathbf{S}(\mathbf{U}) \tag{1.30}$$

where a subscript denotes partial differentiation and where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u^r \\ \rho u^\phi \\ E \end{bmatrix} \quad , \quad \mathbf{F} = \begin{bmatrix} \rho u^r \\ \rho\big(u^r\big)^2 + p \\ \rho u^r u^\phi \\ u^r(E+p) \end{bmatrix} \quad \text{and} \quad \mathbf{S} = \begin{bmatrix} -\frac{1}{r}\rho u^r \\ -\frac{1}{r}\rho\big(u^r\big)^2 + r\rho\big(u^\phi\big)^2 + \rho f^r \\ -\frac{3}{r}\rho u^r u^\phi \\ -\frac{1}{r}u^r(E+p) + \rho u^r f^r \end{bmatrix}$$

1.3 MHD equations

Since conductivity is very high in the high-density star matter, the ideal MHD approximation is adopted. In this case, the displacement current and the effects of electrical resistivity, viscosity and thermal conduction are neglected. As a result, in conservation-law form the ideal MHD equations are written as

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \mathbf{u}) = 0, \tag{1.31}$$

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) + \nabla[\rho \mathbf{u}\mathbf{u}^T + P\mathbf{I} - \mathbf{B}\mathbf{B}^T] = -\mathbf{f}, \tag{1.32}$$

$$\frac{\partial E}{\partial t} + \nabla[(E+P)\mathbf{u} - \mathbf{B}(\mathbf{u}\cdot\mathbf{B})] = -\rho \mathbf{u}\mathbf{f}, \tag{1.33}$$

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla(\mathbf{u}\mathbf{B}^T - \mathbf{B}\mathbf{u}^T) = 0, \tag{1.34}$$

where ρ is the mass density, $\mathbf{u}$ is the velocity, p is pressure, E is the total energy density, $\mathbf{f}$ is the vector of the body force. The total pressure P is defined as

$$P = p + \frac{1}{2}\mathbf{B}\mathbf{B} \tag{1.35}$$

where p is the fluid pressure. The total energy in this case, is given by

$$E = \frac{1}{2}\rho \mathbf{u}\mathbf{u} + \frac{p}{\Gamma - 1} + \frac{1}{2}\mathbf{B}\mathbf{B} \tag{1.36}$$

As for the magnetic field, there holds the divergence-free condition $\nabla \mathbf{B} = 0$. It is important to note that in the case of one-dimensional problems this constraint reduces to B^r=constant.
Equation (1.34) is the flux-freezing one. It is obvious that the MHD equations are in a fact a coupling between the hydrodynamic and Maxwell ones.
The next step is to try to enforce cylindrical symmetry to the system of the ideal MHD equations. Taking into account the effects of gravity one can find that

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial r}\left(\rho u^r\right) = -\frac{1}{r}\rho u^r \tag{1.37}$$

$$\frac{\partial}{\partial t}\left(\rho u^r\right) + \frac{\partial}{\partial r}\left[\rho\left(u^r\right)^2 + p - \left(B^r\right)^2\right] = -\frac{1}{r}\rho\left(u^r\right)^2 + r\rho\left(u^\phi\right)^2 + \rho f^r + \frac{\left(B^r\right)^2 - r^2\left(B^\phi\right)}{r} \tag{1.38}$$

$$\frac{\partial}{\partial t}\left(\rho u^\phi\right) + \frac{\partial}{\partial r}\left(\rho u^r u^\phi - B^r B^\phi\right) = -\frac{3}{r}\left(\rho u^r u^\phi - B^r B^\phi\right) \tag{1.39}$$

$$\frac{\partial B^\phi}{\partial t} + \frac{\partial}{\partial r}\left(B^\phi u^r - B^r u^\phi\right) = -\frac{1}{r}\left(B^\phi u^r - B^r u^\phi\right) \tag{1.40}$$

$$\frac{\partial E}{\partial t}+\frac{\partial}{\partial r}\left[u^r\big(E+p\big) - B^r\big(B^r u^r + r^2 B^\phi u^\phi\big)\right] = -\frac{1}{r}u^r\big(E+p+\frac{1}{2}\big((B^r)^2+(rB^\phi)^2\big)\big)+\rho u^r f^r - B^r\big(B^r u^r + r^2 B^\phi u^\phi \tag{1.41}$$

The total energy is

$$E = \frac{1}{2}\rho\left(\big(u^r\big)^2 + \big(ru^\phi\big)^2\right) + \frac{p}{\Gamma-1} + \frac{1}{2}\left(\big(B^r\big)^2 + \big(rB^\phi\big)^2\right) \tag{1.42}$$

The evolution of the star, idealized as an infinite rotating cylinder and taking into account the effects of the gravitational and magnetic fields, will be described by the solution of the system of these equations. Equations (1.37)-(1.41) can be written in a more compact form as

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_r = \mathbf{S}(\mathbf{U}) \tag{1.43}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u^r \\ \rho u^\phi \\ B^\phi \\ E \end{bmatrix} \quad , \quad \mathbf{F} = \begin{bmatrix} \rho u^r \\ \rho\big(u^r\big)^2 + p - \big(B^r\big)^2 \\ \rho u^r u^\phi - B^r B^\phi \\ B^\phi u^r - B^r u^\phi \\ u^r(E+p) - B^r\big(B^r u^r + r^2 B^\phi u^\phi\big) \end{bmatrix}$$

and

$$\mathbf{S} = \begin{bmatrix} -\frac{1}{r}\rho u^r \\ -\frac{1}{r}\rho\big(u^r\big)^2 + r\rho\big(u^\phi\big)^2 + \rho f^r + \frac{\big(B^r\big)^2 - r^2\big(B^\phi\big)}{r} \\ -\frac{3}{r}\left(\rho u^r u^\phi - B^r B^\phi\right) \\ -\frac{1}{r}\left(B^\phi u^r - B^r u^\phi\right) \\ -\frac{1}{r}u^r\big(E+p+\frac{1}{2}\big((B^r)^2+(rB^\phi)^2\big)\big) + \rho u^r f^r - B^r\big(B^r u^r + r^2 B^\phi u^\phi\big) \end{bmatrix}$$

Chapter 2

Numerical methods

In this chapter the numerical methods are presented. The description of the methods mainly follows Toro (1999) and papers already published and which are noted in each case.

2.1 Conservation laws

An important consideration for numerical solutions to compressible fluid flow is how the numerical method will respond to the presence or formation of shocks, i.e. discontinuities in the fluid variables. A shock is not a solution to the differential fluid equations, but rather a member of a larger class called *weak solutions*. As finite difference approximations are derived from the differential form of the equations, it is not surprising that they typically fail when shocks appear. As shocks form generically from smooth initial data, many special techniques have been developed for the numerical solution of fluid equations. One approach is to introduce an *artificial viscosity* that adds extra dissipation in the vicinity of a shock, spreading the would-be discontinuity over a few points. Artificial viscosity is implemented by adding a viscous term to the pressure

$$p \rightarrow p + Q$$

in the fluid equations, where the viscosity is activated where the flow is compressive. Moreover, the artificial viscosity should only be applied to shocks and not to rarefraction waves. This technique has been widely used and has the advantage of simplicity of implementation and computational efficiency.
A second approach to solving the fluid equations comes from methods developed specifically for conservation laws. These methods, usually variations of Godunov's original idea to use piece-wise solution of the Riemann problem, have proven to be very reliable and robust.
Conservation laws greatly simplify the mathematical description of physical systems by focusing on quantities $\mathbf{Q}_i$, where $\mathbf{Q}_i$ may be a state vector with multiple components, whose

volume-integral does not change in time

$$\partial_t \int_V d\mathbf{Q} = 0$$

While conservation laws are often written in *differential* form it is useful to first consider an *integral* formulation, which is often the more fundamental expression. The average volume of Q within the cell $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ is denoted Q_i and it is

$$Q_i = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}(x,t)dx$$

The change of Q_i with time can be calculated from the flux, $\mathbf{F}(\mathbf{U}(x,t)$ of $\mathbf{U}$ through the cell boundaries. This consideration thus expresses the conservation law

$$\frac{d}{dt}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}(x,t)dx = \mathbf{F}\big(x_{i-\frac{1}{2}},t\big) - \mathbf{F}\big(x_{i+\frac{1}{2}},t\big) \tag{2.1}$$

The conservation law can be written in integral form by integrating (2.1) from an initial time 0 to a final time Δt,

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}(x,\Delta t)dx - \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}(x,0)dx = \int_0^{\Delta t} \mathbf{F}\big(x_{i-\frac{1}{2}},t\big) - \int_0^{\Delta t} \mathbf{F}\big(x_{i+\frac{1}{2}},t\big) \tag{2.2}$$

and the differential form follows from further manipulations if it is assumed that $\mathbf{U}$ is differentiable

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0 \tag{2.3}$$

It should be emphasized that the integral formulation should be viewed as the primary mathematical form for a conservation principle, because it is independent on an assumption of differentiability. For example, at a shock front in a fluid system, $\mathbf{U}$ is not differentiable and the differential form of the conservation law fails, while the integral formulation is still satisfied. Discretizations of conservation equations via finite differences rely on the differential form and artificial viscosity must be added near shock fronts, forcing $\mathbf{U}$ to be differentiable. An alternative strategy is to develop algorithms based directly on the integral formulation of the conservation laws. The Godunov method and its extensions are examples of this latter approach.

2.2 Riemann problem

The Riemann problem for a general $m \times m$ non-linear hyperbolic system with data $\mathbf{U}_l$, $\mathbf{U}_r$ is the initial value problem

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0$$

$$\mathbf{U}(x,0) = \begin{cases} \mathbf{U}_l & \text{if} \quad \text{x} < 0 \\ \mathbf{U}_r & \text{if} \quad \text{x} > 0 \end{cases}$$

The solution of the Riemann problem consists of $m+1$ constant states, separated by m waves, which can be either shock waves, rarefraction waves or contact discontinuities, see Fig. 2.1.

2.3 Godunov's method

Numerical algorithms for conservation laws are typically developed by discretizing the equations in their fundamental integral form. These methods are often derived using a control volume discretization, whereby the domain is divided into computational cells I_i, defined to span the interval $\left[x - \frac{\Delta x}{2}, x + \frac{\Delta x}{2}\right] = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$, where Δx is the spatial discretization scale. Following the derivation of the integral conservation law (2.2) for the computational cell I_i the averaged quantities $\bar{\mathbf{U}}_i^n$ are introduced

$$\bar{\mathbf{U}}_i^n = \frac{1}{\Delta x} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{U}\big(x, t^n\big)$$

The discrete form of the conservation law (2.2) is then

$$\bar{\mathbf{U}}_i^{n+1} = \bar{\mathbf{U}}_i^n - \frac{\Delta t}{\Delta x}\big(\mathbf{F}_{i+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2}}\big) \tag{2.4}$$

where the numerical flux is defined by $\mathbf{F}_{i+\frac{1}{2}}$

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} \mathbf{F}\left(\mathbf{U}\big(x_{i+\frac{1}{2}}\big), t\right) dt \tag{2.5}$$

At first blush, a numerical method based on a discretization of the integral law does not appear promising. The flux integral (2.5) does not appear readily solvable and, generally it is not. However in his seminal work, Godunov devised a technique to approximately evaluate the flux integral by replacing the function $\mathbf{U}(x, t^n)$ with a piece-wise constant function, see Fig. 2.2. In this approach the individual cells (control volumes) are treated as a sequence of shock tubes and a separate Riemann initial value problem is solved at each cell interface. Provided that the waves from neighboring cells do not interact, a proviso that gives a Courant-type condition on the time step, each Riemann problem can be solved exactly to yield the local solution $\mathbf{U}(x,t)$ for each shock tube, see Fig. 2.3. Furthermore, since the solution of each of the local Riemann problems is self-similar, $\mathbf{U}\big(x_{i+\frac{1}{2}}, t\big)$ is a constant in time and the evaluation of the integral (2.5) becomes trivial. Then, explicit expressions for the cell averages at the advanced time $t = T^{n+1}$ can be found via (2.4).

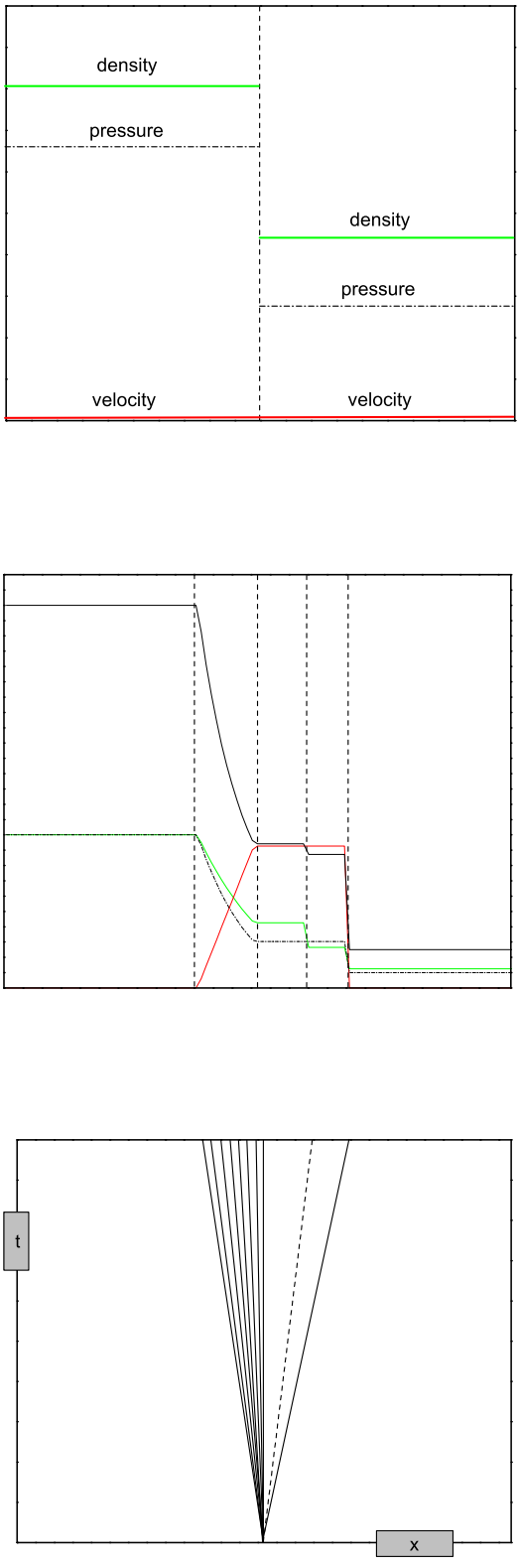


Figure 2.1: The Riemann problem for a shock tube. Initial data, solution and wave patterns.

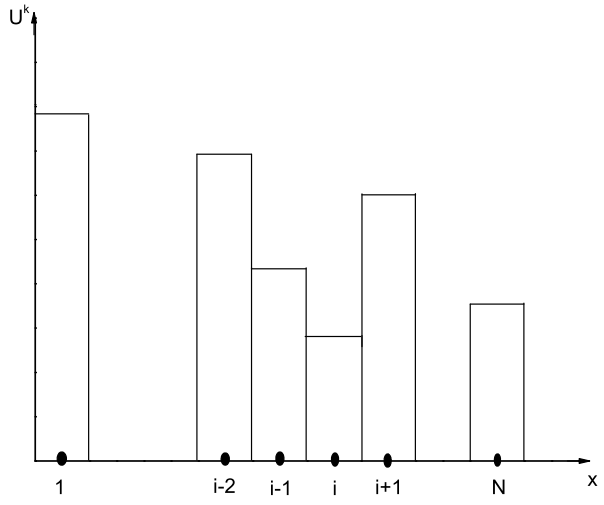


Figure 2.2: Piece-wise constant distribution of data at time level n, for a single component of the vector **U**.

2.4 Grid

Using Godunov type schemes cell averages must be defined over finite volumes. A domain $[0, L] \times [0, T]$ in the x-t plane is descretised as shown in figure 2.4. The spatial domain of length L is subdivided into N finite volumes, called computational cells, given as

$$x_{i-\frac{1}{2}} = (i-1)\Delta x \leq x \leq i\Delta x = x_{i+\frac{1}{2}}$$

The extreme values $x_{i-\frac{1}{2}}$ and $x_{i+\frac{1}{2}}$ define the position of the intercell boundaries at which the corresponding intercell numerical fluxes must be specified. The size of the cell is

$$\Delta x = x_{i-\frac{1}{2}} - x_{i+\frac{1}{2}}$$

2.5 Time step size

The spatial discretisation Δx is chosen on desired accuracy or available computational resources. What is left to be determined is the time step size Δt, according to the CFL condition

$$\Delta t = \frac{C_{cfl}\Delta x}{S^n_{max}}$$

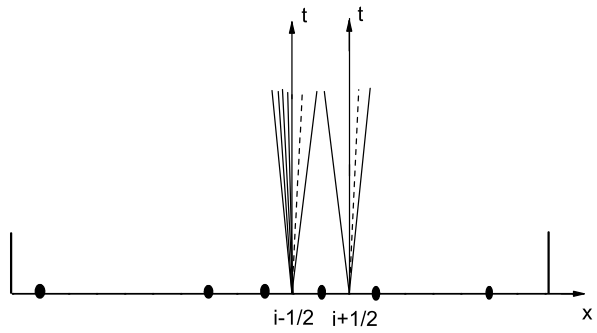


Figure 2.3: Typical wave patterns emerging from soluitons of local Riemann problems at intercell boundaries $i - \frac{1}{2}$ and $i + \frac{1}{2}$.

where C_{cfl} is a Courant coefficient satisfying

$$0 < C_{cfl} \leq 1$$

S^n_{max} is the largest wave speed present throughout the spatial domain at time level t^n and can be estimated using the following formula

$$S^n_{max} = max\left[\left|U^n_i\right| + cs^n_i\right]$$

where cs^n_i is the sound speed and its form depends on the choice of the equation of state.

2.6 High-resolution shock-capturing schemes

The application of high-resolution shock-capturing (HRSC) methods caused a revolution in computational fluid dynamics. These methods satisfy in a quite natural way the basic properties required for any acceptable numerical method

1. high order of accuracy
2. stable and sharp description of discontinuities
3. convergence to the physically correct solution.

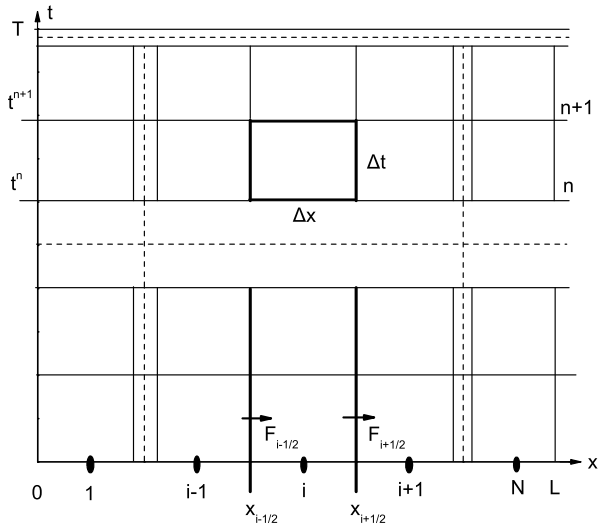


Figure 2.4: Discretization of domain $[0, L]$ into N finite volumes $I_i = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$.

Moreover, HRSC methods are conservative and because of their shock capturing property discontinuous solutions are treated both consistently and automatically whenever and wherever they appear in the flow. These methods manage to avoid spurious oscillations at shocks, which are known as Gibb's phenomena, while retaining a high order of accuracy over the majority of the domain.
As HRSC methods are written in conservation form, the time evolution of zone averaged state vectors is governed by some functions (the numerical fluxes) evaluated at zone interfaces. Numerical fluxes are mostly obtained by means of an exact or approximate Riemann solver. High resolution is usually achieved by using monotonic polynomials in order to interpolate the approximate solutions within numerical cells.
Solving Riemann problems exactly involves time-consuming computations which are particularly costly in the case of multidimensional problems. Therefore, as an alternative, the usage of approximate Riemann solvers has been proposed.

2.6.1 Reconstruction

The Godunov-type numerical methods are based on solutions of the Riemann initial value problem at the interfaces between cells. During an update step functions U_l and U_r are introduced, defined on the intervals $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ to approximate the solution in the control volumes. These functions are created from the cell averages U_i^n and hence are called reconstructions. In the following sections several reconstruction methods are presented.

Piece-wise constant reconstruction

The simplest reconstruction is to appoint to U_l and U_r directly the cell values U_i

$$U(x,0) = \begin{cases} U_l = U_i & if \quad x < 0 \\ U_r = U_{i+1} & if \quad x > 0 \end{cases}$$

where U(x,0) are the initial data for the local Riemann problem defined within the cell $[x_i, x_{i+1}]$. The cell interface $x_{i+\frac{1}{2}}$ is considered to be the point zero along the x-axis at the local Riemann problem. The piece-wise linear reconstruction is in the fact the original Godunov method and results in a numerical scheme in which the spatial derivatives and hence the overall scheme have first order accuracy.

Piece-wise linear reconstruction

The piece-wise linear reconstruction formally results in a scheme with second order convergence. A first choice for the reconstruction algorithm might be to compute the slope, the derivative of the dynamical variable, centered at the cell boundaries

$$s_{i+\frac{1}{2}} = \frac{U_{i+1} - U_i}{x_{i+1} - x_i} \tag{2.6}$$

creating the reconstructed variables

$$U(x,0) = \begin{cases} U_l = U_i + s_i\big(x_{i+\frac{1}{2}} - x_i\big) & if \quad x < 0 \\ U_r = U_{i+1} + s_{i+1}\big(x_{i+\frac{1}{2}} - x_{i+1}\big) & if \quad x > 0 \end{cases}$$

where $x\epsilon\big[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\big]$. In a uniform grid $x_{i+1} - x_i = \Delta x$. However this reconstruction produces spurious oscillations at shocks, making the scheme unstable. This problem can be avoided once a *slope limiter* is used, designed to damp any oscillations near discontinuities. Slope limiters use the *total variation diminishing property* (TVD). If the limited slope is devoted by σ_i then the reconstruction can be defined as

$$U(x,0) = \begin{cases} U_l = U_i + \sigma_i\big(x_{i+\frac{1}{2}} - x_i\big) & if \quad x < 0 \\ U_r = U_{i+1} + \sigma_{i+1}\big(x_{i+\frac{1}{2}} - x_{i+1}\big) & if \quad x > 0 \end{cases}$$

The slope limiter forces the reconstructed variables U_l, U_r to be monotonic near discontinuities and this reduces the local accuracy of the scheme to first order.
The most common slope limiter used in fluid dynamics is the minmod limiter

$$\sigma_i = minmod\big(s_{i-\frac{1}{2}}, s_{i+\frac{1}{2}}\big)$$

where $s_{i+\frac{1}{2}}$ is given by equation (2.6) and the minmod function gives the minimum modulus of two arguments

$$mimod(a,b) = \begin{cases} sgn(a)min(|a|,|b|) & if \quad sgn(a) = sgn(b) \\ 0 & otherwise \end{cases} \tag{2.7}$$

The sign function sgn(a) is

$$sgn(a) = \begin{cases} +1 & if \quad a > 0 \\ -1 & if \quad a > 0 \end{cases}$$

Another common limiter is the monotonized central-difference limiter (MC). The limited slope is

$$\sigma_i = minmod\big(s_i, 2s_{i-\frac{1}{2}}, 2s_{i+\frac{1}{2}}\big)$$

where $s_{i-\frac{1}{2}}$ and $s_{i+\frac{1}{2}}$ are calculated as shown in equation (2.6) and

$$s_i = \frac{U_{i+1} - U_{i-1}}{x_{i+1} - x_{i-1}}$$

The mimod function for three arguments is

$$mimod(a,b,c) = \begin{cases} sgn(a)min(|a|,|b|,|c|) & if \quad sgn(a) = sgn(b) \quad and \quad sgn(b) = sgn(c) \\ 0 & otherwise \end{cases}$$

The most interesting limiter and the one used throughout the computations is a combination of both the minmod and the MC limiters and consists of four different steps

1. Calculation of the jumps $\Delta u = u_{i+1} - u_{i-1}$ in the velocity.

2. Calculation of the variables d_{i-1}, d_i , d_{i+1}, according to

$$d_i = \begin{cases} 1 & if \quad \frac{p_{i+1} - p_{i-1}}{min(p_{i+1}, p_{i-1})} < \epsilon \\ 0 & otherwise \end{cases}$$

   where $\epsilon = 0.01$.

3. Calculation of $d_{min} = min(d_{i-1}, d_i, d_{i+1})$

4. The last step concerns the choice of the appropriate slope limiter. If the conditions $\Delta u > 0$ and $d_{min} = 0$ are met then the minmod limiter will be used, otherwise the MC limiter.

The piece-wise linear reconstruction retains second order accuracy in the smooth parts of the flow, but is only first order accurate in the region of discontinuities due to the application of the limiter. The lack of accuracy is the price to pay in order to damp the oscillations.

MUSCL-Hanckock method

The monotonic upstream centered scheme for conservation laws (MUSCL), as presented in Toro (1999), achieves a second order extension of the Godunov first order upwind scheme. The first step in the MUSCL aproach is to evaluate the *boundary extrapolated values* $U_{l,i}$ and $U_{r,i}$, which are in fact similar to the piece-wise linearly reconstructed values of the previous section. In each computational cell $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ the data cell average values U_i are replaced by piece-wise linear functions

$$\mathbf{U}_{l,i} = \mathbf{U}_i + \boldsymbol{\sigma}_i \frac{\Delta x}{2} \quad and \quad \mathbf{U}_{r,i} = \mathbf{U}_{i+1} - \boldsymbol{\sigma}_{i+1} \frac{\Delta x}{2} \tag{2.8}$$

where σ_i are the limited slopes. The main point in the MUSCL scheme is to evolve the boundary extrapolated values by a time $\frac{\Delta t}{2}$. This procedure takes place in each cell I_i according to

$$\bar{\mathbf{U}}_{l,i} = \mathbf{U}_{l,i} + \frac{1}{2}\frac{\Delta t}{\Delta x}\big(\mathbf{F}(\mathbf{U}_{l,i}) - \mathbf{F}(\mathbf{U}_{r,i})\big)$$

and

$$\bar{\mathbf{U}}_{r,i} = \mathbf{U}_{r,i} + \frac{1}{2}\frac{\Delta t}{\Delta x}\big(\mathbf{F}(\mathbf{U}_{l,i}) - \mathbf{F}(\mathbf{U}_{r,i})\big)$$

The values $\bar{\mathbf{U}}_{l,i}$ and $\bar{\mathbf{U}}_{r,i}$ will be the initial data for the Riemann problem in each cell I_i

$$\mathbf{U}(x,0) = \begin{cases} \mathbf{U}_l = \bar{\mathbf{U}}_{r,i} & if \quad x < 0 \\ \mathbf{U}_r = \bar{\mathbf{U}}_{l,i+1} & if \quad x > 0 \end{cases}$$

The point x=0 in the local coordinate system is equivalent to $x_{i+\frac{1}{2}}$ in the global coordinate system.

PPM

The piece-wise parabolic method was introduced by Collela and Woodward (1984), but the following presentation of the method is based on Martí and Müller (1996). It is a rather complex method and is what differentiates it from the simple piece-wise constant method and the MUSCL method. The first step is to perform a high-order interpolation to obtain preliminary values for the primitive variables $\mathbf{W}$. The second step is about steepening discontinuities and is only applied to contact discontinuities, in order to keep their profile sharp. The third step is to flatten the zone structure near shocks. Another step takes place in order to preserve monotonicity. Finally the derived reconstructed are readjusted, so that they can be used as initial data for the local coordinate Riemann problems.
First interpolated values of the primitive variables are calculated at all zone interfaces using a quartic polynomial. The interpolated values are

$$\mathbf{W}_{i+\frac{1}{2}} = \frac{1}{2}(\mathbf{W}_i + \mathbf{W}_{i+1}) + \frac{1}{8}(\delta_m \mathbf{W}_i + \delta_m \mathbf{W}_{i+1})$$

where

$$\delta_m \mathbf{W}_i = \begin{cases} sgn(\delta \mathbf{W}_i) min\big(|\delta \mathbf{W}_i|, 2|\mathbf{W}_{i+1} - \mathbf{W}_i|, 2|\mathbf{W}_i - \mathbf{W}_{i-1}|\big) & \text{if} \big(\mathbf{W}_{i+1} - \mathbf{W}_i\big)\big(\mathbf{W}_i - \mathbf{W}_{i-1}\big) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

and

$$\delta \mathbf{W}_i = \frac{1}{2}\big(\mathbf{W}_{i+1} - \mathbf{W}_{i-1}\big) \tag{2.10}$$

In smooth parts of the flow, away from extrema, the limiting values of $\mathbf{W}$ are given by

$$\mathbf{W}_{l,i+1} = \mathbf{W}_{r,i} = \mathbf{W}_{i+\frac{1}{2}}$$

Special care has to been taken in the regions where contact discontinuities are present. In the presence of a contact discontinuity only the density values change and hence $\rho_{l,i}$, $\rho_{r,i}$ have to be modified if the condition

$$\Gamma K_0 \frac{|\rho_{i+1} - \rho_{i-1}|}{\min(\rho_{i+1}, \rho_{i-1})} \geq \frac{|p_{i+1} - p_{i-1}|}{\min(p_{i+1}, p_{i-1})}$$

holds. K_0 is a constant. This condition ensures that the discontinuity is a constant one. This procedure is called steepening and takes place in order to ensure sharp profiles. The cell boundary reconstructions of the density are replaced according to

$$\rho_{l,i} = \rho_{l,i}\big(1 - \eta_i\big) + \rho^d_{l,i}\eta_i$$
$$\rho_{r,i} = \rho_{r,i}\big(1 - \eta_i\big) + \rho^d_{r,i}\eta_i$$

with

$$\rho^d_{l,i} = \rho_{i-1} + \frac{\delta_m \rho_{i-1}}{2} \quad , \quad \rho^d_{r,i} = \rho_{i+1} - \frac{\delta_m \rho_{i+1}}{2}$$

where δ_m is given by equation (2.9) and

$$\eta_i = max\big[0, min\big(\eta^{(1)}\big(\tilde{\eta}_i - \eta^{(2)}\big)\big)\big]$$

where $\eta^{(1)}$, $\eta^{(2)}$ are free parameters, while $\tilde{\eta}_i$ is defined as

$$\tilde{\eta}_i = \begin{cases} -\frac{\rho_{i+2} - \rho_{i-2} - 4\delta\rho_i}{12\delta\rho_i} & \text{if} \quad \begin{cases} \delta^2\rho_{i+1} \cdot \delta^2\rho_{i-1} > 0 \\ \big|\rho_{i+1} - \rho_{i-1}\big| - \epsilon^{(1)} min\big(\big|\rho_{i+1}\big|, \big|\rho_{i-1}\big|\big) > 0 \end{cases} \\ 0 & \text{otherwise} \end{cases}$$

where $\epsilon^{(1)}$ is another free parameter and $\delta\rho_i$ is given by (2.10) and

$$\delta^2\rho_i = \frac{\rho_{i+1} - 2\rho_i + \rho_{i-1}}{6\big(\Delta x\big)^2}$$

The next step concerns the flattening of the distribution near the regions where a shock is detected. The order of the method near these regions is reduced in order to avoid spurious postshock oscillations. The shock regions are detected if the conditions

$$\frac{|p_{i+1} - p_{i-1}|}{\min(p_{i+1}, p_{i-1})} > \epsilon^{(2)} \quad \text{and} \quad u_{i-1} > u_{i+1}$$

hold. In these regions the variables $\mathbf{W}_l$ and $\mathbf{W}_r$ are substituted using

$$\mathbf{W}_{l,i}^{\text{flat}} = \mathbf{W}_i f_i + \mathbf{W}_{l,i}(1 - f_i)$$
$$\mathbf{W}_{r,i}^{\text{flat}} = \mathbf{W}_i f_i + \mathbf{W}_{r,i}(1 - f_i)$$

The weight function f_i is given by the maximum of $\tilde{f}_i$ and $\tilde{f}_{i+s_i}$, where

$$\tilde{f}_i = min\left(1, a_i max\left(0, \left(\frac{p_{i+1} - p_{i-1}}{p_{i+2} - p_{i-2}} - \omega^{(1)}\right)\omega^{(2)}\right)\right)$$

The index s_i of $\tilde{f}_{i+s_i}$ is equal to +1 or -1 depending on whether the difference $p_{i+1} - p_{i-1}$ is positive or negative and a_i is given by

$$a_i = \begin{cases} 1 & \text{if} \quad \frac{|p_{i+1}-p_{i-1}|}{\min(p_{i+1},p_{i-1})} > \epsilon^{(2)} \quad \text{and} \quad u_{i-1} > u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

The next step that takes place in the PPM algorithm ensures the monotonicity preservation. In the smooth regions of the flow it is $\mathbf{W}_{l,i+1} = \mathbf{W}_{r,i} = \mathbf{W}_{i+\frac{1}{2}}$, but near discontinuities these values have to be modified in such a manner that the interpolation parabola in each cell zone i is a monotone function that takes values between $\mathbf{W}_{l,i}$ and $\mathbf{W}_{r,i}$. The following modifications are necessary

$$\mathbf{W}_{l,i} = \mathbf{W}_i \quad \text{and} \quad \mathbf{W}_{r,i} = \mathbf{W}_i \quad \text{if} \quad \big(\mathbf{W}r, i - \mathbf{W}_i\big)\big(\mathbf{W}_i - \mathbf{W}_{l,i}\big) \leq 0$$

$$\mathbf{W}_{l,i} = 3\mathbf{W}_i - 2\mathbf{W}_{r,i} \quad \text{if} \quad \big(\mathbf{W}_{r,i} - \mathbf{W}_{l,i}\big)\left(\mathbf{W}_i - \frac{1}{2}\big(\mathbf{W}_{l,i} + \mathbf{W}_{r,i}\big)\right) > \frac{\big(\mathbf{W}_{r,i} - \mathbf{W}_{l,i}\big)^2}{6}$$

$$\mathbf{W}_{r,i} = 3\mathbf{W}_i - 2\mathbf{W}_{l,i} \quad \text{if} \quad \big(\mathbf{W}_{r,i} - \mathbf{W}_{l,i}\big)\left(\mathbf{W}_i - \frac{1}{2}\big(\mathbf{W}_{l,i} + \mathbf{W}_{r,i}\big)\right) < -\frac{\big(\mathbf{W}_{r,i} - \mathbf{W}_{l,i}\big)^2}{6}$$

The final step is to use the reconstructed values $\mathbf{W}_l$ and $\mathbf{W}_r$ in order to construct the effective left and right states of the Riemann problem. In order to accomplish that the knowledge of the eigenvalues $\lambda^\sharp$ is necessary, where $\sharp = -1, 0, 1$. The superscript -1 refers to the eigenvalue $\lambda^{-1} = u - cs$, 0 refers to $\lambda^0 = u$ and +1 to $\lambda^{+1} = u + cs$. Initially the coefficients $\Delta\mathbf{W}$ and $\mathbf{W}_{6,i}$ must be defined

$$\Delta\mathbf{W}_i = \mathbf{W}_{r,i} - \mathbf{W}_{l,i} \quad \text{and} \quad \mathbf{W}_{6,i} = 6\left(W_i - \frac{1}{2}\big(\mathbf{W}_{l,i} + \mathbf{W}_{r,i}\big)\right)$$

Then the left $\tilde{\mathbf{W}}^{\sharp}_{l,i}$ and right $\tilde{\mathbf{W}}^{\sharp}_{r,i}$ are constructed according to

$$\mathbf{W}^{\sharp}_{l,i} = \mathbf{W}_{r,i} - \frac{1}{2}a^{\sharp}\left(\Delta\mathbf{W}_i - \left(1 - \frac{2}{3}a^{\sharp}\right)\mathbf{W}_{6,i}\right)$$

and

$$\mathbf{W}^{\sharp}_{r,i} = \mathbf{W}_{l,i+1} + \frac{1}{2}b^{\sharp}\left(\Delta\mathbf{W}_{i+1} + \left(1 - \frac{2}{3}b^{\sharp}\right)\mathbf{W}_{6,i+1}\right)$$

where

$$a^{\sharp} = \frac{\lambda^{\sharp}_i \Delta t}{\Delta x} \quad \text{and} \quad b^{\sharp} = -\frac{\lambda^{\sharp}_{i+1}\Delta t}{\Delta x}$$

The efficient left and right states of the Riemann problem are

$$\begin{aligned} p_{k,i+\frac{1}{2}} &= p_{k,i+\frac{1}{2}} + C^2_{k,i+\frac{1}{2}}\left(\beta^{+}_{k,i+\frac{1}{2}} + \beta^{-}_{k,i+\frac{1}{2}}\right) \\ u_{k,i+\frac{1}{2}} &= u_{k,i+\frac{1}{2}} + C_{k,i+\frac{1}{2}}\left(\beta^{-}_{k,i+\frac{1}{2}} + \beta^{-}_{k,i+\frac{1}{2}}\right) \\ \rho_{k,i+\frac{1}{2}} &= \left(\frac{1}{\rho_{k,i+\frac{1}{2}}} - \sum_{\sharp=-1,0,+1} \beta^{\sharp}_{k,i+\frac{1}{2}}\right)^{-1} \end{aligned} \tag{2.11}$$

where

$$C^2_{k,i+\frac{1}{2}} = \Gamma p_{k,i+\frac{1}{2}} \rho_{k,i+\frac{1}{2}}$$

for $k = l, r$. The coefficients $\beta^{\pm}_{k,i+\frac{1}{2}}$ are calculated as

$$\beta^{\sharp}_{l,i+\frac{1}{2}} = 0 \quad \text{if} \quad \lambda^{\sharp}_i \le 0$$

$$\beta^{\sharp}_{r,i+\frac{1}{2}} = 0 \quad \text{if} \quad \lambda^{\sharp}_{i+1} \ge 0$$

otherwise

$$\beta^{\pm}_{k,i+\frac{1}{2}} = \mp\frac{1}{2C_{k,i+\frac{1}{2}}}\left(\left(u_{k,i+\frac{1}{2}} - u^{\pm}_{k,i+\frac{1}{2}}\right) \pm \frac{p_{k,i+\frac{1}{2}} - p^{\pm}_{k,i+\frac{1}{2}}}{C_{k,i+\frac{1}{2}}}\right)$$

$$\beta^{0}_{k,i+\frac{1}{2}} = \left(\frac{p_{k,i+\frac{1}{2}} - p^{0}_{k,i+\frac{1}{2}}}{C^2_{i+\frac{1}{2}}} + \frac{1}{\rho_{k,i+\frac{1}{2}}} - \frac{1}{\rho^{0}_{k,i+\frac{1}{2}}}\right)$$

For every passive variable, such as the tangential velocity component v, the effective left and right states of the Riemann problem are

$$v_{l,i+\frac{1}{2}} = v_{l,i+\frac{1}{2}} \quad \text{if} \quad \lambda^{\sharp}_i \le 0$$

$$v_{r,i+\frac{1}{2}} = v_{r,i+\frac{1}{2}} \quad \text{if} \quad \lambda^{\sharp}_{i+1} \geq 0$$

otherwise

$$v_{k,i+\frac{1}{2}} = v^{0}_{k,i+\frac{1}{2}}$$

The data of the Riemann problem are

$$\mathbf{U}(x,0) = \begin{cases} \left(\rho_{l,i+\frac{1}{2}}, \rho_{l,i+\frac{1}{2}} u_{l,i+\frac{1}{2}}, E_{l,i+\frac{1}{2}}\right) & \text{if} \quad x < 0 \\ \left(\rho_{r,i+\frac{1}{2}}, \rho_{r,i+\frac{1}{2}} u_{r,i+\frac{1}{2}}, E_{r,i+\frac{1}{2}}\right) & \text{if} \quad x > 0 \end{cases}$$

The PPM reconstruction is third order accurate in smooth parts of the flow. In the following table the values of the free parameters used throughout the PPM reconstruction algorithm, are presented, as they can be found in the original paper by Woodward and Collela —.

