EXTENSION OF THE ENTROPY VISCOSITY METHOD TO THE MULTI-D EULER EQUATIONS AND THE SEVEN-EQUATION MODEL.

A Dissertation

by

MARC-OLIVIER GERARD DELCHINI

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Co-Chairs of Committee,	Jean C. Ragusa
	Jean Luc Guermond
Committee Members,	Jim Morel
	Yassin Hassan
	Ray Berry
Head of Department,	Yassin Hassan

December 2014

Major Subject: Nuclear Engineering Department

ABSTRACT

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus quam, condimentum quis bibendum eu, sollicitudin eget lacus. Praesent non sodales odio. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nulla ac luctus sapien. Morbi cursus sapien eget lorem fermentum hendrerit. Nam ac erat dui, in cursus velit. Vivamus hendrerit porttitor nisi, ut porttitor lorem volutpat eget. In ligula ligula, euismod ut condimentum sit amet, pulvinar sit amet diam. Pellentesque interdum, ipsum ullamcorper consequat dignissim, sem arcu egestas mauris, vitae interdum sem tortor ut ante. Nunc blandit laoreet nisi, non rutrum lorem hendrerit quis. Cras nunc diam, convallis et feugiat at, auctor id libero. Nunc facilisis massa eu eros imperdiet vestibulum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Donec non velit vitae tortor blandit semper.

Etiam vitae dolor nulla. Ut eros odio, rhoncus eget placerat vitae, elementum ac ante. Proin vitae odio eu nisl pharetra mattis. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Phasellus fermentum lacus consectetur neque consequat ullamcorper. Cras blandit urna non dui consequat molestie. Curabitur viverra nibh at nisi semper faucibus. Nam egestas mauris a enim dignissim nec consectetur tortor rutrum. Mauris at nisi in est luctus congue ut mattis est. Ut pretium, mi quis elementum cursus, ante eros suscipit ligula, ut porttitor elit leo sed turpis. Nam sed dui ligula.

DEDICATION

This is an optional page. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus quam, condimentum quis bibendum eu, sollicitudin eget lacus. Praesent non sodales odio. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nulla ac luctus sapien. Morbi cursus sapien eget lorem fermentum hendrerit. Nam ac erat dui, in cursus velit. Vivamus hendrerit porttitor nisi, ut porttitor lorem volutpat eget. In ligula ligula, euismod ut condimentum sit amet, pulvinar sit amet diam. Pellentesque interdum, ipsum ullamcorper consequat dignissim, sem arcu egestas mauris, vitae interdum sem tortor ut ante. Nunc blandit laoreet nisi, non rutrum lorem hendrerit quis. Cras nunc diam, convallis et feugiat at, auctor id libero. Nunc facilisis massa eu eros imperdiet vestibulum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Donec non velit vitae tortor blandit semper.

Etiam vitae dolor nulla. Ut eros odio, rhoncus eget placerat vitae, elementum ac ante. Proin vitae odio eu nisl pharetra mattis. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Phasellus fermentum lacus consectetur neque consequat ullamcorper. Cras blandit urna non dui consequat molestie. Curabitur viverra nibh at nisi semper faucibus. Nam egestas mauris a enim dignissim nec consectetur tortor rutrum. Mauris at nisi in est luctus congue ut mattis est. Ut pretium, mi quis elementum cursus, ante eros suscipit ligula, ut porttitor elit leo sed turpis. Nam sed dui ligula.

ACKNOWLEDGEMENTS

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Integer lectus quam, condimentum quis bibendum eu, sollicitudin eget lacus. Praesent non sodales odio. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nulla ac luctus sapien. Morbi cursus sapien eget lorem fermentum hendrerit. Nam ac erat dui, in cursus velit. Vivamus hendrerit porttitor nisi, ut porttitor lorem volutpat eget. In ligula ligula, euismod ut condimentum sit amet, pulvinar sit amet diam. Pellentesque interdum, ipsum ullamcorper consequat dignissim, sem arcu egestas mauris, vitae interdum sem tortor ut ante. Nunc blandit laoreet nisi, non rutrum lorem hendrerit quis. Cras nunc diam, convallis et feugiat at, auctor id libero. Nunc facilisis massa eu eros imperdiet vestibulum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Donec non velit vitae tortor blandit semper.

Etiam vitae dolor nulla. Ut eros odio, rhoncus eget placerat vitae, elementum ac ante. Proin vitae odio eu nisl pharetra mattis. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Phasellus fermentum lacus consectetur neque consequat ullamcorper. Cras blandit urna non dui consequat molestie. Curabitur viverra nibh at nisi semper faucibus. Nam egestas mauris a enim dignissim nec consectetur tortor rutrum. Mauris at nisi in est luctus congue ut mattis est. Ut pretium, mi quis elementum cursus, ante eros suscipit ligula, ut porttitor elit leo sed turpis. Nam sed dui ligula.

NOMENCLATURE

P	Pressure
T	Temperature
$\vec{u}$	vector velocity
ρ	Density
E	specific total energy
e	specific internal energy
P_I	interfacial pressure
$\vec{u}_I$	interfacial velocity
μ	viscosity coefficient
κ	viscosity coefficient
α	void fraction
∂_t	partial time derivative
eos	equation of state

This page is optional.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE Page

LIST OF TABLES

TABLE Page

1. INTRODUCTION:

Hyperbolic systems of equations are encountered in various engineering fields (extraction of oil, turbine technology, nuclear reactors, etc ...). Improving numerical solution techniques for such equations are an ongoing topic of research. This is obviously the case for fluid equations. Being able to accurately solve and predict the behavior of a fluid in a turbine or in a reactor, for example, can lead to a safe decrease in conservative safety margins, which translates into a decrease in production cost. Thus, we can see the importance of having a good understanding of the mathematical theory behind these wave-dominated systems of equations, and also, the importance of developing robust and accurate numerical methods.

A large number of theoretical studies has shown the role played by characteristic equations and the corresponding eigenvalues on how and at what speed the physical information propagates: physical shocks or discontinuities can form, leading to unphysical instabilities and oscillations that pollute the numerical solution due to entropy production [55]. Thus, a question rises: how to accurately resolve shocks and conserve the physical solution at the same time? A lot of work is available in the literature and include Riemann solvers, Godunov-type fluxes, flux limiter and artificial viscosity methods. Toro's book [55] gives a good overview of the theory related to hyperbolic systems of equations and focuses on Riemann solvers and Godunov-type fluxes that can be used with discontinuous spatial discretizations: finite volume (FV) and discontinuous Galerkin finite element method (DGFEM). Flux limiters [12, 15] can achieve high-order accuracy with DGFEM [14] but suffer from some drawbacks: difficulties were found to reach steady-state solution when using time-stepping schemes and generalization to unstructured grids is not obvious [11]. The artificial viscosity method was first introduced by Neumann and Richtmeyer [58] but was found over-dissipative and, thus, abandoned. It is only, later with the development of high-order schemes and because of their simple implementation that they have regained interest: Lapidus [35, 20] developed a high-order viscosity method by making the viscosity coefficient proportional to the gradient of the velocity in 1-D. Lohner et al. [41] extended this concept to multi-dimension by introducing a vector that will measure the direction of maximum change in the absolute value of the velocity norm, so that shear layers are not smeared. Pressure-based viscosities were also studied [40] where the viscosity coefficient is set proportional to the Laplace of pressure, allowing to detect curvature changes in the pressure profile. Since pressure is nearly constant but in shocks, it is a good indicator of the presence of a shock. Recently, Reisner et al. [51] introduced the C-method for the compressible Euler equations with artificial dissipative terms: instead of computing the viscosity coefficient on the fly as for Lapidus and pressure-based methods, a partial differential equation (PDE) is added to the original system of equations. This additional PDE allows to solve for the viscosity coefficient and contains a source term that is function of the gradient of velocity. The method seems to yield good results in 1-D for a wide range of tests. Guermond et .al [28, 29, 22] proposed an entropy-based viscosity method for conservative hyperbolic system of equations: artificial dissipative terms are added to the system of equations with a viscosity coefficient modulated by the entropy production that is known to be large in shocks and small everywhere else. The method was successfully applied in various schemes [29, 22, 57] and showed high-order convergence with smooth solutions. Results using the ideal gas equation of state were run for 1-D Sod tube and showed good agreement with the exact solutions. 2-D tests were also performed on unstructured grids and behaved very satisfactorily [57]. The method is fairly simple to implement and is consistent with the entropy inequality.

The objective of this research is to solve hyperbolic system of equations using a continuous Galerkin finite element method (CGFEM) and an implicit temporal discretization under the Moose framework [16]. We are particularly interested in simulating the flow behavior occurring in nuclear reactors. The set of equations that will be considered are the multi-D Euler equations with variable area [55] and the multi-D seven equations model for two-phase fluids [49]. These systems of equations are well defined in a sense that they possess real eigenvalues. To numerically solve these equations, we need to rely on a numerical method that will allow us to resolve shocks and discontinuity that may form. More importantly, we need a method that is accurate for a wide range of Mach numbers and is not restricted to any type of equation of state. These previous requirements can be hard to fulfill. Numerical methods are often tested with the ideal gas equation of state which can not be used for liquid phase. The difficulty rises once a numerical method used for resolving shocks, is required to work for low Mach flows that are isentropic: in this particular case, a compressible model is used to simulate a flow in the incompressible limit. Recent publications [24, 56] highlighted the difficulties related to such a choice: asymptotic studies have shown that some of the numerical methods become ill-scaled in the low Mach limit, making the numerical solution unphysical. For example, the Roe scheme requires a fix in the low Mach limit while conserving its accuracy when shocks occur [39].

We propose to use the entropy-based viscosity method introduced by Guermond et al. to solve for the hyperbolic systems previously enumerated since the method has shown good results when used for solving the multi-D Euler equations with various discretization schemes. More importantly, it is simple to implement, can be used with unstructured grids, and its dissipative terms are consistent with the entropy minimum principle and proven valid for any equation of state under certain conditions [23]. However, a few questions remains: the low Mach limit has not been investigated and the current definition requires an analytical expression of the entropy which can be difficult to obtain for some equation of state. These two issues will be addressed. A particular attention will be brought to the low Mach problem and the available literature related to the asymptotic limit of the Navier-Stokes [45] and Euler equations [24, 56] should be of great insight in order to understand how the dissipative terms behave.

It is also proposed to investigate how the entropy viscosity method can be applied to the multi-D radiation-hydrodynamic equations [43]. These equations are known to develop solutions with shocks [6]. They consist of coupling the multi-D Euler equations with a radiation-diffusion equation through some source terms. Most of the current solvers are based on the study of the hyperbolic terms in order to derive a Riemann-type solver [50]. Flux-limiter technique [30] are also used and suffer from the same drawbacks as for the pure multi-D Euler equations. Therefore, it will be valuable to assess how the entropy viscosity method can adapt to this multi-physics system, and if successful, will offer an alternative to the current available numerical methods.

Thus, the proposal is organized as follows: Section 1 will recall the main feature of the current version of the entropy-based viscosity method. Section 2 will introduce the system of equations under considerations. A short description of the equations and their properties will be given. In Section 3, the problem formulation is discussed for each system of equations and suggestions of test cases are given in order to test the numerical method.

2. HYPERBOLIC CONSERVATION LAWS

In this chapter, some key properties of the hyperbolic conservation laws are recalled. The objective is to introduce the reader to the notion of shock, weak solution and entropy minimum principle by studying a simple hyperbolic scalar equation through a linear progression. First, the mathematical properties of the hyperbolic scalar equation are studied and include the derivation of the eigenvalue and the characteristic equation. Then, it is explained how shock are formed which will bring us to show that the solution is non-unique. The reader will be introduced to the notion of weak solution. Lastly, it will be explained how the entropy condition is used to ensure uniqueness of the weak solution and more importantly, convergence of the numerical solution to the physical one. In the last section of this chapter, the notions introduced for the hyperbolic scalar equation are generalized to the hyperbolic systems of equation.

2.1 Hyperbolic scalar equations:

The study of a hyperbolic scalar equation is detailed in order to provide the reader with a better understanding of its mathematical properties that are useful to the comprehension of shock formation among others.

2.1.1 Eigenvalue and characteristic curves:

For academic purpose, we consider a simple hyperbolic scalar equation with initial conditions to form what is called an *Initial Boundary Value Problem* (IBVP) as shown in Eq. (2.1). We define a computational domain Ω of dimension d, bounded by the boundary Γ of dimension $d-1$. Each variable is assumed to be a function of space,

$\boldsymbol{r} \in R^d$, and time $t \in R_+$.

$$\begin{cases} \partial_t u(\boldsymbol{r},t) + \vec{\nabla}\cdot \boldsymbol{f}(u) = 0, \ (\boldsymbol{r},t) \in R^d \times R_+ \\ u(\boldsymbol{r},0) = u(\boldsymbol{r}_0) \end{cases} \tag{2.1}$$

where u and $\boldsymbol{f}(u)$ are the solution and the inviscid flux, respectively. The inviscid flux $\boldsymbol{f}(u)$ is assumed to be a differential function of the solution u. Two definitions of the inviscid flux will be considered in this chapter in order to illustrate the differences between linear and non-linear hyperbolic scalar systems: a linear flux $\boldsymbol{f}_1$ and a non-linear flux $\boldsymbol{f}_2$, as follows:

$$\partial_t u(\boldsymbol{r},t) + \vec{\nabla}\cdot \boldsymbol{f}_1(u) = \partial_t u(\boldsymbol{r},t) + \vec{\nabla}\cdot(au) = 0 \tag{2.2a}$$

$$\partial_t u(\boldsymbol{r},t) + \vec{\nabla}\cdot \boldsymbol{f}_2(u) = \partial_t u(\boldsymbol{r},t) + \vec{\nabla}\cdot\left(\frac{u^2}{2}\right) = 0 \tag{2.2b}$$

Eq. (2.2a) and Eq. (2.2b) are respectively known as the linear advection and Burger's equations. They have been widely studied in the literature and are well understood (REFS).

The eigenvalue, denoted by λ, of the hyperbolic equation is obtained by looking at the jacobian matrix of the inviscid flux, $\boldsymbol{f}(u)$, with respect to the solution u, and corresponds to the wave propagation speed. When considering the fluxes $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$, it is found that their eigenvalues are $\lambda_1 = a$ and $\lambda_2 = u$, respectively. For the linear advection equation, the wave speed is a constant through the computational domain. On the other hand, the wave speed is a function of space and time for the Burger's equation, since equal to the solution itself.

Once the eigenvalues are determined, the next step consists of deriving the characteristic equation and the characteristic curves. For simplicity purpose, we limit ourself to the $x - t$ plane. Under this assumption, characteristic curves are defined as curves $x = x(t)$ in the plane $x - t$ along with the PDE becomes an ODE [55]. To determine the characteristic curves, Eq. (2.1) is recast as a function of the eigenvalue, λ, by using the chain rule as shown in Fig. 2.3.

$$\begin{aligned}
&\partial_t u(\boldsymbol{r}, t) + \frac{df}{du}\partial_x u = 0 \\
&\partial_t u(\boldsymbol{r}, t) + \lambda\partial_x u = 0 \\
&\partial_t u(\boldsymbol{r}, t) + \frac{dx}{dt}\partial_x u = 0 \text{ along } \frac{dx}{dt} = f'(u) = \lambda
\end{aligned} \tag{2.3}$$

Eq. (2.3) represents the rate of change of the solution u along the curve $\frac{dx}{dt} = f'(u) = \lambda$. It tells us that the solution u is constant along the curve $\frac{dx}{dt} = \lambda$ since its rate of change is zero. The eigenvalue is the slope of the characteristic curve and is referred to as the characteristic speed. For a given characteristic curve, the characteristic speed is a constant, since the solution u is constant as well, and given by the initial condition, $f'(u(x, 0)) = f'(u_0)$ which allows us to integrate in order to obtain an analytical expression for $x(t)$:

$$\begin{aligned}
\frac{dx}{dt} &= f'(u_0) \\
\Leftrightarrow x(t) &= x_0 + f'(u_0)t
\end{aligned} \tag{2.4}$$

where we set $x(t = 0) = x_0$ that can be seen as the initial position of a particle traveling along the characteristic curve. It is current to represent the characteristic curves in a $t-x$ plane and example will be given for the linear advection and Burger's equation. Eq. (2.4) informs us on the position x of a particle carrying the initial value u_0 for each time t. Now, assuming that the initial value of the solution is $u_0(x_0)$ along the characteristic curve passing through the point x_0 given by Eq. (2.4), the solution

$u(x,t)$ at position x and time t can be expressed as follows:

$$u(x,t) = u_0(x_0) = u_0(x - f'(u_0)t) \tag{2.5}$$

Eq. (2.5) can be seen as an analytical solution of the hyperbolic scalar equation (Eq. (2.1)). It is also understood that the derivative of the flux that corresponds to the eigenvalue of the system, has profound consequences on the behavior of the solution as explained in Section 2.1.2.

2.1.2 Formation of shocks and vanishing viscosity equation/solution

Hyperbolic scalar equations are known to develop shocks even from a smooth initial profile. This section aims at detailing how shocks form based on the mathematical properties introduced in Section 2.1.1 and the two examples of Eq. (2.2a) and Eq. (2.2b), i.e., the 1-D linear advection and Burger's equations.