$\eta^{(1)}$	$\eta^{(2)}$	$\omega^{(1)}$	$\omega^{(2)}$	$\epsilon^{(1)}$	$\epsilon^{(2)}$	K_0
20	0.05	0.75	10	0.01	0.01	0.1

Table 2.1: Values of the free parameters in the PPM scheme

WENO

The central weighted essentially non-oscillatory reconstruction (CWENO), as presented by Kurganov and Tadmor (2000), provides a third order accurate interpolant which is built from the given cell averages. This interpolant is initially written as a convex combination of two one-sided linear functions and one centered parabola. In smooth regions this convex combination guarantees the desired third order accuracy. It automatically switches to a second-order, one-sided linear reconstruction in the presence of large gradients.
In general in each cell I_i a quadratic polynomial as a convex combination of three polynomials $P_l(x)$, $P_r(x)$ and P_c is constructed

$$P_i(x) = w_l P_l(x) + w_r P_r(x) + w_c P_C(x)$$

with positive weights $w_k \geq 0$, where $k = c, l, r$ and $\sum_k w_k = 1$. The polynomials $P_l(x)$, $P_r(x)$ correspond to left and right one-sided linear reconstructions respectively, while $P_c(x)$ is a parabola centered around x_i. The efficient left and right state values $\mathbf{U}_l$ and $\mathbf{U}_r$ to be used in the Riemann problem are calculated as

$$\mathbf{U}_{l,i+\frac{1}{2}} = P_i\big(x_{i+\frac{1}{2}}, t\big) \quad , \quad \mathbf{U}_{r,i+\frac{1}{2}} = P_{i+1}\big(x_{i+\frac{1}{2}}, t\big) \tag{2.12}$$

In the CWENO reconstruction algorithm the quadratic polynomial in cell I_i at time t^n can be written as

$$P_i(x, t^n) = A_i + B_i\big(x - x_i\big) + \frac{1}{2} C_i \big(x - x_i\big)^2 \tag{2.13}$$

where the coefficients A_i, B_i, C_i are

$$A_i = \mathbf{U}_i - \frac{w_c}{12}\big(\mathbf{U}_{i+1} - 2\mathbf{U}_i + \mathbf{U}_{i-1}\big)$$

$$B_i = \frac{1}{\Delta x}\left[w_r\big(\mathbf{U}_{i+1} - \mathbf{U}_i\big) + w_c\frac{\mathbf{U}_{i+1} - \mathbf{U}_{i-1}}{2} + w_l\big(\mathbf{U}_i - \mathbf{U}_{i-1}\big)\right]$$

$$C_i = 2w_c\frac{\mathbf{U}_{i-1} - 2\mathbf{U}_i + \mathbf{U}_{i+1}}{\big(\Delta x\big)^2}$$

The weight functions w_k, where $k = c, l, r$ are taken as

$$w_k = \frac{a_k}{\sum_m a_m}$$

where $m = c, l, r$ and

$$a_k = \frac{c_k}{\big(\epsilon + IS_k\big)^p}$$

the values c_l, c_r, c_c are

c_l	c_r	c_c
0.25	0.25	0.2

Table 2.2: Values of the parameters c_c, c_r, c_l used in the CWENO reconstruction scheme.

The parameter ϵ guarantees that the denominator does not vanish and is taken as $\epsilon = 10^{-6}$. The value of p may be chosen to provide the highest accuracy in smooth areas and to ensure the non-oscillatory nature of the solution near the discontinuities and the value $p = 2$ is an effective empirical choice. Finally the smoothness indicators IS_k are defined as

$$IS_l = \big(\mathbf{U}_i - \mathbf{U}_{i-1}\big)^2$$

$$IS_r = \big(\mathbf{U}_{i+1} - U_i\big)^2$$

and

$$IS_c = \frac{13}{3}\big(\mathbf{U}_{i+1} - 2\mathbf{U}_i + \mathbf{U}_{i-1}\big)^2 + \frac{1}{4}\big(\mathbf{U}i + 1 - \mathbf{U}_{i-1}\big)^2$$

After having calculated the smoothness indicators IS_k, the weight function w_k and the coefficients A_i, B_i and C_i of the polynomial (2.13) the effective left and right values of the Riemann problem are

$$\mathbf{U}(x,0) = \begin{cases} \mathbf{U}_{l,i+\frac{1}{2}} = P_i\big(x_{i+\frac{1}{2}}, t\big) = A_i + \frac{\Delta x}{2}B_i + \frac{\big(\Delta x\big)^2}{8}C_i & \text{if} \quad x < 0 \\ \mathbf{U}_{r,i+\frac{1}{2}} = P_{i+1}\big(x_{i+\frac{1}{2}}, t\big) = A_{i+1} - \frac{\Delta x}{2}B_{i+1} + \frac{\big(\Delta x\big)^2}{8}C_{i+1} & \text{if} \quad x > 0 \end{cases}$$

2.6.2 Riemann solvers

ROE

The solver of Roe is probably the most popular approximate Riemann solver and it is presented according to Toro (1999). Roe's solver does not apply straightforward to the set of conservation laws but to their linearized form

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0 \Rightarrow \mathbf{U}_t + \mathbf{A}(\mathbf{U})\mathbf{U}_x = 0$$

where $\mathbf{A}(\mathbf{U}) = \frac{\partial \mathbf{F}}{\partial \mathbf{U}}$ is the Jacobian matrix. In Roe's approach the Jacobian matrix $\mathbf{A}(U)$is replaced by a constant Jacobian matrix $\tilde{\mathbf{A}}(\mathbf{U}) = \tilde{\mathbf{A}}(\mathbf{U}_l, \mathbf{U}_r)$ and so the original system of the PDEs becomes a linear system with constant coefficients

$$\mathbf{U}_t + \tilde{\mathbf{A}}\mathbf{U}_x = 0 \tag{2.14}$$

Using the constant Jacobian matrix valuable information can be extracted for the physical problem. These information concern the eigenvalues $\tilde{\lambda}^j$ of the system, the right eigenvectors $\tilde{\mathbf{R}}^j$ and the left ones $\tilde{\mathbf{L}}^j$ for $j = 1, ..., n$, where n is the number of the equations or equivalently the number of the variables. The numerical flux is obtained as

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{2}\big(\mathbf{F}(\mathbf{U}_l) + \mathbf{F}(\mathbf{U}_r) - \sum_{j=1}^{n} \tilde{a}^j \big|\tilde{\lambda}^j\big|\tilde{\mathbf{R}}^j\big)$$

where $\tilde{a}^j$ are the wave strengths, calculated as

$$\tilde{a}^j = \tilde{\mathbf{L}}^j \Delta\mathbf{U} = \tilde{\mathbf{L}}^j(\mathbf{U}_r - \mathbf{U}_l) \tag{2.15}$$

It must be noted that the constant Jacobian matrix $\tilde{\mathbf{A}}$ is evaluated using the values $\mathbf{U}_l$ and $\mathbf{U}_r$. These values combined in a proper manner provide the so called Roe-averages, $\mathbf{U}_{ROE} = V(\mathbf{U}_r, \mathbf{U}_l)$, where V is the Roe averaging operator. There is a choice of which intermediate state the Jacobian $\tilde{\mathbf{A}}(\mathbf{U}_{ROE})$ should be evaluated at. There are three criteria that ensure the stability and consistency of the numerical flux

1. $\tilde{\mathbf{A}}(\mathbf{U}_{ROE})(\mathbf{U}_r - \mathbf{U}_l) = \mathbf{F}(\mathbf{U}_r) - F(\mathbf{U}_l)$
2. $\tilde{\mathbf{A}}(\mathbf{U}_{ROE})$ is diagonalizable with real eigenvalues
3. $\tilde{\mathbf{A}}(\mathbf{U}_{ROE}) \rightarrow \frac{\partial \mathbf{F}}{\partial \mathbf{U}}$ as $\mathbf{U}_{ROE} \rightarrow \mathbf{U}$

However popular and robust in many cases Roe's approximate state Riemann solver faces serious deficiencies when the problem is not linearizable. It has been shown that if the initial velocity is negative this situation really occurs even though the problem has a solution with positive density and internal energy. The reason for the failure of the linearization is the occurrence of two rarefaction waves in the exact solution to the Riemann problem.

UNO

The uniformly high-order accurate non-oscillatory schemes (UNO) share many desirable properties with total variation diminishing schemes, but TVD schemes have at most first-order accuracy, in the sense of truncation errors, at extrema of the solution. ON the other hand UNO schemes are uniformly second order approximations , since the number of extrema of the discrete solution is not increasing in time. Unlike TVD schemes the non-oscillatory ones are not required to damp the values of each local extremum at every single time step, but are allowed to occasionally accentuate a local extremum. It has to be mentioned that the price to pay for this extra accuracy is the loss of the TVD property. One of the first UNO schemes was introduced by Ami Harten and Stanley Osher (1987) and it will be briefly discussed in the case of hyperbolic conservation laws. This discussion follows the paper by Harten and Osher (1987).
The UNO scheme follows several of the ideas initially used for the derivation of Roe's approximate Riemann solver. As in Roe's algorithm the linear set of equations (2.14) with constants coefficients will be solved. In order to accomplish that Roe's averaging must be used

$$\mathbf{U}_{ROE,i} = V(\mathbf{U}_i, \mathbf{U}_{i+1})$$

The next step is the calculation of the eigenvalues $\tilde{\lambda}^j$, of the wave strengths a^j using equation (2.15), of the right eigenvectors $\tilde{\mathbf{R}}^j$ and of the left ones $\tilde{\mathbf{L}}^j$ using the constant Jacobian matrix $\mathbf{A}(\mathbf{U}_i, \mathbf{U}_{i+1}) = \mathbf{A}(\mathbf{U}_{ROE,i})$ from equation (2.14). The numerical flux is given by

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{2}\big[\mathbf{F}(\mathbf{U}_i) + \mathbf{F}(\mathbf{U}_{i+1}) - \sum_{j=1}^{n} c_i^j R_i^j\big]$$

where n is the number of the variables and

$$\begin{aligned} c_i^j = |\lambda_i^j| a_i^j - max(0, \lambda_i^j)(1 - \frac{\Delta t}{\Delta x}\lambda_{i-1}^j)\hat{S}_i^j \\ + min(0, \lambda_i^j)(1 + \frac{\Delta t}{\Delta x}\lambda_{i+1}^j)\hat{S}_{i+1}^j \end{aligned}$$

$\hat{S}_i^j$ denotes the component of the vector of slopes in the j^{th} characteristic field and is defined as follows

$$\hat{S}_i^j = \frac{m(S_{-,i}^j, S_{+,i}^j)}{1 + \frac{\Delta t}{\Delta x}(\lambda_i^j - \lambda_{i-1}^j)}$$

where $m(x, y)$ is the minmod function———. $S_{\pm,i}^j$ are defined as

$$S_{-,i}^j = a_{i-1}^j + \frac{1}{2}D_i^j \quad and \quad S_{+,i}^j = a_i^j - \frac{1}{2}D_i^j$$

where

$$D_i^j = m(a_{i+1}^j - a_i^j, a_i^j - a_{i-1}^j)$$

The UNO scheme provides satisfactory results even in the cases the Roe's approximate Riemann solver blows up.

Marquina

Marquina's flux formula (Donat and Marquina (1996), Font, Martí, Marquina, Müller (1997)) introduces a dissipative mechanism into the numerical scheme especially designed to eliminate undesired pathologies which flaw most high-order shock-capturing methods, if no excessive smearing is introduced at discontinuities. Contrary to other linearized solvers Marquina's Riemann solver is not based on averaging's being able to solve Riemann problems with different left and right equations of state. It has been designed for general hyperbolic systems of conservation laws.
The presentation of the method is based on the original paper by Donat and Marquina (1996). Marquina's flux formula applied to a system of conservation laws in one dimension, yields a conservative method whose numerical flux function is computed as follows : Given left and right states, the sided local characteristic variables and fluxes are calculated according to

$$\omega_l^j = \mathbf{L}^j(\mathbf{U}_l)\mathbf{U}_l \quad , \quad \phi_l^j = \mathbf{L}^j(\mathbf{U}_l)\mathbf{F}(\mathbf{U}_l)$$
$$\omega_r^j = \mathbf{L}^j(\mathbf{U}_r)\mathbf{U}_r \quad , \quad \phi_r^j = \mathbf{L}^j(\mathbf{U}_r)\mathbf{F}(\mathbf{U}_r)$$

for $j = 1, ..., n$ where n is the number of equations of the system. Here $\mathbf{L}^j(\mathbf{U}_l)$ and $\mathbf{L}^j(\mathbf{U}_r)$ are the normalized left eigenvectors of the Jacobian matrix of the system of conservation laws, calculated in the left and right states.
Let $\lambda^1(\mathbf{U}_l), ..., \lambda^n(\mathbf{U}_l)$ and $\lambda^1(\mathbf{U}_r), ..., \lambda^n(\mathbf{U}_r)$ be the corresponding eigenvalues. Then for every j=1,...,n the procedure is as follows

1. If $\lambda^j(\mathbf{U})$ does not change sign in $\left[\mathbf{U}_l, \mathbf{U}_r\right]$, namely if $\lambda^j(\mathbf{U}_l)\lambda^j(\mathbf{U}_r) > 0$, then the scalar scheme is upwind and the numerical flux is calculated according to the relevant characteristic information If $\lambda^j(\mathbf{U}_l)$¿0, then

   $$\phi_+^j = \phi_l^j \quad , \quad \phi_-^j = 0$$

   else

   $$\phi_+^j = 0 \quad , \quad \phi_-^j = \phi_r^j$$

2. Otherwise, as a way to avoid numerical pathologies, the scalar scheme is switched to the more viscous, entropy-satisfying local Lax-Friedrichs scheme

   $$a^j = max\left[\left|\lambda^j(U_l)\right|, \left|\lambda^j(U_r)\right|\right]$$

   $$\phi_+^j = \frac{\phi_l^j + a^j\omega_l^j}{2} \quad , \quad \phi_-^j = \frac{\phi_r^j - a^j\omega_r^j}{2}$$

Marquina's flux formula is then

$$\mathbf{F}_{i+\frac{1}{2}} = \sum_{j=1}^{n} \left[\phi_{+}^{j}\mathbf{R}^{j}(\mathbf{U}_l) + \phi_{-}^{j}\mathbf{R}^{j}(\mathbf{U}_r)\right]$$

where $\mathbf{R}^j(U_l)$ and $\mathbf{R}^j(U_r)$ are the right normalized eigenvectors of the Jacobian matrices $\mathbf{A}(\mathbf{U}_l), \mathbf{A}(\mathbf{U}_r)$.

HLL

The Harten-Lax-vanLeer (HLL) algorithm is a very simple approximate Riemann solver, presented extensively by Toro (1999). The algorithm assumes that the maximum signal velocities, S_l and S_r for left and right moving waves, respectively, are known. The three states for this solver are the initial left and right states, U_l and U_r, for the region beyond the distance travelled by the fastest left and right moving signals and a single intermediate state U_{lr}

$$U(x,t) = \begin{cases} U_l & if \quad \frac{x}{t} \leq S_l \\ U^{hll} & if \quad S_l \leq \frac{x}{t} \leq S_r \\ U_r & if \quad \frac{x}{t} \geq S_r \end{cases}$$

The middle state U^{hll} is a constant vector determined by requiring energy conservation in the computational cell $\left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ and is given by

$$U^{hll} = \frac{S_r U_r - S_l U_l + F_l - F_r}{S_r - S_l} \tag{2.16}$$

The approximate HLL Riemann solver assumes the existence of just two waves, which separate three constant states. In fact it does not take into account the contact wave, since all the procedures that take place in the interior of the Riemann problem are described by a single constant state. The corresponding flux F^{hll} along the t-axis is

$$F^{hll} = \frac{S_r F_l - S_l F_r + S_l S_r (U_r - U_l)}{S_r - S_l} \tag{2.17}$$

Then the intercell numerical flux for the approximate Godunov method is given by

$$F_{i+\frac{1}{2}}^{hll} = \begin{cases} F_l & if \quad 0 \leq S_l \\ F^{hll} & if \quad S_l \leq 0 \leq S_r \\ F_r & if \quad 0 \geq S_r \end{cases} \tag{2.18}$$

The remaining pendency in order to fully describe the HLL solver are the estimates for the wave speeds S_l and S_r. Two simple estimates are

$$S_l = u_l - cs_l \quad , \quad S_r = u_r - cs_r \tag{2.19}$$

and

$$S_l = min\big[u_l - cs_l, u_r - cs_r\big] \quad , \quad S_r = max\big[u_l + cs_l, u_r + cs_r\big] \tag{2.20}$$

The HLL method is unfortunately very diffusive, but it remains an important approximate Riemann solver. Its simplicity makes it very easy to implement and computationally very efficient. This makes the HLL solver ideal for an initial approach to a problem and as a sanity check for more complicated schemes. It has been demonstrated that the HLL solver, when beginning with physical initial data, always produces a physical intermediate state, a property known as positively conservative. In addition the linear solvers, such as the Roe solver, do not have this property and often produce solutions with negative pressures in regions where the fluid density is low.

HLLC

The HLLC scheme, introduced by Toro, Spruce and Spears (1994), is a modification of the HLL solver described in the previous section, whereby the missing contact discontinuity is restored. The HLLC approximate Riemann solver is given as follows, according to Toro (1999)

$$U(x,t) = \begin{cases} U_l & if \quad \frac{x}{t} \leq S_l \\ U_{*l} & if \quad S_l \leq \frac{x}{t} \leq S_* \\ U_{*r} & if \quad S_* \leq \frac{x}{t} \leq S_r \\ U_r & if \quad \frac{x}{t} \geq S_r \end{cases}$$

The solution vector is

$$U_{*k} = \rho_k \left(\frac{S_k - u_k}{S_k - S_*}\right) \begin{bmatrix} 1 \\ S_* \\ \frac{E_k}{\rho_k} + (S_* - u_k)\big[S_* + \frac{p_k}{\rho_k(S_k - u_k)}\big] \end{bmatrix} \tag{2.21}$$

for $k = l$ and $k = r$. The HLLC flux can be written as

$$F^{hllc}_{i+\frac{1}{2}} = \begin{cases} F_l & if \quad 0 \leq S_l \\ F_{*l} = F_l + S_l(U_{*l} - U_l) & if \quad S_l \leq 0 \leq S_* \\ F_{*r} = F_r + S_r(U_{*r} - U_r) & if \quad S_* \leq 0 \leq S_r \\ F_r & if \quad 0 \geq S_l \end{cases} \tag{2.22}$$

where U_{*l} and U_{*r} are given by (2.21). For any passive scalar q advected with the fluid, such as the tangential velocity component v, the corresponding HLLC state is given by

$$(\rho q)_{*k} = \rho_k \left(\frac{S_k - u_k}{S_k - S_*}\right) q_k \tag{2.23}$$

for $k = l$ and $k = r$. The HLLC algorithm requires information for three different wave speeds, namely S_l, S_r and S_*. If the pressure p_* and the particle velocity u_* are known

within the region the Riemann problem has evolved then the following wave speeds can be chosen

$$S_l = u_l - cs_l p_l \quad , \quad S_* = u_* \quad , \quad S_r = u_r + cs_r p_r \tag{2.24}$$

where

$$p_k = \begin{cases} 1 & if \quad p_* \leq p_k \\ \left[1 + \frac{\Gamma+1}{2\Gamma}\left(\frac{p_*}{p_k} - 1\right)\right]^{\frac{1}{2}} & if \quad p_* \geq p_k \end{cases} \tag{2.25}$$

The choice of wave speeds discriminates between rarefraction and shock waves. The wave relations used are exact but the the pressure ratio across the shock is approximated, since the solution for p_* is an approximation. In order to estimate the value of p_* and u_* it is favourable to use the adaptive non-iterative scheme (ANRS). This scheme is especially designed to calculate the values of p_* and u_* and moreover it can stand by itself as an *approximate-state* Riemann solver. It is a conjunction of three different schemes, namely the primitive variable Riemann solver (PVRS), the two-rarefraction Riemann solver (TRRS), the two-shock Riemann solver (TSRS). These three algorithms are combined in a proper manner. The PVRS approximate Riemann solver gives

$$p_{pv} = \frac{1}{2}(p_l + p_r) - \frac{1}{2}(u_r - u_l)\bar{\rho}\bar{\alpha}$$

$$u_{pv} = \frac{1}{2}(u_l + u_r) - \frac{1}{2}\frac{(p_r - p_l)}{\bar{\rho}\bar{\alpha}}$$

where

$$\bar{\rho} = \frac{1}{2}(\rho_l + \rho_r) \quad , \quad \bar{\alpha} = \frac{1}{2}(\alpha_l + \alpha_r)$$

The TRRS approximate Riemann solver yields

$$p_{tr} = \left[\frac{\alpha_r + \alpha_l - \frac{\Gamma-1}{2}(u_r - u_l)}{\frac{\alpha_l}{p_l^z} + \frac{\alpha_r}{p_r^z}}\right]^{\frac{1}{z}}$$

$$u_{tr} = \frac{\frac{P_{lr}u_l}{\alpha_l} + \frac{u_r}{\alpha_r} + \frac{2(P_{lr}-1)}{\Gamma-1}}{\frac{P_{lr}}{a_l} + \frac{1}{\alpha_r}}$$

where

$$P_{lr} = \left(\frac{p_l}{p_r}\right)^z \quad , \quad z = \frac{\Gamma - 1}{2\Gamma}$$

The TSRS Riemann solver gives

$$p_{ts} = \frac{g_l(p_0)p_l + g_r(p_0)p_r - (u_r - u_l)}{g_l(p_0) + g_r(p_0)}$$

$$u_{ts} = \frac{1}{2}(u_l + u_r) + \frac{1}{2}\big[(p_{ts} - p_r)g_r(p_0) - (p_{ts} - p_l)g_l(p_0)\big]$$

where

$$g_k(p) = \big[\frac{A_k}{p + B_k}\big]^{\frac{1}{2}} \quad , \quad p_0 = max(0, p_{pv}) \tag{2.26}$$

for $k = l$ and $k = r$.
The ANRS algorithm combines the PVRS scheme together with the non-iterative TRRS and TSRS solvers. Initially a a number of parameters has to be defined

$$p_{min} = max(p_l, p_r) \quad , \quad p_{max} = max(p_l, p_r) \quad , \quad p_* = p_{pv} \quad , \quad Q = \frac{p_{max}}{p_{min}}$$

One more parameter used is defined as Q_{user} and the value appointed to it should be $Q_{user} = 2$.
The ANRS algorithm is as follows

1. If the conditions $Q < Q_{user}$ and $p_{min} < p_* < p_{max}$ are met then the PVRS scheme is used.

2. If the above conditions are not met and $p_* < p_{min}$ then the TRRS algorithm is used, otherwise if $p_* > p_{min}$ the TSRS is used.

HLLC-MHD

Recently the HLLC three-state approximate Riemann solver was generalized to ideal MHD by Li (2005). Li' s HLLC solver reduces to the Toro et al. (1994) and Batten et al. (1997) HLLC solver in the HD case, while it is consistent with the integral form of the conservation laws. Even though the exact solution of the MHD Riemann problem involves seven different states (one entropy wave, two Alfvén waves and four magnetosonic waves) the three-state HLLC solver by Li achieves an accuracy close to that of the linearized solver by Roe (1981), but at a reduced computational cost and without the need for an eigen-decomposition.
The system of the MHD equations in conservative form consists of equations (1.31) to (1.34). The conservative variables are $U = \big[\rho, \rho u, \rho v, \rho w, B_x, B_y, B_z, E\big]$. The description of the HLLC-MHD method follows Li's (2005) original paper. The solution vector is

$$U_{*k} = \begin{bmatrix} \rho_k\big(\frac{S_k - u_k}{S_k - S_*}\big) \\ \rho_k\big(\frac{S_k - u_k}{S_k - S_*}\big) S_* \\ \rho_k v_k\big(\frac{S_k - u_k}{S_k - S_*}\big) - \frac{B_x^* B_y^* - B_x B_y}{S_k - S_*} \\ \rho_k w_k\big(\frac{S_k - u_k}{S_k - S_*}\big) - \frac{B_x^* B_z^* - B_x B_z}{S_k - S_*} \\ B_x \\ B_y^{HLL} \\ B_z^{HLL} \\ E_k\big(\frac{S_k - u_k}{S_k - S_*}\big) + \frac{(p_* S_* - p* u_k) - \big(B_x^*(\mathbf{B}^{HLL}\mathbf{u}^{HLL}) - B_x(\mathbf{Bu})\big)}{S_k - S+*} \end{bmatrix} \tag{2.27}$$

where it is

$$S_* = \frac{\rho_r u_r\big(S_r - u_r\big) - \rho_l u_l\big(S_l - u_l\big) + p_l - p_r - B_{xl}^2 + B_{xr}^2}{\rho_r\big(S_r - u_r\big) - \rho_r\big(S_l - u_l\big)} \tag{2.28}$$

and

$$p_* = \rho\big(S_l - u\big)\big(S_* - u\big) + p \tag{2.29}$$

where S_r, S_l are evaluated according to Einfeldt et al. (1991) and p is the total pressure, given by equation (1.35). The variables B_y^* and B_z^* are calculated according to the HLL formula (2.16). The quantity u^{HLL} can be calculated from the conservative variables U^{hll}. The HLLC-MHD flux formula is given by equation (2.18).

Central scheme

According to Kurganov and Tadmor (2000) a fully-discrete central scheme is constructed, by building an intermediate mesh of variable cell length, making use of the *local speed of propagation* at each cell interface $a_{i+\frac{1}{2}}$ defined by

$$a_{i+\frac{1}{2}} = max\big[\rho\big(\frac{\partial \mathbf{F}}{\partial \mathbf{U}}(U_{l,i+1})\big), \rho\big(\frac{\partial \mathbf{F}}{\partial \mathbf{U}}(\mathbf{U}_{r,i})\big)\big]$$

where $\rho(A) = max_i(\big|\lambda^j(A)\big|)$ with $\lambda^j(A)$ being the eigenvalues of the Jacobian matrix $\mathbf{A} = \frac{\partial \mathbf{F}}{\partial \mathbf{U}}$ and $j = 1, ..., n$, where n is the number of the system's equations. In addition, subscripts l and r in the above equation stand for the reconstructed values of $\mathbf{U}$ at the left and right sides of the corresponding numerical cell (i+1 and i, respectively).
The numerical flux function is given by

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{2}\big[\mathbf{F}(\mathbf{U}_{l,i+1}) + \mathbf{F}(\mathbf{U}_{r,i})\big] - \frac{a_{i+\frac{1}{2}}}{2}\big[\mathbf{U}_{l,i+1} - U_{r,i}\big]$$

This numerical flux depends only on the local propagation speeds $a_{i+\frac{1}{2}}$ and due to its simple form, it can be implemented and extended to any spatial order straightforwardly.

2.7 Vacuum tracking algorithm

The vacuum tracking algorithm was initially proposed by Munz (1994) and Munz, Scneider, Gerlinger (1994) and the following discussion is based on their publications. The approximation of gas flow with a gas-vacuum boundary may give rise to severe difficulties in the numerical schemes. This is due to the fact that in the region of vacuum the equations of Euler, which are based on the continuum assumption, are no longer valid and a numerical approximation based on this set of equations will fail. There is always the temptation to replace the vacuum region by a gas of low density and pressure but if this procedure is done in approximate Riemann solvers, then the local wave structure of the solution is

essentially altered. In this case the conservation laws are violated and non-linear instabilities will be generated. This effect has been observed and studied extensively especially for Godunon-type schemes. These difficulties can be avoided if a tracking method is used, since in that case the propagation of gas-vacuum boundary is followed and the information about its actual location is used to determine the numerical flux between the grid zones at the vacuum interface.
From a physical point of view it is important to note that a shock wave can not be adjacent to a vacuum region, since the Rankine-Hugoniot conditions are no longer satisfied in this case. This observation does not stand in the case of a contact discontinuity which is in fact the boundary between the state and the vacuum regions. The solution of the vacuum Riemann problem consists of a rarefraction wave and a contact discontinuity and its initial values, considering that the right state is the one containing the vacuum, are

$$U(x,0) = \begin{cases} \big(\rho_l, \rho_l u_l, E_l\big) & for \quad x \leq 0 \\ (0,0,0) & for \quad x > 0 \end{cases}$$

It must be noted that this problem is no real initial value problem, but a free boundary problem. The complete solution in the case of a right vacuum state can be written as

$$U(x,t) = \begin{cases} \big(\rho_l, \rho_l u_l, E_l\big) & for \quad \frac{x}{t} \leq u_l - cs_l \\ \big(\rho_0, \rho_0 u_0, E_0\big) & for \quad u_l - cs_l \leq \frac{x}{t} \leq u_l + \frac{2cs_l}{\Gamma - 1} \\ (0,0,0) & otherwise \end{cases}$$

with

$$\begin{aligned} u_0 &= \frac{\big[(\Gamma-1)u_l + 2\big(\frac{x}{t} + cs_l\big)\big]}{\Gamma + 1} \\ \rho_0 &= \big[(u_0 - \frac{x}{t})^2 \frac{\rho_l^{\Gamma}}{\Gamma p_l}\big]^{\frac{1}{\Gamma-1}} \\ p_0 &= \frac{{\rho_0}^{\Gamma}}{\rho_l} p_l \end{aligned} \tag{2.30}$$

where cs_l is the speed of the sound and its form depends on the equation of state used and Γ is the adiabatic exponent. For completeness the formula for the vacuum being situated on the left-hand side is also presented

$$\begin{aligned} u_0 &= \frac{\big[(\Gamma-1)u_r + 2\big(\frac{x}{t} + cs_r\big)\big]}{\Gamma + 1} \\ \rho_0 &= \big[(\frac{x}{t} - u_0)^2 \frac{\rho_r^{\Gamma}}{\Gamma p_r}\big]^{\frac{1}{\Gamma-1}} \\ p_0 &= \frac{{\rho_0}^{\Gamma}}{\rho_r} p_r \end{aligned} \tag{2.31}$$

within the region $u_r - \frac{2cs_r}{\Gamma-1} \leq \frac{x}{t} \leq u_r + cs_r$.
Unfortunately in practical applications the use of this solution to the vacuum Riemann

problem provides inaccurate results and it has to be modified so that the movement of the vacuum boundary can also be taken into account. This is the main idea of the vacuum tracking algorithm, in which at a first step there is an estimation of the real movement of the boundary and according to this the numerical flux is determined in a second step in such a way that the gas-vacuum interface remains sharp.
Consider that the location of the vacuum boundary at the time level t^n is known and situated in the i^{th} grid zone. Its location is named x_v^n. Initially it can be set as $x_v^n = r_i$. This is the appropriate point to make a distinction as for the procedure to follow in order to predict the location of the boundary at the time t^{n+1}. If throughout the calculations the simple piece-wise reconstruction has been used, then the left state U_l is given by the formula

$$U_l = a\bar{U}_i^n + (1-a)U_{i-1}^n \tag{2.32}$$

with

$$a = \frac{x_v^n - x_{i-\frac{1}{2}}}{h} \quad and \quad \bar{U}_i^n = \frac{x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}}{x_v^n - x_{i-\frac{1}{2}}} U_i^n \tag{2.33}$$

On the other hand if a reconstruction scheme of higher order has been used, i.e. PPM, then the left states can easily be derived directly by those provided by the reconstruction algorithm and the previous step (equations (2.32) and (2.33)) can be ignored. The location of the boundary at t^{n+1} is given by

$$x_v^{n+1} = x_v^n + (t^{n+1} - t^n)u_v^n \tag{2.34}$$

with

$$u_v^n = u_l + \frac{2}{\Gamma - 1} cs_l \tag{2.35}$$

It is strongly recommended that the above algorithm should be programmed using both the primitive and the conservative variables in order to check which set of variables provides the best results. It has been shown that the primitive formulation can be favourable in several cases when compared to the conservative one. The information obtained about the approximation of the gas-vacuum boundary can be used for the calculation of the numerical flux to the regions adjacent to it. There are the following two cases to distinguish

1. If $\left(x_v^{n+1} - x_{i-\frac{1}{2}}\right) \leq \epsilon h$, where ϵ is a small positive number $\left(\epsilon = 0.01\right)$ then the grid zone interface at $x_{i+\frac{1}{2}}$ lies in the vacuum during the whole time and hence the numerical flux must be zero
$$F_{i+\frac{1}{2}} = 0$$

2. If $\left(x_v^{n+1} - x_{i-\frac{1}{2}}\right) \geq \epsilon h$ then the gas-vacuum boundary moves across the grid zone interface. In this case the flux vector at $x_{i+\frac{1}{2}}$ becomes non-zero and is given by the formula
$$F_{i+\frac{1}{2}} = \begin{cases} F(U_i^n) & if \quad U_i^n - cs_i^n > 0 \\ F(U_0(0,t)) & otherwise \end{cases}$$

with U_0 given by (2.30). F(U) is the vector of the physical fluxes.

The above calculation of the fluxes is based on the exact solution of the Vacuum Riemann problem, but it can be replaced by a much simpler averaged one. Two different approximate flux calculations are presented shortly

$$F_{i+\frac{1}{2}} = \frac{u_v}{u_v - min(0, u_l - cs_l)}\big(F(U_i^n) - min(0, u_l - cs_l)U_i^n\big) \tag{2.36}$$

and

$$\begin{aligned} F_{i+\frac{1}{2}} = F(U_l) &- \frac{1}{\Delta t} min\big[(u_l - cs_l)\Delta t, x_{i+\frac{1}{2}} - x_v^n\big]U_l \\ &- \frac{1}{\Delta t} max\big[0, x_{i+\frac{1}{2}} - x_v^n - (u_l - cs_l)\Delta t\big]U_{lr} \end{aligned} \tag{2.37}$$

with

$$U_{lr} = \frac{F(U_l) - (u_l - cs_l)u_l}{u_v - u_l + cs_l}$$

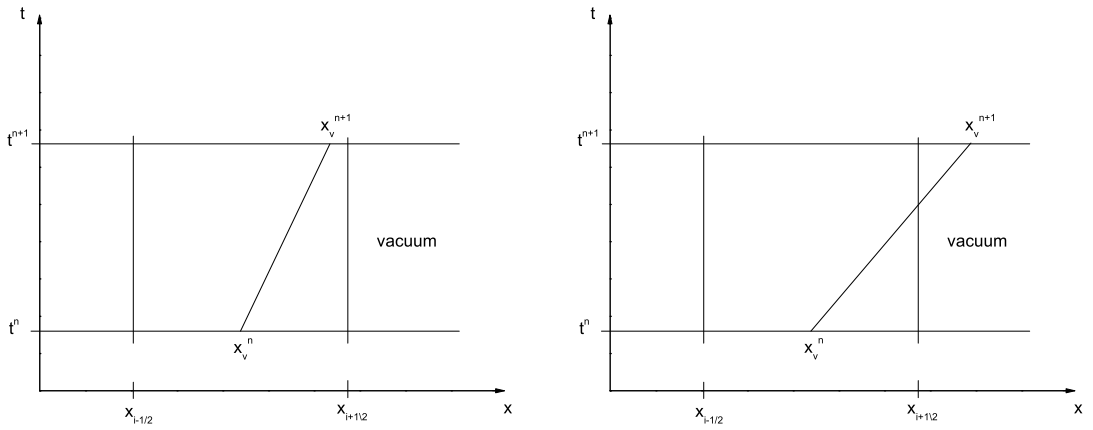


Table 2.3: Tracking of the gas vacuum boundary for cases (1) and (2).

The vacuum boundary method can be adjusted to any of the numerical methods presented previously and in a very easy manner. The solvers have to be modified slightly in order to be able to detect the vacuum region. The procedure is as follows

1. if($\rho_i > 0$ and $\rho_{i+1} > 0$) then use the Riemann solver
2. else if($\rho_i > 0$ and $\rho_{i+1} = 0$) then activate the vacuum-tracking algorithm
3. else vacuum

2.8 Source terms

In many applications source terms are present in the equations and this fact poses new challenging problems that concern their solution. Such is the case of equations —, where a geometric source term vector appears in the right-hand side. A non-linear system of hyperbolic conservation laws with source terms can be written in the form

$$U_t + F(U)_x = S(U) \tag{2.38}$$

where U is the vector of conserved variables, F is the vector of fluxes and S is the vector of source terms, which in general is a function of U or other physical parameters of the problem studied. There are essentially two ways of solving inhomogeneous systems of the form (2.38) namely the *splitting method* and the *method of lines*.

2.8.1 Splitting method

The main idea of this approach, as presented in Toro (1999), is to split equation (2.38), for a time Δt into two different problems. The main goal is to evolve U^n from time $t = t^n$ to the new value U^{n+1} at time $t = t^{n+1}$ in a designated time step Δt. The splitting method consists of two steps. Initially the homogeneous problem has to be solved for a whole time step using the problem's initial data

$$U_t + F(U)_x = 0 \tag{2.39}$$

This can be achieved by using directly any of the methods described earlier in the present chapter. The results obtained after having completed this step for time equal to Δt are denoted as $\bar{U}^{n+1}$ and are used as the initial data for the second step, which consists of the solution of the source problem

$$\frac{d}{dt}U = S(U) \tag{2.40}$$

The solution to the set of the ordinary equations (2.40) is in the fact the solution U^{n+1} to the inhomogeneous problem (2.38). In order to solve the set of ODEs an appropriate method must be used, i.e. Runge-Kutta. The philosophy of the splitting method is restricted to solving two different systems of equations using in each case the appropriate numerical methods and can be expressed in the form

$$U^{n+1} = \mathcal{S}^{(\Delta t)}\mathcal{C}^{(\Delta t)}U^n$$

where $mathcal S^t$ is interpreted as the solution operator for the problem (2.40) applied over the time interval Δt and $\mathcal{C}^t$ for the homogeneous problem over the same time interval. This procedure is only first order in time, when S and C are at least first-order accurate solution operators. As long as $\mathcal{C}$ and $\mathcal{S}$ are at least second order accurate a fully second order scheme is

$$U^{n+1}(x) = \mathcal{S}^{(\frac{1}{2}\Delta t)}\mathcal{C}^{(\Delta t)}\mathcal{S}^{(\frac{1}{2}\Delta t)}U^n(x)$$

The method chosen for the solution of the system of ODEs is a fourth-order, four stage explicit Runge-Kutta

$$\begin{aligned} k_1 &= \Delta t\, s(q^n) \\ k_2 &= \Delta t\, s(q^n + \tfrac{1}{2}k_1) \\ k_3 &= \Delta t\, s(q^n + \tfrac{1}{2}k_2) \\ k_4 &= \Delta t\, s(q^n + k_3) \\ U^{n+1} &= U^n + \tfrac{1}{6}\big[k_1 + 2k_2 + 2k_3 + k_4\big] \end{aligned}$$

The main advantage of the splitting method is that whenever chosen it is easy to choose and implement the best method for each one of the operators $\mathcal{S}$ and $\mathcal{C}$. On the other hand it has been found that it does not provide accurate results when steady state solutions are to be computed since in such a case it breaks the balance between the convection and the source term.

2.8.2 Method of lines

A different approach to the solution of the system of equation (2.38) is the so-called method of lines. In this case the spatial derivatives have to be discretized

$$F(U)_x = \frac{1}{h}\big(F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}\big)$$

in order to express the time-dependent evolution of equation (2.38) in the semi-discrete form

$$\frac{d}{dt}U_i = S_i - \frac{1}{h}\big(F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}\big) \tag{2.41}$$

where h is the grid step and $F_{i+\frac{1}{2}}$,$F_{i-\frac{1}{2}}$ are the numerical fluxes at the cell interface. The term S_i is the average over of S(U) over the cell $\big[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\big]$, which can be defined for example by $S_i \approx S(U_i)$.
In order to achieve high-order accuracy in time, the time integration is done using a high-order total variation diminishing (TVD) Runge-Kutta scheme, which combines the first-order forward Euler steps and involves prediction and correction. For example, the third-order accuracy can be achieved via

$$U^{(1)} = U^n + \Delta t L(U^n)$$

$$U^{(2)} = \frac{3}{4}U^n + \frac{1}{4}U^{(1)} + \frac{1}{4}\Delta t L(U^{(1)})$$

$$U^{n+1} = \frac{1}{3}U^n + \frac{2}{3}U^{(2)} + \frac{2}{3}\Delta t L(U^{(2)})$$

where L(U) is the right-hand side of equation (2.41) and U^{n+1} is the final value after advancing one time step from U^n. The fluxes across the cell interfaces can be calculated using any of the the Riemann solvers previously presented in this chapter. The main

advantage of the method of lines when compared to the splitting method is that it does not face problems if steady state problems are to computed.
However for both of the methods it must be noted that they provide unsatisfactory results if the source terms are stiff. In this case the characteristic scale of the source term is order of magnitude different from the characteristic scale of the homogeneous part. This provides severe numerical burden beyond the classical numerical stability problem, which can usually be removed by using implicit methods for the source term. In most of the cases, unless the numerical calculations fully resolve the small scale, in the sense that the time step and the spatial mesh size is in the order of the small stiff parameter, unphysical solutions appear.

2.9 Poisson's equation

The Newtonian gravitational potentional Φ satisfies Poisson' equation

$$\nabla^2\Phi(r) = 4\pi G\rho \tag{2.42}$$

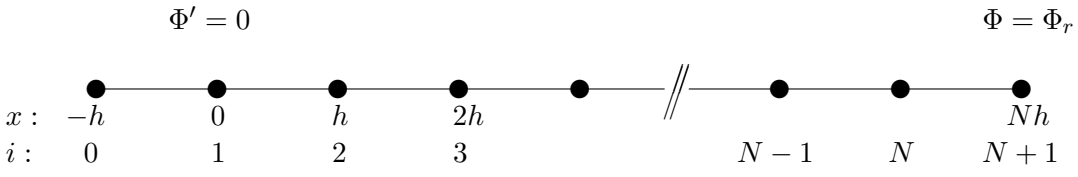
along with the initial conditions

$$\begin{aligned} \Phi'(0) &= 0 \\ \Phi(L) &= \Phi_r \end{aligned} \tag{2.43}$$

within the spatial domain $0 < x < L$. G is the gravitational constant and ρ is the density. Poisson's equation is classified as an elliptic equation and hence it can be treated as a boundary value problem in ordinary differential equations and its solution consists of deriving difference equations and solving all of them simultaneously. Most of the difficulty in solving elliptic equations lies in the solution of these large sets of algebraic equations. A fast tridiagonal solution algorithm has been chosen to carry out this task.
Consider the Poisson's equation in cylindrical coordinates in the one dimensional region

$$\frac{\partial^2\Phi}{\partial r^2} + \frac{1}{r}\frac{\partial\Phi}{\partial r} = 4\pi G\rho \tag{2.44}$$

Let the region be described using a uniform mesh with spacing $h = N^{-1}$ for some integer N.



The conventional finite difference approximations both to the first and second order

derivatives are

$$\frac{\partial \Phi}{\partial r} = \frac{\Phi_{i+1} - \Phi_{i-1}}{2h}$$
$$\frac{\partial^2 \Phi}{\partial r^2} = \frac{\Phi_{i+1} - 2\Phi_i + \Phi_{i-1}}{h^2}$$

Substituting these central difference approximations to (2.44) the following difference equation is derived

$$(2r_i - h)\Phi_{i-1} - 4r_i\Phi_i + (2r_i + h)\Phi_{i+1} = 8\pi G h^2 r_i \rho_i \tag{2.45}$$

Equation (2.45) applies to all the grid points except for $i = 1$ and $i = N + 1$. A special treatment for these points must be carried out using the boundary conditions (2.43). The left boundary condition is equivalent to a symmetry boundary condition. This can easily be proven once a hypothetical grid point $i = 0$ located at $x = -h$ is considered. The finite difference approximation yields for the left boundary condition at $i = 1$

$$\frac{\partial \Phi}{\partial r} \mid_{i=1} = 0 \Rightarrow \frac{\Phi_2 - \Phi_0}{2h} = 0 \Rightarrow$$
$$\Phi_0 = \Phi_2$$

The difference equation (2.45) becomes

$$-\Phi_1 + \Phi_2 = 2\pi G h^2 \rho_1 \tag{2.46}$$

Since $\Phi_{N+1} = \Phi(L) = \Phi_r$ at the right boundary equation (2.45) for $i = N$ is written

$$(2r_N - h)\Phi_{N-1} - 4r_N\Phi_N = 8\pi G h^2 r_N \rho_N - (2r_N + h)\Phi_r \tag{2.47}$$

The set of equations (2.45),(2.46) and (2.47) are written together in the matrix form

$$\begin{pmatrix} B_1 & C_1 & & & & & \\ A_2 & B_2 & C_2 & & & & \\ & A_3 & B_3 & C_3 & & & \\ & & & \dots & & & \\ & & & A_i & B_i & C_i & \\ & & & & & \dots & \\ & & & & & A_N & B_N \end{pmatrix} \begin{pmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \dots \\ \Phi_i \\ \dots \\ \Phi_N \end{pmatrix} = \begin{pmatrix} S_1 \\ S_2 \\ S_3 \\ \dots \\ S_i \\ \dots \\ S_N \end{pmatrix}$$

where $B_1 = -1$, $C_1 = 1$, $D_1 = 2\pi G h^2 \rho_1$, $S_N = 8\pi G h^2 r_N \rho_N - (2r_N + h)\Phi_r$, $A_i = 2r_i - h$, $B_i = -4r_i$, $C_i = 2r_i + h$, $S_i = 8\pi G h^2 r_i \rho_i$. The matrix elements are all zero except along the tree diagonal lines. The solution to the system of the difference equations can be derived using the *tridiagonal solution* algorithm. It is in fact a variant of Gauss elimination and consists of four different steps

1. Initialization of two new variables

$$B_1' = B_1 \quad and \quad D_1' = D_1$$

2. Calculation of the following equations in increasing order of i until $i = N$ is reached

$$R = \frac{A_i}{B'_{i-1}}$$

$$B'_i = B_i - RC_{i-1}$$

$$D'_i = D_i - RD'_{i-1} \quad for \quad i = 2, 3, ..., N-1$$

3. Calculation of the solution for the last unknown by

$$\Phi_N = \frac{D'_N}{B'_N}$$

4. Calculation of the following equations in decreasing order of i

$$\Phi_i = \frac{D'_i - C_i \Phi_{i+1}}{B'_i} \quad for \quad i = N-1, N-2, ..., 1$$

In a computer program, the primed variables B'_i and D'_i need not be distinguished from B_i and D_i respectively because B'_i and D'_i are stored in the same memory spaces as for B_i and D_i. Therefore step (1) is not necessary in real programming.

Chapter 3

Numerical results

In the previous chapter the different numerical methods were presented analytically. Nevertheless just a theoretical presentation of these methods is not sufficient unless it is combined with the release of the results that arise by their application. In the present chapter these methods are tested thoroughly using a series of suitable problems, widely known as shock tube problems. These problems are designed to test the robustness and the efficiency of the HRSC numerical methods and in many cases demonstrate their weaknesses. It is possible that several methods fail to produce satisfactory results when applied to a particular problem and therefore it is important to be aware of their limitations. These test problems, widely known as shock tube problems, are quite simple, but their solution is demanding. As a matter of fact a crowd of different physical phenomena is detected in their solution, such as shock waves, contact discontinuities and rarefraction waves. By checking the results that are presented in this chapter the reader can have a first view concerning the potentials of each one of the methods.
The shock tube problems can be distinguished in several categories, each one of which aims to test different aspects of the numerical methods. There is a first set of 8 different one-dimensional shock tube tests. For these test problems the exact solution is already known and it is always demonstrated along with the numerical results. In the case of the 1D problems the numerical methods are compared with each other in a specific manner. In particular the results of the PPM reconstruction algorithm along with the Riemann solvers of HLL and HLLC are presente d in the same plot, the MUSCL reconstruction algorithm with HLL and HLLC in a different plot, the ROE-type schemes (ROE, UNO, Marquina) are presented in one plot and finally the central scheme, which uses the WENO reconstruction algorithm, is compared with the WENO-HLLC scheme. A two dimensional shock tube problem is used in order to check the efficiency of the source terms algorithms. In this case two different numerical schemes are compared, namely PPM-HLLC and UNO. For the implementation of the source terms the splitting method was used. The next test case is the vacuum boundary shock tube problem, where the efficiency of PPM-HLLC and MUSCL-HLLC in regions adjacent to vacuum is checked. The last test concerns two MHD

shock tube problems, which are resolved using a suitably modified HLLC Riemann soler, along with the PPM reconstruction algorithm.

3.1 1D shock tube problems

As already mentioned 9 different one-dimensional shock tube problems will be solved in this section. The solution to these problems will provide a first view to the abilities of each one of the numerical methods.
These shock tube problems are initial value problems for the time-dependent Euler equations for ideal gases

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0 \tag{3.1}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} \quad , \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(E+p) \end{bmatrix}$$

The equation of state is

$$p = (\Gamma - 1)\rho\epsilon \tag{3.2}$$

In each one of the following problems the polytropic index is set as $\Gamma = 1.4$. The spatial domain is $x = [0, 1]$ and is discretised in 100 computational grid points. Each problem consists of a left and a right initial state, which are separated by a discontinuity at a specific position x_0. The CFL number may vary in several cases and its value will be made known in every case.

3.1.1 First shock tube problem

The first shock tube problem is also known as modified Sod's problem. The left-state initial values are $\mathbf{U}_l = \big[\rho, \rho u, p\big]_l = \big[1, 0.75, 1\big]$ and the right-state ones are $\mathbf{U}_r = \big[\rho, \rho u, p\big]_r = \big[0.125, 0, 0.1\big]$. It is a rather simple problem but the physical phenomena that emerge are quite interesting. In fact its solution consists of a shock wave that is moving to the right side, a rarefraction wave moving to the left and a contact discontinuity that separates these two regions.

PPM reconstruction and HLL, HLLC Riemann solvers

The PPM reconstruction algorithm seems to produce identical results when used either along with the HLL or the HLLC Riemann solver. The free parameters in the PPM algorithm are the ones presented in the previous chapter. A different selection of these parameters could possibly provide better results, especially in the steepening of the contact discontinuity.

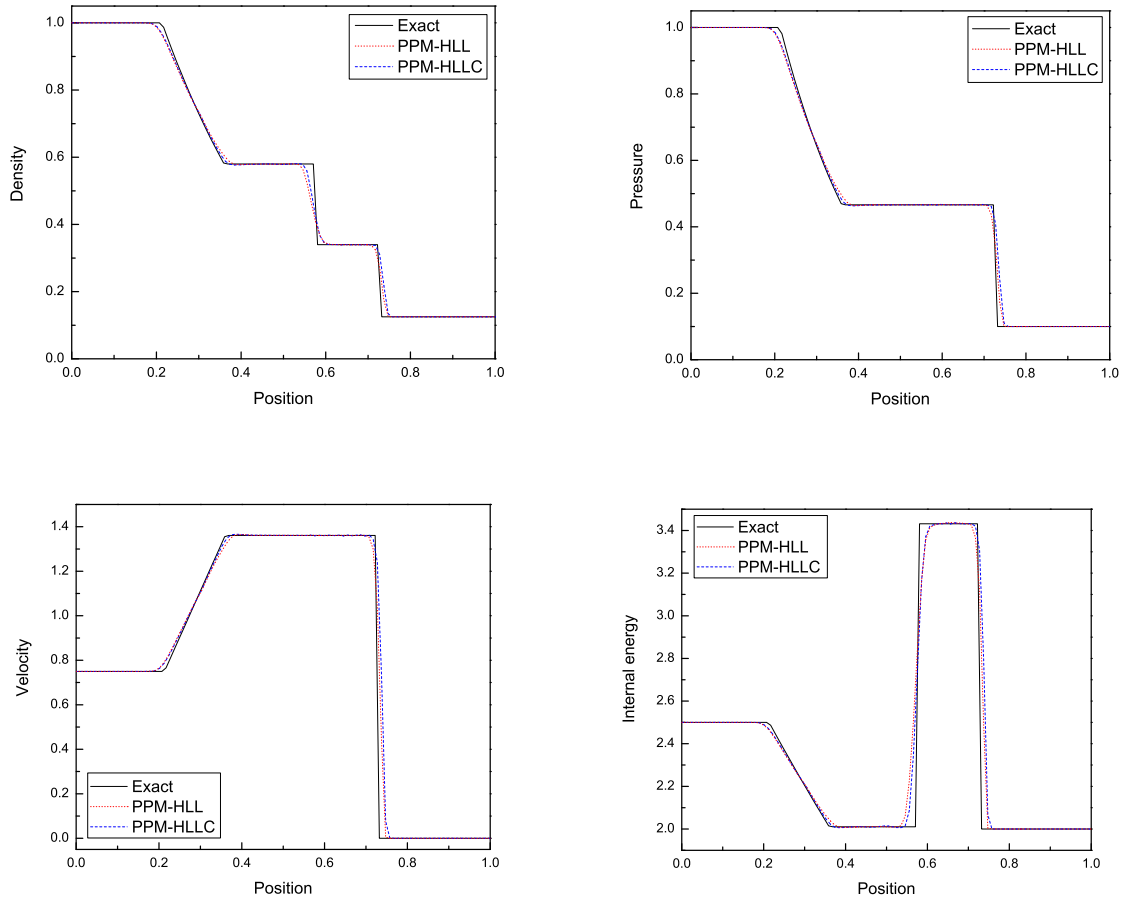


Figure 3.1: PPM reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.3$ and the different schemes are compared at time $t = 0.2$ units. Sod's modified problem as already mentioned offers a perspective of the different wave patterns that can arise in a physical problem and the fact that PPM proves itself efficient should be taken into account.

MUSCL reconstruction and HLL, HLLC Riemann solvers

Both MUSCL along with HLL and MUSCL along with HLLC seem to behave pretty much the same way. MUSCL and HLLC seems to be a little less diffusive, but generally both schemes are similar. Compared to ones by the PPM there seem to be several differences, which are mainly identified at the internal energy plot, where MUSCL produces oscillations.

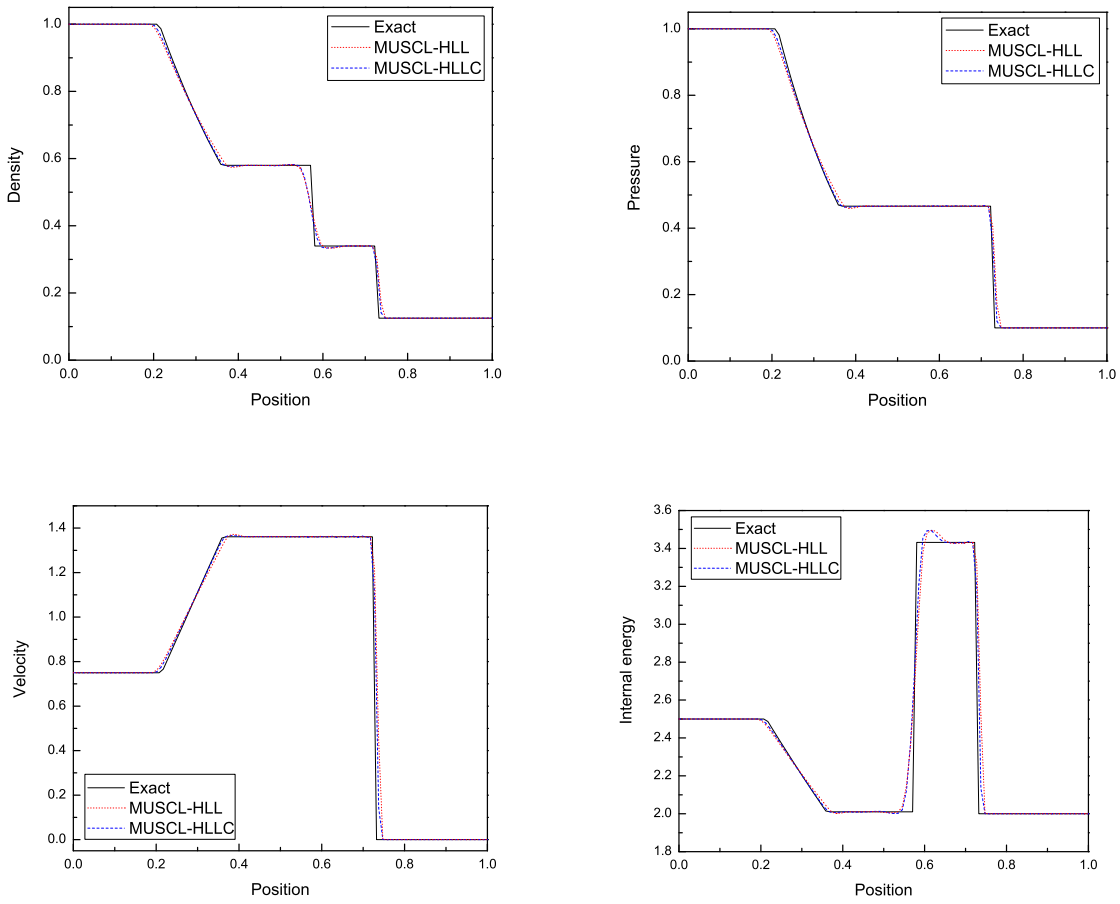


Figure 3.2: MUSCL reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.3$ and the different schemes are compared at time $t = 0.2$ units.

ROE-type schemes (ROE, UNO, Marquina)

Both UNO and Marquina are in fact based on ROE's Riemann solver and are supposed to provide better results. Indeed the results prove that these improved schemes are more efficient. This is obvious at the density and the internal energy plots. It should also be noted that UNO seems to produce small oscillations in the region of the contact discontinuity, while Marquina fails to simulate the rarefraction wave as efficiently as the other two schemes.

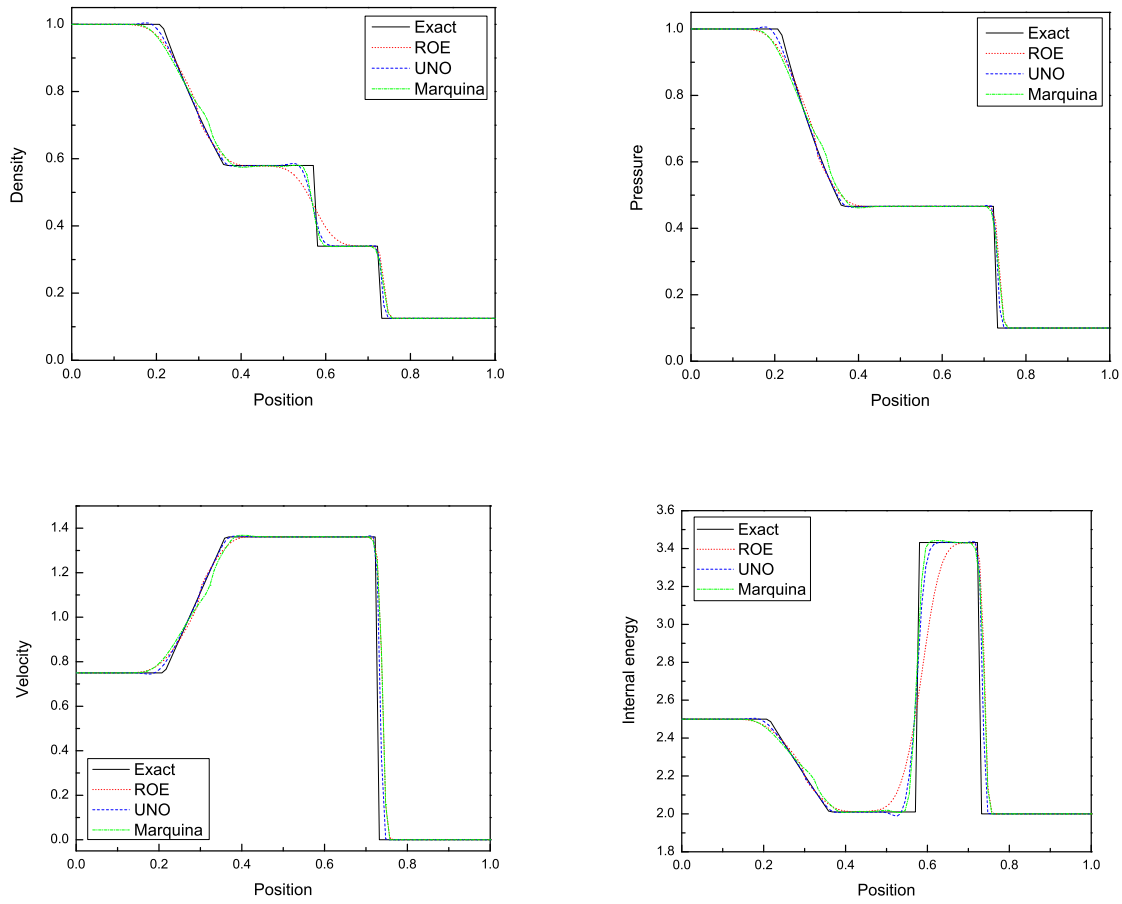


Figure 3.3: ROE, UNO and Marquina Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.3$ and the different schemes are compared at time $t = 0.2$ units.

Central scheme and WENO reconstruction along with HLLC

As already mentioned the reconstruction algorithm used in the central scheme is the WENO one. The differences between the two numerical schemes compared in the following plots are most visible in the internal energy plot. The WENO-HLLC scheme produces oscillations which are not present in the solution provided by the central scheme.

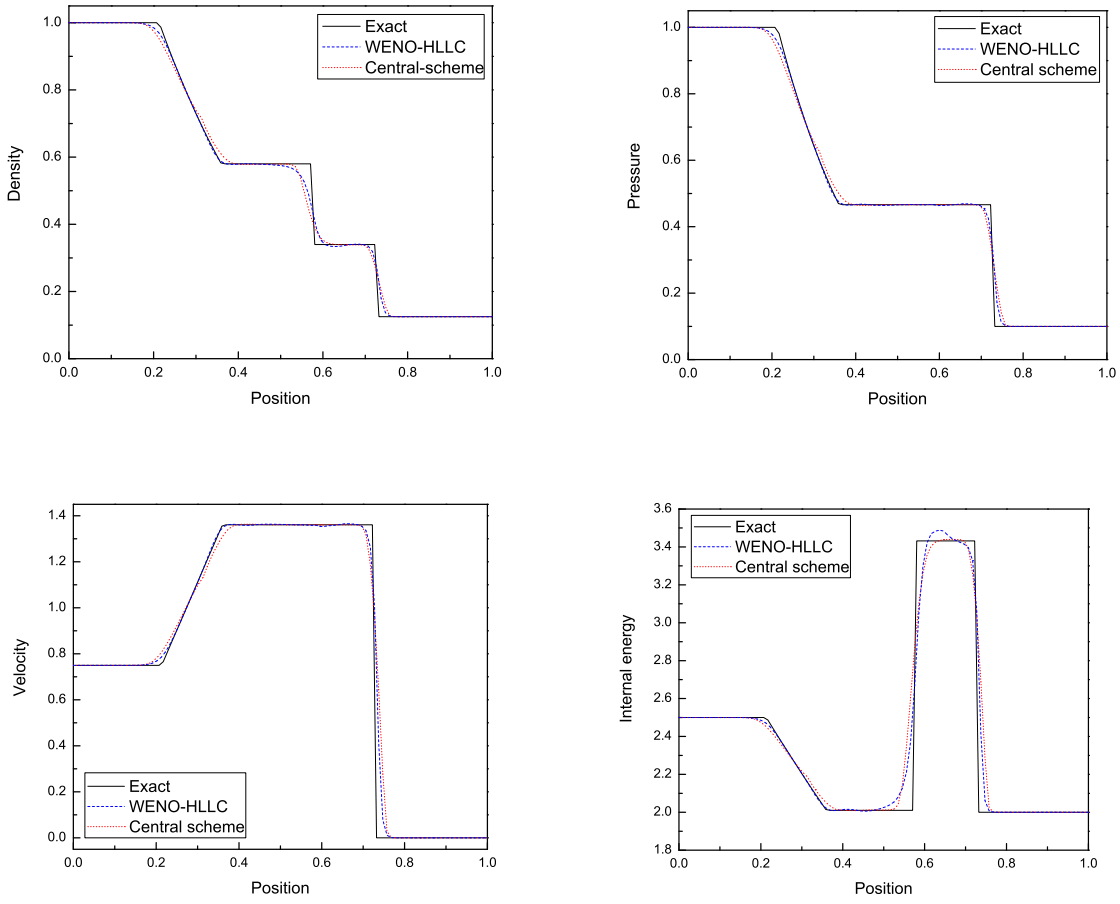


Figure 3.4: Central scheme by Kurganov-Tadmor and WENO reconstruction algorithm along with the HLLC Riemann solver.

The CFL number used was set as 0.9 for the WENO-HLLC scheme and as 0.1 for the central scheme, the initial discontinuity's position was at $x_0 = 0.3$ and the different schemes are compared at time $t = 0.2$ units.

3.1.2 Second shock tube problem

The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[1, -2, 0.4\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[1, 2, 0.4\right]$. This is the so called 123 problem and its exact solution consists of two rarefraction waves. The most interesting aspect of the solution is that a close to vacuum region is generated in the middle of the two waves. The simulation of a low-density state is a demanding procedure for most of the numerical schemes and this particular problem can be used in order to assess the potentials of each one of them.

PPM reconstruction and HLL, HLLC Riemann solvers

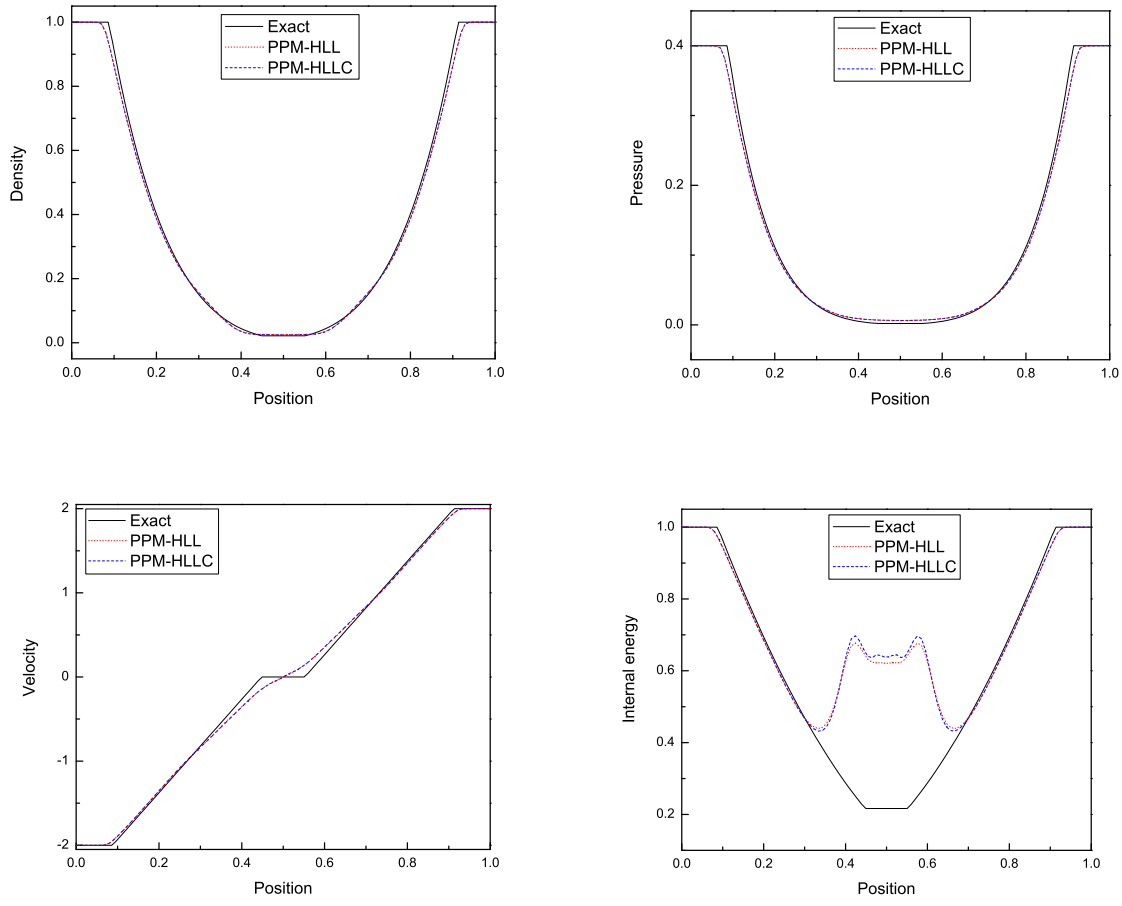


Figure 3.5: PPM reconstruction algorithm and HLL, HLLC Riemann solvers.

It is CFL=0.9, $x_0 = 0.5$ and the different schemes are compared at time $t = 0.15$ units.

MUSCL reconstruction and HLL, HLLC Riemann solvers

MUSCL along with HLL and HLLC seems to provide quite satisfying results. Both the density and pressure plots are adequately represented. Nevertheless the velocity profile is not quite accurate. The divergence observed in the internal energy profile is due to numerical errors. In low-density regions, both density and pressure have small values, so their fraction gives rise to large numbers.

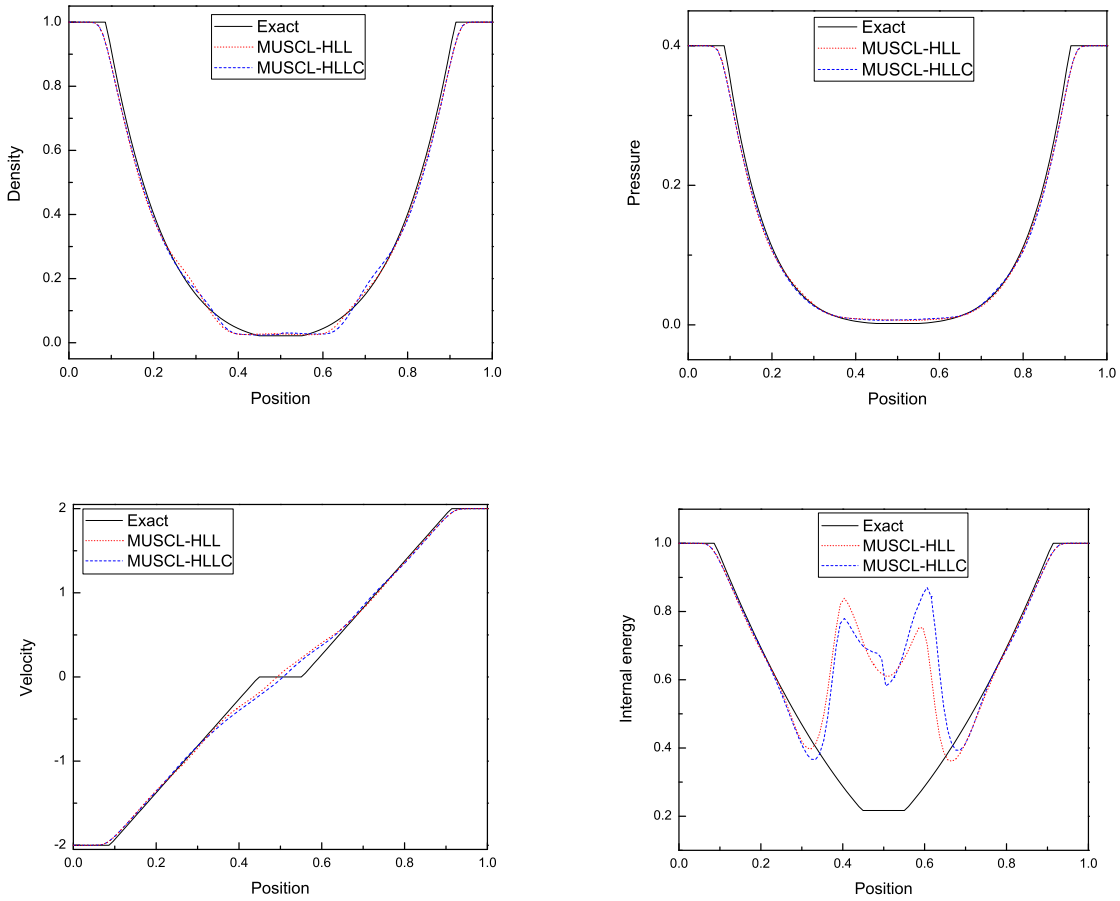


Figure 3.6: MUSCL reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.15$ units.

ROE-type schemes (ROE, UNO, Marquina)

ROE and UNO when applied to the 123 test problem fail to produce results. As already mentioned ROE-type schemes crash in regions of low-density. Nevertheless Marquina manages to provide satisfactory results.

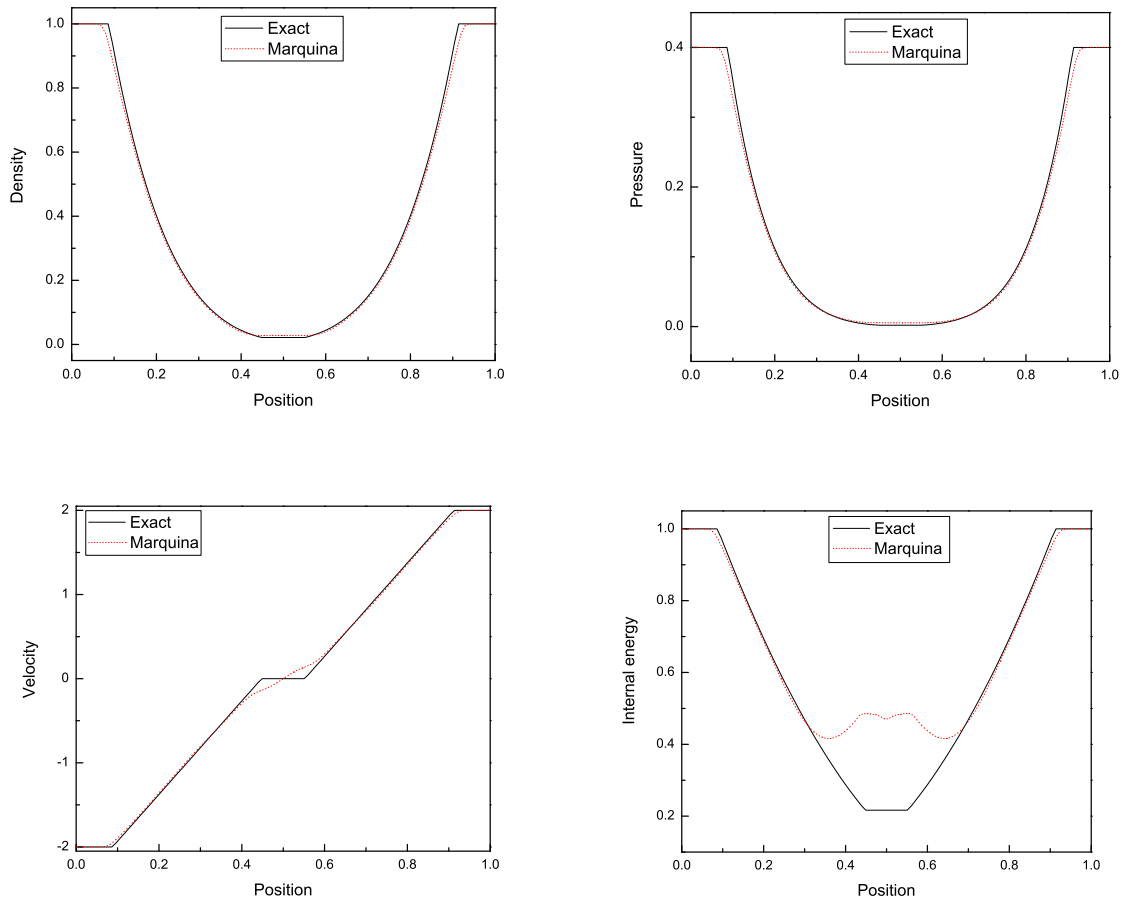


Figure 3.7: Marquina Riemann solver. ROE and UNO fail at this shock tube test.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.15$ units.

Central scheme and WENO reconstruction along with HLLC

Unlike ROE and UNO, both the central and the WENO-HLLC scheme, when applied to the 123 problem, provide results, which are presented in the following plots.