When considering the 1-D linear advection equation recalled in Eq. (2.2a) with the flux $f_1(u) = au$, the eigenvalue is found equal to $\lambda_1 = a$ and constant. Thus, the slope of the characteristic curve remains constant and each particle travels at the same velocity through the computational domain. In other word, the initial profile $u_0(x)$ of the solution is simply translated at speed a to the right if $a \geq 0$ and to the left if $a \leq 0$. Obviously, if $a = 0$, the flux is also null and the solution remains static. A representation of the characteristic curve for the linear advection equation, Eq. (2.2a), is given in Fig. 2.1 in a $t - x$ plane: all of the characteristic curves are parallel since their slope is given by the eigenvalue that is constant for the linear advection equation.

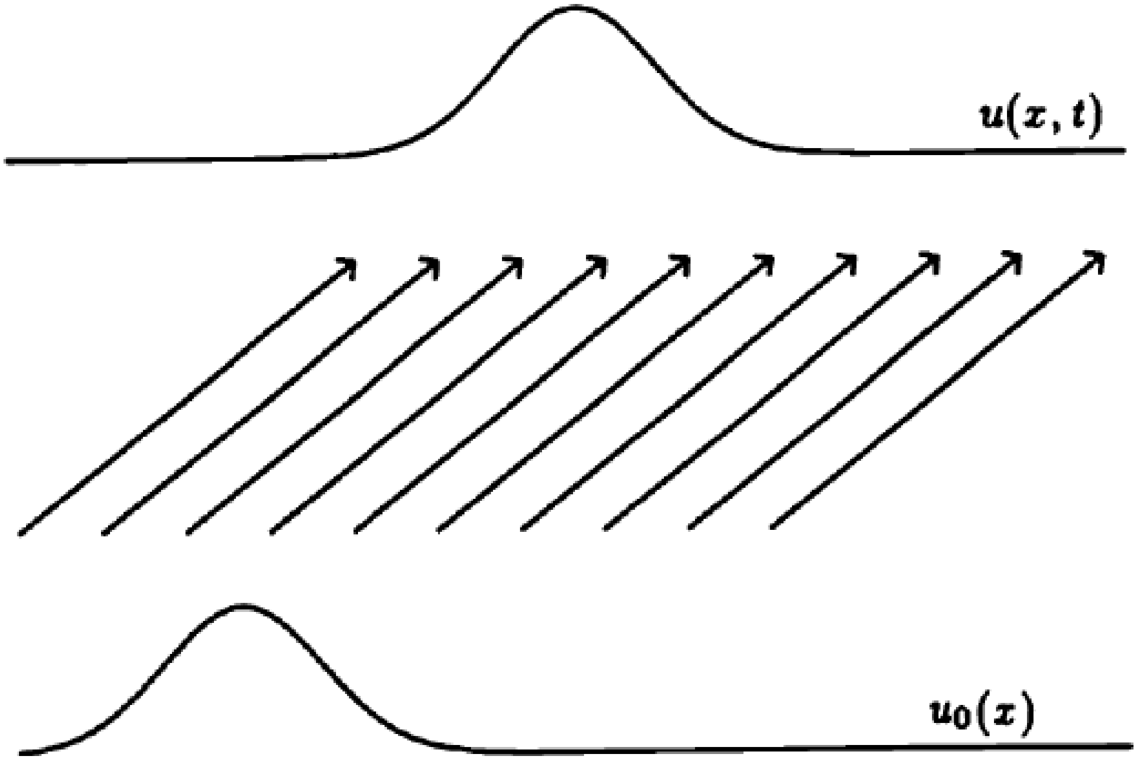


Figure 2.1: Characteristic curves for the linear advection equation.

In the case of the 1-D Burger's equation, the eigenvalue is no longer constant and equal to the solution itself $\lambda_2 = u(x,t)$. The slope of the characteristic curve is now a function of space and more precisely of the initial solution u_0 which requires to look at two distinct cases: a constant and a non-constant initial solution. In the former case, the slope of the characteristic is constant which is the same situation as the linear advection equation previously investigated. In the later case, the characteristic curve will not have the same slope and thus, may intersect. When two characteristic curves intersect, it means that at a given time and position, two values of the solution are allowed (each characteristic curve carries different initial values of the solution): the solution displays an infinite gradient also called shock wave as shown in Fig. 2.2.

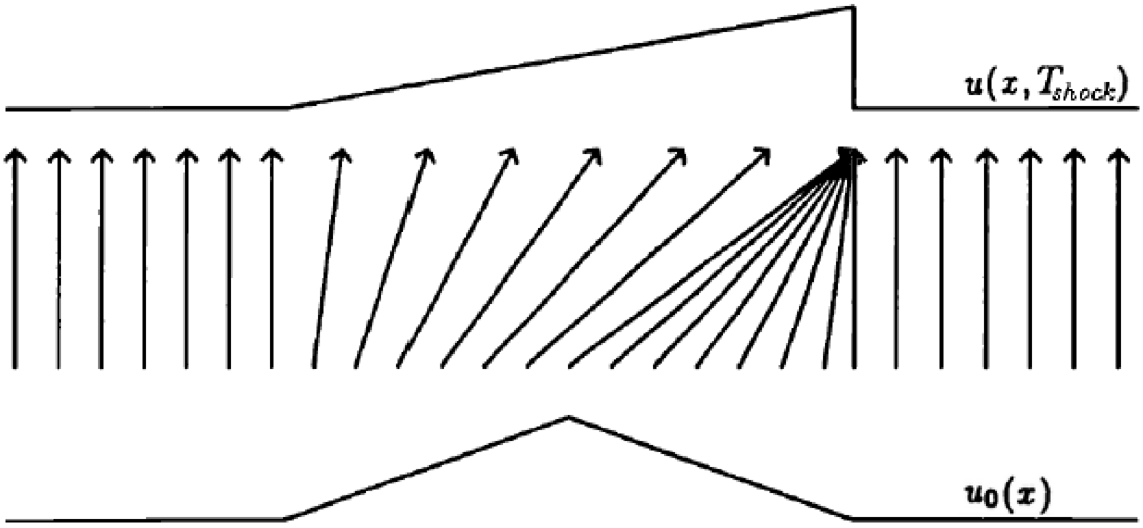


Figure 2.2: Characteristic curves for the 1-D Burger's equation.

The time the shock occurs at, T_{shock}, can be analytically determined. Let consider a 1-D non-linear flux $f(u)$ and two characteristic curves originating from the position x_0 and $x_0 + dx$ carrying the initial values $u_0(x_0)$ and $u_0(x_0 + dx)$, respectively. The characteristic curves are:

$$\begin{aligned} x_1(t) &= x_0 + f'(u_0(x_0))t \\ x_2(t) &= (x_0 + dx) + f'(u_0(x_0 + dx))t \end{aligned} \tag{2.6}$$

Let now assume that the two characteristic curves intersect at time T_{shock}, which implies $x_1(T_{shock}) = x_2(T_{shock})$. Using Eq. (2.6), it yields:

$$x_0 + f'(u_0(x_0))T_{shock} = (x_0 + dx) + f'(u_0(x_0 + dx))T_{shock} \tag{2.7}$$

From Eq. (2.7) and after a few lines of calculation, an expression for T_{shock} can be derived as follows:

$$T_{shock} = \frac{-dx}{f'(u_0(x_0 + dx)) - f'(u_0(x_0))} \tag{2.8}$$

Taking the limit $dx \rightarrow 0$ and using the definition of the derivative, Eq. (2.8) becomes:

$$T_{shock} = \frac{-1}{f''(u_0)u_0'} = \frac{-1}{f'(u_0)}, \tag{2.9}$$

The trivial case $f''(u) = 0$ is ruled out since it was assumed a non-linear flux. However, let assume that the flux is linear. In that case, the second-order derivative is null and by taking the limit of Eq. (2.9), it yields $T_{shock} \rightarrow \infty$ which means that a shock wave never forms. This result is consistent with the conclusion made earlier in this section when studying the linear advection equation (Eq. (2.2a)). Going back to a non-linear flux, breaking first occurs where the derivative of the flux is negative (to ensure positivity of T_{shock}) and its absolute value is maximum (so that $\frac{1}{f'(u_0)}$ is minimal). Once the shock is formed, at a time $t \geq T_{shock}$, more than two characteristic curves may intersect leading to a tripled-value situation as shown in Fig. 2.3.

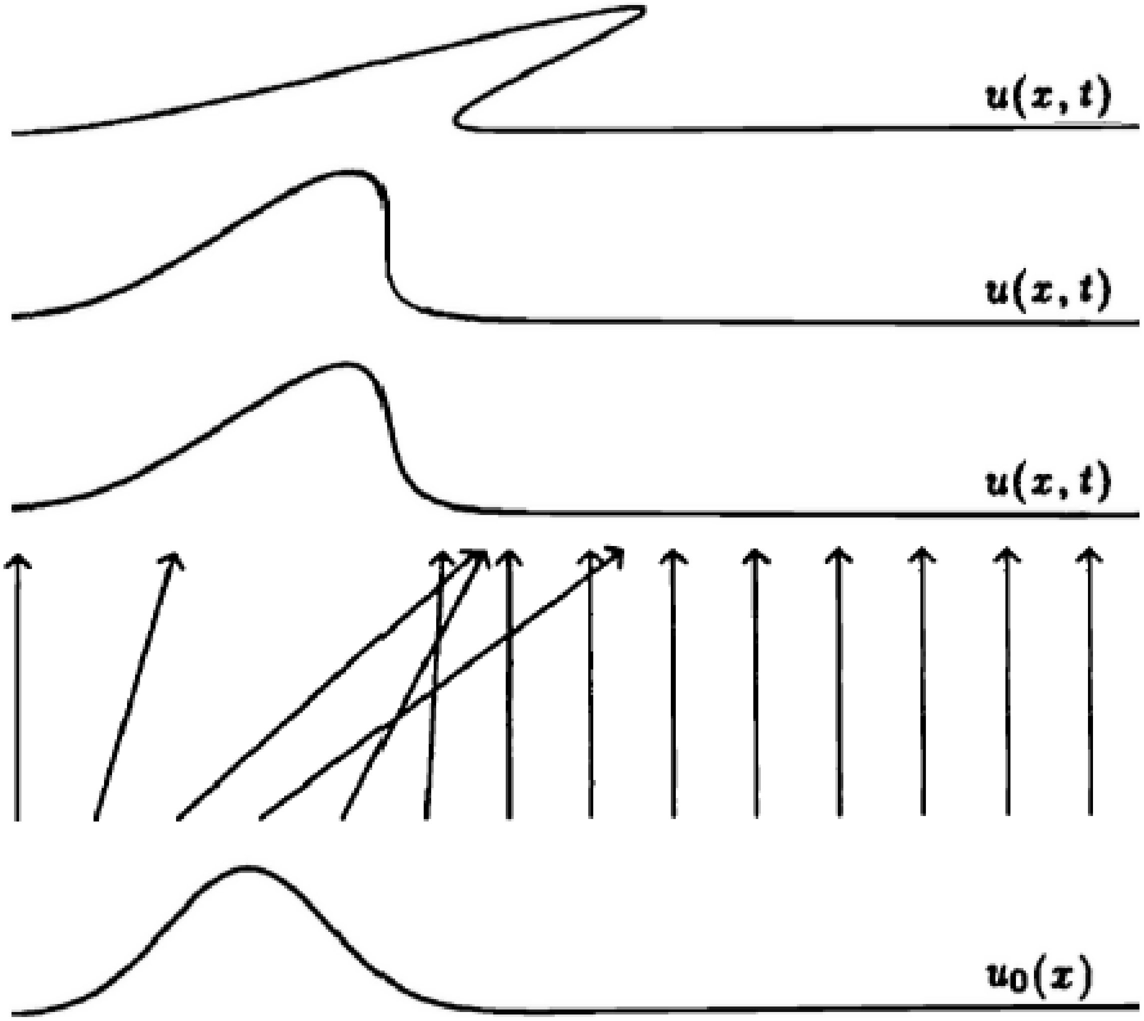


Figure 2.3: Example of a tripled-value situation.

In this case, uniqueness of the solution is not ensured since for a given position and time the solution admits three values. This particular phenomenon makes sense when solving the shallow-water equation that is used to model a breaking wave on a sloping beach. However, when considering a gas flow, uniqueness of the thermodynamic properties is required to ensure a point-wise singled-value density. From the last example, we understand that preventing the tripled-value situation from forming may be the key to obtain the correct physical behavior when solving hyperbolic scalar and system of equations alike Euler equations. A solution to this problem could come from the study of an advection-diffusion equation type that is used, for instance, to model the propagation of particles in a material by both advection and diffusion. This type of equation is known to have unconditionally *smooth solution* for all time and spatial location. A 1-D generic form is given in Eq. (2.10).

$$\partial_t u(xt) + \partial_x f(x,t) = \epsilon \partial_{xx} u(x,t), \tag{2.10}$$

where ϵ is a diffusion coefficient that can be solution-dependent in theory but is assumed constant for the purpose of this section. Since the main difference between the hyperbolic problem given in Eq. (2.1) and Eq. (2.10) lies in the diffusion term $\epsilon \partial_x x u(x,t)$, it is proposed to investigate its effect on the numerical solution. If the solution $u(x,t)$ is smooth, the diffusion term $\epsilon \partial_{xx} u(x,t)$ in Eq. (2.10) is negligible and the numerical solution is driven by the advection term $\partial_x f(u(x,t))$ so that Eq. (2.10) and Eq. (2.1) have similar behaviors. As the solution becomes stiffer, the diffusion terms becomes large enough to influence the behavior of the numerical solution and will prevent the wave from braking as it happens in hyperbolic problems. In other terms, the diffusion term, by monitoring the change of curvature in the numerical solution, locally affects the numerical solution when needed. The diffusion coefficient ϵ can be seen as a tuning coefficient that will also affect the smoothness of the numerical solution as shown in Fig. 2.4. As ϵ goes to zero, the numerical solution becomes sharper and tends to the solution obtained when solving the hyperbolic problem given in Eq. (2.10). On the opposite, with an infinite viscosity coefficient, the shock is smoothed out.

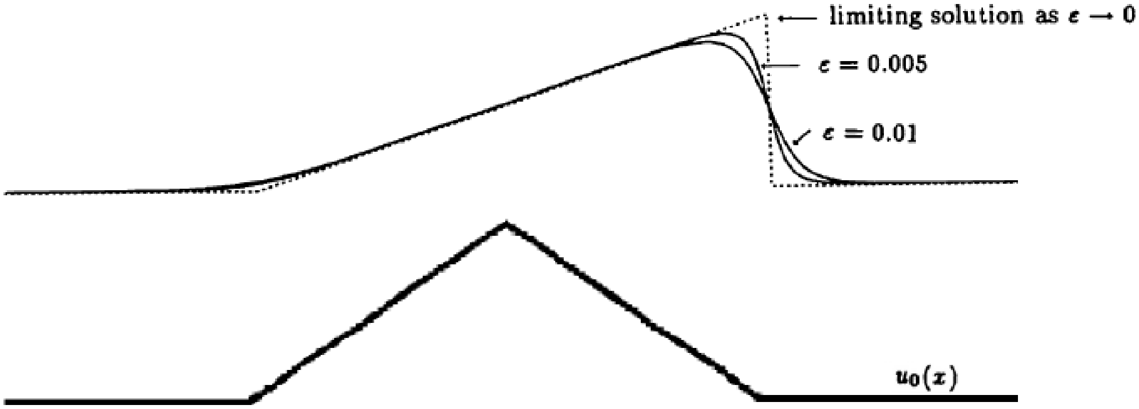


Figure 2.4: Influence of the viscosity coefficient ϵ on the numerical solution stiffness.

Thus, by adding a diffusion term also called viscosity term, the numerical solution remains smooth and single valued, and should allow us to retrieve the correct physical behavior of a hyperbolic problem in the limit $\epsilon \rightarrow 0$. This approach is referred to as a *vanishing viscosity method* and the numerical solution obtained with this method is denoted by $u^\epsilon(x, t)$. We can now define the notion of *generalized solution.*

Definition 2.1.1. *A generalized definition u of a hyperbolic scalar equation conservation law*

$$\partial_t u + \vec{\nabla} \cdot \boldsymbol{f}(u) = 0,$$

is called admissible vanishing viscosity solution if there is a sequence of smooth unique solution u^ϵ of the parabolic equation

$$\partial_t u^\epsilon + \vec{\nabla} f(u^\epsilon) = \epsilon \nabla u^\epsilon,$$

that converges to u as $\epsilon \rightarrow 0$.