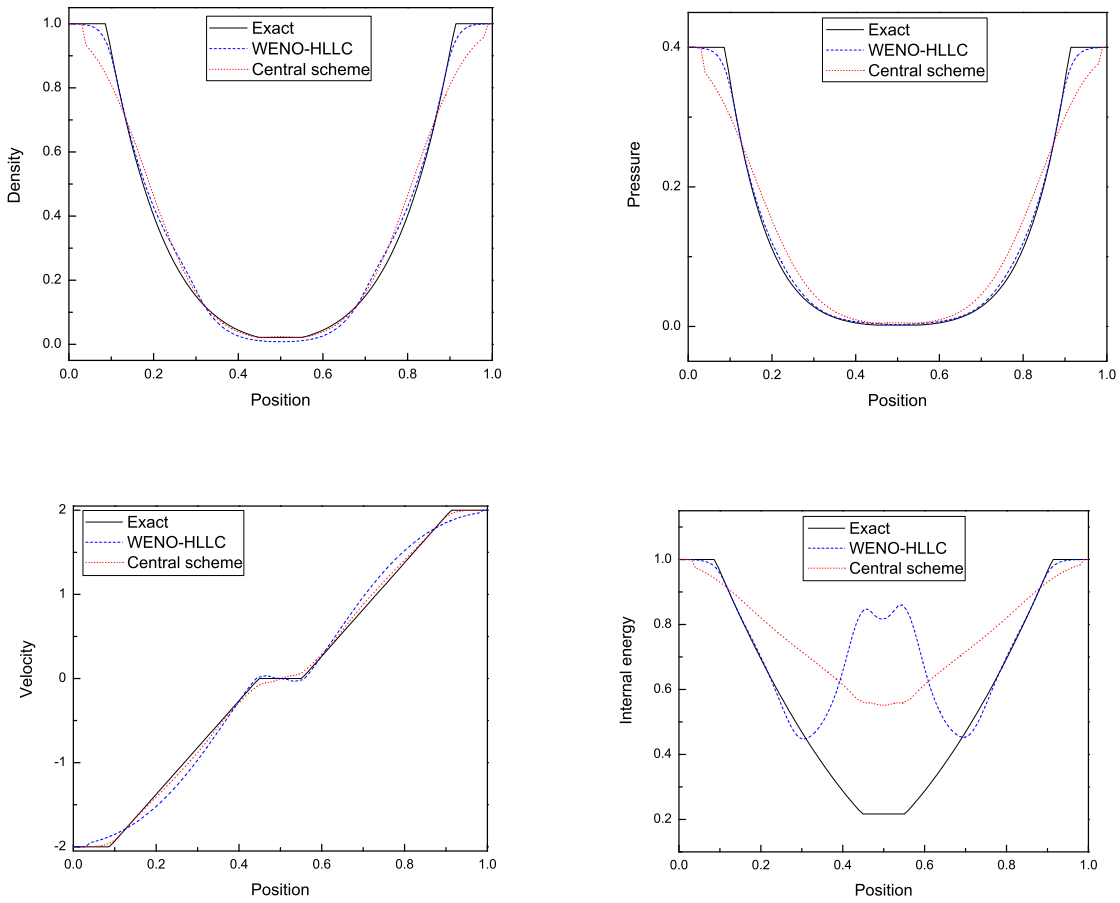


Figure 3.8: Central scheme by Kurganov-Tadmor and WENO reconstruction algorithm along with the HLLC Riemann solver.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.15$ units.

3.1.3 Third shock tube problem

The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[1, 0, 1000\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[1, 0, 0.01\right]$. The solution to this problem consists of a strong shock wave, a contact discontinuity and a rarefraction wave.

PPM reconstruction and HLL, HLLC Riemann solvers

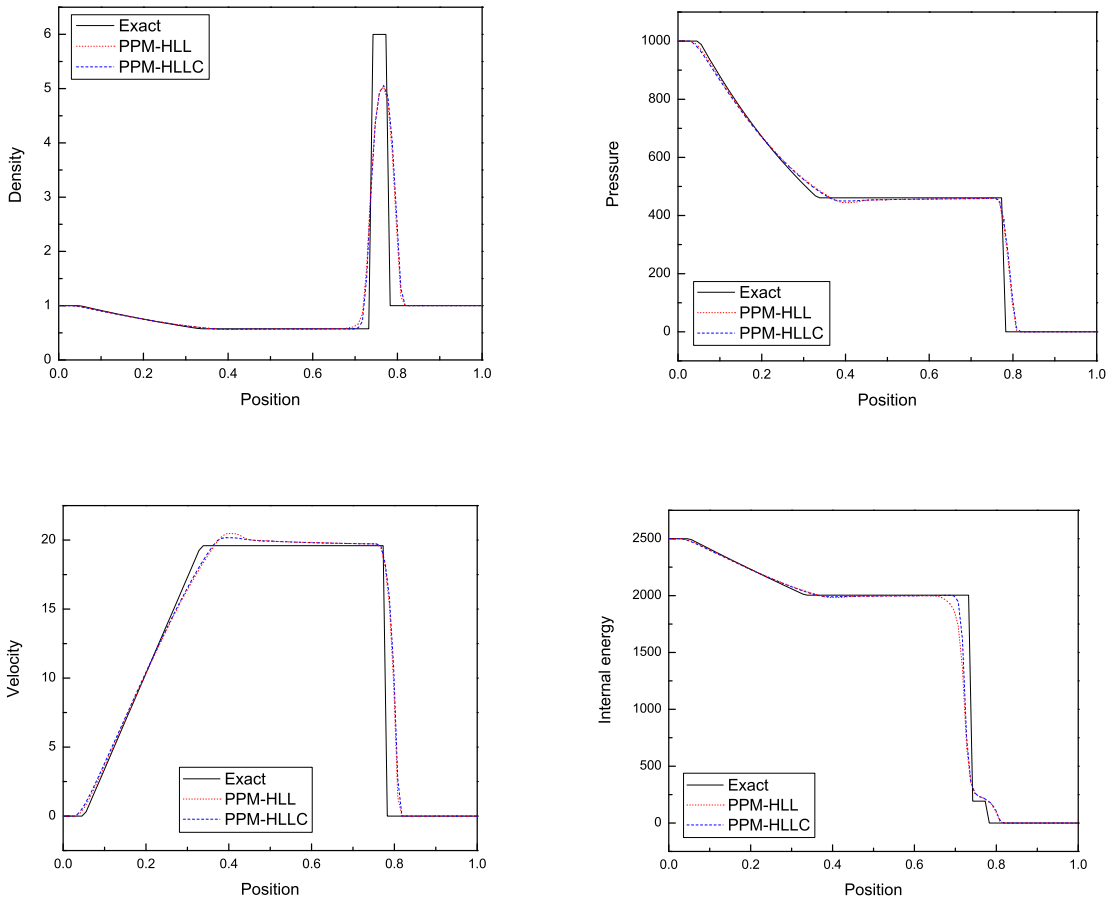


Figure 3.9: PPM reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.012$ units.

MUSCL reconstruction and HLL, HLLC Riemann solvers

MUSCL along with HLL and HLLC seems to provide quite satisfying results. Both schemes fail to attribute the maximum values of the density and moreover in the internal energy plot MUSCL-HLL seems to produce oscillations.

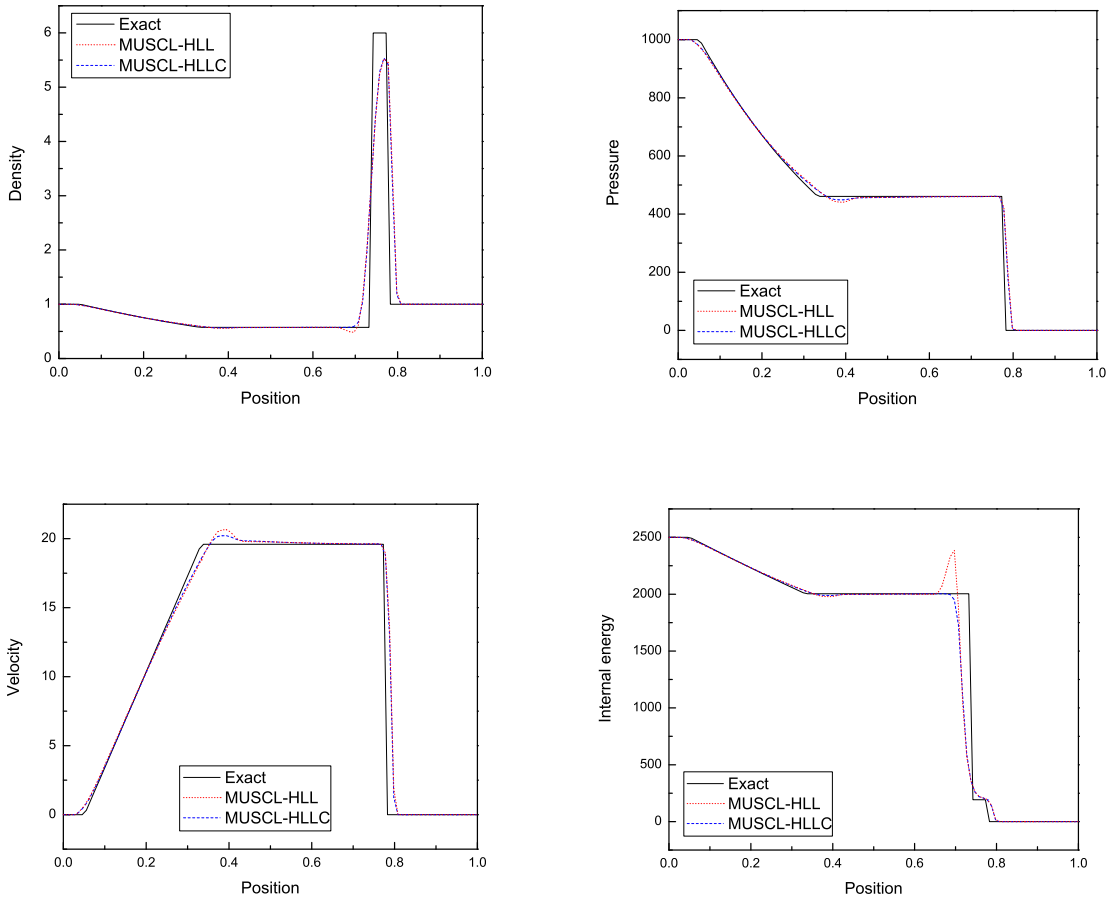


Figure 3.10: MUSCL reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.012$ units.

ROE-type schemes (ROE, UNO, Marquina)

In this case all of the numerical schemes prove themselves efficient and manage to solve the shock tube problem. UNO seems to resolve the density profile better than the rest of the numerical schemes, but in the internal energy plot several oscillations are present.

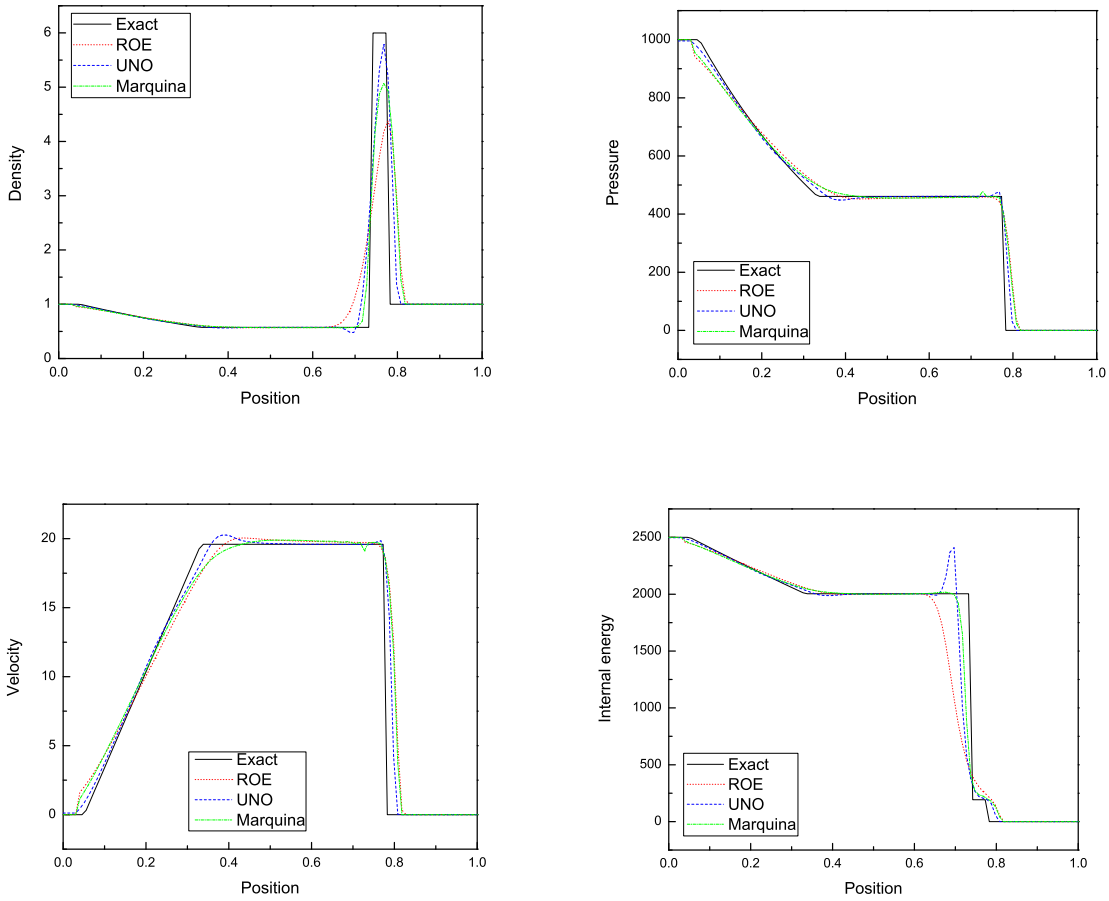


Figure 3.11: ROE, UNO and Marquina Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.012$ units.

Central scheme and WENO reconstruction along with HLLC

The WENO-HLLC scheme fails to produce results in this case. On the other hand the central scheme works, but appears quite diffusive when compared to the previous schemes.

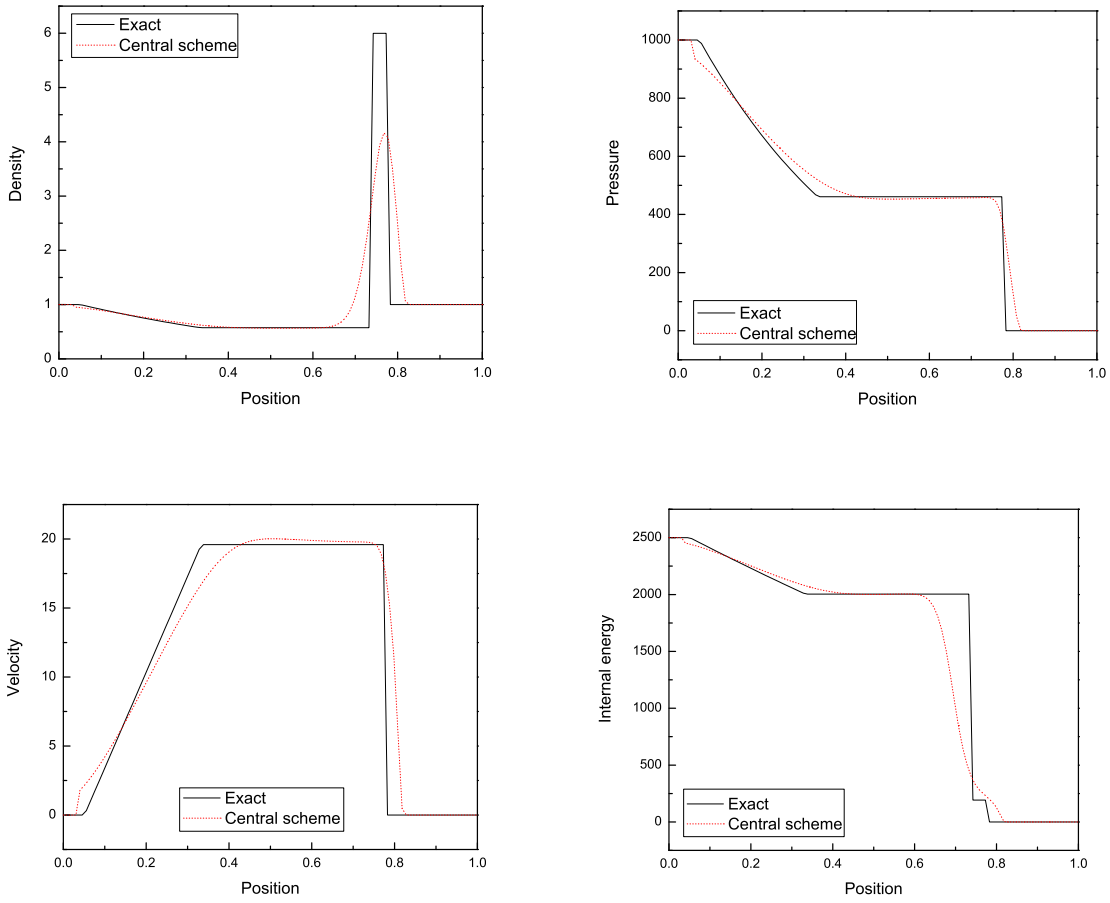


Figure 3.12: Central scheme by Kurganov-Tadmor. The WENO reconstruction algorithm along with the HLLC Riemann solver fails in this case.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.012$ units.

3.1.4 Fourth shock tube problem

The first shock tube problem is also known as Sod's problem. The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[5.99924, 19.5975, 460.894\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[5.99242, -6.19633, 46.0950\right]$. The initial date describe two very strong skock waves travelling towards each other and the solution consists of three different discontinuities travelling to the right.

PPM reconstruction and HLL, HLLC Riemann solvers

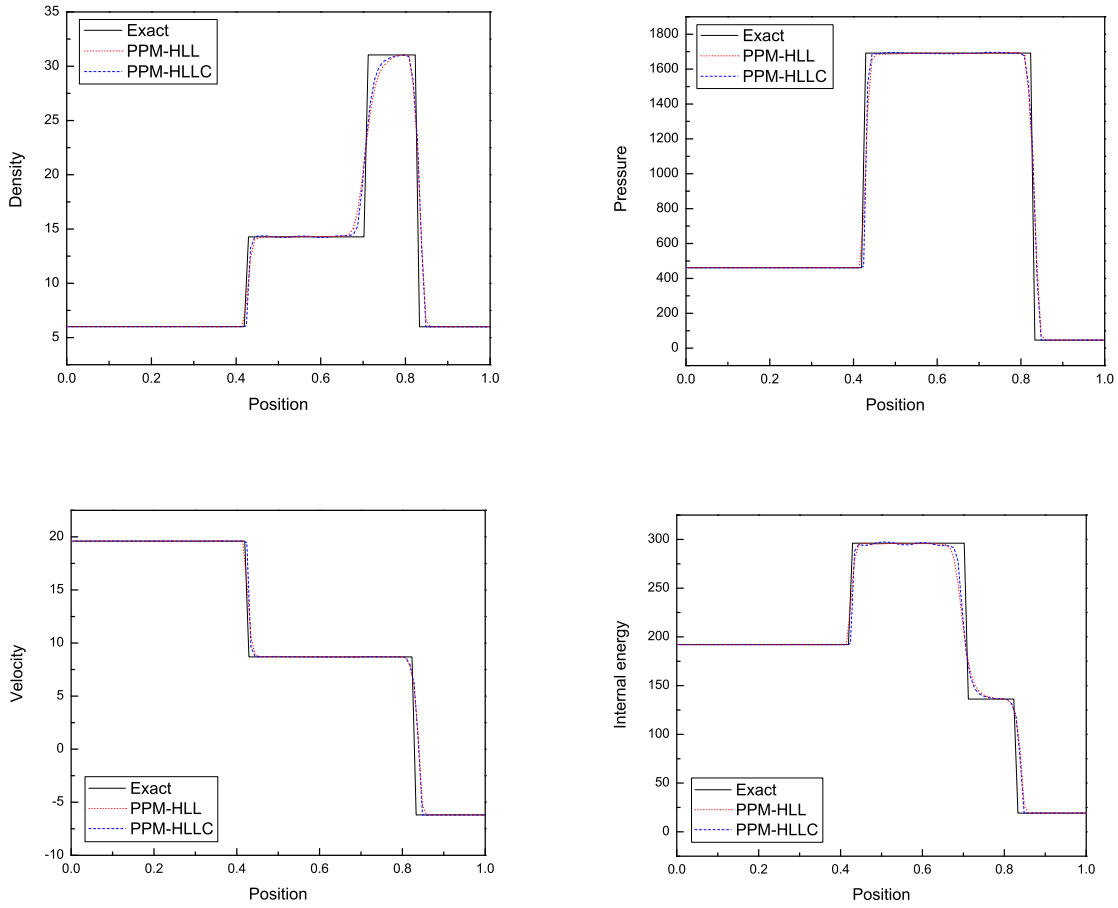


Figure 3.13: PPM reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.4$ and the different schemes are compared at time $t = 0.035$ units.

MUSCL reconstruction and HLL, HLLC Riemann solvers

The results by the MUSCL reconstruction algorithm along with both the HLL and HLLC Riemann solvers are satisfying. The discontinuities are sufficiently resolved. Several oscillations are present in the solution.

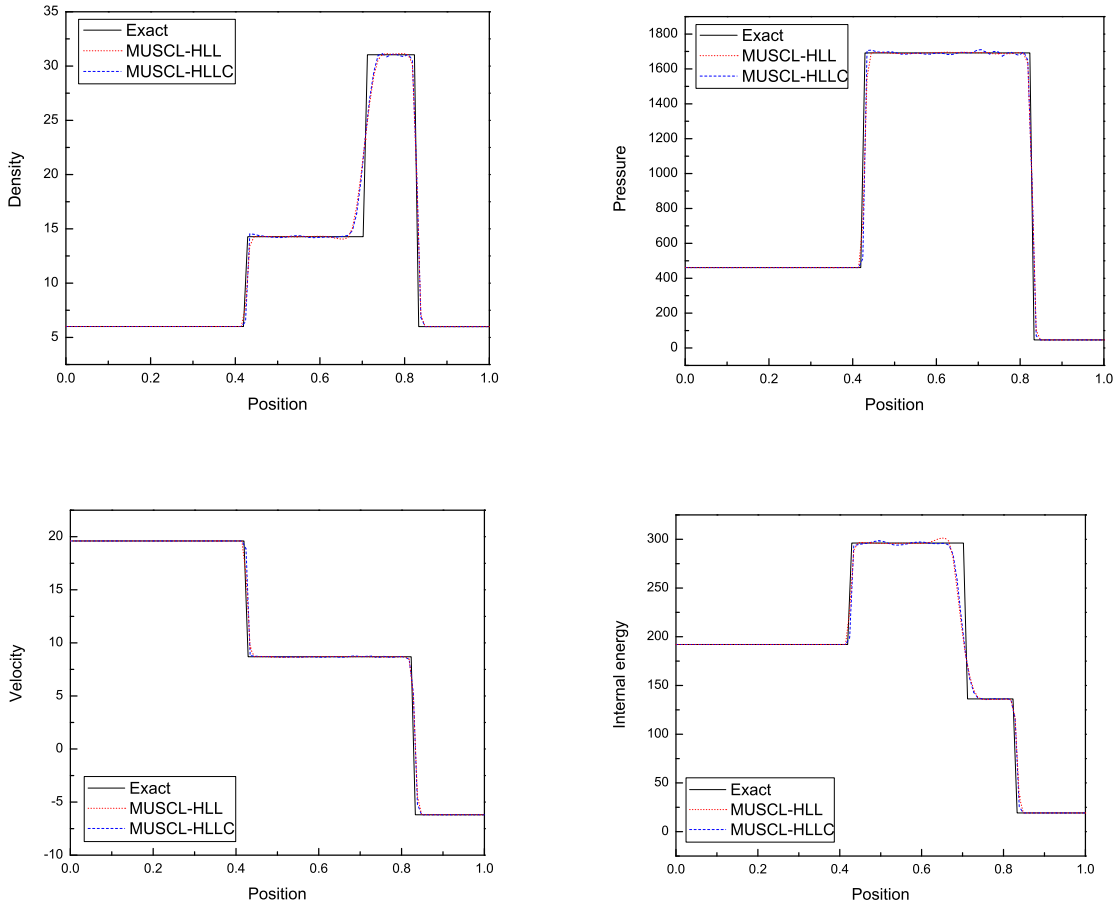


Figure 3.14: MUSCL reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.4$ and the different schemes are compared at time $t = 0.035$ units.

ROE-type schemes (ROE, UNO, Marquina)

The Riemann solvers of ROE, UNO and Marquina also manage to handle this particular shock tube problem. UNO and Marquina are less diffusive than ROE's solver.

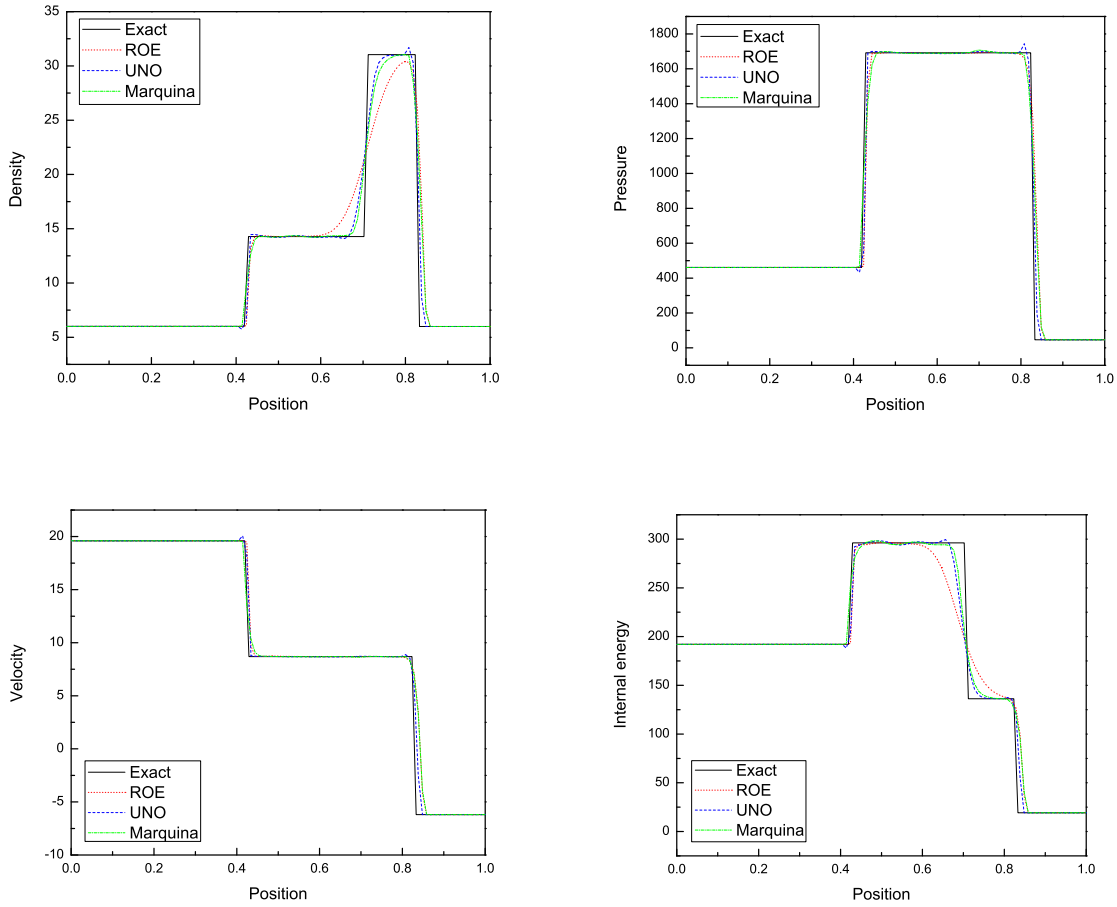


Figure 3.15: ROE, UNO and Marquina Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.4$ and the different schemes are compared at time $t = 0.035$ units.

Central scheme and WENO reconstruction along with HLLC

The central scheme, when compared to the WENO-HLLC scheme, seems to operate in a more proper manner. The central scheme is less diffusive and the discontinuities are better handled.

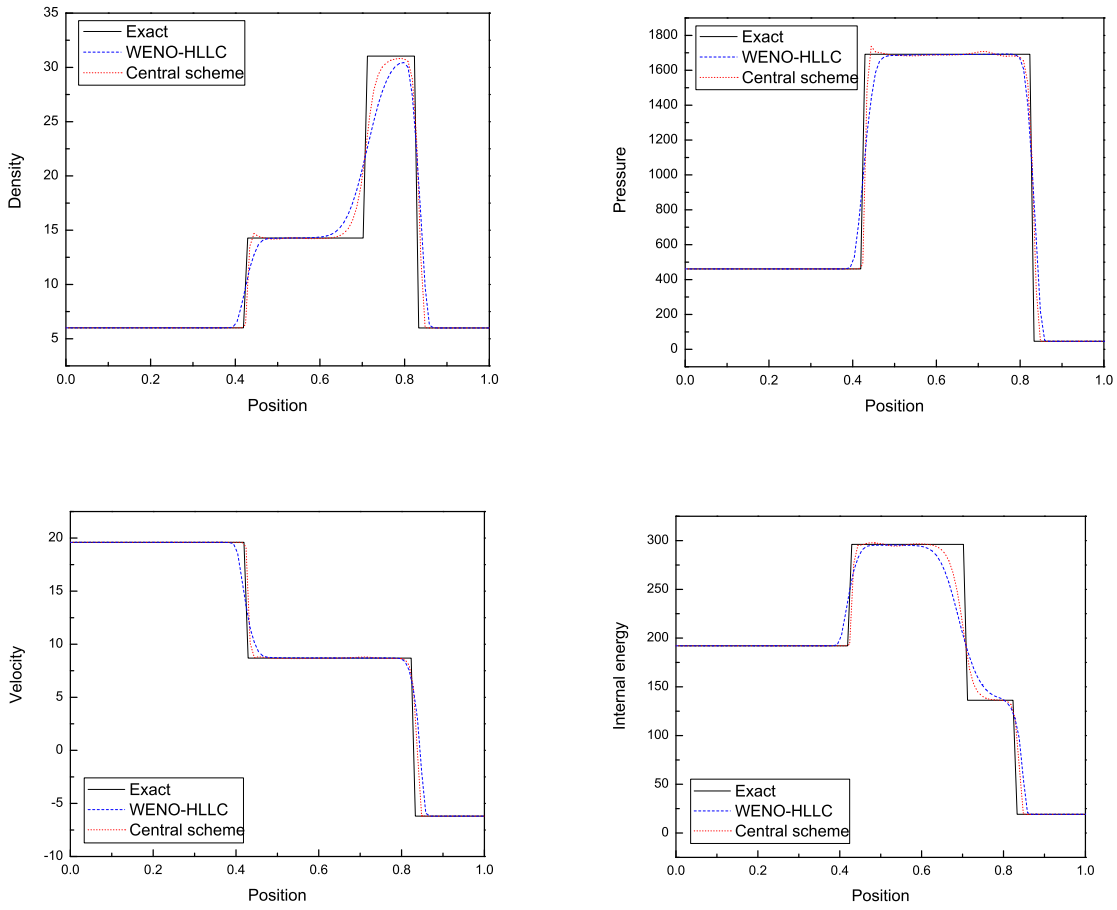


Figure 3.16: Central scheme by Kurganov-Tadmor and WENO reconstruction algorithm along with the HLLC Riemann solver.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.4$ and the different schemes are compared at time $t = 0.035$ units.

3.1.5 Fifth shock tube problem

The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[1, -19.59745, 1000\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[1, -19.59745, 0.01\right]$. The solution of this test consists of a right-travelling shock wave, a left rarefraction wave and a stationary contact discontinuity.

PPM reconstruction and HLL, HLLC Riemann solvers

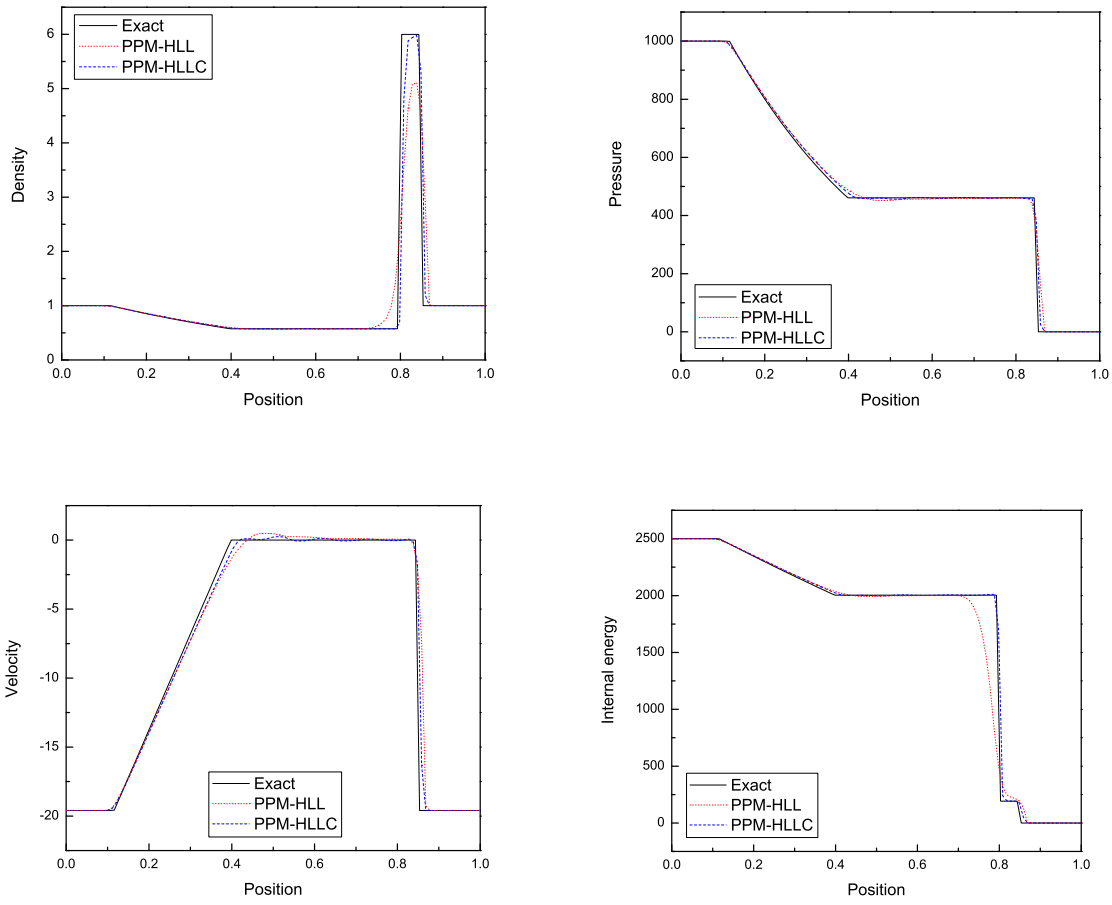


Figure 3.17: PPM reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.8$ and the different schemes are compared at time $t = 0.012$ units.

MUSCL reconstruction and HLL, HLLC Riemann solvers

The MUSCL reconstruction algorithm along with the HLL Riemann solver fails in this particular problem. On the other hand by substituting HLL with HLLC, the solution of this test is quite accurate. Nevertheless the solution is oscillatory and this is easily detected in the pressure, the velocity and the internal energy profiles.

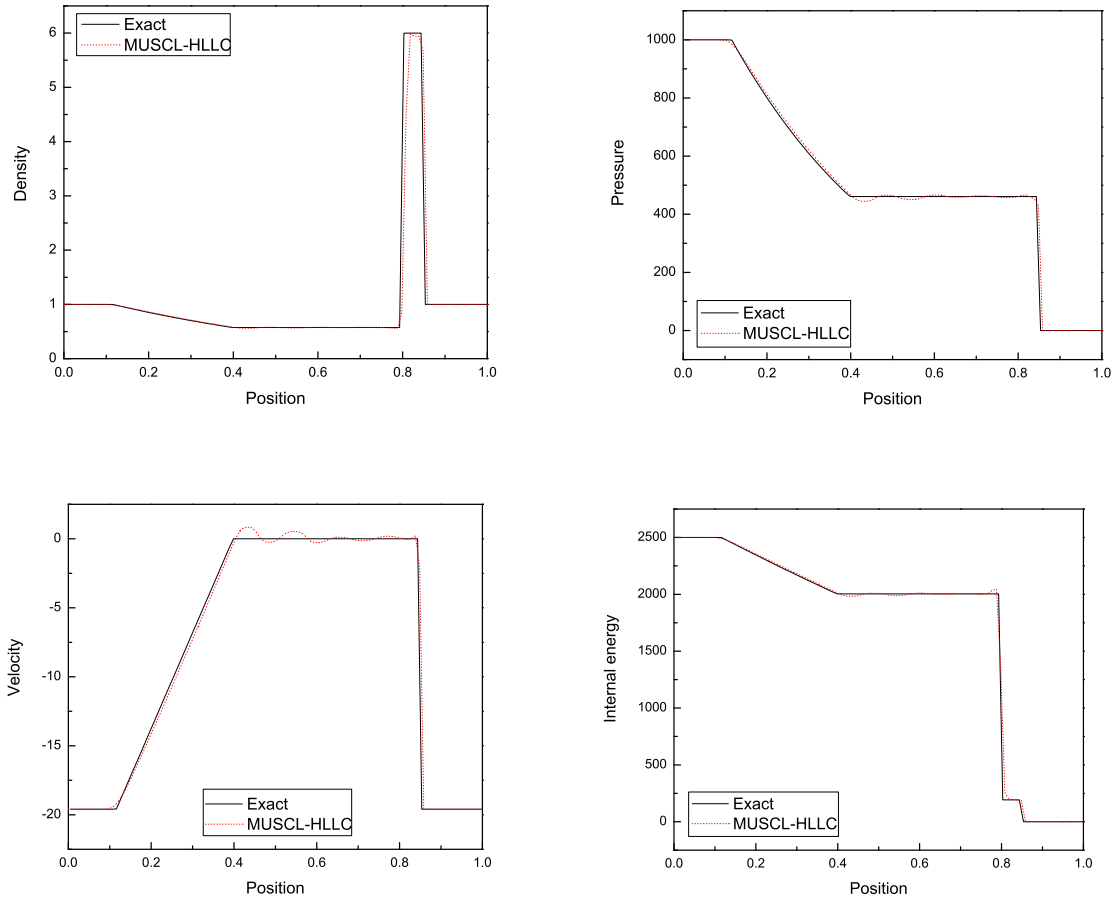


Figure 3.18: MUSCL reconstruction algorithm HLLC Riemann solver. MUSCL-HLL fails.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.8$ and the different schemes are compared at time $t = 0.012$ units.

ROE-type schemes (ROE, UNO, Marquina)

ROE ,UNO and Marquina provide results for this particular problem. The UNO Riemann solver is oscillatory and this is evident in the velocity and pressure plots.

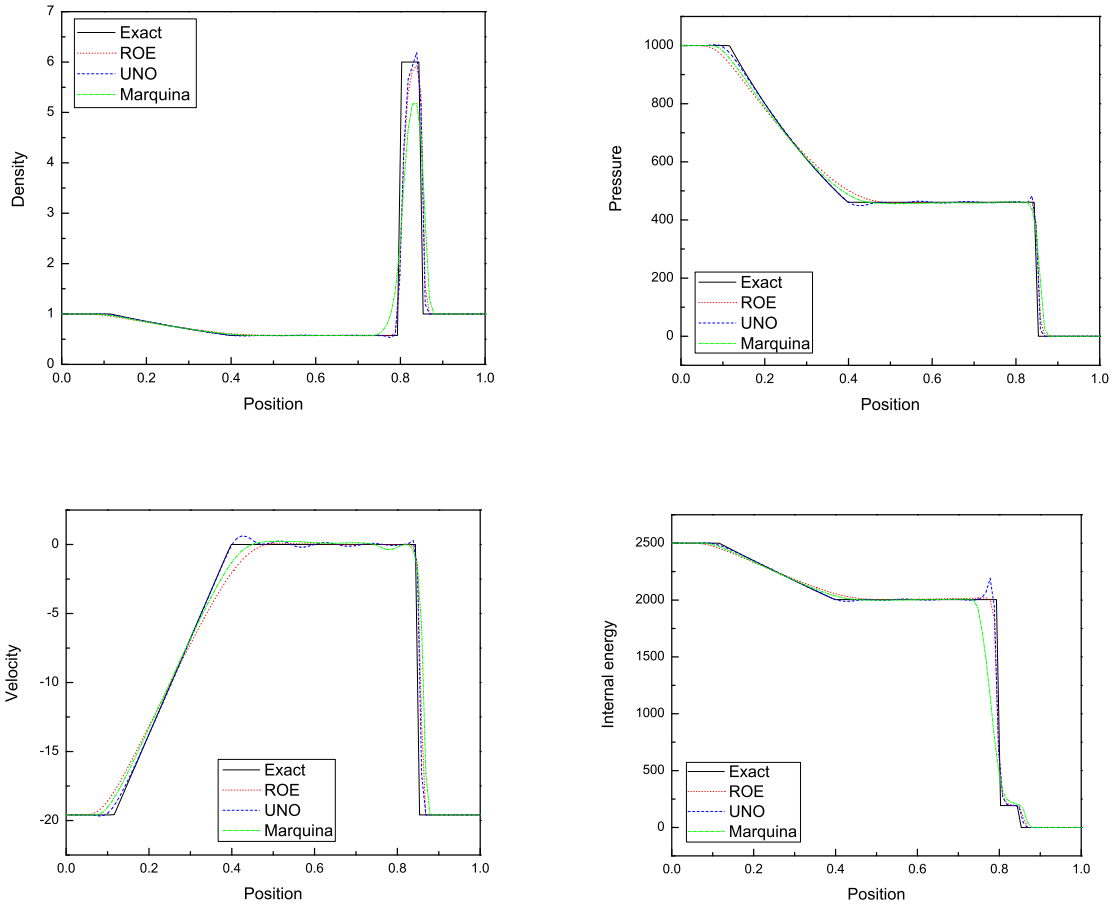


Figure 3.19: ROE, UNO and Marquina Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.8$ and the different schemes are compared at time $t = 0.012$ units.

Central scheme and WENO reconstruction along with HLLC

The WENO reconstruction along with HLLC Riemann solver fails. However the results by the central scheme are not quite satisfactory since the solution is more diffusive than the previous schemes.

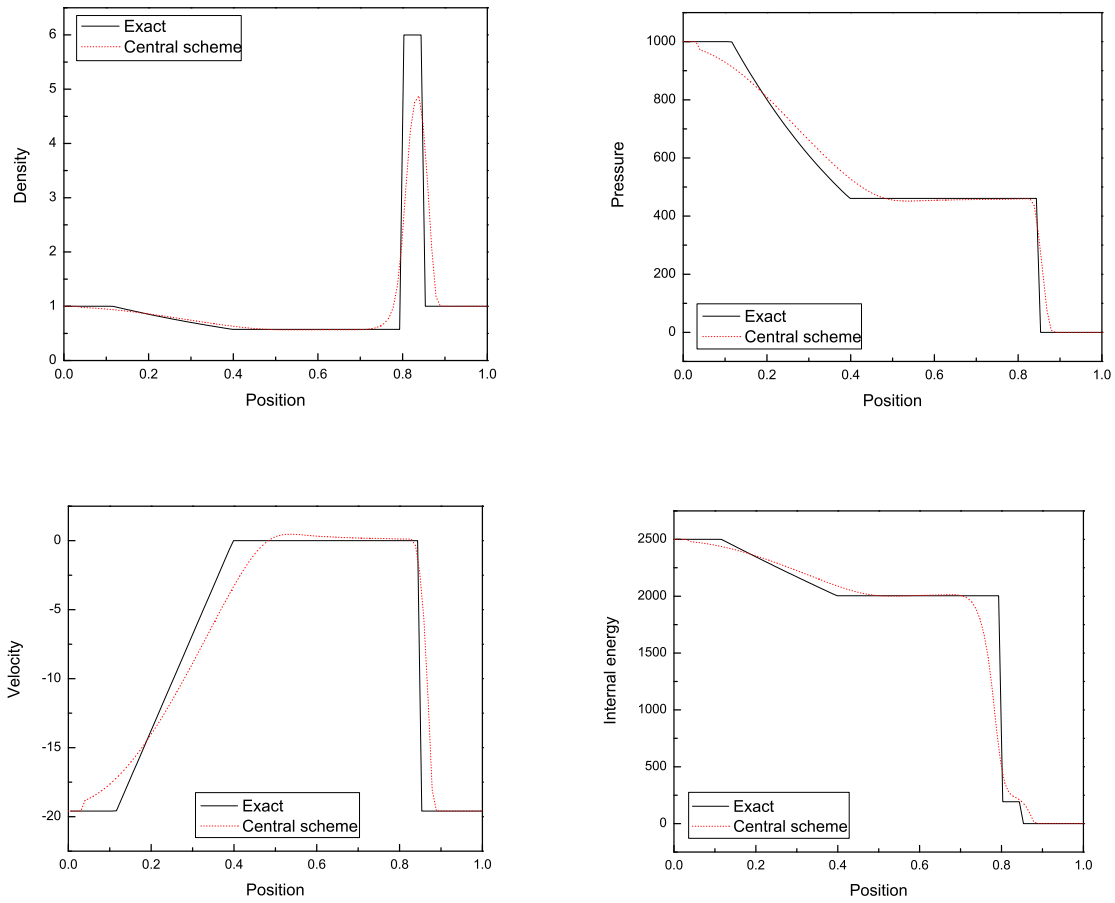


Figure 3.20: Central scheme by Kurganov-Tadmor. The WENO reconstruction algorithm along with the HLLC Riemann solver fails.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.8$ and the different schemes are compared at time $t = 0.012$ units.

3.1.6 Sixth shock tube problem

The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[1.4, 0, 1\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[1, 0, 1\right]$. This test corresponds to an isolated stationary contact wave. It is interesting to note that Marquina failed this particular test. ROE and UNO manage to simulate the contact wave precisely. This is also the case of PPM-HLLC and MUSCL-HLLC, while PPM-HLL and MUSCL-HLL are more diffusive. This test demonstrates the advantage of HLLC over HLL in similar problems. WENO-HLLC and the central scheme are also pretty diffusive.

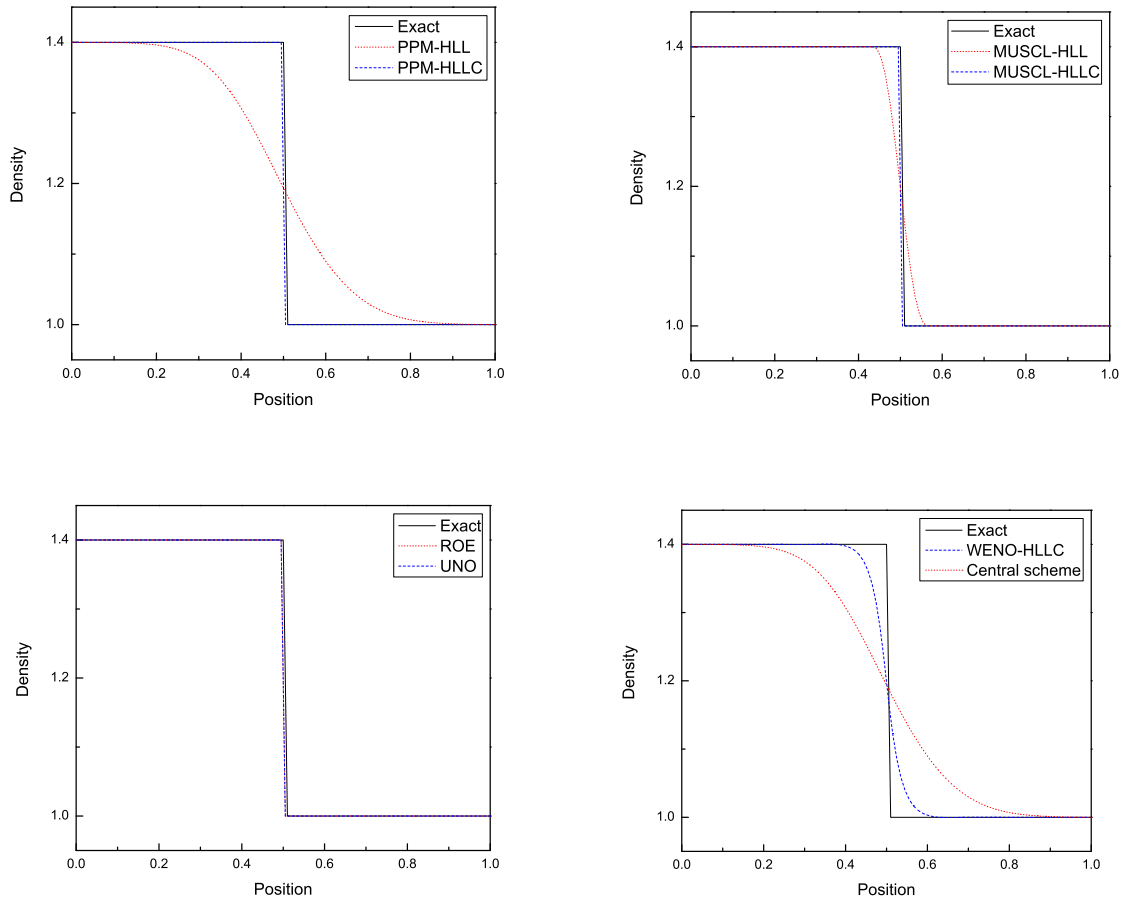


Figure 3.21: Numerical schemes applied to test 6.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 2$ units.

3.1.7 Seventh shock tube problem

The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[1.4, 0.1, 1\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[1, 0.1, 1\right]$. This test corresponds to an isolated contact moving slowly to the right. Marquina also failed this test. It seems that MUSCL in this particular test operates with more precision than PPM.

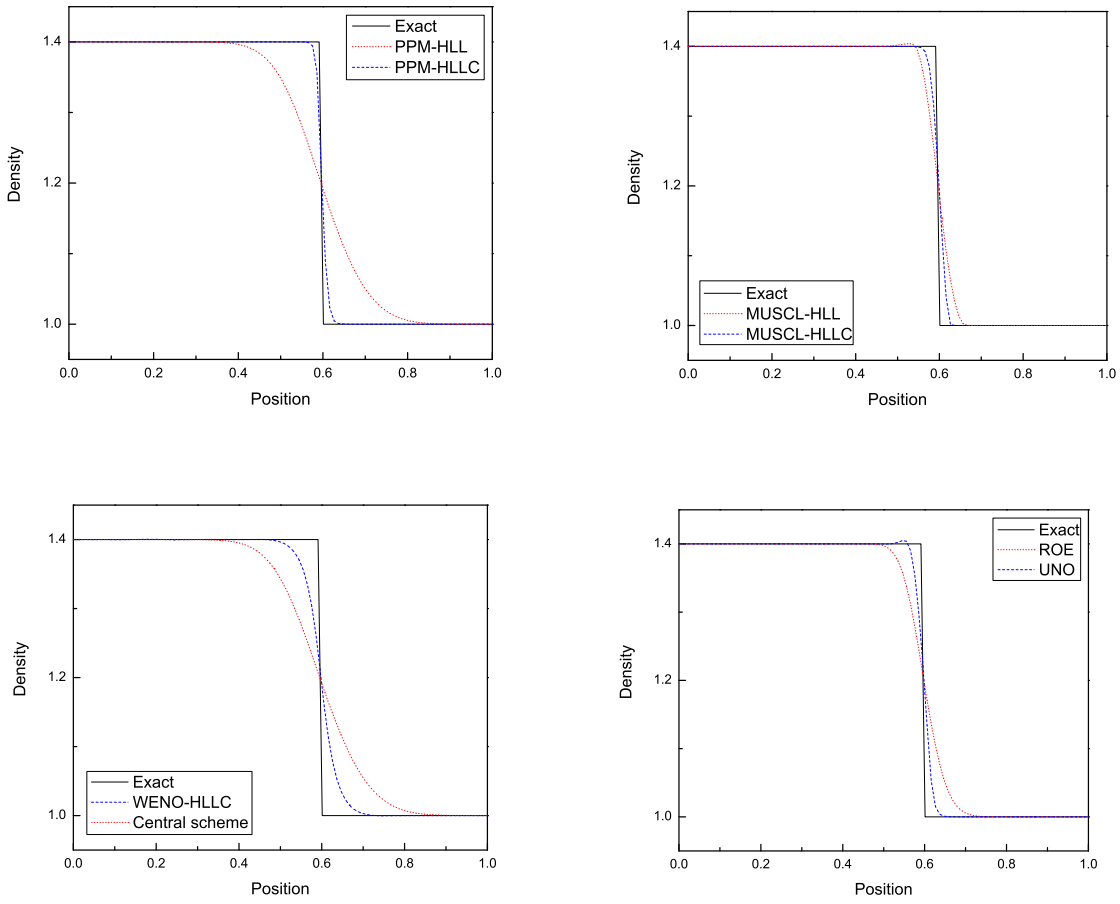


Figure 3.22: Numerical schemes applied to test 7.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 2$ units.

3.1.8 Eigth shock tube problem

The left-state initial values are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[0.445, 0.698, 3.528\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[0.5, 0, 0.571\right]$. This is Lax's problem and it is designed to check the ability of the numerical schemes to resolve moving contact discontinuities. The solution consists of a left rarefraction wave, a right-moving shock wave and a moving contact discontinuity.

PPM reconstruction and HLL, HLLC Riemann solvers

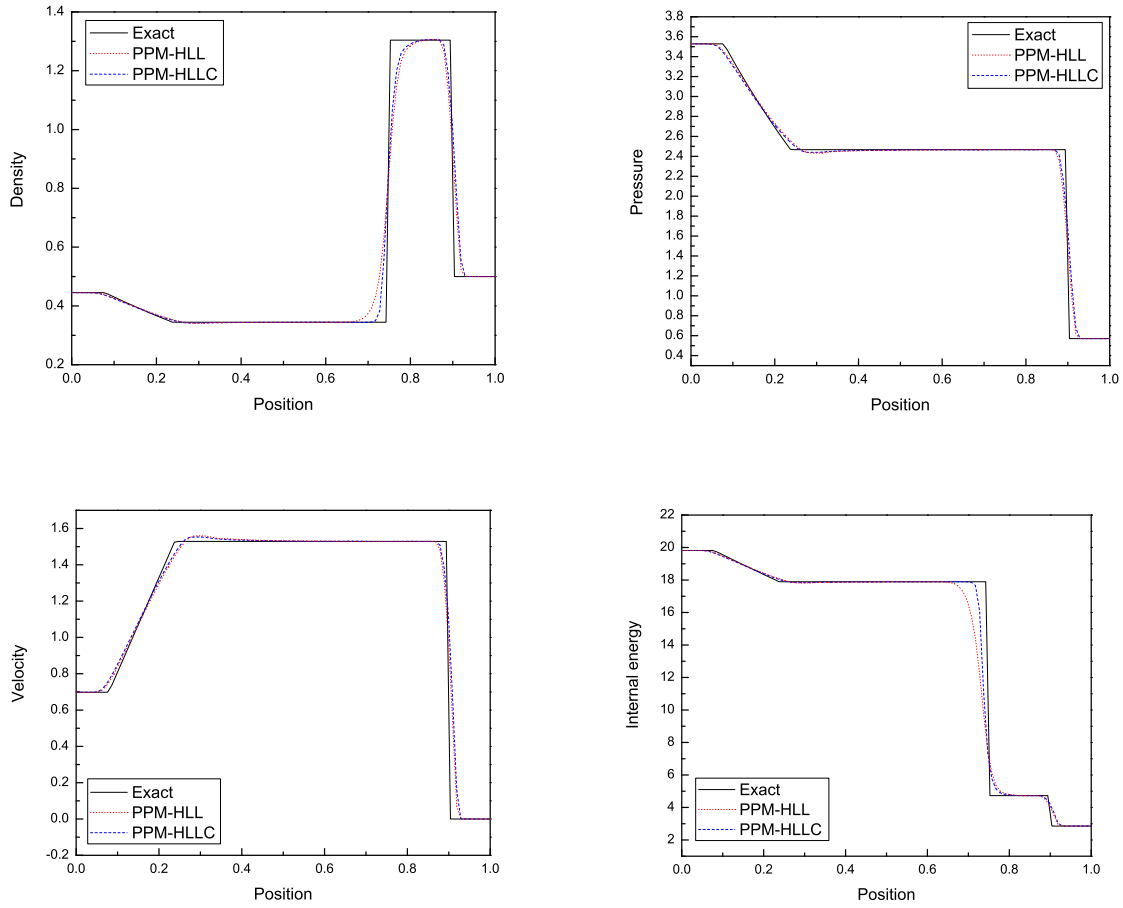


Figure 3.23: PPM reconstruction algorithm and HLL, HLLC Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.16$ units.

MUSCL reconstruction and HLL, HLLC Riemann solvers

Both MUSCL-HLL and MUSCL-HLLC manage to resolve the different wave patterns that are present in the solution of this particular test. However several oscillations are present in the solution The CFL number used was set as 0.9, the initial discontinuity's position

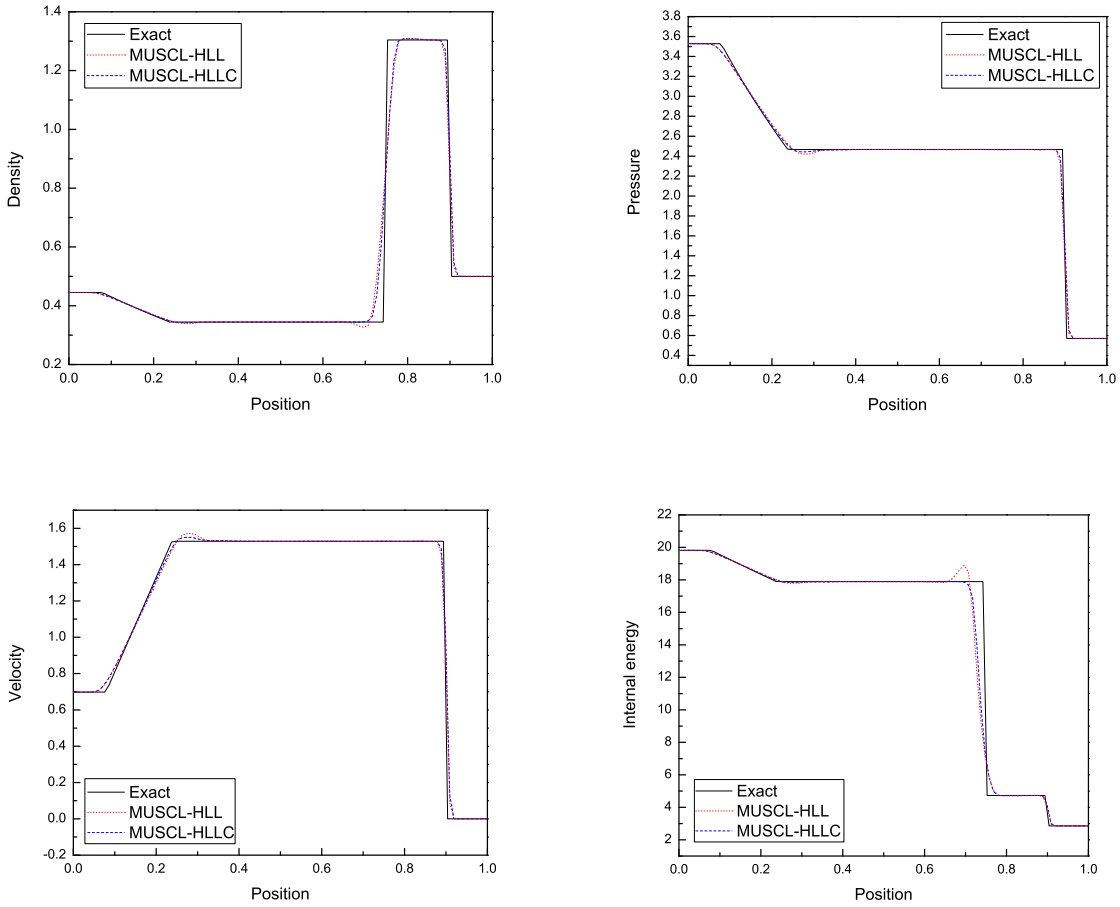


Figure 3.24: MUSCL reconstruction algorithm and HLL, HLLC Riemann solvers.

was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.16$ units.

ROE-type schemes (ROE, UNO, Marquina)

ROE, UNO and Marquina resolve the waves that appear in the solution. Several oscillations are evident in the UNO plots.

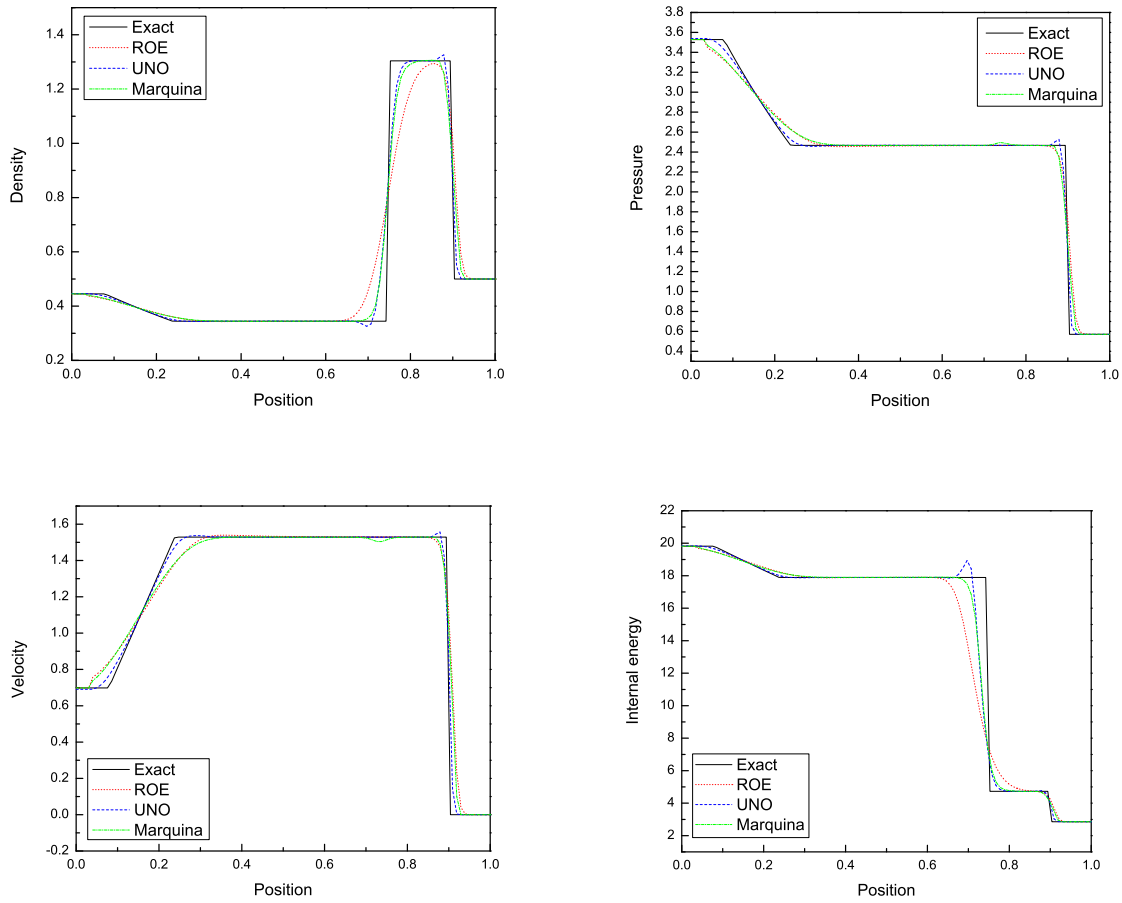


Figure 3.25: ROE, UNO and Marquina Riemann solvers.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.16$ units.

Central scheme and WENO reconstruction along with HLLC

The central scheme appears to be more diffusive than the WENO-HLLC one. This problem could be fixed if a smaller CFL number was initially chosen. As a matter of fact if the CFL number was less than 0.5, then the resolution would be improved.

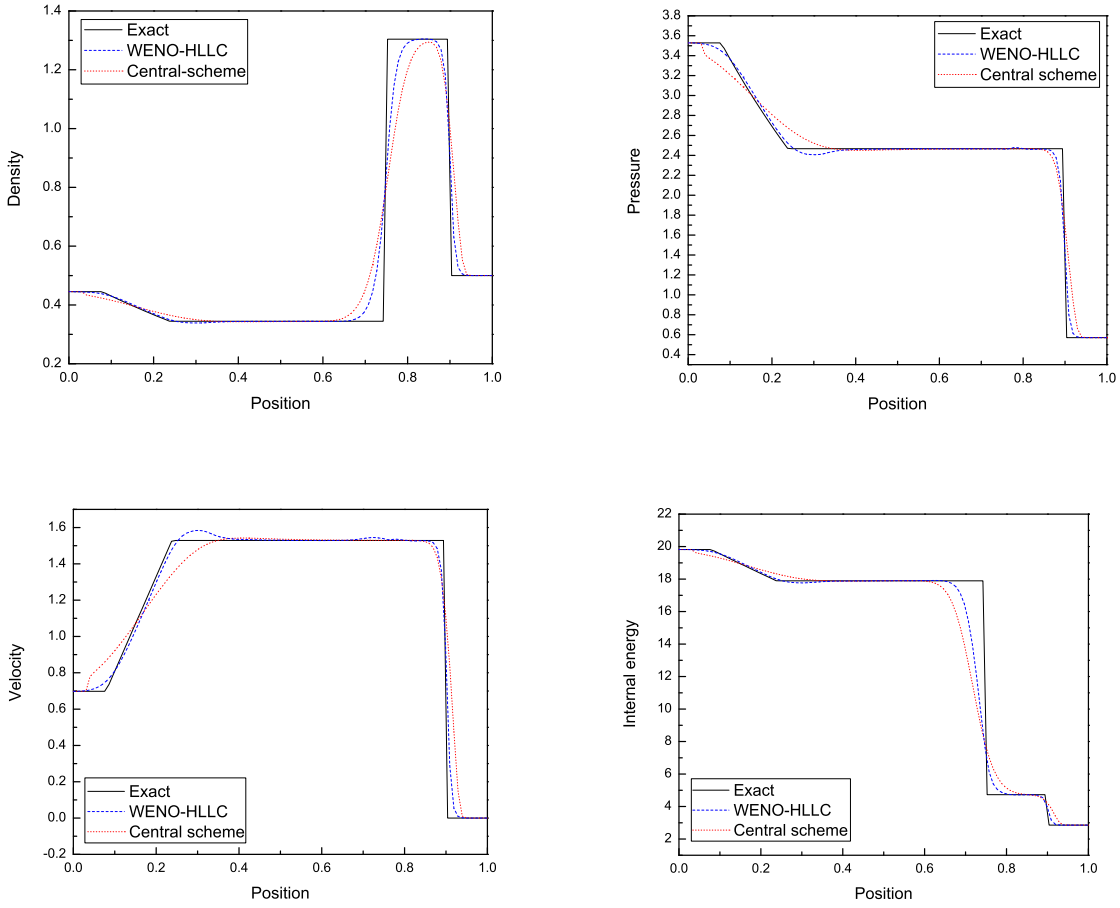


Figure 3.26: Central scheme by Kurganov-Tadmor and WENO reconstruction algorithm along with the HLLC Riemann solver.

The CFL number used was set as 0.9, the initial discontinuity's position was at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.16$ units.

3.2 2D shock tube problem

This part of the chapter covers the application of several of the previously presented numerical methods to a new problem, widely known as the explosion test. Moreover this problem demands the handling of the source terms, which are present in the formulation of the equations. This is an opportunity to check the efficiency of the algorithms, which are suitably designed to handle with source terms.
The term two-dimensional shock tube problem is in fact misleading, since the equations that have to be solved are the Euler ones in one dimension plus some source terms. In fact the multidimensional Euler equations, whenever a cylindrical or spherical symmetry is enforced, can be simplified to the one-dimensional inhomogeneous system

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_r = \mathbf{S}(\mathbf{U}) \tag{3.3}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} \quad , \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(E+p) \end{bmatrix} \quad , \quad \mathbf{S} = -\frac{a}{r} \begin{bmatrix} \rho u \\ \rho u^2 \\ u(E+p) \end{bmatrix}$$

The details about this formulation can be found in TORO ——. Here r is the radial direction, u is the radial velocity and a is a parameter. For $a = 0$ plain one-dimensional flow is reproduced, for $a = 1$ cylindrical symmetry is enforced and the equations are equivalent to the two-dimensional case. If $a = 2$ then spherical symmetry is assumed and the equations are equivalent to the three-dimensional homogeneous Euler equations for ideal gases. As already mentioned the main concern in this problem is to test the efficiency and the robustness of the numerical methods that solve equations with source terms.
In order to solve the equations with terms two different numerical schemes were chosen, namely PPM-HLLC and UNO. Each one of these schemes cooperates with the so called splitting numerical scheme for source terms, already presented in the previous chapter.
The left-state initial values for the explosion test are $\mathbf{U}_l = \big[\rho, \rho u, p\big]_l = \big[1, 0, 1\big]$ and the right-state ones are $\mathbf{U}_r = \big[\rho, \rho u, p\big]_r = \big[0.125, 0, 0.1\big]$. Once cylindrical symmetry is enforced the parameter a is set as equal to unity. It has to be noted that no exact solution is presented in the plots. The reader will have to take a look at Toro's ————.
A comparison between the results of the two different schemes, PPM-HLLC and UNO, proves that PPM-HLLC is more efficient. UNO still causes oscillations in different regions of the solution. The results of MUSCL-HLLC were omitted, because in fact they match in a respectable degree the ones by the PPM-HLLC numerical scheme.
The polytropic index is set as $\Gamma = 1.4$. The spatial domain is $x = [0, 1]$ and is discretised in 100 computational grid points. The CFL number is set as 0.9, the initial discontinuity's position is at $x_0 = 0.5$ and the different schemes are compared at time $t = 0.25$ units.

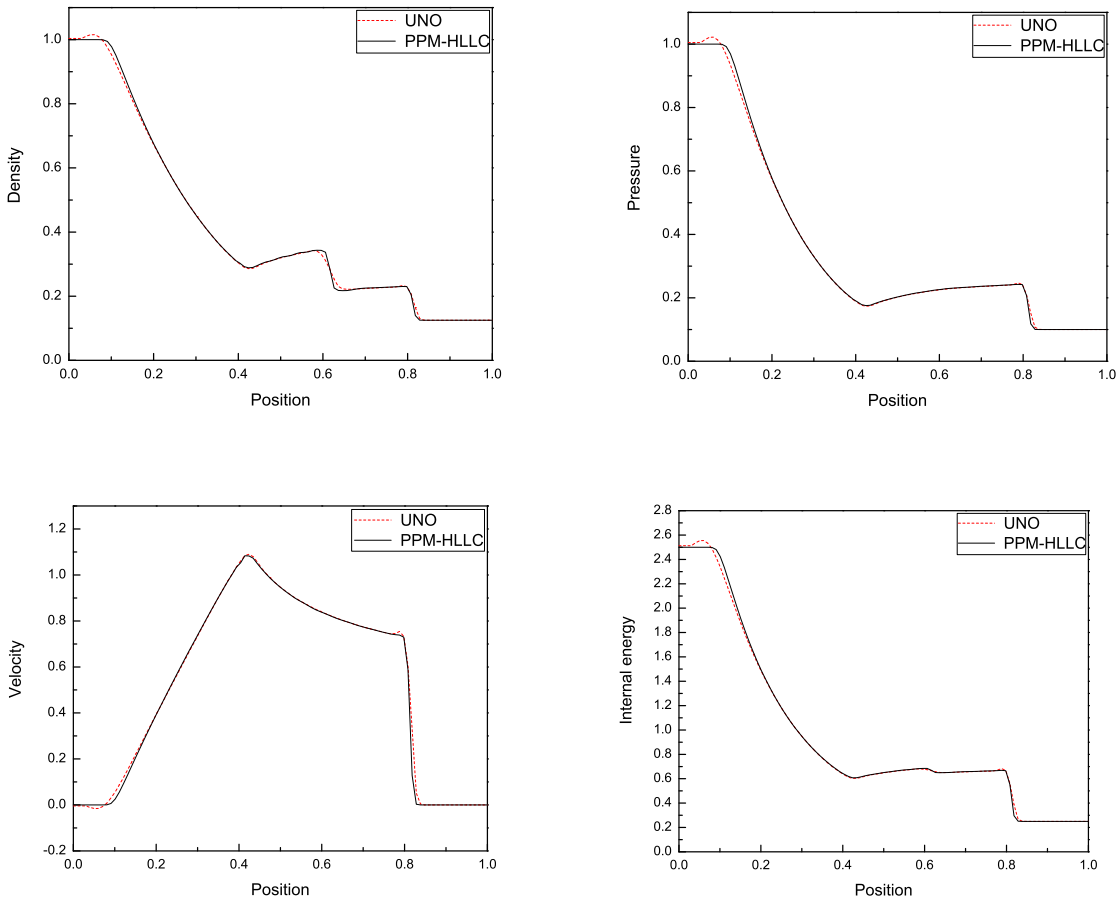


Figure 3.27: UNO and PPM-HLLC applied to the explosion test.

3.3 Vacuum boundary problem

The vacuum boundary problem was presented extensively in the previous chapter. The main task here is to check whether the algorithm devised by Munz —— is indeed robust and efficient enough to solve problems, which assume vacuum as their initial condition. Such a problem is the gas-vacuum expansion test applied to the one-dimensional homogeneous Euler equations for ideal gases. The left-state initial values for this test are $\mathbf{U}_l = \left[\rho, \rho u, p\right]_l = \left[1, 0, 2.5\right]$ and the right-state ones are $\mathbf{U}_r = \left[\rho, \rho u, p\right]_r = \left[0, 0, 0\right]$. The solution of this problem consists of an expansion wave which travels to the right. The exact solution can be found in MUNZ——. Two different numerical methods were used, namely PPM-HLLC and MUSCL-HLLC, suitably combined with the vacuum boundary tracking algorithm. The results show that the two methods are quite equivalent, although in the momentum plot the expansion wave and especially its head, is resolved in a different manner. It has to be noted that in the vacuum tracking algorithm, whenever the sound speed is required, the following formula is used

$$c_s = \sqrt{\Gamma \rho^{\Gamma - 1}} \tag{3.4}$$

In fact equation (3.4) is derived using the isentropic approximation and it works better than the usual formula $c_s = \sqrt{\Gamma \frac{p}{\rho}}$.

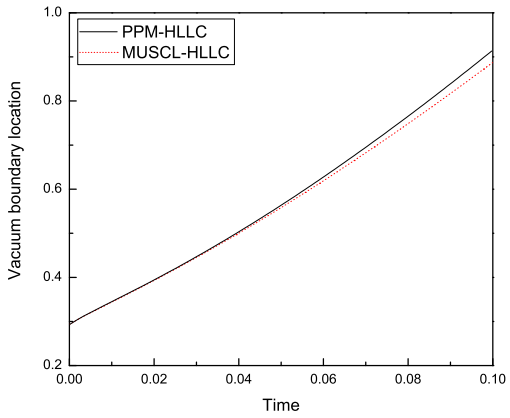


Figure 3.28: Location of the vacuum boundary as a function of time.

The polytropic index is set as $\Gamma = 1.4$. The spatial domain is $x = [0, 1]$ and is discretised in 100 computational grid points. The CFL number is set as 0.9, the initial discontinuity's position is at $x_0 = 0.3$ and the different schemes are compared at time $t = 0.1$ units.

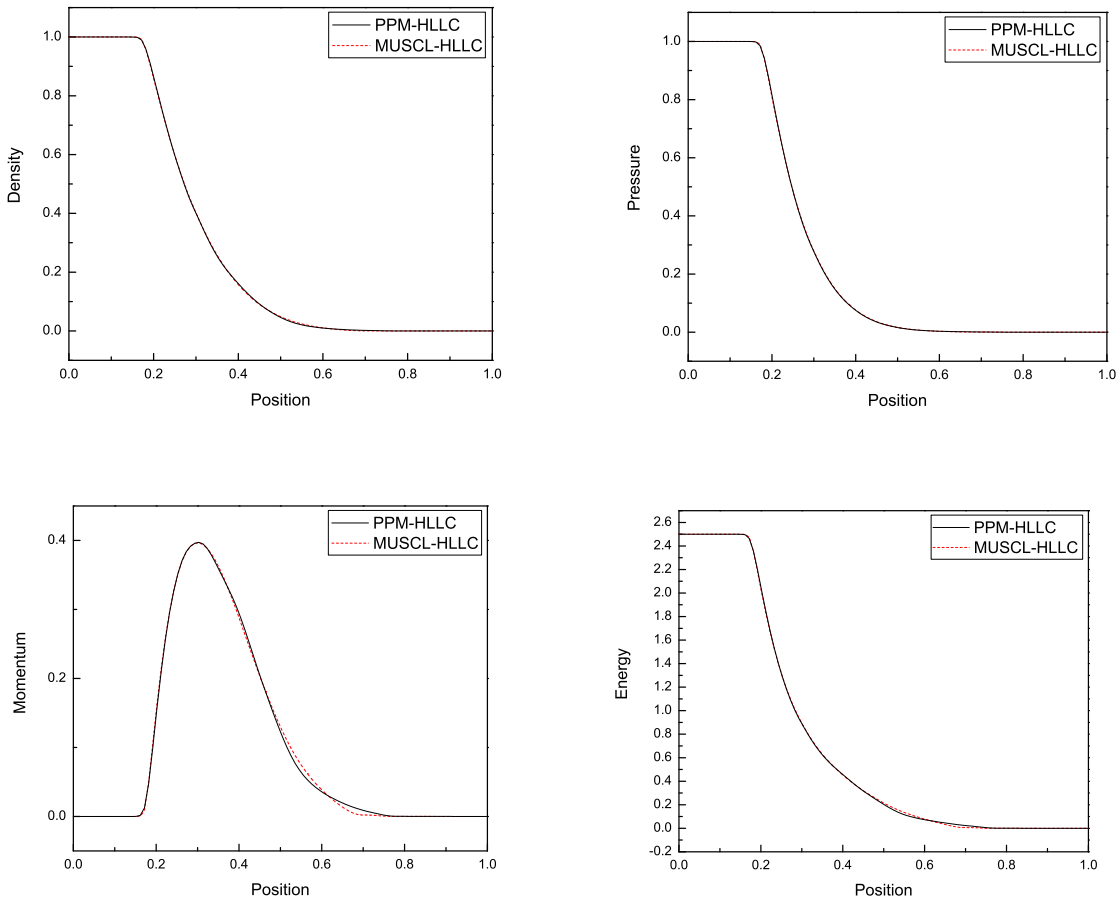


Figure 3.29: PPM-HLLC and MUSCL-HLLC applied to the gas-vacuum expansion test.