It is now clear that by adding a viscosity term to an hyperbolic equation as previously explained, a generalized solution can be obtained with the vanishing viscosity approach. We will see in Section 2.1.3, that a generalized solution can be also defined by the use of a mathematical technique resulting in a weak formulation of the hyperbolic scalar equation. Before doing so, it is proposed to investigate one more property of a shock wave: its speed. Knowing the breaking time T_{shock} and position is not sufficient information to track a shock wave once it is formed. An useful information will be to derive an expression that provides us with the speed of the shock. One of the reason for deriving such expression is, to obtain an analytical solution that can be used for comparison against numerical solutions in order to assess their accuracy. To do so, we consider, one again, a general 1-D hyperbolic scalar equation for simplicity, as shown in Eq. (2.11):

$$\partial_t u(x,t) + \partial_x f(u(x,t)) = 0 \tag{2.11}$$

We assume that the position of the shock is given by a function of time denoted by $s(t)$ and that the associated speed is $S = \frac{ds}{dt}$. At this particular position, the derivatives of the solution u and the flux $f(u)$ are no longer continuous. We also define a control volume $[x_1; x_2]$ that contains the shock wave so that $x_1 \leq s(t) \leq x_2$. Eq. (2.11) is integrated over the control volume as shown in Eq. (2.12):

$$\frac{d}{dt}\int_{x_1}^{s(t)} u(x,t)dx + \frac{d}{dt}\int_{s(t)}^{x_2} u(x,t)dx + f(x_2,t) - f(x_1,t) = 0 \tag{2.12}$$

The first two integrals can be recast by using the following chain rule:

$$\frac{d}{dt}\int_{y_1(t)}^{y_2(t)} g(y,t)dy = \int_{y_1(t)}^{y_2(t)} g(y,t)dy + g(y_2,t)\frac{dy_2(t)}{dt} - g(y_1,t)\frac{dy_1(t)}{dt}, \tag{2.13}$$

which yields by noticing that x_1 and x_2 are fixed and thus, not function of time:

$$\int_{x_1}^{s(t)} \partial_t u(x,t)dx - \int_{s(t)}^{x_2} \partial_t u(x,t)dx + \big(u(s^-(t)) - u(s^+(t))\big)\, S + f(x_2,t) - f(x_1,t) = 0 \tag{2.14}$$

where $u(s^-(t),t)$ and $u(s^+(t),t)$ are the values of the solution u before and after the shock, respectively. Assuming that the x_1 and x_2 approach the shock position $s(t)$ from the left and right, respectively, and that $\partial_t u$ is bounded, the two integrals vanish to yield the following expression for the speed of shock S:

$$S = \frac{f(x_2,t) - f(x_1,t)}{u(s^-(t)) - u(s^+(t))} = \frac{\Delta f}{\Delta u} \tag{2.15}$$

The above expression computing the speed of shock (Eq. (2.15)) is known as the Rankine-Hugoniot jump condition. The hyperbolic scalar equation given Eq. (2.11) is only valid in smooth parts of the solution, and thus, require the use of the Rankine-Hugoniot jump condition in order to solve for the shock region.

2.1.3 Weak solution and entropy condition:

As mentioned in Section 2.1.2, another way to define a generalized solution is to use a mathematical technique that consists of multiplying the hyperbolic scalar equation by a smooth test function $\phi \in C_0^1$ that are continuously differentiable within a contact support. Then, integration per part is performed in order to transfer the derivative off the solution u and onto the test function ϕ. The resulting equation involves fewer derivative on u and, hence, requires less smoothness.

2.1.4 Entropy viscosity method for the multi-D Burger equation:

This section aims at making the link between the theoretical results presented from Section 2.1.1 through Section 2.1.3, and the entropy viscosity method that is the focus of this dissertation. For academic purpose, we limit our scope to the multi-D Burger's equation and rely on the work by Guermond et al. [28, 29, 22, 57]. However, the reader must keep in mind that the objective of this dissertation is to apply the entropy-viscosity method to hyperbolic system of equation.

In the previous sections, it was demonstrated the importance of ensuring uniqueness of the weak solution: it is achieved (i) by adding a viscosity term to the hyperbolic scalar equation in order to prevent tripled-value point from forming, and (ii) by using an entropy condition. When applied to the viscid multi-D Burger's equation, it yields:

$$\partial_t u(\boldsymbol{r},t) + \vec{\nabla}\cdot\left(\frac{u(\boldsymbol{r},t)^2}{2}\vec{n}\right) = \vec{\nabla}\cdot\left(\mu(\boldsymbol{r},t)\vec{\nabla}u(\boldsymbol{r},t)\right) \tag{2.16a}$$

$$R(\boldsymbol{r},t) = \partial_t\eta(\boldsymbol{r},t) + \vec{\nabla}\cdot\Phi \leq 0 \tag{2.16b}$$

where $\mu(\boldsymbol{r},t)$ is a spatial-dependent viscosity coefficient and $R(\boldsymbol{r},t)$ denotes the entropy residual. Assuming that μ is constant, Eq. (2.10) is retrieved. All of the other variables in Eq. (2.16) were defined previously. It was shown in Section 2.1.3 that the sign of the entropy residual, Eq. (2.16b), is conditioned to the convexity of the entropy function η, to the positivity of the viscosity coefficient $\mu(\boldsymbol{r},t)$, and also to the form of the viscous term $\vec{\nabla}\cdot\left(\mu(\boldsymbol{r},t)\vec{\nabla}u(\boldsymbol{r},t)\right)$. In other word, the entropy condition could be used to derive the proper viscous term that will ensure the correct sign for the entropy residual in the shock region. In the case of the multi-D Burger's equation, the choice of the viscous term is obvious and probably unique. However, when considering hyperbolic system of equation alike the multi-D Euler equations, deriving the viscous terms consistent with the entropy condition is no longer trivial and can lead to long and fastidious derivation. This aspect of the method is detailed in Section 2.2.

Once the viscous term is derived and known to be consistent with the entropy condition, it remains to define the viscosity coefficient $\mu(\boldsymbol{r}, t)$ that is known to be positive. Such a step is crucial and should not be underestimated since it will determine the accuracy of the numerical method. The easiest definition we can think of, is to set μ to a constant value. By doing so, dissipation will be added to the shock region, preventing the wave from braking, and also to the smooth region of the solution that does not need dissipation. Such a behavior is not ideal and will not accurately resolve the shock. Another option would be to to track the shock position in order to only add a significant amount of dissipation in the shock region. We also require the viscosity coefficient to be smooth based on the work from REF that a discontinuous viscosity coefficient could yield instabilities in the numerical solution. Such a behavior can be easily understood by considering the following example. Let us assume that the viscosity coefficient jumps from zero to a large value as closing to the shock region. Because the dissipative term is conservative, the gradient of the solution, $\vec{\nabla} u$, will have to experience the same discontinuity as the viscosity coefficient, thus, yielding the same type of behavior in the solution itself. Defining a smart viscosity capable of detecting and tracking a shock is not a straightforward task and needs to rely on a good understanding of the theory related to the shock formation. For example, we can think of monitoring the gradient of the solution itself that will become large in the shock region. Following this reasoning, a possible definition in 1-D would be $\mu(x,t) \propto |\partial_x u(x,t)|$. Another approach consists of using the entropy residual R_e defined in Section 2.1.3. The entropy residual was initially studied to ensure uniqueness of the weak solution, but its variations are intimately related to the solution: R_e is small as the solution is smooth and become large (in absolute value) in the shock region. Thus, by monitoring the variation of the entropy residual, the shock can be detected and also tracked. This approach was used by Guermond et al. [28, 29, 22] to solve for hyperbolic scalar equations such as the multi-D Burger's equation and named after the Entropy Viscosity Method (EVM). Their method requires the definition of three viscosity coefficients: a high-order viscosity coefficient, $\mu_e(\boldsymbol{r}, t)$, defined proportional to the absolute value of the entropy residual R_e, a first-order viscosity coefficient denoted by $\mu_{max}(\boldsymbol{r}, t)$ set proportional to the local maximum eigenvalue, and the viscosity coefficient $\mu(\boldsymbol{r}, t)$ such as $\mu(\boldsymbol{r}, t) = \min\left(\mu_{max}(\boldsymbol{r}, t), \mu_e(\boldsymbol{r}, t)\right)$, used in the dissipative term $\vec{\nabla} \cdot \left(\mu(\boldsymbol{r}, t) \vec{\nabla} u(\boldsymbol{r}, t)\right)$. The idea is to detect the entropy production characteristic of a shock wave. By defining $\mu_e(\boldsymbol{r}, t)$ proportional to $|R_e|$, the high-order viscosity will be large in the shock region and small anywhere else. The first-order viscosity serves as an upper bound for $\mu(\boldsymbol{r}, t)$ and its definition lies onto two criteria:

1. the definition of $\mu_{max}(\boldsymbol{r}, t)$ is determined so that the viscous regularization in Eq. (2.16a) is equivalent to the upwind-scheme when assuming $\mu(\boldsymbol{r}, t) = \mu_{max}(\boldsymbol{r}, t)$, which yields: $\mu_{max}(\boldsymbol{r}, t) = \frac{h}{2} f'(u(\boldsymbol{r}, t))$. Derivation of the expression for μ_{max} can be found in REF and is easily done in 1-D when discretizing Burger's equation with a finite difference method.

2. the first-order viscosity coefficient is related to the Curant-Friedrichs-Lewy number (CFL) and more precisely to the stability of the numerical solution when using temporal explicitly schemes.

Proof of a maximum principle for the EVM is under investigation and thus, will affect the definition of the first-order viscosity coefficient μ_{max}. Based on the definition of the high- and first-order viscosity coefficients, the variation of the $\mu(\boldsymbol{r}, t)$ are the followings: as the solution is smooth, the entropy production measured by the entropy residual, R, is small and thus $\mu(\boldsymbol{r}, t) = \mu_e(\boldsymbol{r}, t)$. In the shock region, the entropy residual is peaked and the high-order viscosity coefficient saturates to the first-order viscosity coefficient that is known to be over-dissipative since equivalent to upwind-scheme. With such definition, the viscosity coefficient $\mu(\boldsymbol{r}, t)$ is peaked in the shock region and small anywhere else, while experiencing smooth variations.

It remains now to complete the definition of the high-order viscosity coefficient $\mu_e(\boldsymbol{r}, t)$ which can be achieved by performing a dimensional analysis. Noticing that the units of $\mu(\boldsymbol{r}, t)$ are $m^2 \cdot s^{-1}$, a suitable definition for $\mu_e(\boldsymbol{r}, t)$ is:

$$\mu_e(\boldsymbol{r}, t) = h^2 \frac{|R(\boldsymbol{r}, t)|}{norm(s)}$$

where $norm(s)$ is a normalization function of the same unit as the entropy s, and h is a characteristic length of the mesh. At this stage, the choice of the normalization function is unclear. Guermont et al. proposed to use $norm(s) = ||s - \bar{s}||_\infty$ where $\bar{s}$ is the average value of the entropy function over the computational domain, and $|| \cdot ||_\infty$ denotes the infinity norm. Their definition of the EVM when applied to hyperbolic scalar equation is the following:

$$\partial_t u(\boldsymbol{r}, t) + \vec{\nabla} \cdot (f(u(\boldsymbol{r}, t))\vec{n}) = \vec{\nabla} \cdot \left(\mu(\boldsymbol{r}, t) \vec{\nabla} u(\boldsymbol{r}, t) \right) \tag{2.17a}$$

$$R(\boldsymbol{r}, t) = \partial_t \eta(\boldsymbol{r}, t) + \vec{\nabla} \cdot \Phi \tag{2.17b}$$

$$\begin{cases} \mu(\boldsymbol{r}, t) = \min\left(\mu_e(\boldsymbol{r}, t), \mu_{max}(\boldsymbol{r}, t)\right) \\ \mu_{max}(\boldsymbol{r}, t) = \frac{h}{2} \left| f'(u(\boldsymbol{r}, t)) \right| \\ \mu_e(\boldsymbol{r}, t) = h^2 \frac{\max(|R(\boldsymbol{r}, t), J)|}{||s - \bar{s}||_\infty} \end{cases} \tag{2.17c}$$

where h is defined as the gird size and J denotes the jump of the entropy flux Φ at the interface of each element. Information relative to the computation of the jump with continuous Galerkin finite element method (CGFEM) will be detailed in Section 3.1.1. For the case of discontinuous schemes, the reader can refer to [57]. Numerical results for the multi-D Burger's equation solved with the EVM with

CGFEM are presented in Section 4.1.

Remark. *The definition of the viscosity coefficients given in Eq. (2.17c) requires a isentropic mesh in order to be able to define the gird size h. An alternative definition without h is under investigation for the case of hyperbolic scalar equations.*

2.2 Hyperbolic system of equations:

In this section, application of the EVM to non-linear hyperbolic system of equations is discussed. The reader can refer to Toro [55] and Leveque for an extension to the non-linear hyperbolic conservation laws, of the theoretical notions (eigenvalues, characteristic curves, . . .) introduced in Section 2.2. The objective of this section, is to provide the reader with a methodology on how to apply the EVM to any hyperbolic system of equations. For academic purpose, we will rely on the latest published version of the EVM [57] for the multi-D Euler equations in order to understand the main steps of the method. Other more recent publications will be also used. [23, 29] are good pre-requites to [57] to see the theoretical evolution of the EVM. For hyperbolic system of equation, it is custom of using the physical solution s that is of opposite sign as the mathematical entropy denoted by η.

To the best of our knowledge, the EVM was successfully applied to one hyperbolic system of equation: the multi-D Euler equations with the Ideal gas equation of state [23, 57]. Good agreements with the exact solutions were obtained in 1- and 2-D results when using discontinuous schemes (finite volume and discontinuous Galerkin finite element methods). We now recall the details of the latest version of the method [57] and remind the reader that this version will be used as a starting point and modified during this dissertation. The viscous regularization derived from the entropy condition for the multi-D Euler equations with the Ideal Gas equation of state, is the following:

$$\partial_t \rho + \vec{\nabla} \cdot (\rho \vec{u}) = \vec{\nabla} \cdot \left(\mu \vec{\nabla} \rho \right) \tag{2.18a}$$

$$\partial_t (\rho \vec{u}) + \vec{\nabla} (\rho \vec{u} \otimes \vec{u}) + \vec{\nabla} P = \vec{\nabla} \cdot \left(\rho \mu \vec{\nabla}^s \vec{u} \right) \tag{2.18b}$$

$$\partial_t (\rho E) + \vec{\nabla} \cdot [\vec{u} (\rho E + P)] = \vec{\nabla} \cdot \left(\kappa \vec{u} \vec{\nabla}^s \vec{u} + \kappa \vec{\nabla} T \right) \tag{2.18c}$$

$$P = (\gamma - 1) \rho e = (\gamma - 1) C_v \rho T \tag{2.18d}$$

where ρ, $\rho\vec{u}$ and ρE are the fluid density, momentum and total energy, respectively, and will be referred to as the conservative variables. The pressure P and the temperature T are computed from the Ideal gas equation of state (IGEOS) recalled in Eq. (2.18d) which is a function of the density ρ and the specific internal energy e. The heat capacity C_v is constant by definition. The viscosity coefficients μ and κ are spatial and temporal dependent and are defined proportional to an entropy residual R as follows:

$$\begin{cases} \mu (\boldsymbol{r}, t) = \min \left(\mu_e (\boldsymbol{r}, t), \mu_{max} (\boldsymbol{r}, t) \right) \\ \kappa (\boldsymbol{r}, t) = \frac{\gamma \mathrm{Pr}}{\gamma - 1} \mu (\boldsymbol{r}, t), \end{cases} \tag{2.19a}$$

and

$$\begin{aligned} \mu_{max} (\boldsymbol{r}, t) &= \frac{h}{2} \left(||\vec{u} (\boldsymbol{r}, t)|| + c (\boldsymbol{r}, t) \right) \\ \mu_e (\boldsymbol{r}, t) &= C_E h^2 \frac{\max \left(|R (\boldsymbol{r}, t)|, J \right)}{||s (\boldsymbol{r}, t) - \bar{s}(t)||_\infty} \\ R (\boldsymbol{r}, t) &= \partial_t s (\boldsymbol{r}, t) + \vec{u} \cdot \vec{\nabla} s (\boldsymbol{r}, t), \end{aligned} \tag{2.19b}$$

where C_E is a constant coefficient of order one, h is the grid size, $c = \sqrt{\gamma P / \rho}$ is the speed of sound and Pr is a Prandtl number taken in the interval $\in \left[0; \frac{1}{4}\right]$. The variable J denotes the jump of the entropy flux $\vec{u}s$ and is pice-wise constant. It is computed at the interfaces between each cell and its direct neighbors. Further details can be found in [57]. The entropy s is function of the density and the pressure:

$$s\left(\rho, P\right) = C_v \ln\left(\frac{P}{\rho^\gamma}\right), \tag{2.20}$$

and can also be recast as a function of the density and the internal energy using the IGEOS. It remains to define the symmetric gradient $\vec{\nabla}^s \vec{u}$ with the following entries $\vec{\nabla}^s \vec{u}_{i,j} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$. The current definition of the EVM suffers from some theoretical gap. The normalization parameters $||s\left(\boldsymbol{r}, t\right) - \bar{s}(t)||_\infty$ used in the definition of the high-order viscosity coefficient μ_e in Eq. (2.19b) does not currently have any theoretical justification. The same remark can be made for the Prandtl number that is set by the user based on its experience. Plus, the viscous regularization given in Eq. (2.18) is only valid for the IGEOS which is limiting for engineering applications. However, new developments in the theory extended the validity of the method to any equation of state [23] which makes it a good candidate for nuclear reactor applications for instance. Thus, based on the work done in [23] and with the experience of [29, 57] the following methodology is proposed. We consider the generic non-linear hyperbolic system in order to detail our methodology:

$$\partial_t \boldsymbol{U}(\boldsymbol{r}, t) + \vec{\nabla} \cdot \boldsymbol{F}\left(\boldsymbol{U}\right) = 0, \tag{2.21}$$

where $\boldsymbol{U} = \left(U_1, \ldots, U_n\right)$ is a vector solution and $\boldsymbol{F}\left(\boldsymbol{U}\right)$ is a hyperbolic flux whom eigenvalues are denoted by $\left(\lambda_1, \ldots, \lambda_n\right)$.