3.4 MHD shock tube problems

Up to now the different numerical methods were used to solve the system of the Euler equations for ideal gas dynamics. Both the cases of the homogeneous and the inhomogeneous system were treated using shock tube problems as initial conditions. The next step is to apply the appropriate numerical method to a more complex system of equations, namely the ideal MHD equations, which in fact describe the coupling of Euler's ideal gas dynamics equations and Maxwell's equations. The system of the MHD equations in conservative form is as follows

$$\mathbf{U}_t + \mathbf{F}(\mathbf{U})_x = 0 \tag{3.5}$$

where

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ B_y \\ B_z \\ E \end{bmatrix} \quad , \quad \mathbf{F} = \begin{bmatrix} \rho u_x \\ \rho u_x^2 + p^* - B_x^2 \\ \rho u_x u_y - B_x B_y \\ \rho u_x u_z - B_x B_z \\ B_y u_x - B_x u_y \\ B_z u_x - B_x u_z \\ u_x(E + p^*) - B_x\big(B_x u_x + B_y u_y + B_z u_z\big) \end{bmatrix}$$

A Γ-law equation of state is assumed

$$p = (\Gamma - 1)\rho\epsilon \tag{3.6}$$

where p is the gas pressure. The total pressure and the total energy are

$$p^* = p + \frac{1}{2}\big(B_x^2 + B_y^2 + B_z^2\big) \quad , \tag{3.7}$$

$$E = \frac{1}{2}\rho\big(u_x^2 + u_y^2 + u_z^2\big) + \rho\epsilon + \frac{1}{2}\big(B_x^2 + B_y^2 + B_z^2\big) \tag{3.8}$$

In order to solve numerically the MHD equations the HLLC Riemann solver, suitably modified as proposed by LIU—– and already presented in the previous chapter, was used. The reconstruction method chosen was PPM.
The results concerning two different MHD shock tube problems are presented in the following pages. In each case the polytropic index is set as $\Gamma = \frac{5}{3}$. The spatial domain is $x = [0, 1]$ and is discretised in 512 computational grid points. For every problem the CFL number, the position of the initial discontinuity and the total time will be noted.

3.4.1 First MHD shock tube problem

The left-state initial values for the first MHD shock tube test are $\mathbf{U}_L = \left[\rho, u_x, u_y, u_z, B_y, B_z, p\right]_L = \left[1, 10, 0, 0, \frac{5}{\sqrt{4\pi}}, 0, 20\right]$ and the right-state ones are $\mathbf{U}_R = \left[\rho, u_x, u_y, u_z, B_y, B_z, p\right]_R = \left[1, -10, 0, 0, \frac{5}{\sqrt{4\pi}}, 0, 1\right]$ with $B_x = \frac{5}{\sqrt{4\pi}}$. The solution consists of a pair of two fast shock waves, a slow rarefraction wave moving to the left, a slow sock wave travelling to the right and a contact discontinuity. The exact solution of this problem can be found in RYU——.

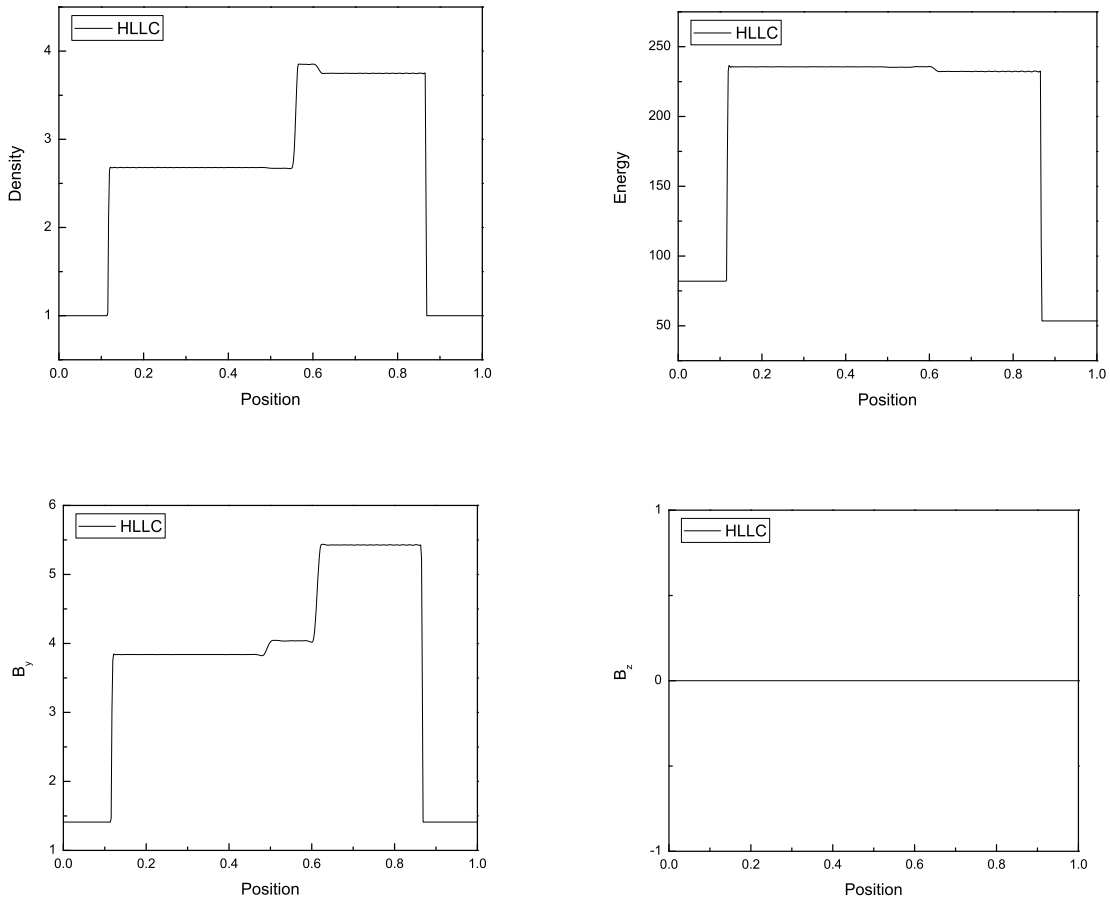


Figure 3.30: MHD-HLLC Riemann solver along with PPM reconstruction applied to the first MHD shock tube test.

The CFL number is set as 0.8, the initial discontinuity's position is at $x_0 = 0.5$ and the total time is $t = 0.08$ units.

The results presented concern the density, the energy ,B_y, B_z, the gas pressure p, u_x, u_y, u_z. The HLLC Riemann solver, suitably modified for the ideal MHD equations, proves itself to be efficient and robust.

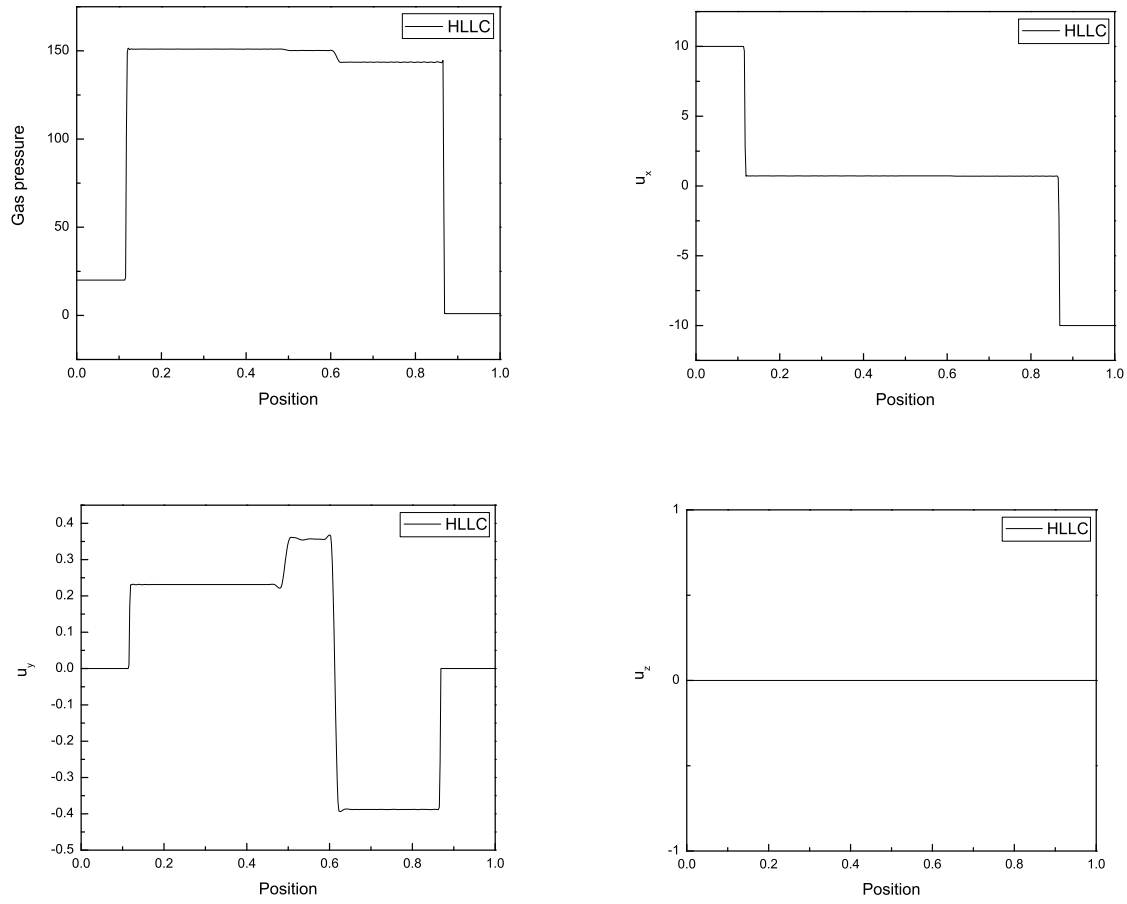


Figure 3.31: MHD-HLLC Riemann solver along with PPM reconstruction applied to the first MHD shock tube test.

3.4.2 Second MHD shock tube problem

The left-state initial values for the first MHD shock tube test are $\mathbf{U}_L = \left[\rho, u_x, u_y, u_z, B_y, B_z, p\right]_L = \left[1, 0, 0, 0, \frac{6}{\sqrt{4\pi}}, 0, 1\right]$ and the right-state ones are $\mathbf{U}_R = \left[\rho, u_x, u_y, u_z, B_y, B_z, p\right]_R = \left[0.1, 0, 2, 1, \frac{1}{\sqrt{4\pi}}, 0, 10\right]$ with $B_x = \frac{3}{\sqrt{4\pi}}$. The solution of this problem consists of a pair of a fast shock wave, a rotational discontinuity wave and a slow shock wave that travel from the left side of the contact discontinuity and a slow rarefraction wave, a rotational discontinuity and a slow rarefraction wave travelling to the right. The exact solution of this problem can be found in RYU——.

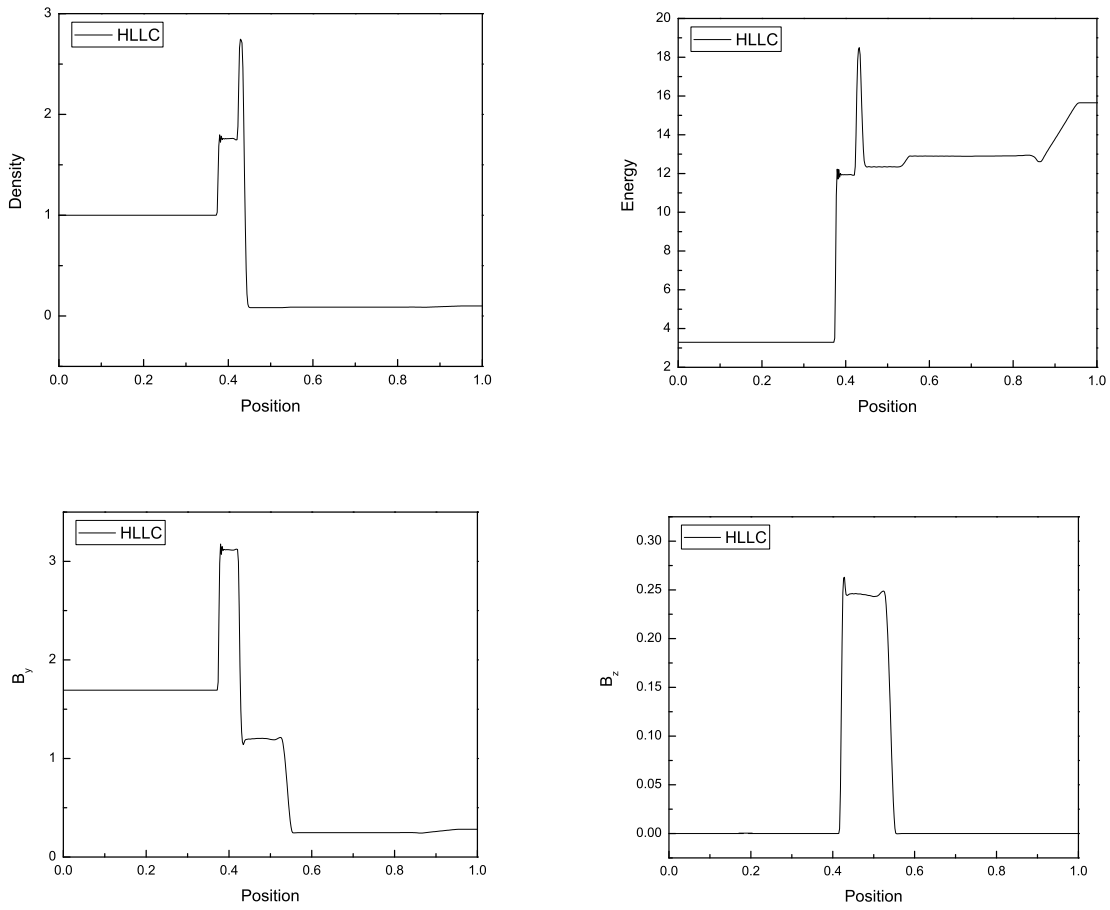


Figure 3.32: MHD-HLLC Riemann solver along with PPM reconstruction applied to the second MHD shock tube test.

The results presented concern the density, the energy ,B_y, B_z, the gas pressure p, u_x, u_y, u_z. The HLLC Riemann solver, suitably modified for the ideal MHD equations manages to resolve in an efficient way the different wave patterns present in the solution. Nevertheless several oscillations are present in the solution, especially in the region of the first fast shock wave.

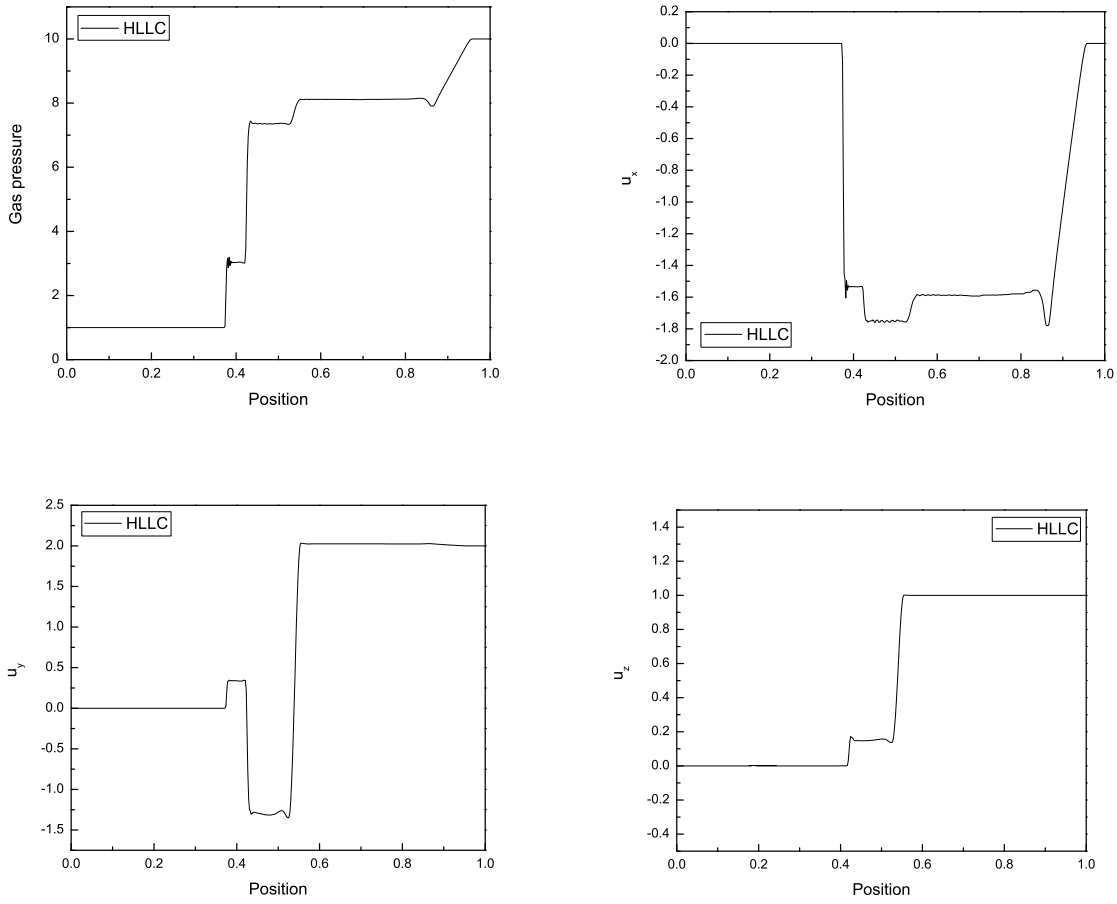


Figure 3.33: MHD-HLLC Riemann solver along with PPM reconstruction applied to the second MHD shock tube test.

The CFL number is 0.8, the initial discontinuity is at $x_0 = 0.5$ and the total time is $t = 0.035$ units.

Chapter 4

Dynamical simulations

The dynamical evolution of a star is described by the appropriate set of equations, which can either be the hydrodynamic ones (equations (1.23)-(1.26)) or the MHD ones (equations (1.37)-(1.41)). The solution of these equations is a demanding procedure, since it incorporates the usage of different numerical schemes, to each one of which a different task is assigned. It was quite disappointing that most of the schemes that were presented at the previous chapter failed to produce satisfactory results. The numerical code which was developed for the simulations uses the HLLC Riemann solver in order to solve the homogeneous one dimensional equations along with the PPM reconstruction algorithm. The splitting method was used for the source terms. The Poisson equation was solved using the fast tridiagonal solution algorithm. All these methods have already been thoroughly described at chapter 2.
It is important to note that the vacuum tracking algorithm has also been incorporated to the numerical code. In fact most of the simulations are meant to check the behavior of this particular algorithm when it comes to more serious problems than the simple Riemann ones. The presence of a vacuum region around a star is an innovation and its results have to be taken seriously into account. Up to now the simulations concerning the dynamical evolution of a star assume the presence of an artificial atmosphere, which surrounds the stellar object. This atmosphere is necessary for the numerical calculations, especially when Riemann solvers are used. Nevertheless the introduction of an atmosphere, which is not present when it comes to real life, causes problems. The Riemann problem which has to be solved at the boundary of the star and its surrounding region is different in the case vacuum is present or if on the other hand an artificial atmosphere exists.
Initially the simulations concern the solution of the one dimensional hydrodynamic Euler equations in the case of a static, uniformly rotating and differentially rotating fluid. These cases have been examined using both approaches, the vacuum and the presence of an atmosphere. The next step is the solution of the MHD equations. In this case the results concern just static stars and assume an atmosphere surrounding the star.
Several very important aspects of the problem are also presented in the present chapter,

concerning mainly the nondimensional formulation used, the initial and boundary conditions, the gravitational potential and the Lane-Emden formulation.

4.1 Nondimensional formulation

Nondimensional variables are introduced in order to facilitate the numerical calculations. By expressing mass, length, time using a different set of units, other than the Gaussian ones (gram, centimeter, second) it is possible to work with numbers which are closer to unity, eliminating significantly the presence of truncation errors. The fundamental scale for mass used throughout the simulations is the mass of the sun

$$M_0 = 1.99 \times 10^{33} \quad \text{gr} \tag{4.1}$$

for length it is the typical radius of a neutron star

$$R = 10\text{km} = 10^6 \quad \text{cm} \tag{4.2}$$

and for time it is the millisecond

$$t_0 = 1\text{ms} = 10^{-3} \quad \text{s} \tag{4.3}$$

The typical central density of a neutron star in Gaussian units is $\rho_c = 2 \times 10^{15}$ gr cm^{-3}. By expressing the density according to the new units its value is $\rho_c = 1.0052\ M_0\ R^{-3}$. It is evident that it is preferable to run numerical calculations using the latter value of the central density than numbers which are as large as 10^{15}.
Using the above fundamental scales for mass, length and time it is possible to derive nondimensional expressions for the variables which will be dynamically evolved during the simulations. Thus the nondimensional distance is

$$r = \bar{r}R \Rightarrow \bar{r} = \frac{r}{R}$$

the density is

$$\rho = \bar{\rho}\frac{M_0}{R^3} \Rightarrow \bar{\rho} = \rho\frac{R^3}{M_0}$$

time is expressed as

$$t = \bar{t}t_0 \Rightarrow \bar{t} = t\frac{1}{t_0}$$

the nondimensional radial velocity is

$$u^r = \bar{u}^r\frac{R}{t_0} \Rightarrow \bar{u}^r = u^r\frac{t_0}{R}$$

the angular velocity is

$$u^\phi = \bar{u}^\phi\frac{1}{t_0} \Rightarrow \bar{u}^\phi = u^\phi t_0$$

the nondimensional pressure is

$$p = \bar{p}\frac{M_0}{Rt_0^2} \Rightarrow \bar{p} = p\frac{Rt_0^2}{M_0}$$

and finally the body force is

$$f^r = \bar{f}^r\frac{R}{t_0^2} \Rightarrow \bar{f}^r = f^r\frac{t_0^2}{R}$$

All other quantities can be computed from the above ones directly in the dimensionless system. As an example the total energy in the hydrodynamic case is

$$\bar{E} = \frac{1}{2}\bar{\rho}\left(\left(\bar{u}^r\right)^2 + \left(\bar{u}^\phi\right)^2\right) + \frac{\bar{p}}{\Gamma - 1}$$

The next step is to substitute the above nondimensional variables to the system of the one dimensional hydrodynamic Euler equations. The system derived is the subsequent one

$$\frac{\partial\bar{\rho}}{\partial\bar{t}} + \frac{\partial}{\partial\bar{r}}\left(\bar{\rho}\bar{u}^r\right) = -\frac{1}{\bar{r}}\bar{\rho}\bar{u}^r \tag{4.4}$$

$$\frac{\partial}{\partial\bar{t}}\left(\bar{\rho}\bar{u}^r\right) + \frac{\partial}{\partial\bar{r}}\left[\bar{\rho}\left(\bar{u}^r\right)^2 + \bar{p}\right] = -\frac{1}{\bar{r}}\bar{\rho}\left(\bar{u}^r\right)^2 + \bar{r}\bar{\rho}\left(\bar{u}^\phi\right)^2 + \bar{\rho}\bar{f}^r \tag{4.5}$$

$$\frac{\partial}{\partial\bar{t}}\left(\bar{\rho}\bar{u}^\phi\right) + \frac{\partial}{\partial\bar{r}}\left(\bar{\rho}\bar{u}^r\bar{u}^\phi\right) = -\frac{3}{\bar{r}}\bar{\rho}\bar{u}^r\bar{u}^\phi \tag{4.6}$$

$$\frac{\partial\bar{E}}{\partial\bar{t}} + \frac{\partial}{\partial\bar{r}}\left[\bar{u}^r\left(\bar{E} + \bar{p}\right)\right] = -\frac{1}{\bar{r}}\bar{u}^r\left(\bar{E} + \bar{p}\right) + \bar{\rho}\bar{u}^r\bar{f}^r \tag{4.7}$$

It can be easily observed that the system of the hydrodynamic equations remains the same. In the case of the MHD simulations the components of the magnetic field must also be derived in their nondimensional form. It is for the radial component

$$B^r = \bar{B}^r\frac{M_0^{\frac{1}{2}}}{R^{\frac{1}{2}}t} \Rightarrow \bar{B}^r = B^r\frac{R^{\frac{1}{2}}t}{M_0^{\frac{1}{2}}}$$

and for the angular component

$$B^\phi = \bar{B}^\phi\frac{M_0^{\frac{1}{2}}}{R^{\frac{3}{2}}t} \Rightarrow \bar{B}^\phi = B^\phi\frac{R^{\frac{3}{2}}t}{M_0^{\frac{1}{2}}}$$

By substituting the nondimensional quantities to the one dimensional MHD equations, the system remains unchanged

$$\frac{\partial\bar{\rho}}{\partial\bar{t}} + \frac{\partial}{\partial\bar{r}}\left(\bar{\rho}\bar{u}^r\right) = -\frac{1}{\bar{r}}\bar{\rho}\bar{u}^r \tag{4.8}$$

$$\frac{\partial}{\partial \bar{t}}\left(\bar{\rho}\bar{u}^r\right)+\frac{\partial}{\partial \bar{r}}\left[\bar{\rho}\left(\bar{u}^r\right)^2+\bar{p}-\left(\bar{B}^r\right)^2\right]=-\frac{1}{\bar{r}}\bar{\rho}\left(\bar{u}^r\right)^2+\bar{r}\bar{\rho}\left(\bar{u}^\phi\right)^2+\bar{\rho}\bar{f}^r+\frac{\left(\bar{B}^r\right)^2-\bar{r}^2\left(\bar{B}^\phi\right)}{\bar{r}} \quad (4.9)$$

$$\frac{\partial}{\partial \bar{t}}\left(\bar{\rho}\bar{u}^\phi\right)+\frac{\partial}{\partial \bar{r}}\left(\bar{\rho}\bar{u}^r\bar{u}^\phi-\bar{B}^r\bar{B}^\phi\right)=-\frac{3}{\bar{r}}\left(\bar{\rho}\bar{u}^r\bar{u}^\phi-\bar{B}^r\bar{B}^\phi\right) \quad (4.10)$$

$$\frac{\partial \bar{B}^\phi}{\partial \bar{t}}+\frac{\partial}{\partial \bar{r}}\left(\bar{B}^\phi\bar{u}^r-\bar{B}^r\bar{u}^\phi\right)=-\frac{1}{\bar{r}}\left(\bar{B}^\phi\bar{u}^r-\bar{B}^r\bar{u}^\phi\right) \quad (4.11)$$

$$\frac{\partial \bar{E}}{\partial \bar{t}}+\frac{\partial}{\partial \bar{r}}\left[\bar{u}^r\left(\bar{E}+\bar{p}\right)-\bar{B}^r\left(\bar{B}^r\bar{u}^r+\bar{r}^2\bar{B}^\phi\bar{u}^\phi\right)\right]=-\frac{1}{\bar{r}}\bar{u}^r\left(\bar{E}+\bar{p}+\frac{1}{2}\left((\bar{B}^r)^2+(\bar{r}\bar{B}^\phi)^2\right)\right)+\bar{\rho}\bar{u}^r\bar{f}^r-\bar{B}^r\left(\bar{B}^r\bar{u}^r+\bar{r}^2\bar{B}^\phi\bar{u}^\phi \quad (4.12)$$

In the following the carets will be omitted for simplicity.

4.2 Lane - Emden formulation

It is obvious that the variables used in the Lane - Emden equation are different than those which have to be used in the one dimensional hydrodynamic and MHD systems of equations. Therefore the variables ξ, θ and ζ have to be appropriately transformed, using the necessary equations, in order to provide the initial data for the density, the angular velocity and the pressure of the dynamical simulations. If it is R the radius of polytrope, ξ_1 its radius in Lane - Emden units, then

$$R=a\xi_1 \Rightarrow a=\frac{R}{\xi_1}$$

The distance can be calculated using equation (1.8) and the density of the star by (1.7). Since the central density is expressed in Gaussian units (gr cm^{-3}) the values of the distance and the density are also expressed in the same units. Therefore the nondimensional formulation of the previous section must be taken into account.
Concerning the rotation number ζ, since it is u^ϕ=Ω, the transformation takes place as follows, according to (1.14)

$$u^\phi=\sqrt{\zeta\pi G\rho_c}$$

and since G and ρ_c are expressed in Gaussian units, u^ϕ has to be transformed to a nondimensional variable.

4.3 Potential

The values of the potential Φ are given by Poisson's equation

$$\nabla^2\Phi = 4\pi G\rho$$

Since cylindrical coordinates are used throughout the simulations, the solution for the potential outside the star is not the usual one as in spherical coordinates. The Laplacian in cylindrical coordinates is

$$\nabla^2\Phi = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial\Phi}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2\Phi}{\partial\phi^2} + \frac{\partial^2\Phi}{\partial z^2}$$

In the one dimensional case the potential Φ does not depend on the coordinates ϕ and z. Substituting the Laplacian in Poisson's equation and integrating, yields equation (1.4)

$$\frac{d\Phi}{dr} = \frac{4\pi G}{r}\int_0^r \rho r dr \Rightarrow$$

$$\frac{d\Phi}{dr} = \frac{2G}{r}\mu(r) \tag{4.13}$$

where

$$\mu(r) = 2\pi\int_0^r \rho r dr \tag{4.14}$$

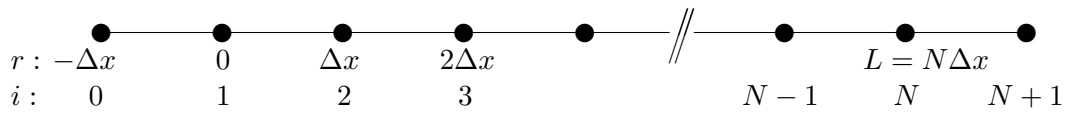
In fact $\mu(r)$ is the mass per unit length interior to radius r. Integrating equation (4.13) inward from a radius r_0 outside the cylinder gives

$$\Phi(r) = 2\mu_t \ln\left(\frac{r}{r_0}\right) \quad \text{for} \quad r \geq R \tag{4.15}$$

where μ_t=$\mu(R)$ is the total mass per unit length of the star. The values of Φ outside the cylinder increase logarithmically with r. This expression for the potential is used in the simulations. The potential inside the star is defined by the solution of Poisson's equation and its values outside the star are given by equation (4.15).

4.4 Initial and boundary conditions

Since the simulation stake place using a finite grid, it is essential to define the boundary conditions which will be used. The boundary conditions describe the behavior of the variables both at the center of the star and at the end of the computational grid. It must be noted that it is necessary for computational reasons to introduce ghost points.



These points are situated on the left of the first point of the grid, which is labelled as i=1. They are used throughout the calculations since the boundary conditions at the center of the star (x=0) are denoted by derivatives. In fact the boundary conditions for the conservative variables $U = \big(\rho, \rho u^r, \rho u^\phi, E\big)$ and the potential Φ at the center are

$$\frac{\partial \rho}{\partial r}\Big|_{r=0} = 0, \quad \frac{\partial^2 \big(\rho u^r\big)}{\partial r^2}\Big|_{r=0} = 0, \quad \frac{\partial \big(\rho u^\phi\big)}{\partial r}\Big|_{r=0} = 0,$$

$$\frac{\partial E}{\partial r}\Big|_{r=0} = 0 \quad \text{and} \quad \frac{d\Phi}{dr}\Big|_{r=0} = 0$$

Using central differences the derivative for the density can be written as

$$\frac{\partial \rho}{\partial r}\Big|_{r=0,i=1} = 0 \Rightarrow \frac{\rho_2 - \rho_0}{2\Delta x} = 0 \Rightarrow$$

$$\rho_2 = \rho_0$$

The derivatives for ρu^ϕ, E and Φ are treated in the same manner and it is evident that these variables appear to be symmetric at the center of the star. On the other the derivative of the momentum ρu^r is of second order and central differences yield

$$\frac{\partial^2 \big(\rho u^r\big)}{\partial r^2}\Big|_{r=0,i=1} = 0 \Rightarrow \frac{\big(\rho u^r\big)_0 - 2\big(\rho u^r\big)_1 + \big(\rho u^r\big)_2}{\big(\Delta x\big)^2} = 0 \Rightarrow$$

$$\big(\rho u^r\big)_0 - 2\big(\rho u^r\big)_1 + \big(\rho u^r\big)_2 = 0 \tag{4.16}$$

Because the cylinder is axisymmetric, the radial velocity must vanish for all time, namely $u^r(t,0) = 0$. So it is $u^r_1 = 0$ and equation (4.16) yields

$$\big(\rho u^r\big)_0 = -\big(\rho u^r\big)_2$$

Apparently the momentum ρu^r is antisymmetric at the center of the star.
The choice of the point r=0 as the initial grid point imposes difficulties, which are evident in the case of the hydrodynamic equations (equations (1.23)-(1.26)). The term $\frac{1}{r}$ which is present in the source terms is driven to infinity when r=0. In order to deal with this singularity a Taylor approximation of the radial velocity u^r is used

$$u^r = u^r(t,0) + \frac{\partial u^r}{\partial r} r$$

Since $u^r(t,0) = 0$ it is

$$u^r = \frac{\partial u^r}{\partial r} r$$

and the term $\frac{\partial u^r}{\partial r}$ can be approximated using central differences. The source terms at the center of the star can be written as

$$\mathbf{S} = \begin{bmatrix} -\rho\frac{\partial u^r}{\partial r} \\ -\rho\left(\frac{\partial u^r}{\partial r}\right)^2 r + r\rho\big(u^\phi\big)^2 + \rho f^r \\ -3\rho u^\phi \frac{\partial u^r}{\partial r} \\ -\frac{\partial u^r}{\partial r}(E+p) + \rho r \frac{\partial u^r}{\partial r} f^r \end{bmatrix}$$

where the total energy is

$$E = \frac{1}{2}\rho\left(\big(r\frac{\partial u^r}{\partial r}\big)^2 + \big(ru^\phi\big)^2\right) + \frac{p}{\Gamma - 1}$$

For the rest of the grid points the source terms are given by equation (1.30). It is important to note that the source terms as presented here, are expressed using the primitive variables $W = \big(\rho, u^r, u^\phi, p\big)$. When using conservative numerical schemes the matrix of the source terms has to be rearranged using the conservative ones, $U = \big(\rho, \rho u^r, \rho u^\phi, E\big)$. The procedure is exactly the same when it comes to the MHD equations (equations (1.37)-(1.41)).
At the boundary which is situated at r=L and which coincides with the end of the computational grid, transmissive boundary conditions are used. In fact it is

$$\frac{\partial \rho}{\partial r}\Big|_{r=L} = 0, \quad \frac{\partial\big(\rho u^r\big)}{\partial r}\Big|_{r=L} = 0, \quad \frac{\partial\big(\rho u^\phi\big)}{\partial r}\Big|_{r=L} = 0,$$

$$\frac{\partial E}{\partial r}\Big|_{r=L} = 0 \quad \text{and} \quad \frac{d\Phi}{dr}\Big|_{r=L} = 0$$

A ghost point is added at the end of the computational region. These conditions allow the boundary to be transparent to the passage of the waves and does not impose any effect on them. When a wave reaches the boundary it just crosses it and does not affect the rest of the simulation.

4.5 Dynamical evolution

The solution of the equations, whether it is the hydrodynamic or the MHD ones, is carried out computationally. In both cases the reconstruction algorithm used is the PPM one, with its parameters as specified in chapter 2. The source terms are resolved using the splitting method. The Poisson equation is solved using the fast tridiagonal solution algorithm. Two different cases are examined for the region outside the star. In the first case an artificial atmosphere is assumed to surround the stellar object and in the second one a vacuum region is incorporated and the dynamical evolution takes place using the vacuum algorithm. The Riemann solver used throughout the numerical simulations is the HLLC one or its MHD variant.
The computational region is $r = [0, 4]$ in dimensionless units. The star extends from r=0,

where its center is situated, to r=1. The rest of the computational region is either a vacuum one or it is occupied by the atmosphere. The extent of the region surrounding the star has been chosen so in order to allow the propagation of the waves which may appear in the solution. The number of grid points varies according to the simulation. In fact results are presented for 200, 400, 800 and 1600 points. In each case the computational grid is uniform. The time step is calculated using the CFL condition, as described in chapter 2.
For each one of the simulations different parameters have to be defined. These parameters can either concern the physical characteristics of the star, such as the polytropic index n and the rotation number ζ or the numerical methods used, such as the CFL number. In each one of the cases that will be examined all these parameters are specified.
The initial data for all the following configurations are those that arise from the solution of the Lane-Emden equation. Static, uniformly rotating and differentially rotating stars are examined. In some cases perturbations of the initial data are introduced in order to check their effect on the evolution of the stars.

4.5.1 Hydrodynamic stars

The evolution of hydrodynamic stars is governed by the one-dimensional Euler equations, in which the effects of gravity are also incorporated. It is the system of equations (1.23) to (1.26), along with the Poisson equation and an equation of state (1.28). Three different cases are examined, namely a static star, a uniformly rotating and a differentially rotating. It is also distinct whether there is a vacuum region surrounding the fluid or an atmosphere.

Static star with atmosphere

In this case a static star $\big(\zeta = 0\big)$, surrounded by an artificial atmosphere, is studied. The polytropic index is n=1. The atmosphere surrounding the star is a low-density region. In fact it can be considered as a part of the star, which exceeds as far as the computational space. Its initial density is set equal to the density value of the last grid point which belongs to the star. The pressure of this region is given by the polytropic equation of state (1.2), where the polytropic constant has already been calculated while solving the Lane-Emden equation. When the values of the density in the atmosphere get larger than a multiplicate of their initial value, then they are reset to the initial density. In this particular simulation the gravitational field does not evolve during the simulation and the potential is kept stable.
The CFL number is set as 0.1 and the number of grid points is 1600. The total time of the simulation is 8 ms.The plots presented concern the total energy E, the pressure p, the radial velocity u^r, the Newtonian gravitational potential Φ. The density profile is also presented, along with a detail of the central star region. In all the plots the results of the dynamical evolution are compared with the initial data. It is important to note that the main purpose is to check how the numerical methods manage to handle the initial equilibrium configuration.

It is evident that the the simulation of the static star surrounded by an artificial atmosphere provides satisfactory results. It must be noted once more that the potential does not evolve throughout the simulation.

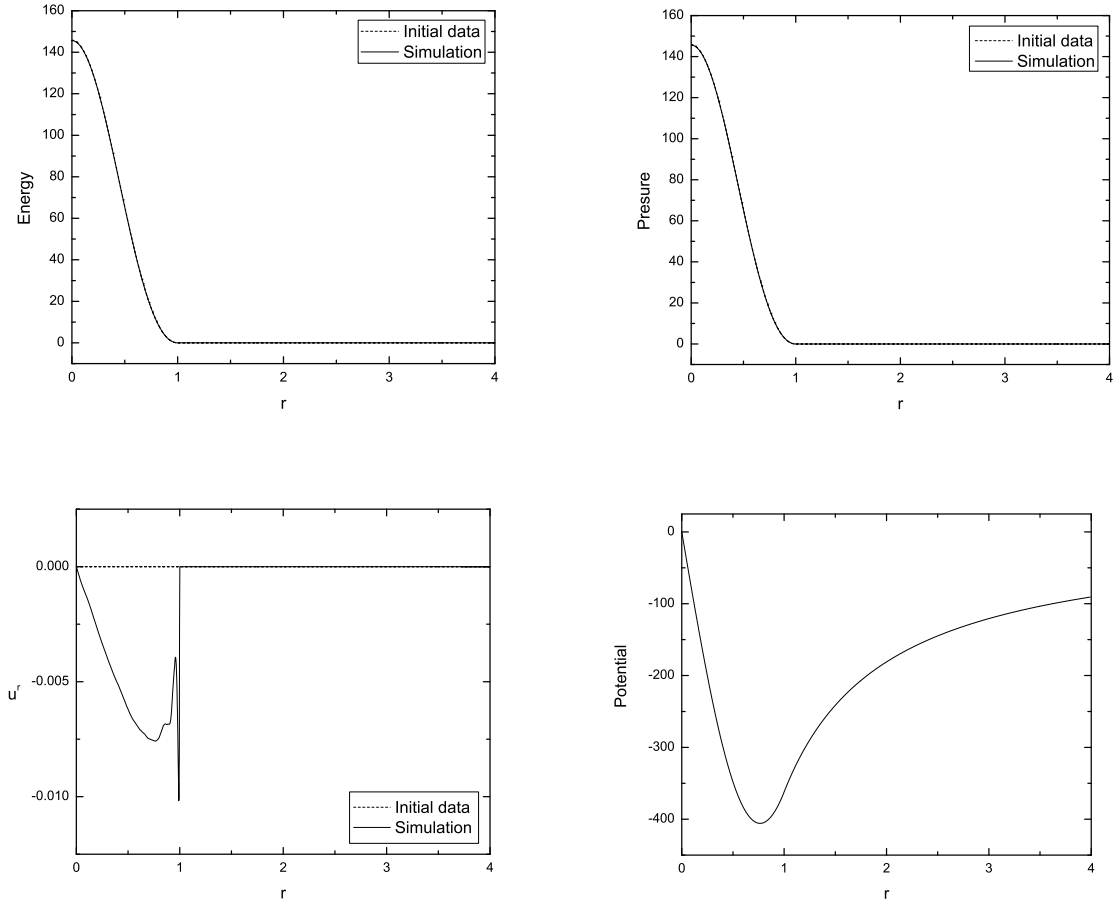


Figure 4.1: Energy, pressure, radial velocity and potential profiles for a static star with a surrounding vacuum region. The data of the simulation at t=8 ms are compared with the initial one.

The numerical methods do not seem to be able to capture sufficiently the central density of the star. In fact the results of the simulation diverge from the initial data, since the value of the central density is not equal to the initial one and the shape of the curve is different.

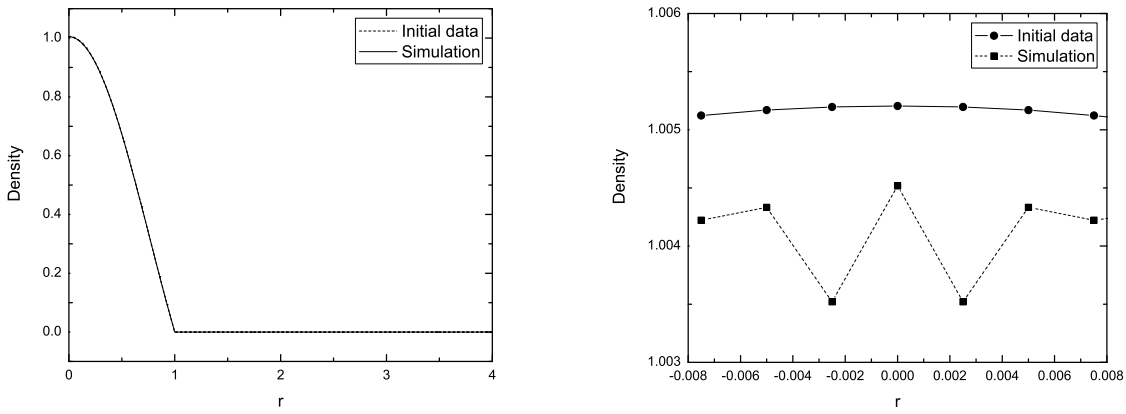


Figure 4.2: Density profile and detail at the center of the star at t=8 ms.

The next plot presents the behavior of the relative difference of the central density. It is ρ_0 the initial central density and ρ_c is the value of the central density at each time step, as it arises from the simulation.

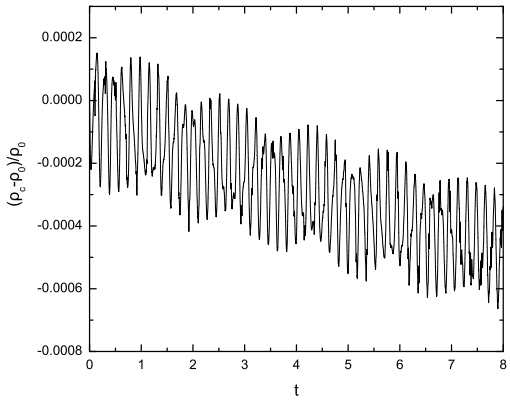


Figure 4.3: Plot of the relative difference of the central density versus time in ms.

It must be noted that the variables that are plotted are the nondimensional ones.

Static star with vacuum

In this case a static star $\left(\zeta = 0\right)$, surrounded by a vacuum region, is studied. The polytropic index is n=1. At the vacuum region all the variables ρ, u^r, u^ϕ, p are set equal to zero. In order to determine and calculate the boundary between the star and the void, the vacuum tracking algorithm is used, as described in chapter 2. It has to be noted that an extra criteria has been added to the program which allows the vacuum boundary to move backward and forward. This criteria states that if the vacuum boundary is situated before the surface of the star, then its value is reset to that of the grid point which is situated just before the surface. If $r_{surface}$ is the position of the surface of the star at each time step and $x_v(t)$ is the position of the vacuum boundary at the same time step, then if $x_v(t) < r_{surface} - \frac{\Delta x}{2}$ all the conservative variables are reset to zero and it is $x_v(t) = r_{surface} - \Delta x$. This criteria is used in all the simulations that concern stars surrounded by a vacuum region.
The CFL number is set as 0.1 and the number of grid points is 1600. The total time of the simulation is 8 ms.The plots presented concern the total energy E, the pressure p, the radial velocity u^r, the Newtonian gravitational potential Φ. The density profile is also presented, along with a detail of the central star region. In all the plots the results of the dynamical evolution are compared with the initial data.
It is obvious that the energy, the pressure, the radial velocity and the potential profiles are quite accurate. It has to be noted that in the case of the static star with a vacuum region surrounding it the potential evolves in time. In fact it is difficult to distinguish the initial configuration from the simulation's results. The distinction between the initial state and the one that has arisen after 8 ms is more evident in the case of the density plot of the center of the star. The ghost points are also obvious in this case. The point of this plot is to show in what a manner can the numerical schemes reproduce the behavior of the density at the central region. The results seem to be satisfactory since the simulation even after 8 ms approaches the initial equilibrium configuration and moreover the shape of the density plots is identical. The final result, when compared to the one of the static star with an artificial atmosphere, is more accurate.
The next important feature in the case of the static star with a vacuum region, is the behavior of the relative difference of the central density, namely $\frac{\rho_c - \rho_0}{\rho_0}$,where ρ_0 is the initial central density and ρ_c the central density at each time step. The plot of the relative difference versus time shows that the numerical schemes in the case of the vacuum manage to handle with greater efficiency the values at the center of the star. Finally the different positions x_v of the vacuum boundary are plotted versus time.
All the parameters that are plotted are the nondimensional ones.

In the following plots, the profiles of the variables are compared to the data of the initial equilibrium configuration.

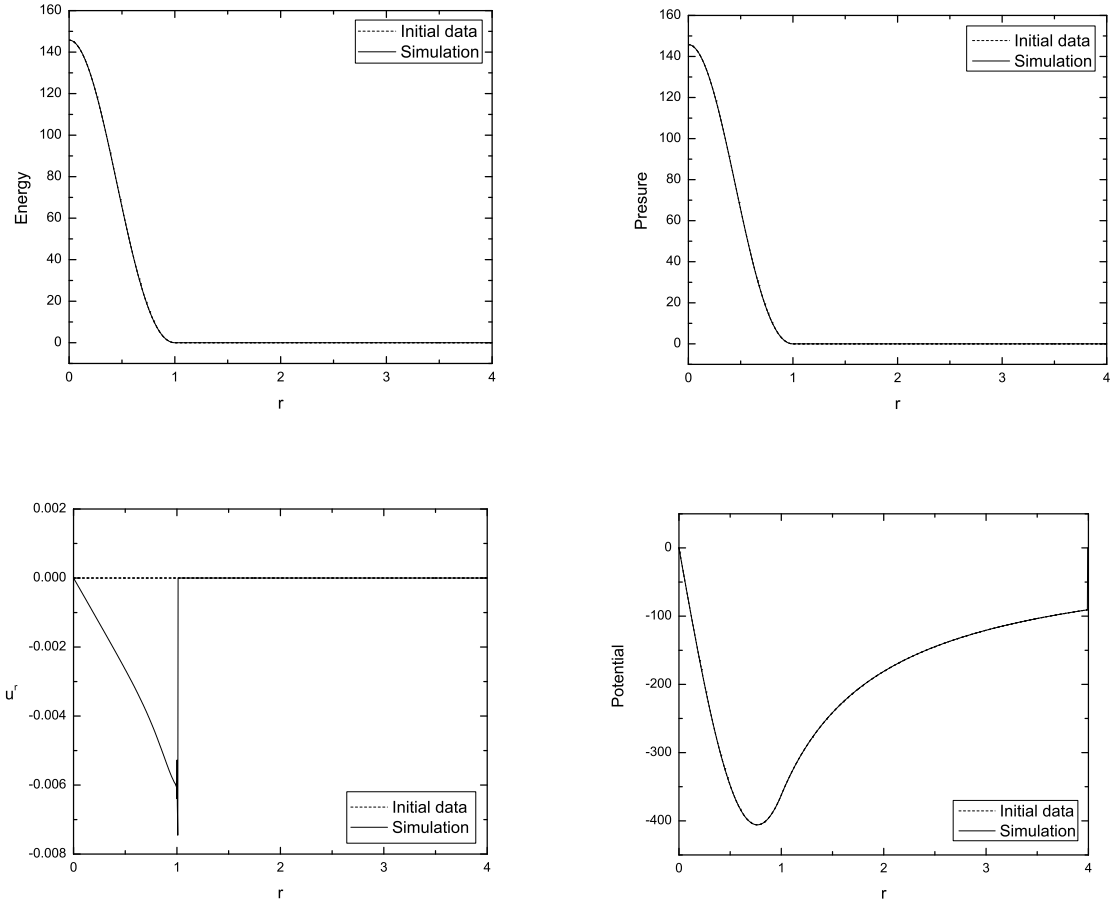


Figure 4.4: Energy, pressure, radial velocity and potential profiles for a static star with a surrounding vacuum region. The data of the simulation at t=8 ms are compared with the initial one.

The detail of the density profile shows that even after 8 ms the numerical schemes manage to provide robust results. It is important that the maxima of the density at the center is preserved and that the neighboring points do not diverge.

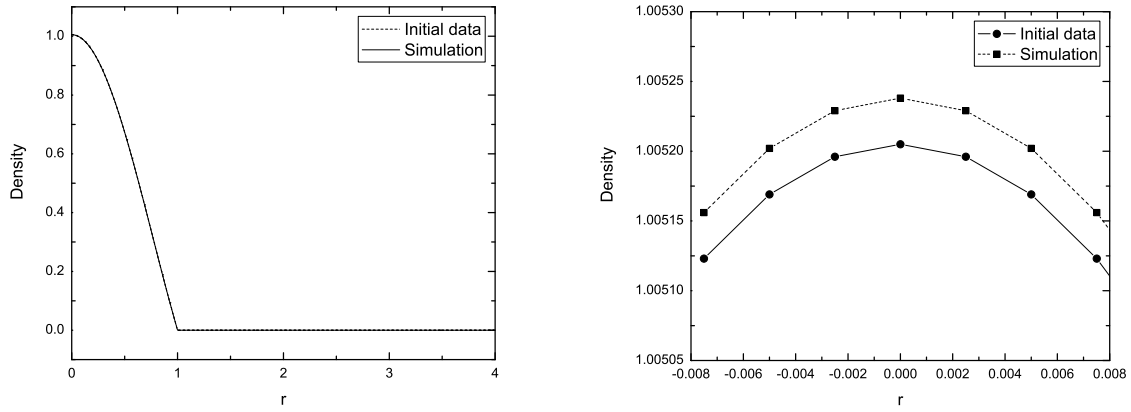


Figure 4.5: Density profile and detail at the center of the star at t=8 ms.

The behavior of the central density in the scheme with the vacuum region is oscillatory. In fact it seems to oscillate around the initial value ρ_0. This result is improved when compared to the one of the static star with an atmosphere.

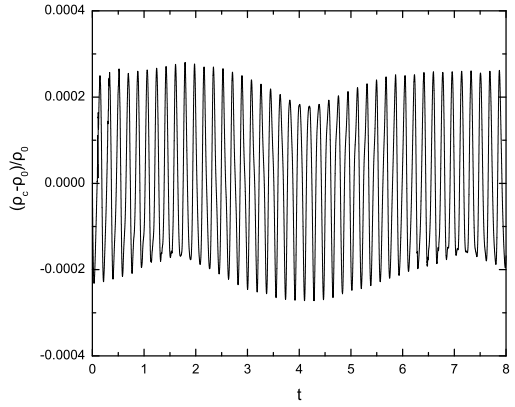


Figure 4.6: Plot of the relative difference of the central density versus time in ms.

The following plot presents the position of the vacuum boundary x_v as the simulation evolves.

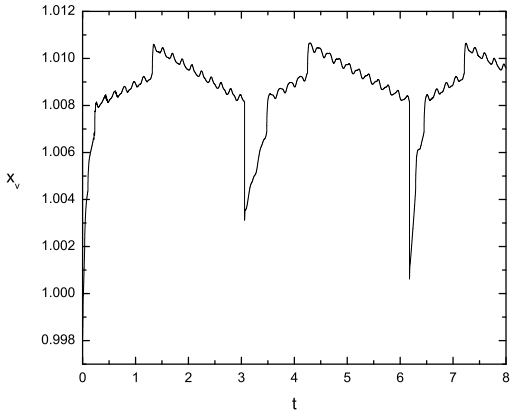


Figure 4.7: Position of the vacuum boundary x_v.

Comparison between the results for a static star with a vacuum region and an atmosphere

As it has already been mentioned the main purpose is to check which configuration, the one with the vacuum region or the one with the artificial atmosphere, can simulate in a proper manner the initial equilibrium state. For both cases the same numerical schemes were used. The Riemann solver was HLLC, the reconstruction algorithm was PPM, the source terms were resolved using the splitting method. Moreover if a vacuum region is present the vacuum boundary algorithm has to be added to the program. It has to be noted that for the static star with the atmosphere the Poisson equation is solved once, using the initial data and the simulation takes place by using these values of the potential. On the other hand, in the presence of a vacuum region, the Poisson equation is solved in each time step.
Two different plots are presented. The first one is a comparison of the density of the central region. The ghost points are also presented. In this particular plot the initial data are compared to the results of the dynamical evolutions of the star with a vacuum region and to those of the star with an atmosphere. Both results are at t=12 ms. It is obvious that the vacuum region case produces more robust result, since the shape and the values of the density profile match better the initial profile.
The second plot concerns the behavior of the central density. The relative difference of the density in the atmosphere case oscillates, but not around a fixed point. In fact the relative difference increases by time due to the augmentation of the numerical errors. Because the

introduction of an artificial atmosphere is not a physical phenomena the solution is biased. This procedure leads to larger numerical errors. On the other hand these problems do not seem to appear in the vacuum case. The relative difference of the density oscillates around a fixed value and these oscillations seem to have a steady amplitude.

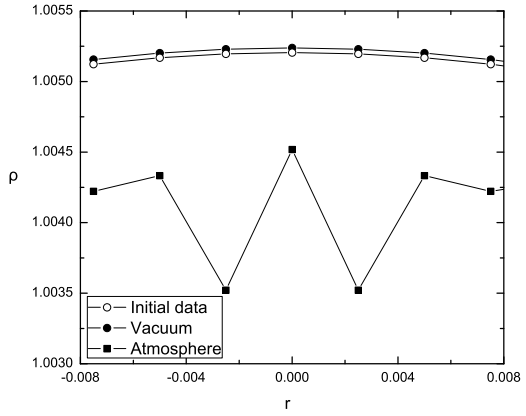


Figure 4.8: Detail of the density profiles at the center of the static star at t=12 ms.

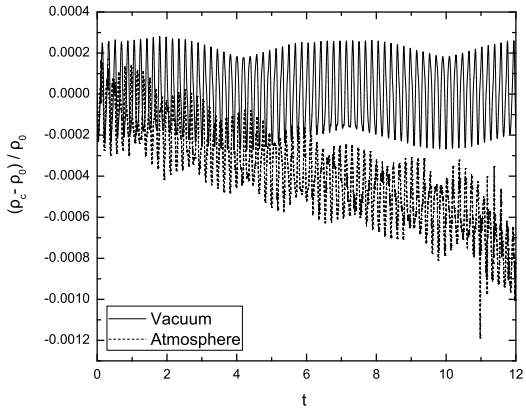


Figure 4.9: Plots of the relative difference of the central density versus time in ms.

Static stars results for different grid points

Up to now the results presented concerned simulations which had taken place using 1600 grid points. It is interesting to check the results of the dynamical evolutions for different numbers of grid points. Two different plots are presented. The first one shows the relative difference of the central density for a static star with an atmosphere for uniform grids with 200, 400, 600 and 1600. The second one displays the relative difference for the case of a static star with a vacuum region. The time is in ms.

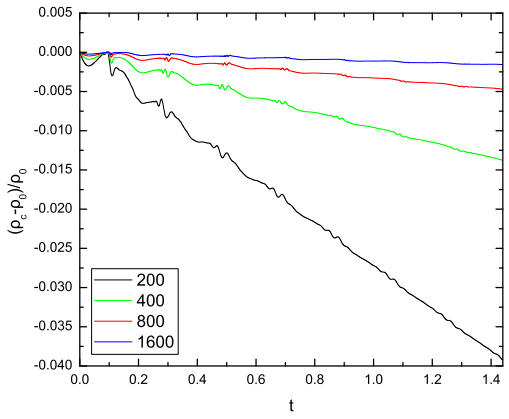


Figure 4.10: Plot for a static star with atmosphere for different grid points.

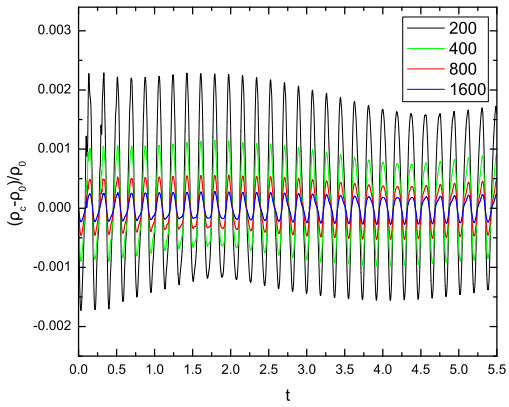


Figure 4.11: Plot for a static star with vacuum for different grid points.

Perturbed static star with vacuum

In this case a static star $\big(\zeta = 0\big)$, with a vacuum region surrounding it, is studied. The polytropic index is n=1. The main difference with the previous simulations is that perturbations are introduced. In fact the initial data used for the dynamical simulation are not exactly the same with the ones used in the previous examples. Once more the Lane-Emden equation is solved, but the initial pressure which is used for the dynamical evolution is not the one that the equilibrium configuration predicts. The pressure is given by the isentropic equation of state

$$p = K\rho^{\Gamma}$$

where K is the polytropic constant and $\Gamma = 1 + \frac{1}{n}$. The value of the polytropic constant is defined using equation (1.9), which yields

$$K = \frac{4\pi G a^2}{(n+1)\rho_c^{\frac{1}{n}-1}}$$

where $a = \frac{R}{\xi_1}$ and ξ_1 is the radius in Lane-Emden units. After having calculated K, a fraction of its value is used to define the pressure. In this particular simulation the initial pressure is

$$p = \frac{95}{100} K\rho^{\Gamma} \tag{4.17}$$

In the previous cases it was important to check the ability of the numerical schemes to simulate the initial equilibrium configuration. Now the main purpose is to check how the perturbations are handled.

The vacuum tracking algorithm is used along with the criteria described in the section for an unperturbed static star with a surrounding vacuum region. The numerical methods are the same as in the previous simulations, namely the HLLC Riemann solver, the PPM reconstruction algorithm, the splitting method.

The CFL number is equal to 0.1 and the number of grid points is 1600. The total time of the simulation is t=1.44 ms. The density, pressure, radial velocity, potential and internal energy profiles are presented. Moreover the relative difference of the central difference and the position of the vacuum boundary are plotted.

In the case of the perturbed static star with a vacuum region beyond its surface it is expected that the data of the simulation will differ from the initial one. In fact the differences are obvious in the following plots.

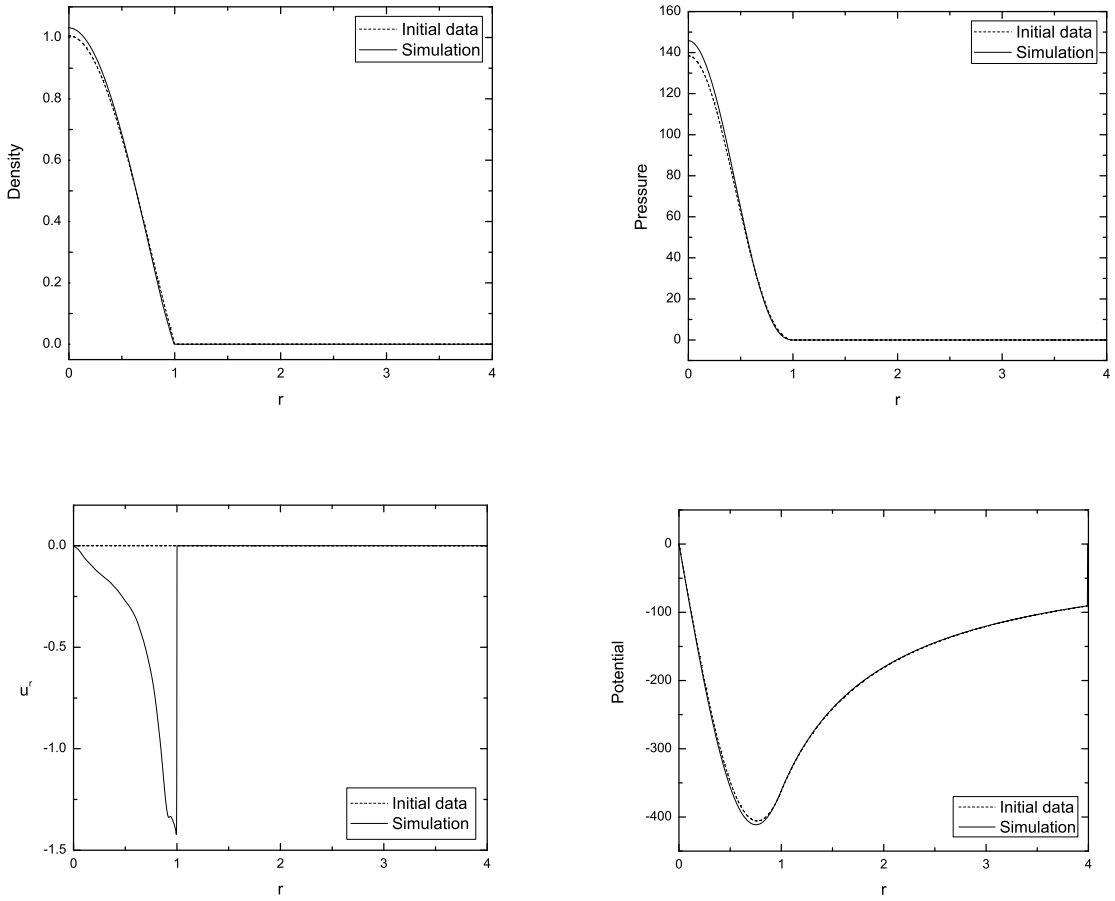


Figure 4.12: Density, pressure, radial velocity and potential profiles for a perturbed static star with a surrounding vacuum region. The data of the simulation at t=1.44 ms are compared with the initial one.

The internal energy is plotted and a detail of the plot near the surface is presented. It is obvious that in the region of the surface shock waves are created.

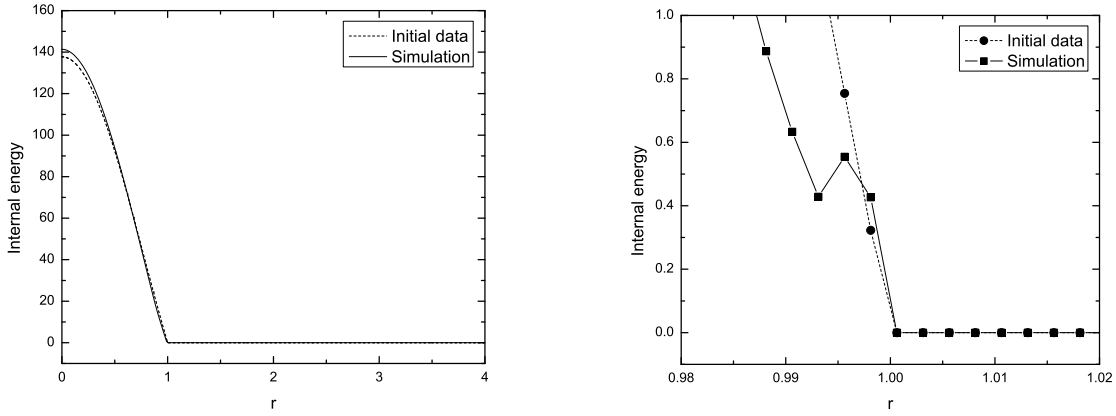


Figure 4.13: Internal energy profile and detail at the surface of the star at t=1.44 ms.

The perturbed star is driven to a new equilibrium state with a new central density, different from the initial new. The values of the central density oscillate round this new equilibrium value.

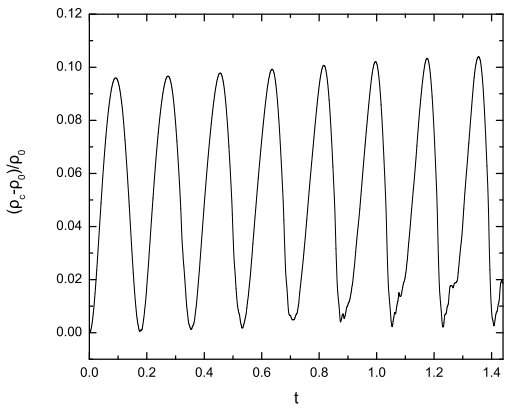


Figure 4.14: Plot of the relative difference of the central density versus time in ms.

The position x_v of the vacuum boundary is constantly changing, but it seems to follow a pattern. In fact the position seems to oscillate round a fixed value, which is the initial surface position. These oscillations have an increasing as the simulation evolves.

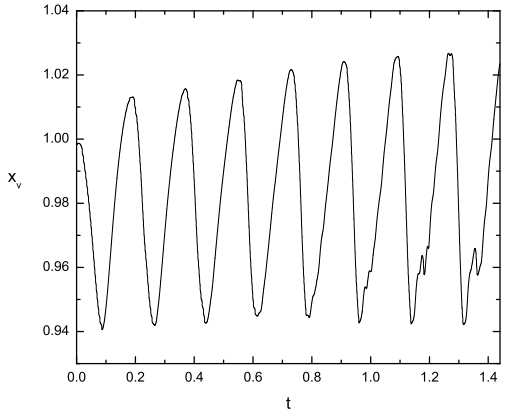


Figure 4.15: Position of the vacuum boundary x_v.

Uniformly rotating star with a vacuum region

In this section a uniformly rotating star with $\zeta = 0.4$ and n=1, is studied. This particular value of the rotating number was chosen because it is close to the maximum shedding limit. The uniformly rotating star is studied in the isentropic case. This means that the equation of state throughout the simulation is the isentropic one (equation (1.2)). This approach makes the energy equation (1.26) redundant since the value of the pressure is known throughout the simulation and the values of the energy can be defined at each time step through the equation

$$E = \frac{1}{2}\rho\left((u^r)^2 + (ru^\phi)^2\right) + \frac{K\rho^\Gamma}{\Gamma - 1} \tag{4.18}$$

The isentropic uniformly rotating star is surrounded by a vacuum region. The hydrodynamic equations that have to be solved are (1.23) to (1.25). The numerical methods are the same as in the previous static models. The CFL number is 0.1 and the grid points are 1600. The total time of the simulation is 1.6 ms.

The following plots concern the density, pressure, radial velocity and potential profiles for an isentropic, uniformly rotating star with $\zeta = 0.4$.

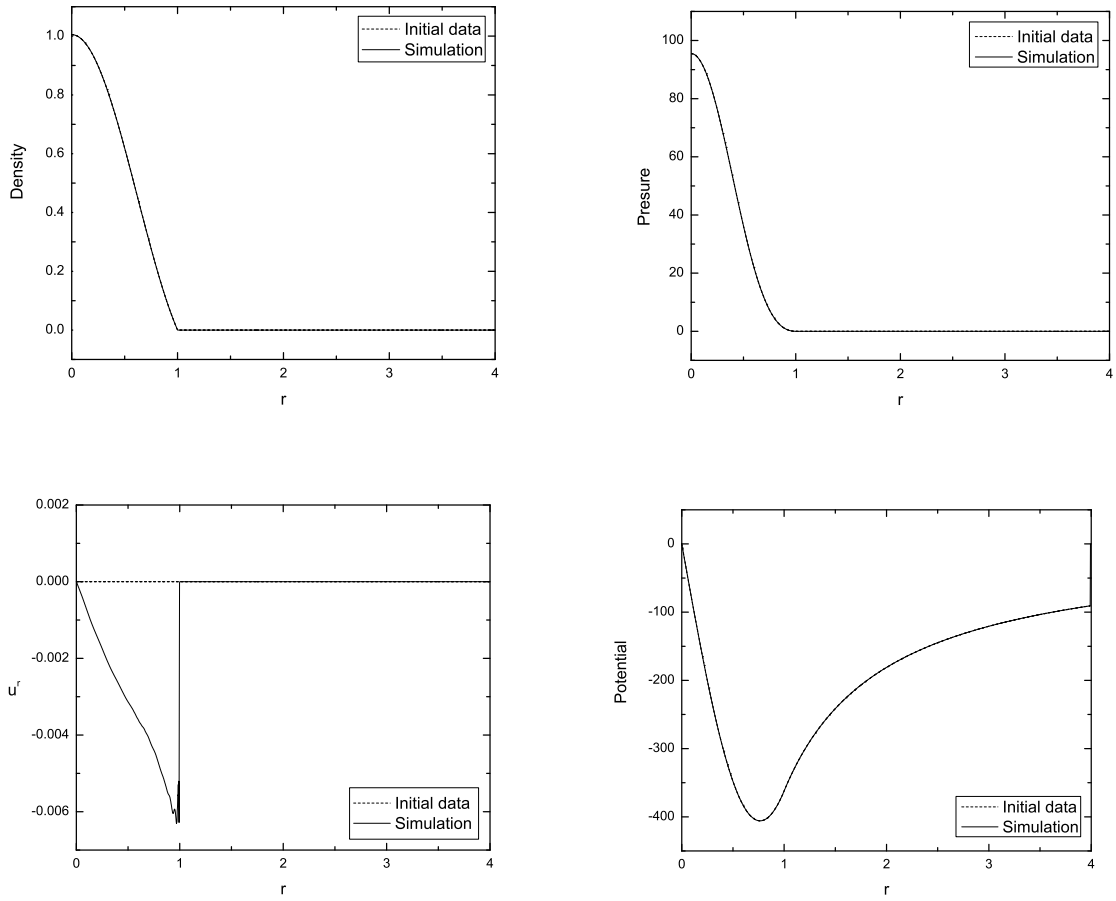


Figure 4.16: Density, pressure, radial velocity and potential profiles for a uniformly rotating star ($\zeta = 0.4$) with a surrounding vacuum region. The data of the simulation at t=1.6 ms are compared with the initial one.

One of the important issues in the case of a uniformly rotating star is to check whether the numerical methods manage to simulate adequately the angular velocity and the angular momentum profiles. The corresponding plots are presented along with details. The velocity u^{ϕ} profile is smoother at the surface of the star than the initial one.

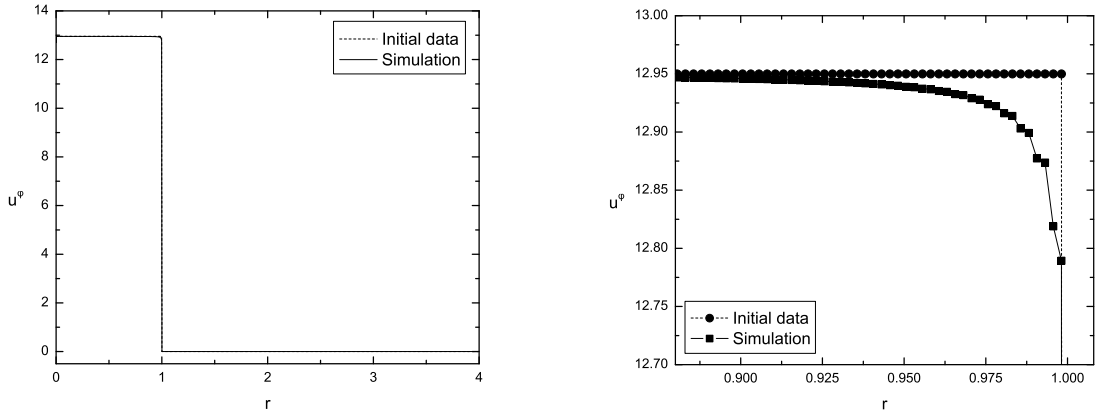


Figure 4.17: Angular velocity profile and detail at the surface of the star at t=1.6 ms.

The angular momentum (ρu^{ϕ}) profile seems to diverge in the center of the star, since the central momentum value differs from the initial one.

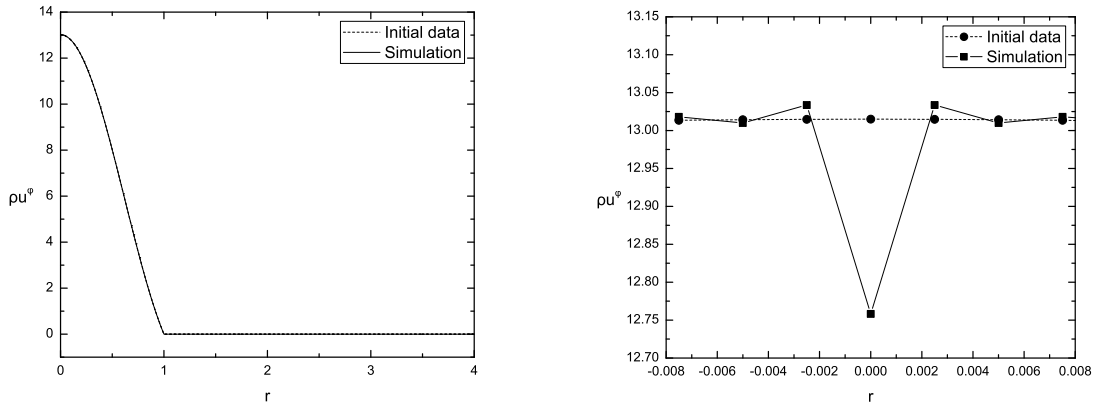


Figure 4.18: Angular momentum profile and detail at the center of the star at t=1.6 ms.

The following plot concerns the behavior of the central density. The relative difference of the density in the case of an isentropic, uniformly rotating star oscillates, but the oscillations differ in amplitude.

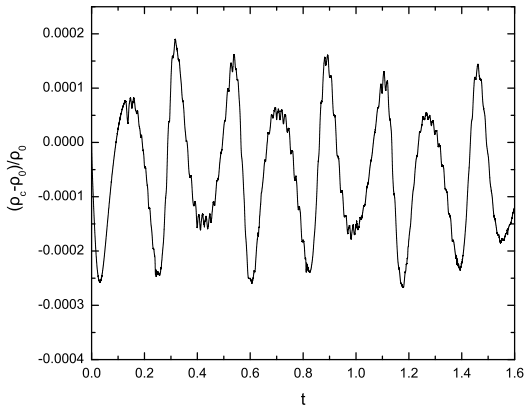


Figure 4.19: Plot of the relative difference of the central density versus time in ms.

The relative difference of the central angular momentum is plotted. It has to be noted that $\rho_0 u_0^\phi$ is the initial value of the momentum and $\rho_c u_c^\phi$ is its value at each time step. It is obvious that the central angular momentum decreases in time.

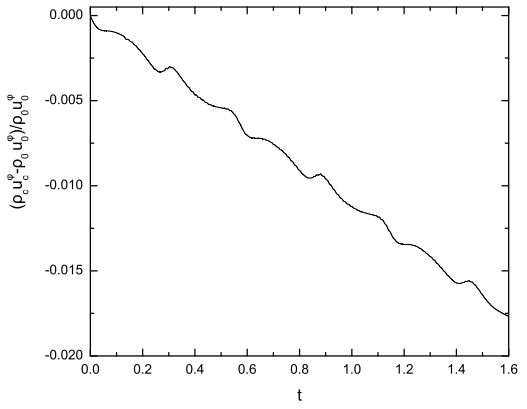


Figure 4.20: Plot of the relative difference of the central angular momentum versus time.

The position x_v of the vacuum boundary is constantly changing. It does not seem to follow a pattern, since the position is constantly changing and when the criteria for the vacuum region at the surface is applied, its value is reset to the previous one.