1. The first step consists of deriving an entropy equation for an entropy function

denoted by $s\left(\boldsymbol{U}\right)$ of the form:

$$R\left(\boldsymbol{U}\right)=\partial_t s\left(\boldsymbol{U}\right)+\boldsymbol{F}_{\boldsymbol{U}}\cdot\vec{\nabla}s\left(\boldsymbol{U}\right)=0, \tag{2.22}$$

where $\boldsymbol{F}_{\boldsymbol{U}}=\frac{\partial\boldsymbol{F}}{\partial\boldsymbol{U}}$ is the jacobian matrix of the hyperbolic flux $\boldsymbol{F}\left(\boldsymbol{U}\right)$. This entropy equation/residual is obtained from the hyperbolic system of equation given in Eq. (2.21) either by multiplying by the matrix $s_{\boldsymbol{U}}=\frac{\partial s(\boldsymbol{U})}{\partial\boldsymbol{U}}$ and using the chain rule, or by doing combination of the equations of the hyperbolic system. This step is well documented for the multi-D Euler equation [55]. The objective is to understand the steps that lead to the derivation of the entropy equations since the same steps will be used to obtain the viscous terms.

2. We now want to derive the dissipative terms consistent with the entropy condition. The method is inspired of what is done for the multi-D Euler equations in [23]. To do so, we first modify Eq. (2.21) by adding a viscous flux $\boldsymbol{G}\left(\boldsymbol{U}\right)$:

$$\partial_t\boldsymbol{U}(\boldsymbol{r},t)+\vec{\nabla}\cdot\boldsymbol{F}\left(\boldsymbol{U}\right)=\vec{\nabla}\cdot\boldsymbol{G}\left(\boldsymbol{U}\right), \tag{2.23}$$

Then, the entropy residual is derived again:

$$\begin{aligned}R\left(\boldsymbol{U}\right)=\partial_t s\left(\boldsymbol{U}\right)+\boldsymbol{F}_{\boldsymbol{U}}\cdot\vec{\nabla}s\left(\boldsymbol{U}\right) &= s_{\boldsymbol{U}}\vec{\nabla}\cdot\boldsymbol{G}\left(\boldsymbol{U}\right)\\ R\left(\boldsymbol{U}\right)=\partial_t s\left(\boldsymbol{U}\right)+\boldsymbol{F}_{\boldsymbol{U}}\cdot\vec{\nabla}s\left(\boldsymbol{U}\right) &=\\ \vec{\nabla}\cdot\left(s_{\boldsymbol{U}}\boldsymbol{G}\left(\boldsymbol{U}\right)\right) &- \boldsymbol{G}\left(\boldsymbol{U}\right)\cdot\vec{\nabla}s_{\boldsymbol{U}}.\end{aligned} \tag{2.24}$$

To prove that the entropy residual R remains positive, the non-conservative terms of the right hand-side has to be positive. Thus, positivity of the entropy residual is tied to the definition of the viscous term $\boldsymbol{G}\left(\boldsymbol{U}\right)$ and the entropy function s. For the multi-D Euler equation, Guermond et al. [23] proved that the entropy function s needs to be concave ($-s$ is convex) in order to ensure positivity of the entropy residual for any equation of state. This condition is tied to a particular choice of the dissipative terms that will be detailed in Section 5.2, and also positivity of the viscosity coefficients. In the general case, the parabolic regularization (REF) can be used and consists of dissipating on the vector solution $\boldsymbol{U}$ as shown in Eq. (2.25):

$$\partial_t \boldsymbol{U}(\boldsymbol{r},t) + \vec{\nabla} \cdot \boldsymbol{F}\left(\boldsymbol{U}\right) = \vec{\nabla} \cdot \left(\mu \vec{\nabla}\left(\boldsymbol{U}\right)\right), \tag{2.25}$$

where μ is a positive viscosity coefficient. Eq. (2.25) obeys to the entropy condition under the condition of having a concave entropy s [5]. Using the entropy condition, other viscous regularizations can be found with multiple viscosity coefficients (see [23] for the case of the multi-D Euler equation). However, it is expected that they all degenerate to the parabolic regularization when assuming that all viscosity coefficients are equal to each other. Plus, for consistency with the parabolic regularization [5], the entropy function s is required to be concave. In other terms, the parabolic regularization can be used as a hint in order to derive a viscous regularization with multiple viscosity coefficients. We will see in Section 5.3.2 that having a viscous regularization with two viscosity coefficients is required for the multi-D Euler equation in order to have well-scaled viscous terms in the non-isentropic low Mach limit. Before moving forward to the next step it is proposed to recast the entropy residual R as a function of the conservative variables. This step is justified by the difficulty encountered to obtain an analytical expression of the entropy function. This is particular true for the multi-D Euler equation when dealing with complex equation of states. We assume that the entropy residual was successfully recast as a function of the conservative variables and that the new entropy residual is denoted by $\tilde{R}$. The reader can refer to Section 5.3.1 as an example.

3. Once the entropy residual is proven to be positive, it remains to define the viscosity coefficient(s). Assuming that a viscous regularization was derived in step 2 with n viscosity coefficients denoted by μ_i, $i \in [1, \ldots, n]$, a general definition can be given under the form:

$$\begin{cases} \mu_i(\boldsymbol{r}, t) = \max\left(\mu_{i,e}(\boldsymbol{r}, t), \mu_{max}(\boldsymbol{r}, t)\right) \\ \mu_{max}(\boldsymbol{r}, t) = \frac{h}{2} \max_{i \in [1,n]} |\lambda_i(\boldsymbol{r}, t)| \\ \mu_{i,e}(\boldsymbol{r}, t) = h^2 \frac{\max\left(\tilde{R}(\boldsymbol{r},t), J\right)}{norm_i(\boldsymbol{r},t)}, \end{cases}$$

where $\mu_{e,i}$ and μ_{max} are the high- and first-order viscosity coefficients, respectively. The high-order viscosity coefficient $\mu_{e,i}$ is defined proportional to the local entropy residual $R(\boldsymbol{r}, t)$ and also function of a normalization parameter $norm_i(\boldsymbol{r}, t)$ that will be further detailed. The first-order viscosity coefficient $\mu_{max}(\boldsymbol{r}, t)$ is set proportional to the maximum eigenvalue and is unique for all viscosity coefficients $\mu_i(\boldsymbol{r}, t)$. h still denoted the grid size. In order to have a complete definition for the $\mu_e(\boldsymbol{r}, t)$, the normalization parameter $norm_i(\boldsymbol{r}, t)$ needs to be defined. It is well known that hyperbolic system of equations suffer from ill-scaled dissipative term in some particular asymptotic limit. This is particular true for the stabilization methods used for the multi-D Euler equations, that require a fix in the low Mach asymptotic limit in order to yield the correct asymptotic behavior [24, 56]. Thus, by non-dimensionalizing the equations, the normalization parameter $\mathrm{norm}_i(\boldsymbol{r}, t)$ can be determined for each viscosity coefficients $\mu_i(\boldsymbol{r}, t)$ to ensure well-scaled dissipative terms.

This three steps process is applied to the multi-D Euler equations with variable area Chapter 5, the seven-equation model Chapter 6 and the radiation-hydrodynamic equations Chapter 7 to determine the dissipative terms and to define the viscosity coefficients. Details about the implementation of the EVM with continuous Galerkin finite element method are provided in Section 3.1.1. The jump J is given on a case by case basis since its definition depends on the variables involved in the expression of the new entropy residual $\tilde{R}$.

3. DISCRETIZATION METHOD AND IMPLEMENTATION DETAILS OF THE ENTROPY VISCOSITY METHOD.

3.1 Spatial and Temporal Discretization Method:

Because the stabilization of the numerical solution of hyperbolic equation systems is intimately connected with the temporal and spatial discretization, before discussing the entropy viscosity stabilization method used in RELAP-7, a short summary of the numerical methods it employs is given.

3.1.1 Spatial Discretization Algorithm

RELAP-7 currently employs the continuous finite element method implemented via the INL MOOSE framework. For simplicity of exposition this section focuses on the weak statement associated to the strong form of the one-dimensional, variable cross-sectional area form of the Euler equations EQUATION summarized in SECTION. Start by writing the equations in “vector” form as

$$\vec{R}(\vec{U}) \equiv \frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{F}}{\partial x} - \vec{S} = \vec{0} \tag{3.1}$$

where

$$\vec{U} \equiv \begin{bmatrix} \rho A \\ \rho u A \\ \rho E A \end{bmatrix}, \qquad \vec{F} \equiv \begin{bmatrix} \rho u A \\ \left(\rho u^2 + p\right) A \\ \rho u H A \end{bmatrix} \tag{3.2}$$

and $\vec{S} = \vec{S}(\vec{U})$ consists of the remaining source terms. Note that $\vec{U}$ and $\vec{F}$ are identical to their meanings in the “constant-area” equations, up to multiplication by the area, A. The variational statement proceeds by dotting (3.1) by an “admissible” vector test function $\vec{W}$ (more details of which will be subsequently given), integrating over the domain Ω, and applying the divergence theorem. Solutions $\vec{U}$ are sought such that

$$\int_{\Omega} \left(\frac{\partial \vec{U}}{\partial t} \cdot \vec{W} - \vec{F} \cdot \frac{\partial \vec{W}}{\partial x} - \vec{S} \cdot \vec{W} \right) \mathrm{d}\Omega + \int_{\Gamma} \left(\vec{F} \cdot \vec{W} \right) \vec{n}_x \, \mathrm{d}\Gamma = 0 \tag{3.3}$$

holds for *all* admissible $\vec{W}$. Note that the test function $\vec{W}$ is not chosen arbitrarily. In particular, it is required that $\vec{W}$ come from the space of vector functions

$$\vec{W} \in \left\{ \begin{bmatrix} w \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ w \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ w \end{bmatrix} \right\} \tag{3.4}$$

where $w \in \mathcal{W}$ is a scalar test function. In the present work, and in general practice, the space $\mathcal{W}$ is taken to be (a subspace of) the Hilbert space $H^1(\Omega)$. This choice, for instance, guarantees enough smoothness that (3.3) makes sense. The approximate problem then proceeds by selecting only test functions from a finite-dimensional subspace of $\mathcal{W}$, denoted by $\mathcal{W}^h$, which is spanned by the basis $\{\phi_i\}$, $i = 1, \ldots, N$. We then seek $\vec{U}^h$ with components in the same space as $\mathcal{W}^h$, satisfying the boundary conditions, and such that

$$\int_{\Omega} \left(\frac{\partial \vec{U}^h}{\partial t} \cdot \vec{W}^h - \vec{F}^h \cdot \frac{\partial \vec{W}^h}{\partial x} - \vec{S}^h \cdot \vec{W}^h \right) \mathrm{d}\Omega + \int_{\Gamma} \left(\vec{F}^h \cdot \vec{W}^h \right) \vec{n}_x \, \mathrm{d}\Gamma = 0 \tag{3.5}$$

holds for all $\vec{W}^h$ defined analogously to (3.4), with components in $\mathcal{W}^h$. Note that (3.5) has been placed in a "continuous" setting, that is, a mesh and finite element discretization has been introduced requiring a continuous solution. Equation (3.5) remains a "weak" restatement of the "strong" equations (3.1) in the sense that derivatives of the solution and its flux need not be continuous. More will be said of this subsequently, in the upcoming section on stabilization methods. Written out in component form, and denoting the components of $\vec{U}^h$ by U_0^h, U_1^h, and U_2^h, (3.5) expands to:

$$\int_\Omega \left(\frac{\partial U_0^h}{\partial t} \phi_i - U_1^h \frac{\partial \phi_i}{\partial x} \right) \mathrm{d}\Omega + \int_\Gamma U_1^h \vec{n}_x \phi_i \, \mathrm{d}\Gamma = 0 \tag{3.6}$$

$$\begin{aligned} &\int_\Omega \left[\left(\frac{\partial U_1^h}{\partial t} - U_0^h g_x + \frac{f}{2d_h} U_1^h \left| \frac{U_1^h}{U_0^h} \right| - p^h \frac{\partial A}{\partial x} \right) \phi_i - \left(\frac{(U_1^h)^2}{U_0^h} + p^h A \right) \frac{\partial \phi_i}{\partial x} \right] \mathrm{d}\Omega \\ &\qquad + \int_\Gamma \left(\frac{(U_1^h)^2}{U_0^h} + p^h A \right) \vec{n}_x \phi_i \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{3.7}$$

$$\begin{aligned} &\int_\Omega \left[\left(\frac{\partial U_2^h}{\partial t} + H_w a_w (T^h - T_w) A - U_1^h g_x \right) \phi_i - U_1^h H^h \frac{\partial \phi_i}{\partial x} \right] \mathrm{d}\Omega \\ &\qquad + \int_\Gamma U_1^h H^h \vec{n}_x \phi_i \, \mathrm{d}\Gamma = 0 \end{aligned} \tag{3.8}$$

Equations (3.6)–(3.8) must hold for $i = 1, \ldots, N$. Note that the approximate pressure, p^h, temperature, T^h, and enthalpy, H^h are functions of the conserved variables U_0^h, U_1^h, U_2^h. As mentioned, a continuous Galerkin formulation is employed, and therefore the unknowns are expressed in the same basis used for the test functions, i.e.

$$U_0^h = \sum_j (U_0)_j \phi_j \tag{3.9}$$

$$U_1^h = \sum_j (U_1)_j \phi_j \tag{3.10}$$

$$U_2^h = \sum_j (U_2)_j \phi_j \tag{3.11}$$

The coefficients $(U_0)_j$, $(U_1)_j$, and $(U_2)_j$ vary in time only, and comprise the solution vector at each iteration. Note that (3.6)–(3.8) are so-called “semi-discrete” equations: they have been discretized in space, but the temporal derivatives remain in continuous form. In SECTION the various time discretization methods employed in RELAP-7, to approximately time-integrate the equations, are summarized. Furthermore, it is well-known that a continuous Galerkin discretization of this set of hyperbolic equations is equivalent to a central difference method for a certain choice of integration rule, and therefore will exhibit oscillatory instabilities unless some artificial diffusion is added to stabilize the method. In SECTION, stabilization will be discussed further, and in particular the new entropy viscosity stabilization scheme will be presented in detail.

3.1.2 Backward Euler

The backward Euler method [9] is a well-known, first-order, A-stable implicit time integration method. Given a generic semi-discrete equation in a form similar to (3.6)–(3.8),

$$\int_\Omega \left(\frac{\partial u^h}{\partial t} + G(u^h) \right) \phi_i \,\mathrm{d}\Omega = 0 \tag{3.12}$$

the backward Euler method results in the temporal discretization

$$\int_\Omega \left(\frac{u^{n+1} - u^n}{\Delta t} + G(u^{n+1}) \right) \phi_i \,\mathrm{d}\Omega = 0 \tag{3.13}$$

where Δt is the timestep, $t^{n+1} = t^n + \Delta t$, and $u^n \equiv u^h(t^n)$ is a shorthand notation used to refer to the finite element solution at time level n. Equation (3.13) is a fully-discrete (possibly nonlinear) equation which must be satisfied for each i.

Note that the backward Euler method, when applied to the linear convection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{3.14}$$

yields a leading-order truncation error term of the form

$$\begin{aligned}\left.\frac{\partial u}{\partial t}\right|_{t^{n+1}} &= \frac{u^{n+1}-u^n}{\Delta t} + \frac{\Delta t}{2}\left.\frac{\partial^2 u}{\partial t^2}\right|_{t^{n+1}} + \mathcal{O}(\Delta t^2) \\ &= \frac{u^{n+1}-u^n}{\Delta t} + \frac{a^2\Delta t}{2}\left.\frac{\partial^2 u}{\partial x^2}\right|_{t^{n+1}} + \mathcal{O}(\Delta t^2) \end{aligned} \tag{3.15}$$

where (3.15) follows from differentiating the continuous equation (3.14) with respect to time:

$$\frac{\partial^2 u}{\partial t^2} = -a\frac{\partial}{\partial t}\left(\frac{\partial u}{\partial x}\right) = -a\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial t}\right) = -a\frac{\partial}{\partial x}\left(-a\frac{\partial u}{\partial x}\right) = a^2\frac{\partial^2 u}{\partial x^2}\ . \tag{3.16}$$

Rearranging terms in (3.15) and adding $a\frac{\partial u}{\partial x}$ to both sides allows us to write

$$\frac{u^{n+1}-u^n}{\Delta t} + a\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} - \frac{a^2\Delta t}{2}\frac{\partial^2 u}{\partial x^2} + \mathcal{O}(\Delta t^2) \tag{3.17}$$

where all the continuous derivatives are assumed to be evaluated at time level t^{n+1}. Thus, the semi-discrete form of the linear convection on the left-hand side of (3.17) is equal to the continuous parabolic partial differential equation on the right-hand side, which includes "artificial" diffusion or viscosity of $\mathcal{O}(\frac{a^2\Delta t}{2})$, to within $\mathcal{O}(\Delta t^2)$. For this reason, we often say that the backward Euler time discretization is inherently stabilizing for the hyperbolic equation (3.14). Obviously, the artificial viscosity for the complete scheme is a composite of the artificial viscosity of both the time and spatial discretization.

The backward Euler time integration method may generate excessive artificial viscosity and should, therefore, only be used for transients with RELAP-7 as an initial scoping calculation, or if only the steady-state solution is of interest. For accurate transient solutions with RELAP-7, the BDF2 time integration method, described next, is highly recommended because it is a second-order (in time) discretization.

3.1.3 BDF2

The backward differentiation formula (BDF) is a family of implicit methods for numerically integrating ordinary differential equations. Some notable members of this family include BDF1, which is equivalent to the backward Euler [4] method discussed in Section 3.1.2, and BDF2, which is the highest-order BDF method which is still A-stable. For fixed step-size Δt, the BDF2 method applied to the ordinary differential equation

$$\frac{\partial u}{\partial t} = f(t, u) \tag{3.18}$$

$$u(t = 0) = u_0 \tag{3.19}$$

yields the update step:

$$u^{n+1} = \frac{4}{3}u^n - \frac{1}{3}u^{n-1} + \frac{2}{3}\Delta t f(u^{n+1}, t^{n+1}) \tag{3.20}$$

Dividing through by $\frac{2}{3}\Delta t$, equation (3.20) can be alternatively written as

$$\frac{\frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}}{\Delta t} = f(u^{n+1}, t^{n+1}) \tag{3.21}$$

The left-hand side of (3.21) can be interpreted as a backward-difference approximation to the continuous time derivative $\frac{\partial u}{\partial t}$, and may be employed in a manner analogous to (3.13) to derive a fully-discrete system of equations:

$$\int_\Omega \left(\frac{\frac{3}{2}u^{n+1} - 2u^n + \frac{1}{2}u^{n-1}}{\Delta t} + G(u^{n+1}) \right) \phi_i \, \mathrm{d}\Omega = 0 \tag{3.22}$$

based on the semi-discrete equations (3.6)–(3.8). Since BDF2 requires two old timesteps, the method must be "bootstrapped" by a lower-order method, such as backward Euler, when starting. This means that a much smaller time step size should be used for start-up, at the beginning of a transient. The BDF2 method is recommended for most transient simulations with RELAP-7.

3.1.4 Jacobian-Free Newton Krylov Solver

The RELAP-7 code solves coupled multi-physics problems using the Jacobian-Free Newton Krylov (JFNK) approach via the MOOSE framework. Field equations solved in the current RELAP-7 code include PDEs to describe one-dimensional fluid flow in pipe systems and heat conduction in solids, as well as ODEs to describe physics in zero-dimensional components and the point kinetics equations.

The JFNK method is a fully-coupled, multi-level method for solving large nonlinear equation systems. In general, it consists of at least two levels: the outer Newton loop for the nonlinear solve and the inner Krylov loop for the linear systems of equations associated to Newton iteration. The JFNK method has become an increasingly popular option for solving large nonlinear equation systems arising from multi-physics problems over the last 20 years, and has branched out into a number of different disciplines [32].

In what follows, a brief description of the JFNK method as it applies to the RELAP-7 application is given. The FEM-discretized field equations are first written as

$$\vec{F}(\vec{u}) = \vec{0} \tag{3.23}$$

where $\vec{F}$ represents the nonlinear equation system and $\vec{u}$ is the solution vector. Newton's method requires an initial guess, $\vec{u}^0$, to start the iteration process. For the transient problems of interest here, the solution at a previous time step is generally used as the initial guess for the method. At the k^{th} iteration, the residual vector is defined as

$$\vec{r}^k \equiv \vec{F}(\vec{u}^k)\,. \tag{3.24}$$

Clearly if $\vec{u}^k$ satisfies (3.23) *exactly*, the k^{th} residual will be zero. To update the solution vector, the following equation is solved for the update vector, $\delta\vec{u}^{k+1}$:

$$J(\vec{u}^k)\delta\vec{u}^{k+1} = -\vec{r}^k \tag{3.25}$$

where $J(\vec{u}^k)$ is the Jacobian matrix evaluated at $\vec{u}^k$. In index notation,

$$J_{ij} \equiv \frac{\partial F_i}{\partial u_j}\ . \tag{3.26}$$

After $\delta\vec{u}^{k+1}$ is obtained, the $(k+1)^{st}$ solution iterate is computed by

$$\vec{u}^{k+1} = \vec{u}^k + \delta\vec{u}^{k+1}\ . \tag{3.27}$$

The Newton iteration is terminated when one of the following conditions is met:

1. The residual vector norm, $|\vec{r}^k|$, is sufficiently small.

2. The relative residual vector norm $\frac{|\vec{r}^k|}{|\vec{r}^0|}$ is sufficiently small.

3. The step size norm, $|\delta\vec{u}^{k+1}|$ is sufficiently small.

Note that (3.25) represents a large linear system of equations. In the JFNK method, we need not explicitly form the matrix J: only its action on a vector (via matrix-vector product) is required. Effective preconditioning is generally required for Krylov subspace methods to be efficient, i.e., for the method to converge in a reasonable number of iterations. A preconditioned version of equation (3.25) can be expressed as (using right preconditioning as an example),

$$J^k P^{-1} \left(P \delta \vec{u}^{k+1} \right) = -\vec{r}^k \tag{3.28}$$

where P is the preconditioning matrix. In the approach current used in RELAP-7, an analytical Jacobian matrix is computed according to (3.26), and passed to the underlying numerical solver library as the matrix P for preconditioning purposes.

3.2 Hyperbolic system of equation and boundary conditions:

- describe general way for bc: waves, characteristic equations, variables, ...
- Look at Burger's equations.
- Do multi-D Euler equations.

3.3 Implementation of the entropy viscosity method (EVM) with continuous Galerkin finite element method:

After describing the theoretical approach that leads to the derivation of the dissipative terms consistent with the entropy minimum principle and the definition of the viscosity coefficient, this section focuses on the implementation of the method in a Galerkin continuous finite element scheme. Details are given on how to implement and compute the jump, the entropy residual and the dissipative terms, for instance. Special attention is required for the jump since their definition is scheme dependent. A non-uniform 2-D mesh family Ω is considered. Each member of this family is called element, k, and the set of its faces is denoted by $\delta k = \{\delta k_j\}$, where j is the number of faces. To integrate the integral over each element e and the boundaries δk, a quadrature rule, $Q = \{q\}$ is used.

For academic purpose, the multi-D Burger's equations are considered and recalled here along with the definition of the viscosity coefficients Eq. (3.29):

$$\partial_t u(\vec{r},t) + \vec{\nabla}\cdot\left[\vec{n}\frac{u(\vec{r},t)^2}{2}\right] = \vec{\nabla}\cdot\Big(\mu(\vec{r},t)\vec{\nabla}u(\vec{r},t)\Big) = \vec{\nabla}\cdot\vec{g} \tag{3.29a}$$

$$R_e(\vec{r},t) = \partial_t\left(s(\vec{r},t)\right) + \vec{\nabla}\left(\vec{n}\Phi(\vec{r},t)\right) \leq 0 \tag{3.29b}$$

$$\mu(\vec{r},t) = \max\left(\mu_e(\vec{r},t), \mu_{max}(\vec{r},t)\right) \tag{3.29c}$$

$$\mu_{max}(\vec{r},t) = \frac{h}{2}|u(\vec{r},t)| \tag{3.29d}$$

$$\mu_e(\vec{r},t) = h^2\frac{\max\left(R_e(\vec{r},t), J\right)}{||s-\bar{s}||_\infty} \tag{3.29e}$$

where $u(\vec{r},t)$ is a conservative variable that depends on both space and time. The entropy function s and the conservative flux in the entropy residual R_e are defined as $s(\vec{r},t) = \frac{u(\vec{r},t)}{2}$ and $\Phi = \frac{u(\vec{r},t)^3}{3}$, respectively. The Burger's equation is known to admit an unique eigenvalue $\lambda = u(\vec{r},t)$. The vector $\vec{n}$ has the same definition as in SECTION. The jump J is assumed piecewise constant and details regarding its evaluation are given later in this SECTION. The normalization parameter $||s-\bar{s}||_\infty$ used in Eq. (3.29e) denotes the infinite norm over the entire computational domain of the quantity $s-\bar{s}$ where $\bar{s}$ is the average entropy over the computational domain as well.

The first step in the implementation of the EVM is the integration of the dissipative terms over each element of the mesh. The continuous finite element approach consists of multiplying each term by a test function and then, integrating over the computational domain. Since the dissipative terms are second-order spatial derivatives, an integration per part is performed leading to:

$$\vec{\nabla}\cdot\vec{g} \rightarrow \int_\Omega W^h \cdot \vec{\nabla}\cdot\vec{g}^h \mathrm{d}\Omega = \int_\Omega \vec{\nabla} W^h \cdot \vec{g}^h \mathrm{d}\Omega - \int_\Gamma \hat{n} \cdot \vec{g}^h W^h \mathrm{d}\Gamma \tag{3.30}$$

In Eq. (3.30), the integral over the domain Ω is transformed into a sum over the elements and evaluated by using the quadrature rule. The other term, consists of an integral over the boundary of the computational domain Γ. It requires to compute the dissipative flux g, and thus, the viscosity coefficient μ at the boundary. The evaluation of the integral over Γ can be avoided by assuming that the viscosity coefficient is zero at the boundary which yields:

$$\vec{\nabla}\cdot\vec{g} \rightarrow \int_\Omega W^h \cdot \vec{\nabla}\cdot\vec{g}^h \mathrm{d}\Omega = \int_\Omega \vec{\nabla} W^h \cdot \vec{g}^h \mathrm{d}\Omega \tag{3.31}$$

The dissipative term $\vec{g}^h$ is function of the viscosity coefficient and the derivative of the conservative variable u that need to be evaluated at the quadrature points. Obtaining the derivative values at the quadrature points with a continuous finite element discretization type is straightforward. On the other hand, computing the viscosity coefficient at the same quadrature points require a little bit more of computational work and is explained in the following.

The next step consists of determining the viscosity coefficient μ that is not obtained by solving a PDE, but computed on the fly from the definition recalled in Eq. (3.29c). The definition of the viscosity coefficient $\mu(\vec{r}, t)$ involves two other viscosity coefficients: a first-order viscosity coefficient $\mu_{max}(\vec{r}, t)$ that is an upper bound, and a high-order viscosity coefficient often also called entropy-viscosity coefficient that is denoted by $\mu_e(\vec{r}, t)$. A common element to the definition of $\mu_{max}(\vec{r}, t)$

and $\mu_e(\vec{r}, t)$ is the mesh size h that can vary through the computational domain and is defined as the shortest distance between two nodes of an element. Thus, when considering a 1-D mesh with linear test function, the local mesh size is simply Δx. For a shape regular mesh, the mesh size is finite and usually available through a function call. For instance, when using libMesh (REF), a function can be called in order to get the mesh size or diameter of the cell under consideration. Once the mesh size h is available, it remains to compute the local maximum eigenvalue, the residual R_e and the jump J.

Obtaining the local maximum eigenvalue is self explanatory using the test functions, $u_k^q = \sum_j u_{k,j} \phi_j^q$, and has to be done for each quadrature point in a given element k: at this point, the first-order viscosity coefficient μ_{max} is available at each quadrature point and is referred to as $\mu_{max,k}^q = \frac{h_k}{2} |u_k^q|$. The high-order viscosity coefficient is trickier to compute since it involves the entropy residual R_e at the quadrature points and the jumps J at the interface between cells. The entropy residual R_e is a PDE but is not discretized in a finite element sense. Instead, each term of the entropy residual is locally computed using the test functions but without integration over the computational domain as follows:

$$R_{e,k}^{q,n} = w_0 s_k^{n,q} + w_1 s_k^{n-1,q} + w_2 s_k^{n-2,q} + \vec{n} \cdot \sum_j \Phi_{k,j} \vec{\nabla} \phi_j^q \tag{3.32}$$

when considering three successive solutions $s_k^{n,q}$, $s_k^{n-1,q}$ and $s_k^{n-2,q}$ in time. The weights w_0, w_1 and w_2 are defined in (SECTION). The values of the entropy function s at the quadrature points is computed using the test function: $s_k^q = \sum_j s_{k,j} \phi_j^q = \sum_j \frac{u_{k,j}^2}{2} \phi_j^q$, which requires to access the values of u^2 at the nodes j. The same method is used for the conservative flux Φ. It is noted that the entropy residual can be recast under a non-conservative form as shown in Eq. (3.33) that can be easier to evaluate depending on what is available to the user.

$$R_e(\vec{r},t) = \partial_t \left(\frac{u(\vec{r},t)^2}{2} \right) + u(\vec{r},t)^2 \vec{\nabla} \left(\vec{n} u(\vec{r},t) \right) \tag{3.33}$$

It remains, now, to compute the jump J that is set constant in each element. In Galerkin continuous finite elements, the variables are continuous at the faces, but their derivative are discontinuous. Thus, the jump of the gradient of a variable to choose, seems to be a good entropy production indicator since it will inform us on the presence of a sharp discontinuity. In the remaining of this section, a generic method is detailed to compute the jump of the gradient of a variable when using Galerkin finite element method. Then, the jump used in the definition of the viscosity coefficient for solving Burger's equation is given.

To be more specific, let us consider an element k and its set of n boundaries $\delta k = \{\delta k_1, \cdots, \delta k_n\}$. We also assume that the outward normal $\hat{n}_i$ to each boundary δk_i is available to us. The objective is to compute the jump J_k of the gradient of the variable $v(\vec{r},t)$ for the element k. Since an element k shares n boundaries with n other elements of the computational domain, a jump $J_{k.i}$ can be computed for each boundary δk_i, and is defined as follows:

$$J_{k,i} = | \left(\vec{\nabla} v(\vec{r},t)_{k,i} - \vec{\nabla} v(\vec{r},t)_{neighbor,i} \right) \cdot \hat{n}_i | \tag{3.34}$$

where the quantity $\vec{\nabla} v(\vec{r},t)_{neighbor,i}$ denotes the gradient of $v(\vec{r},t)$ in the neighbor cell to the element k sharing the interface δk_i. The difference of gradients between the two elements sharing the interface δk_i is multiplied by the outward normal vector $\hat{n}_i$ to obtain the jump normal to the interface. Once all the jumps $J_{k,i}$ are computed for each face i of the element k (a loop over the faces i of element k applies), the jump J_k is computed by choosing the maximum over the $J_{k,i}$:

$$J_k = \max_i \left(J_{k,i} \right) \tag{3.35}$$

With the definition given in Eq. (3.35), the jump J_k is constant in each element k of the computational domain Ω. From this point, the entropy residual R_e and the jump J are known in the element k, at a given time n and at every quadrature points q. It remains to compute the normalization parameters $||s - \bar{s}||_\infty$ that is obtained from a post processing for every new non-linear iteration of the solver and thus is a function of time. The average value of the entropy function of the computational domain is computed from an integral as follows:

$$\bar{s} = \frac{1}{\Omega} \int_\Omega s(\vec{r}, t) \mathrm{d}\Omega \tag{3.36}$$

The high-order viscosity coefficient $\mu_{e,k}^{q,n}$ can now be computed at a given quadrature potions q and given time n:

$$\mu_{e,k}^{q,n} = h_k^2 \frac{\max \left(R_{e,k}^{q,n}, J_k^n \right)}{||s - \bar{s}||_\infty^n} \tag{3.37}$$

The definition of the viscosity coefficient μ from Eq. (3.38) follows:

$$\mu_k^{q,n} = \min \left(\mu_{e,k}^{q,n}, \mu_{max,k}^{q,n} \right) \tag{3.38}$$

From this point, all variables are known to compute the integral of the dissipative term $\int_k \mu \vec{\nabla} u(\vec{r}, t) \vec{\nabla} \phi = \sum_q \mu_k^{q,n} \vec{\nabla} u_k^{q,n} \vec{\nabla} \phi^q$.

4. APPLICATION OF THE ENTROPY VISCOSITY METHOD TO THE MULTI-D BURGER'S EQUATION:

The multi-D Burger's equation is solved using the entropy viscosity method described in (SECTION). The equation with the viscous regularization and the definition of the viscosity coefficients are recalled, and treatment of the boundary condition in also explained in Section 4.1. 1- and 2-D numerical results are presented in Section 4.2. The objective of this section is to present numerical results obtained with the entropy viscosity method for the simple hyperbolic scalar Burger's equation before dealing with hyperbolic system of equations. The multi-physics framework Moose [16] was to implement the multi-D Burger's equation. The code name is Badger.

4.1 The multi-D Burger's equation:

We recall the multi-D Burger's equation (Eq. (4.1a)) with the viscous regularization and the definition of the viscosity coefficient (Eq. (4.1b)).

$$\partial_t u(\vec{r},t) + \vec{\nabla}\cdot\left(\frac{u(\vec{r},t)^2}{2}\vec{n}\right) = \vec{\nabla}\cdot\Big(\mu(\vec{r},t)\vec{\nabla}u(\vec{r},t)\Big) \tag{4.1a}$$

$$\begin{cases} \mu(\vec{r},t) = \min\left(\mu_{max}(\vec{r},t), \mu_e(\vec{r},t)\right) \\ \mu_{max}(\vec{r},t) = \frac{h}{2}|u(\vec{r},t)| \\ \mu_e(\vec{r},t) = h^2 \frac{\max(R_e(\vec{r},t),J)}{||s(\vec{r},t)-\bar{s}(t)||_\infty} \end{cases} \tag{4.1b}$$

$$\begin{cases} R_e(\vec{r},t) = \partial_t s(\vec{r},t) + \vec{\nabla}\left(u(\vec{r},t)s(\vec{r},t)\right) \\ J = [u(\vec{r},t)s(\vec{r},t)] \end{cases} \tag{4.1c}$$

where $\vec{n}$ is a vector whom definition depends on the dimension of the domain: $\vec{n} = (1,0,0)$ in 1-D, $\vec{n} = (1,1,0)$ in 2-D and $\vec{n} = (1,1,1)$ in 3-D. The entropy function is denoted by s and is taken equal to the convex function $s(\vec{r},t) = u(\vec{r},t)^2/2$ for the two examples presented in Section 4.2. The continuous Galerkin finite element method described in (SECTION) along with the second-order temporal integrator $BDF2$ are used to discretize Eq. (4.1a). Such discretization requires to compute the flux at the boundary of the computational domain (EQUATION). Our implementation of the boundary condition for Burger's equation is based on the sign of the dot product $u(\vec{r},t)\vec{n}\cdot\hat{n}$ at the boundary, where $\hat{n}$ is the outward normal to the boundary. For Burger's equation it was demonstrated in SECTION that the eigenvalue is the solution itself $\lambda = u$. Being at the boundary, two cases have to be distinguished:

- $u(\vec{r},t)\vec{n}\cdot\hat{n}$ is negative: the wave enters the computational domain and thus, information needs to be supplied to the code. This boundary condition can be enforced either weakly or strongly. In the former case, the boundary value is specified in the input file, for instance, and used to compute the flux at the boundary. In the later case, the boundary value is still specified but strongly enforce with Dirichlet method.