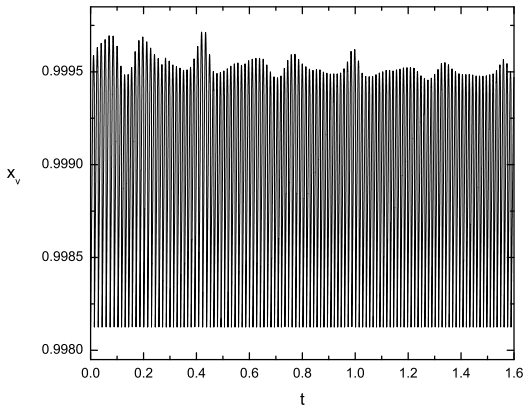


Figure 4.21: Plot of the relative difference of the central density versus time in ms.

Perturbed uniformly rotating star with a vacuum region

In this section an isentropic uniformly rotating star is studied, with $\zeta = 0.4$ and n=1. The main difference with the previous case is that a perturbation is introduced to the initial data. In fact the initial pressure is just a fraction of the one calculated, when solving the Lane-Emden equation. It is

$$p = \frac{95}{100} K \rho^{\Gamma}$$

The alteration of the initial data drives the star to a new equilibrium state, with a different central density than the initial one. The new central density is larger than the previous one. In this case the star is surrounded by a vacuum region, which is handled using the vacuum tracking algorithm and the criteria already described.
The CFL number is 0.1 and the grid points are 1600. The results of the simulation concern the values of the density, the pressure, the radial velocity, the potential, the angular velocity and the angular momentum. The profiles of these variables are plotted at t=1.6 ms. Moreover the relative differences of the central density and the central angular momentum are plotted. Finally the position of the vacuum boundary at each time step is presented.
It has to be noted that since the uniformly rotating star is studied in the isentropic case, no shock waves at the surface can be expected.

It is obvious that the density profile at t=1.6 ms differs significantly from the initial data. The central density is higher. The pressure behaves in a similar way. The differences in the potential profiles are not that obvious.

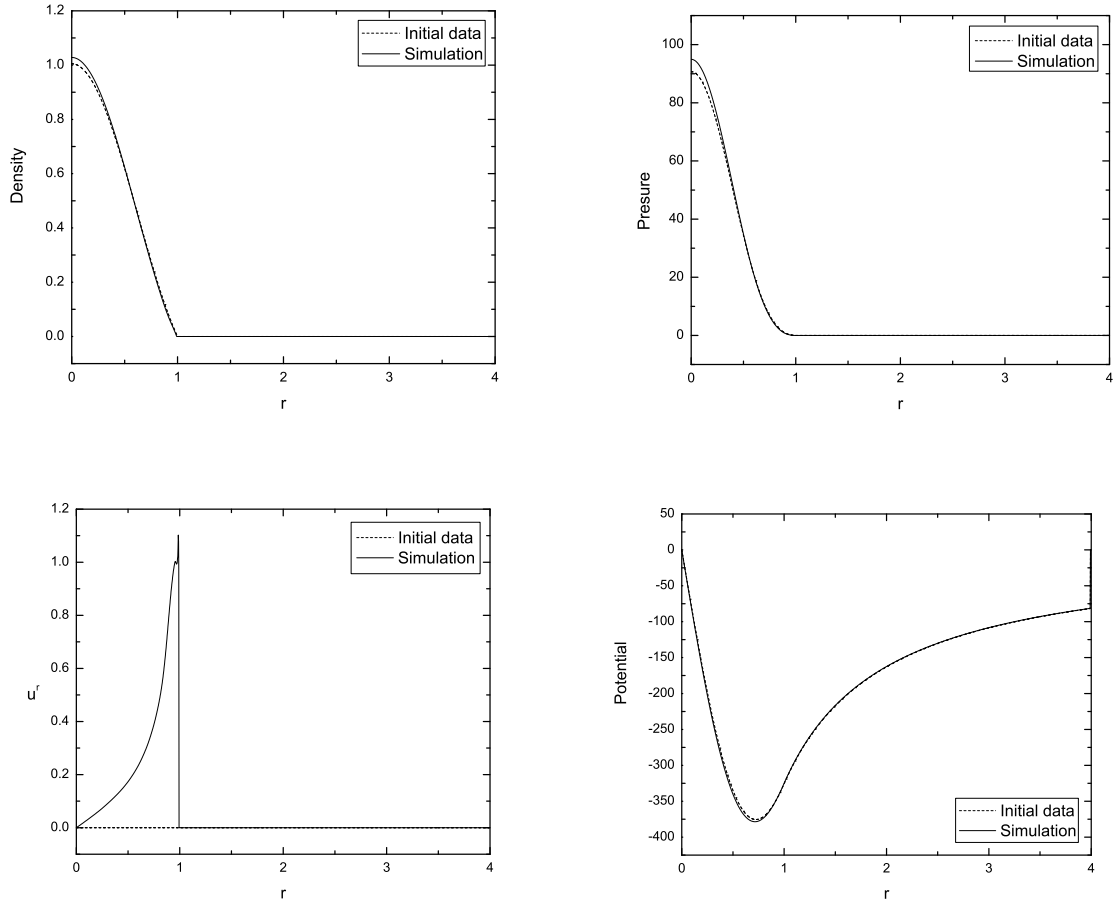


Figure 4.22: Density, pressure, radial velocity and potential profiles for a perturbed uniformly rotating star ($\zeta = 0.4$) with a surrounding vacuum region. The data of the simulation at t=1.6 ms are compared with the initial one.

The angular velocity and angular momentum profiles are presented at t=1.6 ms. The profiles of the variables after the dynamical evolution are different than the initial ones.

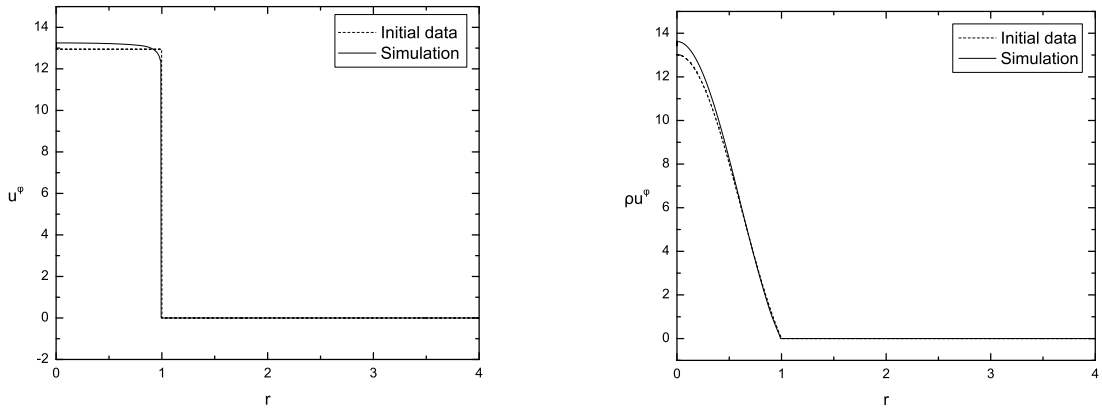


Figure 4.23: Angular velocity and angular momentum profiles at t=1.6 ms.

The plot of the relative difference of the density makes it clear that the star has been driven to a new equilibrium state. The central density increases up to a certain value and starts to oscillate round this value.

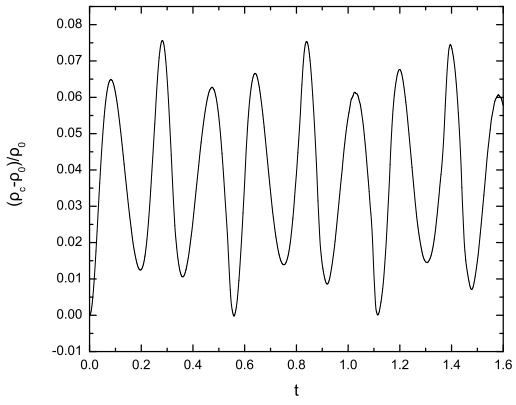


Figure 4.24: Plot of the relative difference of the central density versus time in ms.

The relative difference of the central angular momentum versus time is plotted. The central value oscillates around a certain point, which is different than the initial value and which states the new equilibrium configuration.

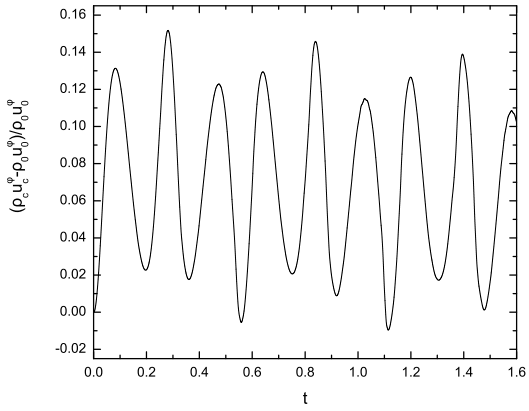


Figure 4.25: Plot of the relative difference of the central angular momentum versus time.

The following plot presents the position of the vacuum boundary as the simulation evolves. The surface of the star is not located at a fixed point and is different at each time step.

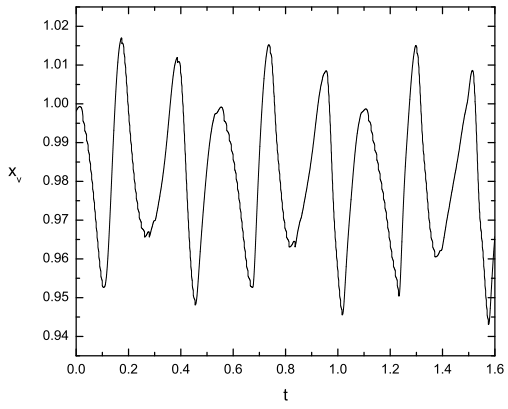


Figure 4.26: Position of the vacuum boundary x_v.

Differentially rotating star with vacuum

In this section the case of a differentially rotating star is studied. The polytropic index is n=1. The initial equilibrium configuration is obtained by solving the Lane-Emden equation for cylindrical rotating polytropes (equation (1.15)). The rotation number ζ is no longer a constant one, as for the uniformly rotating cylindrical stars, since the angular velocity follows a specific profile, which is given by equation (1.17),

$$\Omega(r) = \frac{\Omega_0}{2}\left[1 + cos\left(\frac{\pi r^2}{R^2}\right)\right]$$

According to this profile the maximum value of Ω is found at the center of the star and it decreases monotonically until its surface. The maximum value is defined by the parameter Ω_0, as it can be seen in the following plot

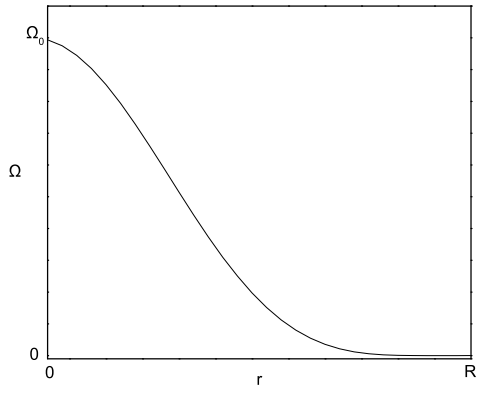


Figure 4.27: Profile of the angular velocity.

This angular velocity profile is used throughout the calculations. According to the Lane-Emden notation, equation (1.18) is introduced, where ζ_0 has to defined. It is $\zeta_0 = 0.1$. The numerical schemes are the same as in the static case and the isentropic uniformly rotating one. It has to be noted that the the differentially rotating star is not isentropic and the energy conservation equation (1.26) is solved. In fact the one-dimensional Euler equations (1.23) - (1.26) are solved, along with the Γ-law equation of state (1.28). The gravitational potential Φ is calculated at each time step, solving the Poisson equation (1.3).
The CFL number is 0.1 and the grid points are 1600. The results of the simulation concern the values of the density, the pressure, the radial velocity, the angular velocity and the angular momentum. The profiles of these variables are plotted at t=1.44 ms. Moreover the relative differences of the central density and the central angular momentum are plotted. Finally the position of the vacuum boundary at each time step is presented.

The plots that are presented concern the density, the pressure, the radial and angular velocity profiles at t=1.44 ms. These profiles are compared to the initial data and they seem to match in a fine way.

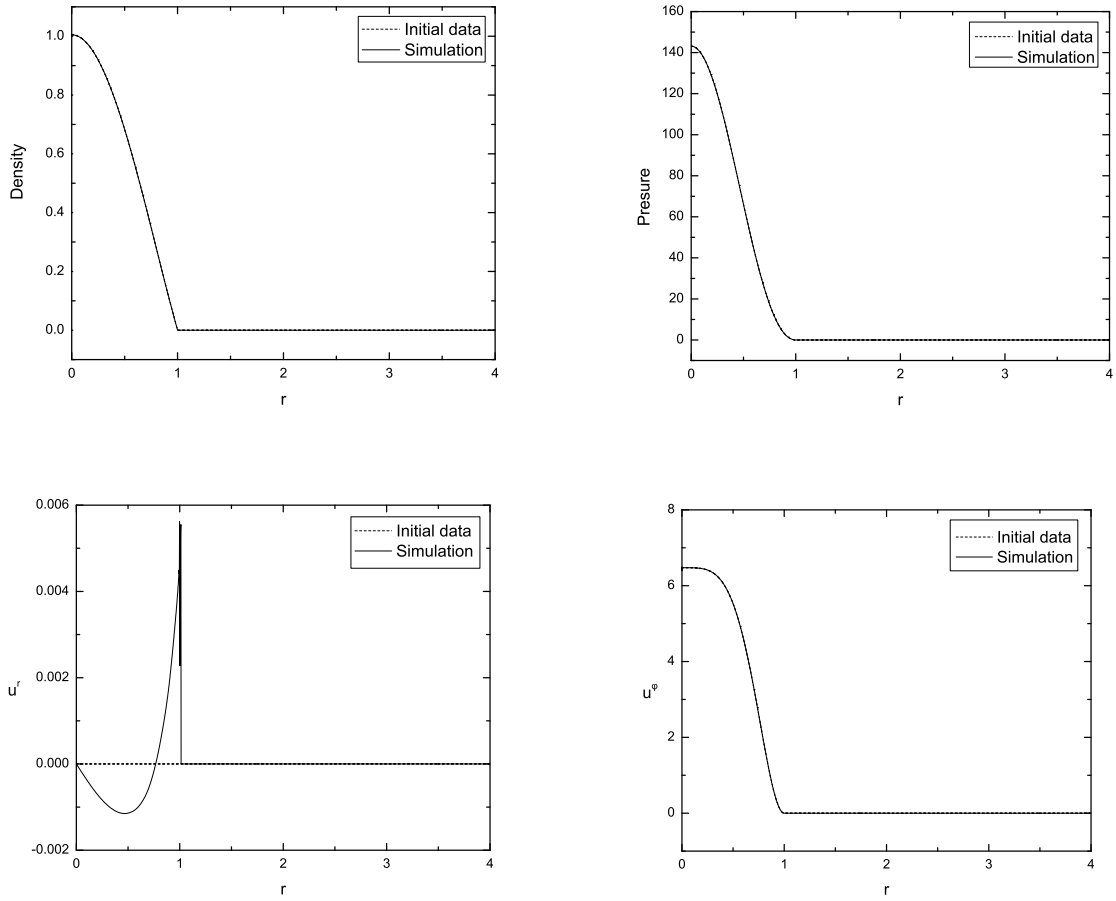


Figure 4.28: Density, pressure, radial and angular velocity profiles for a differentially rotating star with a surrounding vacuum region. The data of the simulation at t=1.44 ms are compared with the initial one.

The profile of the angular momentum ρu^{ϕ} is presented along with a detail at the center of the star. The numerical methods in this case can not describe precisely the behavior of the density at this region.

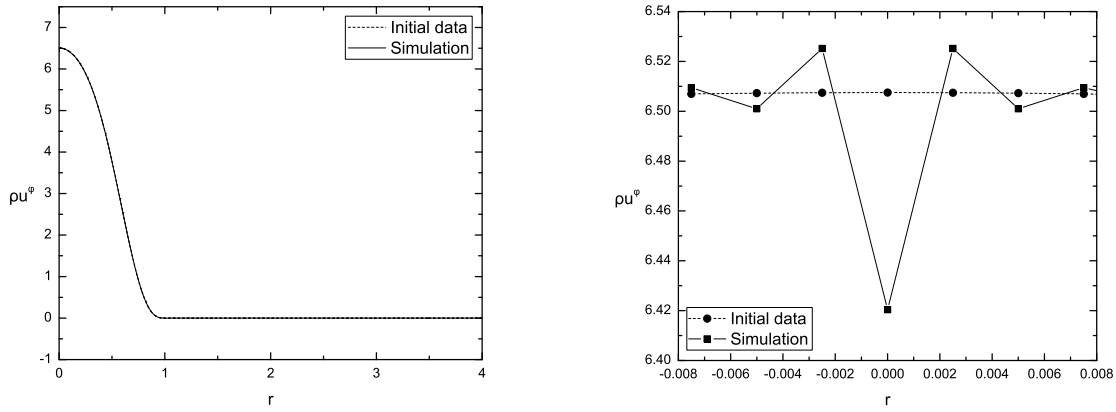


Figure 4.29: Angular momentum profile and detail at the surface of the star at t=1.44 ms.

The following plot concerns the relative difference of the density. The central density oscillates round the initial value and the amplitude of these oscillations is quite small.

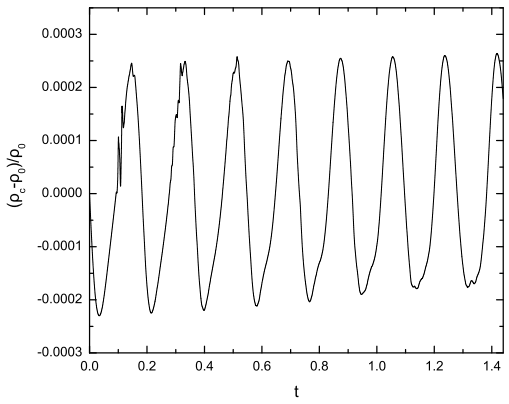


Figure 4.30: Plot of the relative difference of the central density versus time in ms.

In the next plot the relative difference of the central angular momentum is plotted. The central angular momentum decreases in time.

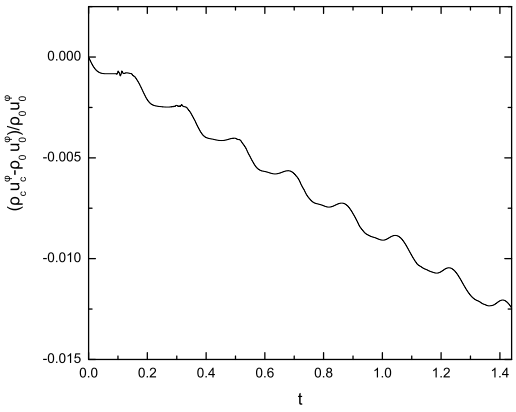


Figure 4.31: Plot of the relative difference of the central angular momentum versus time.

The next plot presents the position of the vacuum boundary at different time steps.

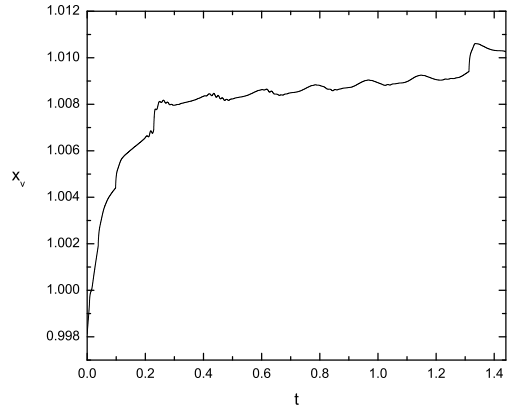


Figure 4.32: Position of the vacuum boundary x_v.

Perturbed differentially rotating star with vacuum

In this section a perturbed differentially rotating star with vacuum is studied. The polytropic index is n=1 and $\zeta = 0.1$. The main difference to the previous section is that the initial data are altered before they are used for the dynamical evolution. A fraction of the pressure is used as initial data, according to the equation (4.17). The initial data do not describe anymore an equilibrium state and therefore the star, as it evolves during the simulation, is driven to a new equilibrium. The CFL number is 0.1 and the grid points are 1600.

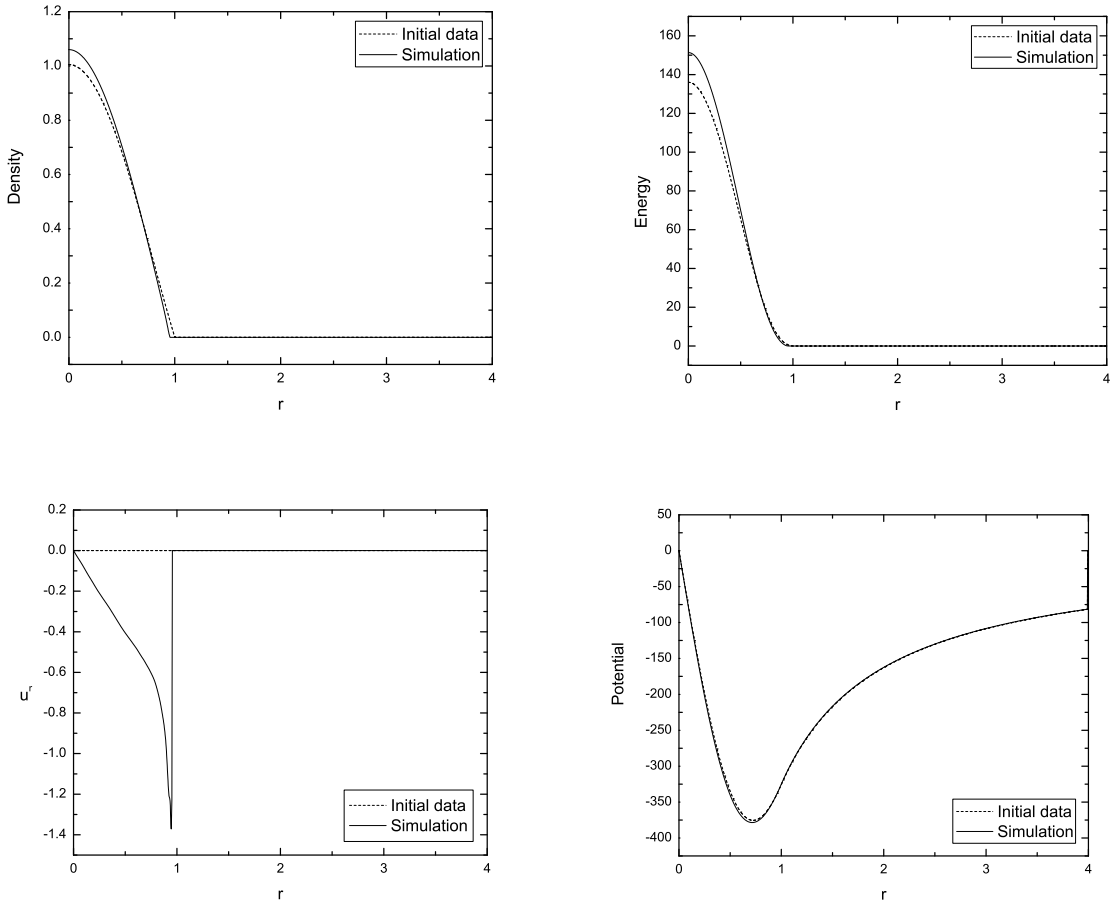


Figure 4.33: Density, energy, radial velocity and potential profiles for a perturbed differentially rotating star with a surrounding vacuum region. The data of the simulation at t=1.44 ms are compared with the initial one.

The angular velocity and momentum profiles are presented at t=1.44 ms.

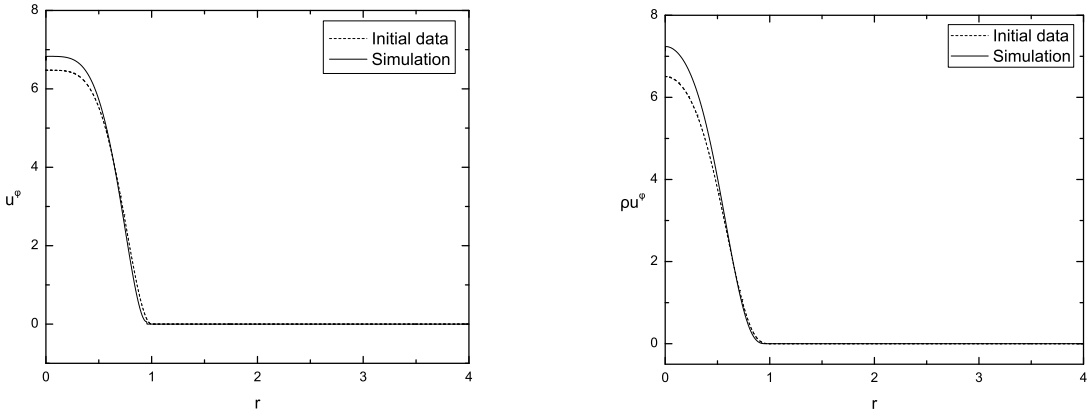


Figure 4.34: Angular velocity and angular momentum profiles at t=1.6 ms.

The following plot presents the relative difference of the central density at each time step.

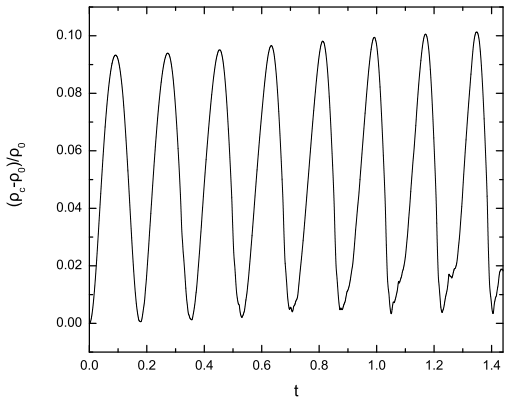


Figure 4.35: Plot of the relative difference of the central density versus time in ms.

The following plot presents the relative difference of the central angular momentum at each time step.

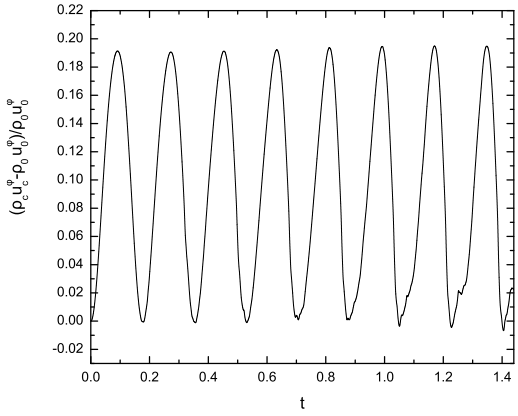


Figure 4.36: Plot of the relative difference of the central angular momentum versus time.

The next plot presents the position of the vacuum boundary at different time steps.

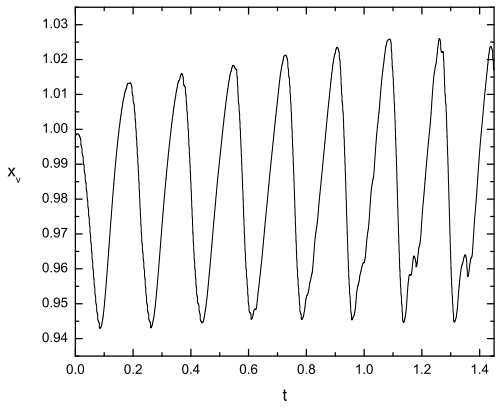


Figure 4.37: Position of the vacuum boundary x_v.

4.5.2 MHD stars

The evolution of MHD stars is governed by the one-dimensional MHD equations, namely equations (1.37) to (1.38). The effects of the Newtonian gravitational field are taken into account. The case of a static MHD star surrounded by an atmosphere is studied. The polytropic index is n=1. Since the star is not rotating, only the radial component B^r of the magnetic field is present in the solution. The magnetic field satisfies Maxwell's constraint equation

$$\nabla \mathbf{B} = 0 \quad \Rightarrow \quad \frac{1}{r}\frac{\partial}{\partial r}\big(rB^r\big) = 0 \quad \Rightarrow \quad \frac{B^r}{r} + \frac{\partial B^r}{\partial r} = 0 \quad \Rightarrow$$

$$B^r = \frac{B_0}{r} \tag{4.19}$$

where B_0 is the value of the field at the surface of the star at t=0. It is obvious that the magnetic field exhibits a singularity at the point r=0. The value of the radial component B^r at the center of the star for the initial data is found by using extrapolation. During the dynamical evolution this value is kept stable.
The magnetic field also satisfies the flux-freezing equation

$$\frac{\partial \mathbf{B}}{\partial t} = \nabla \times \big(\mathbf{u} \times \mathbf{B}\big) \tag{4.20}$$

The radial component of the magnetic field is independent of time, since according to the previous equation it is

$$\frac{\partial B^r}{\partial t} = \frac{1}{r}\frac{\partial\big[\big(\mathbf{u}\times\mathbf{B}\big)_z\big]}{\partial \phi} - \frac{\partial\big[\big(\mathbf{u}\times\mathbf{B}\big)_\phi\big]}{\partial z} = 0$$

The MHD-HLLC Riemann solver is used for the numerical calculations, along with the splitting method for the source terms. The Newtonian gravitational potential is calculated at each time step. An artificial atmosphere is present in the initial configuration. The atmosphere's density is equal to that of the star's surface and its pressure is given by the polytropic equation of state (1.2). If the density of the atmosphere increases and exceeds a multiplicate of its initial value, then it is reset to that initial value. This procedure takes place because the precision of the numerical methods does not exceed numbers smaller than 10^{-7} and otherwise the numerical code would crash.
The CFL number is 0.1 and the number of the grid points is 1600. The plots, which are presented concern the gas pressure, the radial velocity, the radial magnetic field and the potential profiles at t=1.44 ms. Moreover the density profile and a detail of the star's center are plotted. Finally the relative difference of the central density at each time step is plotted.

The following plots concern the radial component of the magnetic field, the gas pressure, the radial velocity and the potential profiles for a static MHD star, with an artificial atmosphere, at t=1.44 ms.

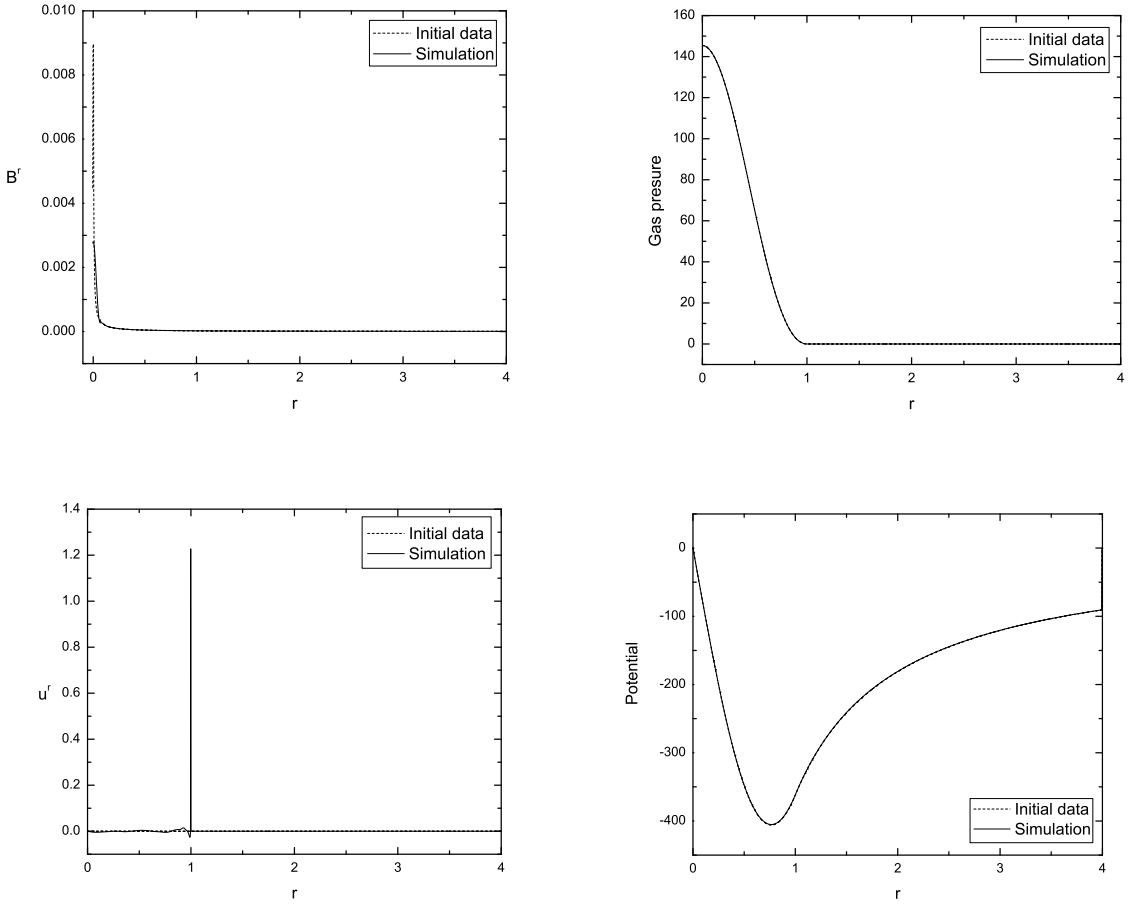


Figure 4.38: Radial magnetic field component, gas pressure, radial velocity and potential profiles for a static MHD star with a surrounding atmosphere. The data of the simulation at t=1.44 ms are compared with the initial one.

The density profile is plotted and a detail at the center of the star is presented.

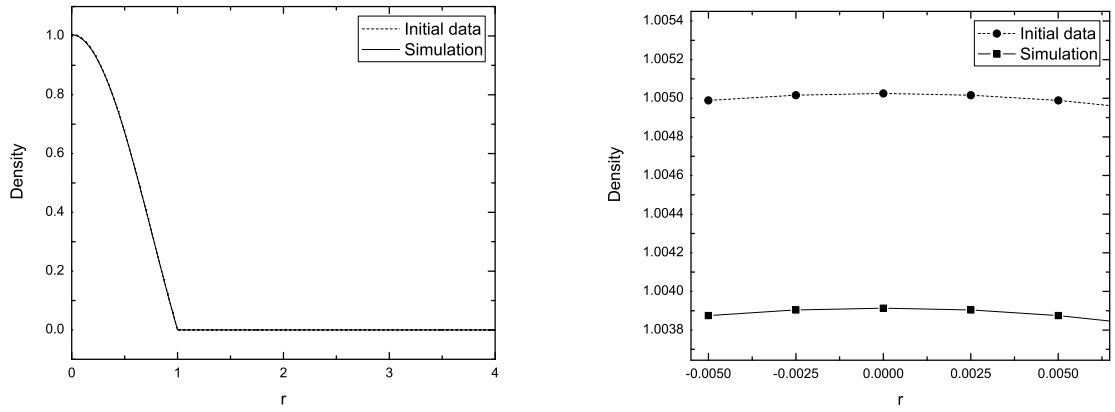


Figure 4.39: Density profile and detail at the center of the star at t=1.44 ms.

The next plot concerns the relative difference of the central density.

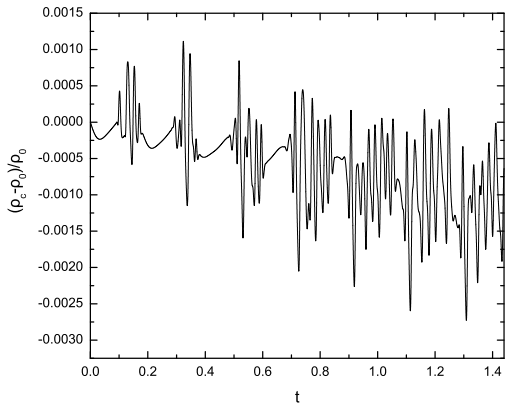


Figure 4.40: Plot of the relative difference of the central density versus time in ms.

Bibliography

[1] Batten, P., Clarke, N., Lambert, C., Causon, D., SIAM J. Sci. Comput. 18, 153 (1997)

[2] Collela, P., Woodward, P. R., J. Comp. Phys., 54, 174 (1984)

[3] Cook, J. N., Shapiro, S. L., Stephens, B. C., Astrophys. J., 599, 1272 (2003)

[4] Donat, R., Font, J. A., Ibáñez, J. M., Marquina, A., J. Comp.Phys., 146, 58 (1998)

[5] Donat, R., Marquina, A., J. Comp. Phys., 125, 42 (1996)

[6] Einfeldt, B., Munz, C. D., Roe, P. L., Sögreen, B., J. Comp. Phys., 92, 273 (1991)

[7] Ferziger, J. H., Peric; M. *Computational Methods for Fluid Dynamics*, Springer (1999)

[8] Flannery, B. P., Press, W. H., Teukolsky, S. A., Vetterling, W. T., *Numerical Recipes in C*, Cambridge Press (2002)

[9] Font, J. A., Martí, J. M., Marquina, A., Müller, E., Astrophys. J., 479, 151 (1997)

[10] Godunov, S. K., Mat. Sb., 47, 357 (1959)

[11] Harten, A., Osher, S., SIAM J. Numer. Anal. 24, 279 (1987)

[12] Jackson, J. D. *Classical Electrodynamics*, John Wiley and Sons, Inc. (1999)

[13] Kokkotas, K. D., Stergioulas, N., in *Proceedings of the 5th International Workshop "New Worlds in Astroparticle Physics"*, Faro, Portugal, gr-qc/0506083 (2005)

[14] Kurganov, A., Levy, D., SIAM J. Sci. Comput., 22, 1461 (2000)

[15] Laney, C.B. *Computational Gasdynamics*, Cambridge Press (1998)

[16] Li, S., J. Comp. Phys., 203, 344 (2003)

[17] Liu, Y. T., Shapiro, S. L., Phys. Rev. D, 69, 044009 (2004)

[18] Martí, J. M., Müller, E., J. Comp. Phys., 123, 1 (1996)

[19] Munz, C. D., Mathematical Methods in Applied Sciences, 17, 597 (1994)

[20] Munz, C. D., Schneider, R., Gerlinger, O., *Numerical Methods for the Navier-Stokes Equations*, Vieweg-Verlag (1994)

[21] Nakamura, S. *Applied Numerical Methods in C*, PTR Prentice Hall (1993)

[22] Roe, P. L., J. Comput. Phys., 43, 357 (1981)

[23] Ryu, D., Jones, T. W., Astrophys. J., 442, 228 (1995)

[24] Shapiro, S. L., Astrophys. J., 544, 397 (2000)

[25] Stergioulas, N., Living Rev. Relativity, 6, 3 (2003)

[26] Stergioulas, N., Apostolatos, T. A., Font, J. A., Mon. Not. R. Astron. Soc., 352, 1089 (2004)

[27] Toro, E.F. *Riemann Solvers and Numerical Methods For Fluid Dynamics*, Springer (1999)

[28] Toro, E. F., Spruce, M., Speares, W., Shock Waves, 4, 25 (1994)