- $u(\vec{r},t)\vec{n}\cdot\hat{n}$ is positive: the wave exits the computational domain. The flux is computed with the value of the solution from the last Krylov iteration supplied by the temporal implicit solver. Because of the iterative process, the information normally carried by the waves, is transmitted to the boundary. This approach is only valid with an implicit solver.

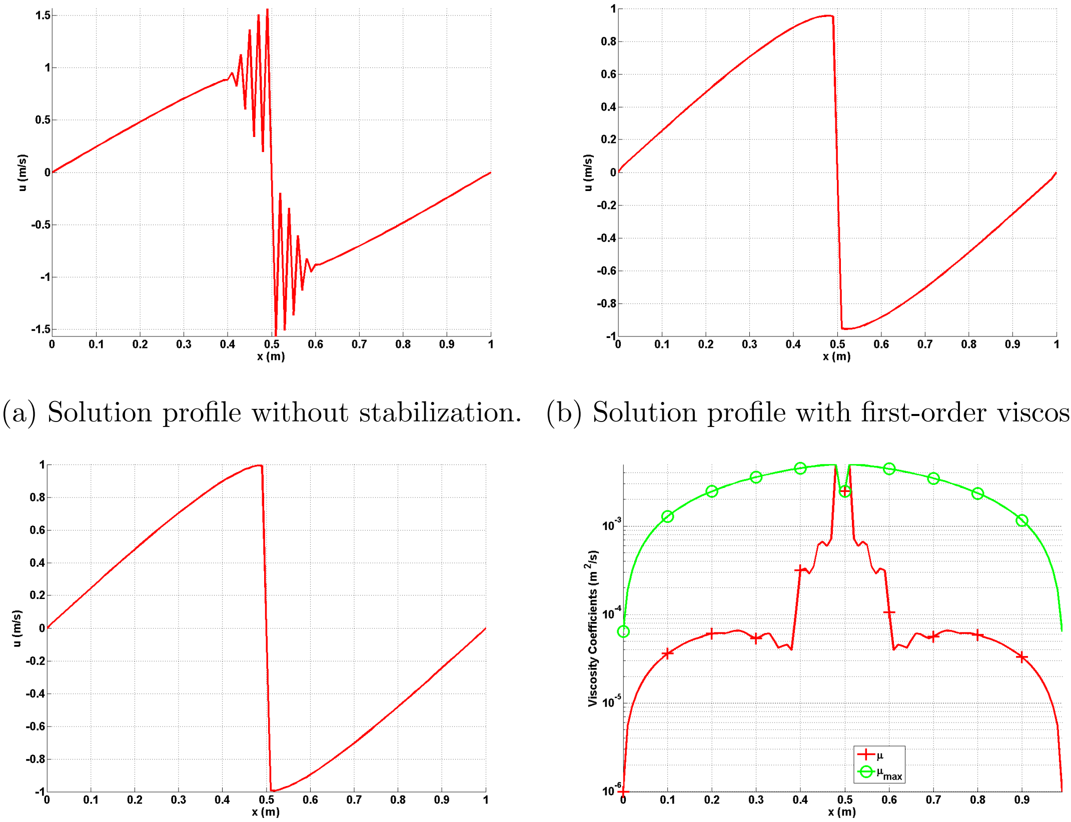
The above method for is a-dimensional and was used to obtain the numerical results presented in Eq. (4.2).

4.2 Numerical results:

Two typical numerical tests are presented in order to illustrate the main features of the entropy viscosity method when applied to the multi-D Burger's equation.

4.2.1 1-D numerical result:

We consider a 1-D computational domain of length $L = 1\ m$ discretized by an uniform mesh of 100 elements. The initial condition consists of a smooth sinusoidal function $u(x, 0) = \sin(2\pi x)$. The values at the left and right boundaries are set to zero and enforced by Dirichlet method. The numerical solution is run until $t = 0.2\ s$ with a CFL of one. In order to investigate the effect of the entropy viscosity method onto the numerical solution, three tests are performed. In the first test, the numerical solution is run with first-order viscosity coefficient which implies $\mu(x, t) = \mu_{max}(x, t)$ at all point of the computational domain and for all time. Then, the same run is performed using the definition of $\mu(x, t)$ recalled in Eq. (4.1b). Lastly, the code is run without stabilization, $\mu(x, t) = 0$. The objective of running the three previous tests, it to demonstrate the usefulness of the stabilization method and also, the gain in accuracy of a high-order stabilization method. Numerical results are shown from Fig. 4.1a through Fig. 4.1d.



(a) Solution profile without stabilization. (b) Solution profile with first-order viscosity.

(c) Solution profile with the EVM.

(d) Viscosity coefficient profiles.

In Fig. 4.1a, none stabilization method is used and numerical instabilities are observed in the shock region. When run with the over-dissipative first-order viscosity coefficient, the solution does not display any instabilities but the shock amplitude is smoothed as shown in Fig. 4.1b. Lastly, the numerical solution obtained with the EVM in Fig. 4.1c, is very close to the exact solution: the shock amplitude is preserved and the solution is stable. The viscosity coefficients are shown in Fig. 4.1d on a log-scale: the hight-order viscosity coefficient μ_e is peaked in the shock region and is small anywhere else. This behavior is expected and corresponds to the theoretical approach detailed in SECTION. It was demonstrated in [57] that high-order accuracy is preserved with the EVM when the solution is smooth (i.e. away from the shock region). It is also noticed the difference of order of magnitude between the high- and first-order viscosity coefficients away from the shock region.

4.2.2 2-D numerical result: Riemann problem.

We now consider a typical 2-D benchmark problem known as Riemann problem. The computational domain consists of a $1 \times 1\ m^2$ square and the following initial conditions are used:

$$u(\vec{r},0) = u_0 = \begin{cases} +0.5 \text{ for } x \leq 0.5 \text{ and } y \leq 0.5 \\ +0.8 \text{ for } x \geq 0.5 \text{ and } y \leq 0.5 \\ -0.2 \text{ for } x \leq 0.5 \text{ and } y \geq 0.5 \\ -1. \text{ for } x \geq 0.5 \text{ and } y \geq 0.5 \end{cases}$$

An uniform mesh of 100×100 elements is used. The solution is run until $t = 0.5\ s$ with a CFL of 0.5. The numerical solution and the viscosity coefficient profiles are plotted at $t = 0.2$ and $t = 0.5\ s$.

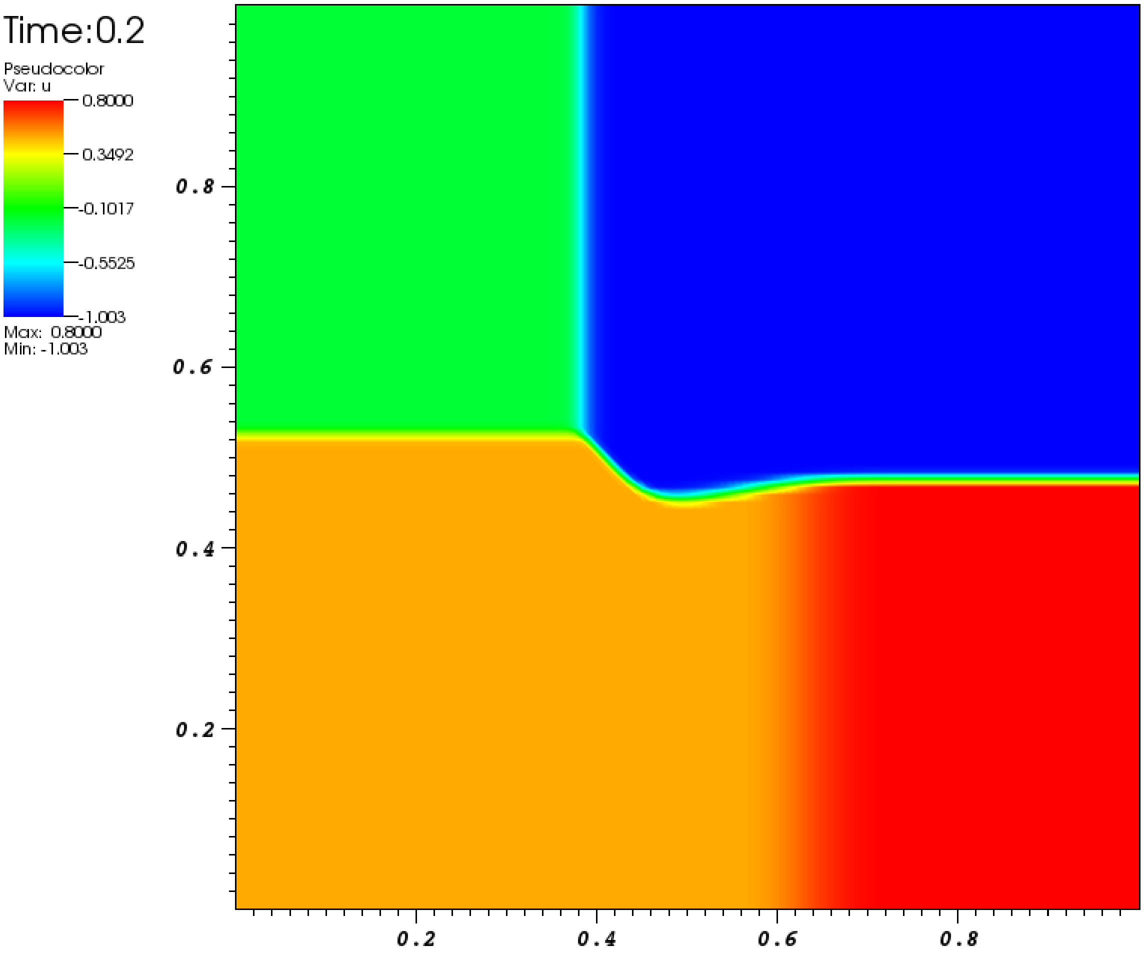


Figure 4.1: Solution profile at $t = 0.2\ s$.

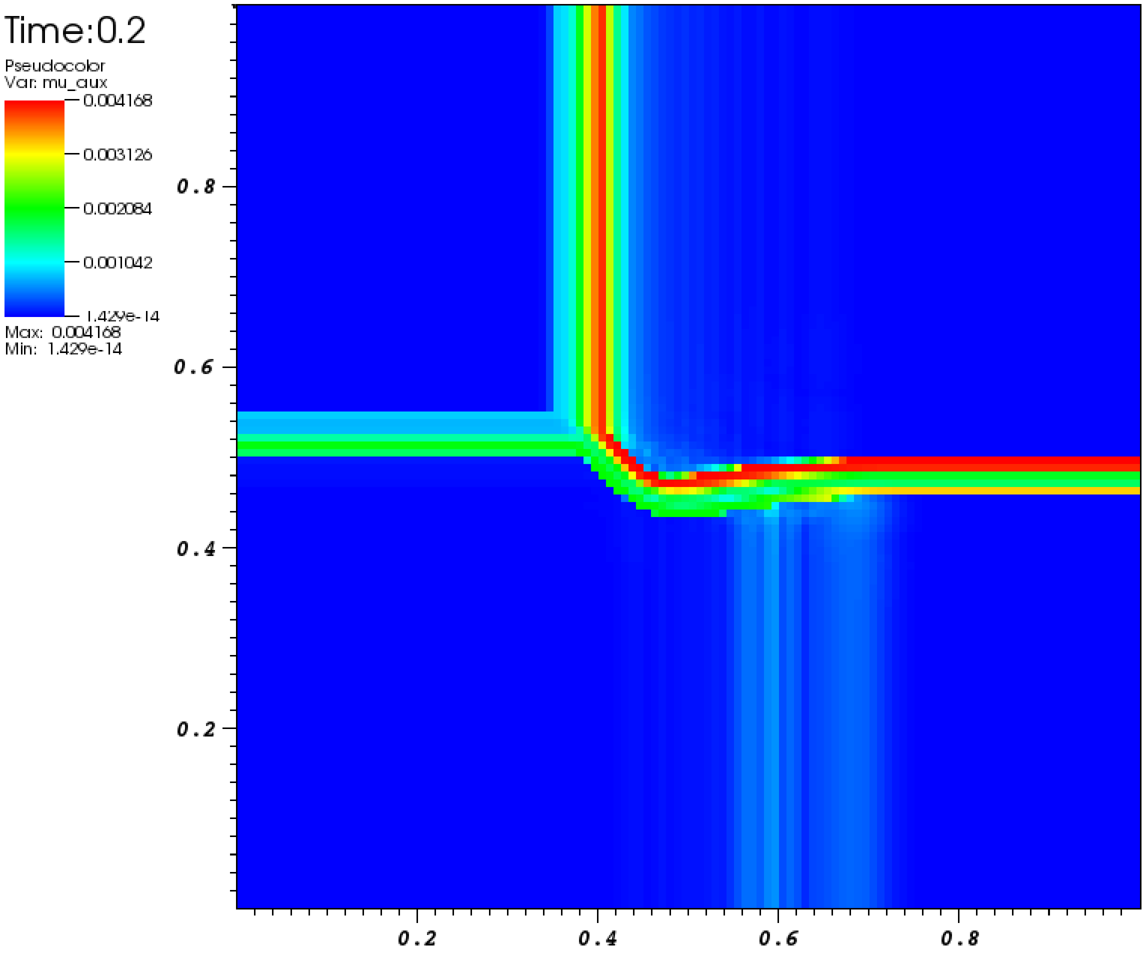


Figure 4.2: Solution profile at $t = 0.2$ s.

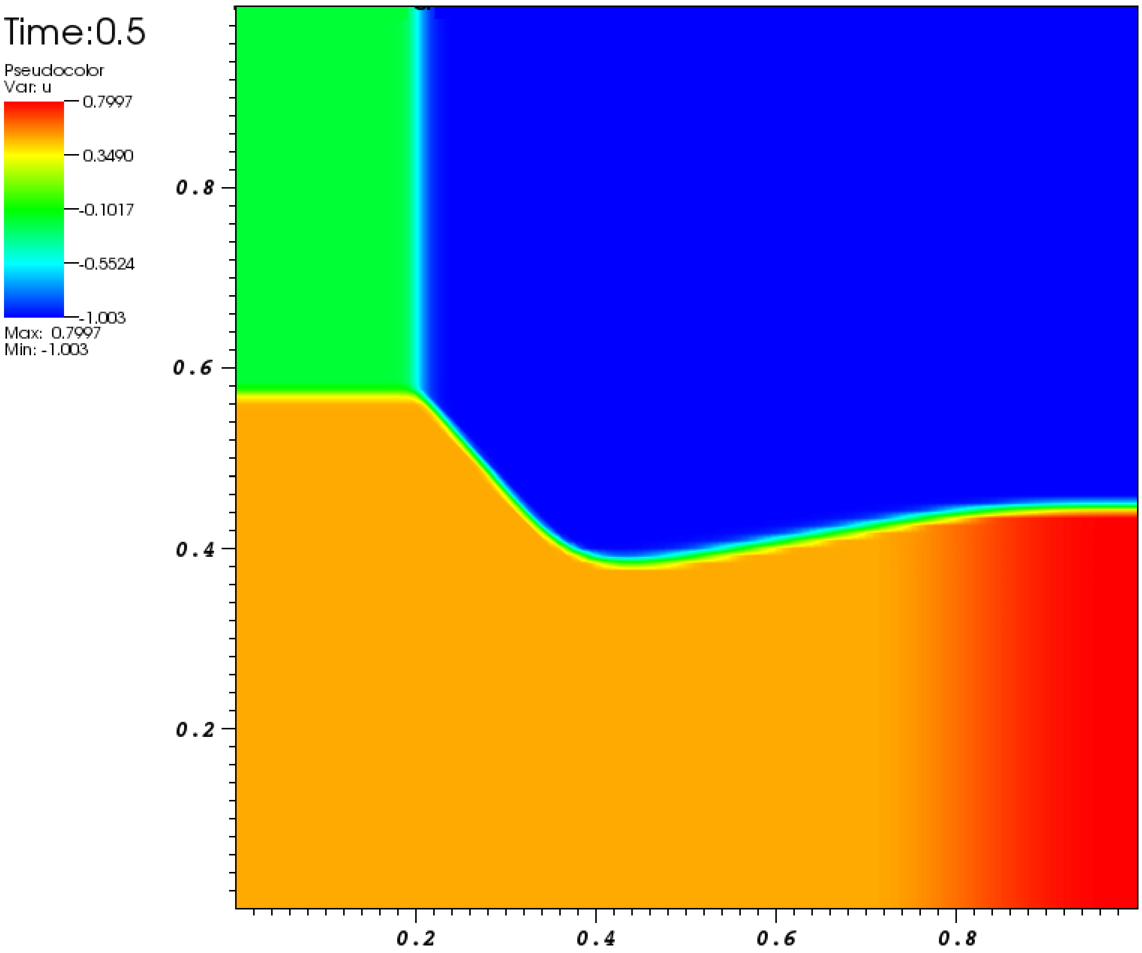


Figure 4.3: Solution profile at $t = 0.5$ s.

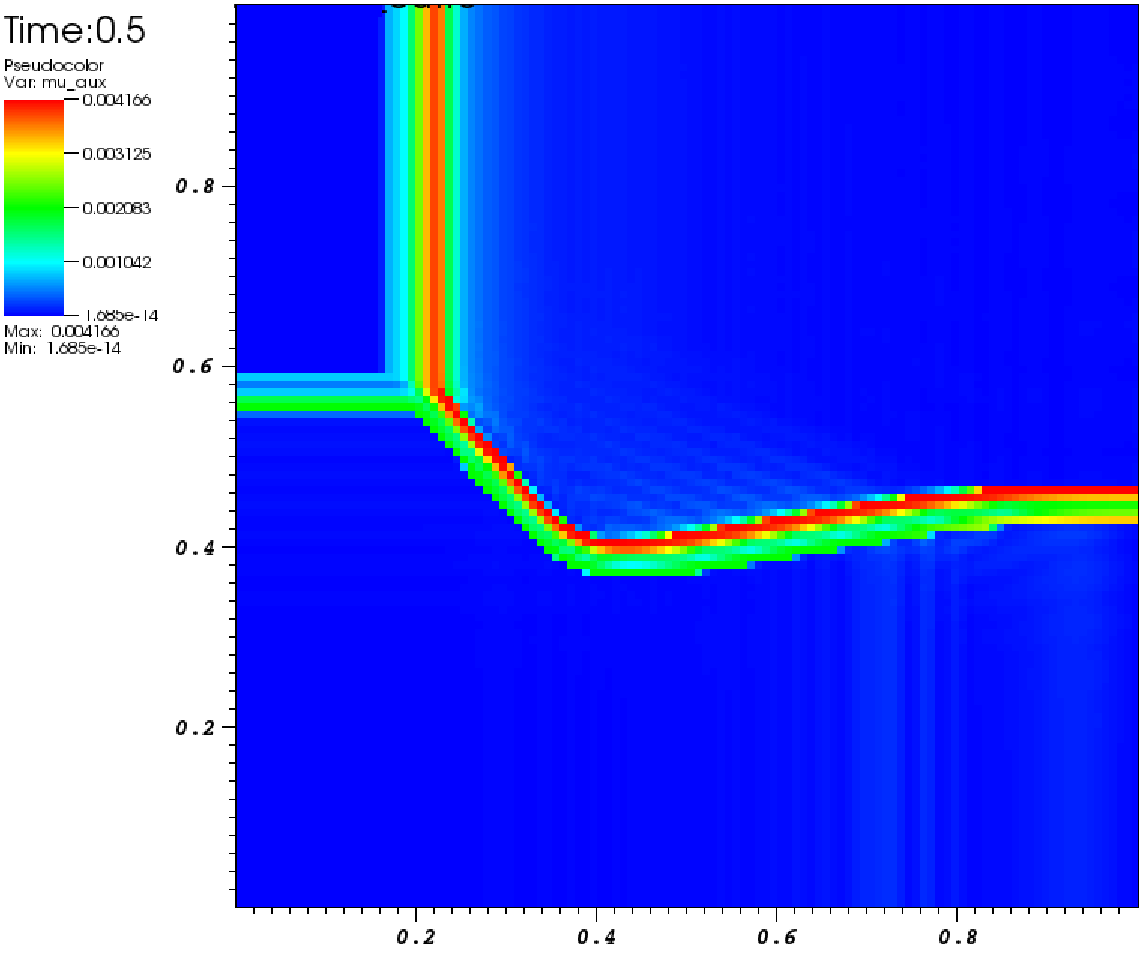


Figure 4.4: Solution profile at $t = 0.5$ s.

The numerical solution is plotted in Fig. 4.1 and Fig. 4.3 at $t = 0.2$ and $t = 0.5$ s, respectively. The numerical solution does not display any oscillations and the shocks are well resolved. The high-order viscosity coefficient is showed in Fig. 4.2 and Fig. 4.4: the shock is well tracked by the EVM and sufficient dissipation is only added in the shock regions, where saturation to the first-order viscosity is achieved.

The above examples are a simple illustration of the capabilities of the EVM when applied to hyperbolic scalar equations. We now focus our attention to the application of the EVM to various hyperbolic systems of equation.

5. APPLICATION OF THE ENTROPY VISCOSITY METHOD TO THE MULTI-D EULER EQUATIONS WITH VARIABLE AREA:

5.1 Introduction

Over the past years an increasing interest raised for computational methods that can solve both compressible and incompressible flows. In engineering applications, there is often the need to solve for complex flows where a near incompressible regime or low Mach flow coexists with a supersonic flow domain. For example, such flow are encountered in aerodynamic in the study of airships. In the nuclear industry, flows are nearly the incompressible regime but compressible effects cannot be neglected because of the heat source and thus needs to be accurately resolved.

When solving the multi-D Euler equations for a wide range of Mach numbers, multiple problems have to address: stability, accuracy and acceleration of the convergence in the low Mach regime. Because of the hyperbolic nature of the equations, shocks can form during transonic and supersonic flows, and require the use of the numerical methods in order to stabilize the scheme and correctly resolve the discontinuities. The literature offers a wide range of stabilization methods: flux-limiter [13, 14], pressure-based viscosity method ([40]), Lapidus method ([35, 41, 20]), and the entropy-viscosity method([28, 29]) among others. These numerical methods are usually developed using simple equation of states and tested for transonic and supersonic flows where the disparity between the acoustic waves and the fluid speed is not large since the Mach number is of order one. This approach leads to a well-known accuracy problem in the low Mach regime where the fluid velocity is smaller that the speed of sound by multiple order of magnitude. The numerical dissipative terms become ill-scaled in the low Mach regime and lead to the wrong numerical solution by changing the nature of the equations solved. This behavior is well documented in the literature [24, 56, 34] and often treated by performing a low Mach asymptotic study of the multi-D Euler equation. This method was originally used [24] to show convergence of the compressible multi-D Euler equations to the incompressible ones. Thus, by using the same method, the effect of the dissipative terms in the low Mach regime, can be understood and, when needed, a fix is developed in order to ensure the convergence of the equations to the correct physical solution. This approach was used as a fixing method for multiple well known stabilization methods alike Roe scheme ([39]) and SUPG [34] while preserving the original stabilization properties of shocks.

We propose, through this paper, to investigate how the entropy viscosity method, when applied to the multi-D Euler equations with variable area, behaves in the low Mach regime. This method was initially introduced by Guermond et al. to solve for the hyperbolic systems and has shown good results when used for solving the multi-D Euler equations with various discretization schemes. More importantly, it is simple to implement, can be used with unstructured grids, and its dissipative terms are consistent with the entropy minimum principle and proven valid for any equation of state under certain conditions [23].

5.2 The Entropy Viscosity Method

5.2.1 Background

The Euler equations are given by

$$\partial_t \rho + \vec{\nabla} \cdot (\rho \vec{u}) = 0 \tag{5.1a}$$

$$\partial_t\left(\rho\vec{u}\right) + \vec{\nabla}\cdot\left(\rho\vec{u}\otimes\vec{u} + P\mathbb{I}\right) = 0 \tag{5.1b}$$

$$\partial_t\left(\rho E\right) + \vec{\nabla}\cdot\left[\vec{u}\left(\rho E + P\right)\right] = 0 \tag{5.1c}$$

where ρ, $\rho\vec{u}$ and ρE are the density, the momentum and the total energy, respectively, and will be referred to as the conservative variables. $\vec{u}$ is the fluid velocity and its specific internal energy is denoted by $e = E - \frac{u^2}{2}$. An equation of state, dependent upon ρ and e, is used to compute the pressure P. The tensor product $\vec{a}\otimes\vec{b}$ is such that $(\vec{a}\otimes\vec{b})_{i,j} = a_i b_j$. The identity tensor is denoted by $\mathbb{I}$.

Next, the entropy viscosity method [28, 29, 22, 57] applied to Eq. (5.1) is recalled. The method consists of adding dissipative terms with a viscosity coefficient modulated by the entropy production; this allows for a high-order accuracy when the solution is smooth (provided that the spatial and temporal discretizations also are high order). The derivation of the viscous regularization (or dissipative terms) is carried out to be consistent with the entropy minimum principle; details and proofs of the derivation can be found in [23]. The viscous regularization thus obtained is valid for any equation of state as long as the physical entropy function s is such that $-s$ is a convex function with respect to the internal energy e and the specific volume $1/\rho$. Euler equations with viscous regularization are given in Eq. (5.2):

$$\partial_t\rho + \vec{\nabla}\cdot\left(\rho\vec{u}\right) = \vec{\nabla}\cdot\left(\kappa\vec{\nabla}\rho\right) \tag{5.2a}$$

$$\partial_t\left(\rho\vec{u}\right) + \vec{\nabla}\cdot\left(\rho\vec{u}\otimes\vec{u} + P\mathbf{I}\right) = \vec{\nabla}\cdot\left(\mu\rho\vec{\nabla}^s\vec{u} + \kappa\vec{u}\otimes\vec{\nabla}\rho\right) \tag{5.2b}$$

$$\partial_t\left(\rho E\right) + \vec{\nabla}\cdot\left[\vec{u}\left(\rho E + P\right)\right] = \vec{\nabla}\cdot\left(\kappa\vec{\nabla}\left(\rho e\right) + \frac{1}{2}||\vec{u}||^2\kappa\vec{\nabla}\rho + \rho\mu\vec{u}\vec{\nabla}\vec{u}\right) \tag{5.2c}$$

where κ and μ are positive viscosity coefficients. $\vec{\nabla}^s\vec{u}$ denotes the symmetric gradient operator that guarantees the method to be rotationally invariant [23]. The viscosity coefficients are key ingredients in the viscous regularization of Eq. (5.2). Other stabilization approaches have been proposed in the literature, for instance, the Lapidus method [20, 35] or pressure-based viscosity methods [40]. Here, we follow the work of Guermond et al. and define the viscosity coefficients, κ and μ, based on the local entropy production. These coefficients are numerically evaluated using the local entropy residual $R_e(\vec{r},t)$ defined in Eq. (5.3); $R_e(\vec{r},t)$ is known to be peaked in shocks and vanishingly small elsewhere [55].

$$R_e(\vec{r},t) := \partial_t s + \vec{u}\cdot\vec{\nabla} s \tag{5.3}$$

In the current version of the method, the ratio of κ to μ is defined as a numerical Prandlt number, $\mathrm{Pr} = \kappa/\mu$. Pr is a user-defined parameter and is usually taken in the range $[0.001; 1]$. Since the entropy residual $R_e(\vec{r},t)$ may be extremely large in shocks, the definition of the viscosity coefficients also includes a first-order viscosity coefficient that serves as upper bound for the entropy-based viscosity coefficients. The first-order viscosity coefficients, denoted by $\mu_{\max}$ and $\kappa_{\max}$, are chosen so that the numerical scheme becomes equivalent to the upwind scheme when the first-order coefficients are employed. The upwind scheme is known to be over-dissipative but guarantees monotonicity [55]. In practice, the viscosity coefficients only saturate to the first-order viscosity coefficients in shocks and are much smaller elsewhere, hence avoiding over-dissipation due to the upwind method. The first-order viscosity coefficients $\mu_{\max}$ and $\kappa_{\max}$ are equal and set proportional to the largest local eigenvalue $||\vec{u}|| + c$:

$$\mu_{\max}(\vec{r},t) = \kappa_{\max}(\vec{r},t) = \frac{h}{2}\left(||\vec{u}(t,\vec{r})|| + c(t,\vec{r})\right), \tag{5.4}$$

where h denotes the local grid size (for higher than linear finite element representations, h is defined as the ratio of the grid size to the polynomial order of the test functions used, see Eq. 2.4 in [57]). For simplicity, the first-order viscosity coefficient will only be referred to as the $\kappa_{\max}(\vec{r}, t)$. In practice, these quantities are evaluated within a given cell K at quadrature points:

$$\kappa^K_{\max}(\vec{r}_q, t) = \frac{h_K}{2}\left(||\vec{u}(t, \vec{r}_q)|| + c(t, \vec{r}_q)\right), \tag{5.5}$$

where $\vec{r}_q$ denotes the position of a quadrature point. As stated earlier, the entropy viscosity coefficients, which we denote by κ_e and μ_e, are set proportional to the entropy production evaluated by computing the local entropy residual R_e. The definitions also include the inter-element jump of the entropy flux $J[s]$, allowing for the detection of discontinuities other than shocks (e.g., contact).

$$\mu^K_e(\vec{r}_q, t) = h^2_K \frac{\max\left(|R^K_e(\vec{r}_q, t)|, J^K[s](t)\right)}{||s - \bar{s}||_\infty} \tag{5.6a}$$

$$\kappa^K_e(\vec{r}_q, t) = Pr\, \mu^K_e(\vec{r}_q, t) \tag{5.6b}$$

where $||\cdot||_\infty$ and $\bar{\cdot}$ denote the L_∞-norm and the average operator over the entire computational domain, respectively. The definition of the entropy jump $J[s]$ is spatial discretization-dependent and examples of definitions can be found in [57] for discontinuous Galerkin discretization. For continuous finite element methods (FEM), the jump of a given quantity is defined as the inter-element change in its normal derivative ($\partial_n = \vec{\nabla} \cdot \vec{n}$) along the common face separating the two elements. We take the largest value over all faces f present on the boundary ∂K of element K:

$$J^K[s](t) = \max_{f \in \partial K} \max_{\vec{r}_q \in f} [\![\vec{\nabla} s(\vec{r}_q, t) \cdot \vec{n}(\vec{r}_q)]\!]_f\,, \tag{5.7}$$

where $[\![a(\vec{r}_q)]\!]_f$ denotes the inter-element jump in $a(\vec{r})$ at quadrature point $\vec{r}_q$ on face f. The denominator $||s-\bar{s}||_\infty$ is used for dimensionality purposes. Currently, there are no theoretical justification for choosing the denominator beyond a dimensionality argument. Finally, the viscosity coefficients μ and κ are as follows:

$$\mu(\vec{r},t) = \min\Big(\mu_e(\vec{r},t)\,,\,\mu_{\max}(\vec{r},t)\Big) \quad \text{and} \quad \kappa(\vec{r},t) = \min\Big(\kappa_e(\vec{r},t)\,,\,\kappa_{\max}(\vec{r},t)\Big). \tag{5.8}$$

Given these definitions, we have the following properties. In shock regions, the entropy viscosity coefficients will experience a peak because of entropy production and thus will saturate to the first-order viscosity. The first-order coefficients are known to be over-dissipative and will smooth out any oscillatory behavior. Elsewhere in the domain, entropy production will be small and the viscosity coefficients μ and κ will remain small. High-order accuracy for entropy-based viscous stabilization was demonstrated using several 1-D shock tube examples and various 2-D tests [28, 29, 57].

5.2.2 Issues in the Low-Mach Regime

In the low-Mach Regime, the flow is known to be isentropic, resulting in very little entropy production. Since the entropy viscosity method is directly based on the evaluation of the local entropy production, it is of interest to study how the entropy viscosity coefficients μ_e and κ_e scale in the low-Mach regime. In practice, the entropy residual R_e will be very small in that regime and so will be the denominator $||s-\bar{s}||_\infty$, thus making the definition of the viscosity coefficients in Eq. (5.6) undetermined and likely ill-scaled. One possible approach would consist of expanding the numerator and denominator in terms of the Mach number and deriving its limit when the Mach number goes to zero. Such derivation may not be straightforward, especially for general equations of state. However, this can be avoided by noting that the entropy residual R_e can be recast as a function of pressure, density, velocity, and speed of sound as shown in Eq. (5.9) of Section 5.3.1. This alternate entropy residual definition is the basis for the low-Mach analysis carried out in this paper and possesses several advantages that are detailed next.

5.3 An All-speed Reformulation of the Entropy Viscosity Method

In this section, the entropy residual R_e is recast as a function of pressure, density, velocity and speed of sound. Then, a low-Mach asymptotic study is carried out for the Euler equations with viscous regularization in order to derive an appropriate normalization parameter that is valid in the low-Mach regime as well as for transonic and supersonic flows.

5.3.1 New Definition of the Entropy Production Residual

The first step in defining viscosity coefficients that behave well in the low-Mach limit is to recast the entropy residual in terms of thermodynamic variables. This provides physical insight on possible normalization choices that can be valid in both low-Mach and transonic flows. The alternate definition of the entropy residual is given in Eq. (5.9). The derivation that leads to this equation is provided in Appendix B.

$$R_e(\vec{r},t) := \partial_t s + \vec{u}\cdot\vec{\nabla} s = \frac{\mathrm{D}s}{\mathrm{D}t} = \frac{s_e}{P_e}\left(\underbrace{\frac{\mathrm{D}P}{\mathrm{D}t} - c^2\frac{\mathrm{D}\rho}{\mathrm{D}t}}_{\widetilde{R_e}(\vec{r},t)}\right), \tag{5.9}$$

where $\frac{\mathrm{D}}{\mathrm{D}t}$ denotes the material derivative ($\frac{\mathrm{D}}{\mathrm{D}t} := \frac{\partial}{\partial t} + \vec{u}\cdot\vec{\nabla}$), and x_y is the standard shortcut notation for the partial derivative of x with respect to y, e.g., $P_e := \frac{\partial P}{\partial e}$. The entropy residuals R_e and $\widetilde{R_e}$ are proportional to one another and will experience similar variations in space and time. Thus, one may elect to employ $\widetilde{R_e}$ instead of R_e

for the evaluation of the local entropy residual. The new expression presents several advantages:

- an analytical expression of the entropy function s is no longer needed: the residual $\widetilde{R_e}$ is evaluated using the local values of pressure, density, velocity and speed of sound. Deriving an entropy function for some complex equation of states may be difficult;
- suitable normalizations for the residual $\widetilde{R_e}$ can be devised. Examples include the pressure itself or combinations of the density, the speed of sound and the norm of the velocity, i.e., ρc^2, $\rho c||\vec{u}||$ or $\rho||\vec{u}||^2$.

Denoting the normalization of $\widetilde{R_e}$ by norm_P, the entropy-based viscosity coefficients μ_e and κ_e can be re-defined as follows:

$$\mu_e^K(\vec{r},t) = h_K^2 \frac{\max\Big(|\widetilde{R_e}^K(\vec{r}_q,t)|\,,||\vec{u}(\vec{r}_q,t)||J^K[P](t)\,,||\vec{u}(\vec{r}_q,t)c^2(\vec{r}_q,t)||J^K[\rho](t)\Big)}{\text{norm}_P^\mu}\,, \tag{5.10a}$$

and

$$\kappa_e^K(\vec{r},t) = h_K^2 \frac{\max\Big(|\widetilde{R_e}^K(\vec{r}_q,t)|\,,||\vec{u}(\vec{r}_q,t)||J^K[P](t)\,,||\vec{u}(\vec{r}_q,t)c^2(\vec{r}_q,t)||J^K[\rho](t)\Big)}{\text{norm}_P^\kappa}\,. \tag{5.10b}$$

Note that now the jump operator acts on the variables appearing in $\widetilde{R_e}$, namely, pressure and density. The μ and κ coefficients are kinematic viscosities (units of m^2/s); the normalization parameters norm_P are thus in units of pressure, hence the use of the subscript P. Note also that we are not requiring the same normalization for both μ_e and κ_e so the entropy viscosity coefficients can be different. The low-Mach asymptotic study presented next will determine the proper normalization.

5.3.2 Asymptotic Study in the Low-Mach Regime

The Euler equations with viscous stabilization, Eq. (5.6), bear some similarities with the Navier-Stokes equations in the sense that dissipative terms (containing second-order spatial derivatives) are present in both sets of equations. An abundant literature exists regarding the low-Mach asymptotics of the Navier-Stokes equations [24, 56, 34, 45]. The asymptotic study presented here is inspired by the work of Muller et al. [45] where an asymptotic derivation for the Navier-Stokes was presented. We remind the reader that the objective is to determine appropriate scaling for the entropy viscosity coefficients so that the dissipative terms remain well-scaled for two limit cases: (i) the isentropic limit where Euler equations degenerate to an incompressible system of equations in the low-Mach limit and (ii) the non-isentropic limit with formation of shocks. The isentropic limit of Euler equations with viscous regularization should yield the incompressible fluid flows results in the low-Mach limit, namely, that the pressure fluctuations are of the order M^2 and that the velocity satisfies the divergent constraint $\vec{\nabla}\cdot\vec{u}_0 = 0$ [24, 56, 34]. For non-isentropic situations, shocks may form for any value of Mach number and the minimum entropy principle should still be verified so that numerical oscillations, if any, be controlled by the entropy viscosity method independently of the value of the Mach number. Our objective is to determine the appropriate scaling for the Reynolds and Péclet numbers, Re_∞ and Pé_∞, in these two limit cases.

In this Section, we are interested in the isentropic limit, the non-isentropic case is treated later. The first step in the study of the limit cases (i) and (ii) is to re-write Eq. (5.2) in a non-dimensional manner. In order to do so, the following variables are introduced:

$$\rho^* = \frac{\rho}{\rho_\infty},\ u^* = \frac{u}{u_\infty},\ P^* = \frac{P}{\rho_\infty c_\infty^2},\ E^* = \frac{E}{c_\infty^2},$$
$$x^* = \frac{x}{L_\infty},\ t^* = \frac{t}{L_\infty/u_\infty},\ \mu^* = \frac{\mu}{\mu_\infty},\ \kappa^* = \frac{\kappa}{\kappa_\infty}, \quad (5.11)$$

where the subscript ∞ denote the far-field or stagnation quantities and the superscript $*$ stands for the adimensional variables. The far-field reference quantities are chosen such that the dimensionless flow quantities are of order 1. The reference Mach number is given by

$$M_\infty = \frac{u_\infty}{c_\infty}, \quad (5.12)$$

where c_∞ is a reference value for the speed of sound. Then, the scaled Euler equations with viscous regularization are:

$$\partial_{t^*}\rho^* + \vec{\nabla}^*\cdot(\rho^*\vec{u}^*) = \frac{1}{\text{Pé}_\infty}\vec{\nabla}^*\cdot(\kappa^*\vec{\nabla}^*\rho^*) \quad (5.13a)$$

$$\partial_{t^*}(\rho^*\vec{u}^*) + \vec{\nabla}^*\cdot(\rho^*\vec{u}^*\otimes\vec{u}^*) + \frac{1}{M_\infty^2}\vec{\nabla}^*P^* = \frac{1}{\text{Re}_\infty}\vec{\nabla}^*\cdot\left(\rho^*\mu^*\vec{\nabla}^{s,*}\vec{u}^*\right)$$
$$+ \frac{1}{\text{Pé}_\infty}\vec{\nabla}^*\cdot\left(\vec{u}^*\otimes\kappa^*\vec{\nabla}^*\rho^*\right) \quad (5.13b)$$

$$\partial_{t^*}(\rho^*E^*) + \vec{\nabla}^*\cdot[\vec{u}^*(\rho^*E^* + P^*)] = \frac{1}{\text{Pé}_\infty}\vec{\nabla}^*\cdot\left(\kappa^*\vec{\nabla}^*(\rho^*e^*)\right)$$
$$+ \frac{M_\infty^2}{\text{Re}_\infty}\vec{\nabla}^*\cdot\left(\vec{u}^*\rho^*\mu^*\vec{\nabla}^{s,*}\vec{u}^*\right) + \frac{M_\infty^2}{2\text{Pé}_\infty}\vec{\nabla}^*\cdot\left(\kappa^*(u^*)^2\vec{\nabla}^*\rho^*\right), \quad (5.13c)$$

where the numerical Reynolds (Re_∞) and Péclet (Pé_∞) numbers are defined as follows:

$$\mathrm{Re}_\infty = \frac{u_\infty L_\infty}{\mu_\infty} \text{ and } \mathrm{P\acute{e}}_\infty = \frac{u_\infty L_\infty}{\kappa_\infty}\,. \tag{5.14}$$

Note that the Prandlt number used in the original version of the entropy viscosity method is simply given by

$$\mathrm{Pr}_\infty = \mathrm{P\acute{e}}_\infty / \mathrm{Re}_\infty\,. \tag{5.15}$$

For simplicity, we use here the ideal gas equation of state; its non-dimensionalized expression is given by

$$P^* = (\gamma - 1)\,\rho^* \left(E^* - \frac{1}{2} M_\infty^2 (u^*)^2 \right) = (\gamma - 1)\,\rho^* e^*\,. \tag{5.16}$$

The numerical Reynolds and Péclet numbers defined in Eq. (5.14) are related to the entropy viscosity coefficients μ_∞ and κ_∞. Thus, once a scaling (in powers of M_∞) is obtained for Re_∞ and $\mathrm{P\acute{e}}_\infty$, the corresponding normalization parameters norm_P^μ and norm_P^κ will automatically be set. For brevity, the superscripts * are omitted in the remainder of this section.

In the low-Mach isentropic limit, shocks cannot form and the compressible Euler equations are known to converge to the incompressible equations when the Mach number tends to zero. When adding dissipative terms, as is the case with the entropy viscosity method, the main properties of the low-Mach asymptotic limit have to be preserved. We begin by expanding each variable in powers of the Mach number. As an example, the expansion for the pressure is given by:

$$P(\vec{r}, t) = P_0(\vec{r}, t) + P_1(\vec{r}, t) M_\infty + P_2(\vec{r}, t) M_\infty^2 + \dots \tag{5.17}$$

By studying the resulting momentum equations for various powers of M_∞, we observe the following: the leading order and first-order pressure terms, P_0 and P_1, are spatially constant if and only if $\text{Re}_\infty = \text{Pé}_\infty = 1$. In this case, we have, at order M_∞^{-2}:

$$\vec{\nabla} P_0 = 0 \tag{5.18a}$$

and, at order M_∞^{-1},

$$\vec{\nabla} P_1 = 0\,. \tag{5.18b}$$

Using the scaling $\text{Re}_\infty = \text{Pé}_\infty = 1$, the leading-order expressions for the continuity, momentum, and energy equations are:

$$\partial_t \rho_0 + \vec{\nabla}\cdot(\rho\vec{u})_0 = \vec{\nabla}\cdot(\kappa\vec{\nabla}\rho)_0 \tag{5.19a}$$

$$\partial_t(\rho\vec{u})_0 + \vec{\nabla}\cdot(\rho\vec{u}\otimes\vec{u})_0 + \vec{\nabla}P_2 = \vec{\nabla}\cdot(\rho\mu\vec{\nabla}^s\vec{u} + \kappa\vec{u}\otimes\vec{\nabla}\rho)_0 \tag{5.19b}$$

$$\partial_t(\rho E)_0 + \vec{\nabla}\cdot\left[\vec{u}(\rho E + P)\right]_0 = \vec{\nabla}\cdot(\kappa\vec{\nabla}(\rho e))_0 \tag{5.19c}$$

where the notation $(fg)_0$ means that we only keep the 0^{th} order terms in the product fg. The leading-order of the equation of state is given by

$$P_0 = (\gamma - 1)(\rho E)_0. \tag{5.20}$$

Using Eq. (5.20), the energy equation can be recast as a function of the leading-order pressure, P_0, as follows:

$$\partial_t P_0 + \gamma\vec{\nabla}\cdot\left(\vec{u}P\right)_0 = \vec{\nabla}\cdot(\kappa\vec{\nabla}(P))_0 \tag{5.21}$$

From Eq. (5.18a), we infer that P_0 is spatially constant. Thus, Eq. (5.21) becomes

$$\frac{1}{\gamma P_0}\frac{dP_0}{dt} = -\vec{\nabla}\cdot\vec{u}_0 \tag{5.22}$$

and, at steady state, we have

$$\vec{\nabla}\cdot\vec{u}_0 = 0\,. \tag{5.23}$$

That is, the leading-order of velocity is divergence-free. The same reasoning can be applied to the leading-order of the continuity equation (Eq. (5.19a)) to show that the material derivative of the density is zero:

$$\frac{\mathrm{D}\rho_0}{\mathrm{D}t} := \partial_t\rho_0 + \vec{u}_0\cdot\vec{\nabla}\cdot\rho_0 = 0\,. \tag{5.24}$$

Therefore, we conclude that by setting the Reynolds and Péclet numbers to one, the incompressible fluid results are retrieved in the low-Mach limit when employing the compressible Euler equations with viscous regularization terms present. In addition, the scaling of the Prandtl number can also be obtained using Eq. (5.15), hence clarifying the use of the numerical Prandtl in the original entropy viscosity method [28].

5.3.3 Scaling of Re_∞ and $Pé_\infty$ for non-isentropic flows

Next, we consider the non-isentropic case. Recall that even subsonic flows can present shocks (for instance, a step initial condition in the pressure will trigger shock formation, independently of the Mach number). The non-dimensionalized form of the Euler equations given in Eq. (5.13) provides some insight on the dominant terms as a function of the Mach number. This is particular obvious in the momentum equation, Eq. (5.13b), where the gradient of pressure is scaled by $1/M_\infty^2$. In the non-isentropic case, we no longer have $\frac{\vec{\nabla} P}{M_2} = \vec{\nabla} P_2$ and this pressure gradient term may need to be stabilized by some dissipative terms of the same scaling so as to prevent spurious oscillations from forming. This leads to the following three possible requirements regarding the non-dimensionalized Reynolds and Péclet numbers for non-isentropic flows: (a) $\text{Re}_\infty = M_\infty^2$ and $\text{Pé}_\infty = 1$, (b) $\text{Re}_\infty = 1$ and $\text{Pé}_\infty = M_\infty^2$, or (c) $\text{Re}_\infty = \text{Pé}_\infty = M_\infty^2$. Any of these choices will also affect the stabilization of the continuity and energy equations. For instance, using a Péclet number equal to M_∞^2 may effectively stabilize the continuity equation in the shock region but this may also add an excessive amount of dissipation for subsonic flows at the location of the contact wave. Such a behavior may not be suitable for accuracy purpose, making options (b) and (c) inappropriate. The same reasoning, left to the reader, can be carried out for the energy equation (Eq. (5.13c)) and results in the same conclusion. The remaining choice, option (a), has the proper scaling: in this case, only the dissipation terms involving $\vec{\nabla}^{s,*}\vec{u}^*$ scale as $1/M_\infty^2$ since $\text{Re}_\infty = M_\infty^2$, leaving unaffected the regularization of the continuity equation since $\text{Pé}_\infty = 1$.

5.3.4 New normalization for the entropy residual

The study of the above limit cases yields two different possible scalings for the Reynolds number: $\text{Re}_\infty = 1$ in the isentropic case and $\text{Re}_\infty = M_\infty^2$ for non-isentropic case, whereas the numerical Péclet number always scales as one. In order to have a stabilization method valid for a wide range of Mach numbers, including situations with shocks, these two scalings should be combined in a unique definition. The non-dimensionalized entropy residual, $\widetilde{R_e}^*$, scales differently flow types (isentropic and non-isentropic) [2]. For isentropic flows, the non-dimensionalized entropy residual is known to scale as the Mach number. For non-isentropic flows, the non-dimensionalized entropy residual is large and presents a peak at the location of the shock. Thus, by inspecting the *local* variation of the non-dimensionalized entropy residual, an appropriate transition for the scaling of the Reynolds number can be obtained:

$$\mathrm{Re}_\infty = \begin{cases} M^2 & \text{if } \left|\widetilde{R_e}^*\right| \geq M \text{ (i.e., non-isentropic flow)} \\ 1 & \text{otherwise} \end{cases} . \tag{5.25}$$

Now that we have determined a scaling for Re_∞ and Re_∞, the normalization parameters norm_P^μ and norm_P^κ can be finalized. For brevity, only the steps leading to the derivation of norm_P^κ are provided; the algebra for norm_P^μ is similar. Using the definition of the viscosity coefficients given in Eq. (5.10) and the scaling of Eq. (5.11), it can be shown that:

$$\kappa_\infty = \frac{\rho_\infty c_\infty^2 u_\infty L}{\mathrm{norm}_{P,\infty}^\kappa} , \tag{5.26}$$

where $\mathrm{norm}_{P,\infty}$ is the reference far-field quantity for the normalization parameter norm_P. Substituting Eq. (5.26) into Eq. (5.14) and recalling that the numerical Péclet number scales as unity, we obtain:

$$\mathrm{norm}_{P,\infty}^\kappa = \mathrm{P\acute{e}}_\infty \rho_\infty c_\infty^2 = \rho_\infty c_\infty^2 . \tag{5.27}$$

Eq. (5.27) provides a proper normalization factor to define the κ viscosity coefficient. Similarly, the normalization parameter norm_P^μ for the μ viscosity coefficient is derived for the two cases given in Eq. (5.25):

$$\mathrm{norm}_P^\mu = \mathrm{Re}_\infty \rho_\infty c_\infty^2 = \begin{cases} \rho \|\vec{u}\|^2 & \text{if } \left|\widetilde{R_e}^*\right| \geq M \text{ (i.e., non-isentropic flow)} \\ \rho c^2 = \mathrm{norm}_P^\kappa & \text{otherwise} \end{cases} . \tag{5.28}$$

Finally, we summarize the definition of the viscosity coefficients μ and κ devised here:

$$\mu(\vec{r},t) = \min\Big(\mu_{\max}(\vec{r},t), \mu_e(\vec{r},t)\Big) \text{ and } \kappa(\vec{r},t) = \min\Big(\mu_{\max}(\vec{r},t), \kappa_e(\vec{r},t)\Big) \tag{5.29a}$$

where the first-order viscosity is given by

$$\kappa_{\max}(\vec{r},t) = \mu_{\max}(\vec{r},t) = \frac{h}{2}\Big(||\vec{u}|| + c\Big) \tag{5.29b}$$

and the entropy viscosity coefficients by

$$\kappa_e(\vec{r},t) = \frac{h^2 \max(\widetilde{R_e}, J)}{\rho c^2} \text{ and } \mu_e(\vec{r},t) = \frac{h^2 \max(\widetilde{R_e}, J)}{\text{norm}_P^{\mu}} \tag{5.29c}$$

with the jumps given by

$$J = ||\vec{u}|| \max\Big([[\vec{\nabla} P \cdot \vec{n}]], c^2[[\vec{\nabla}\rho \cdot \vec{n}]]\Big) \tag{5.29d}$$

where norm_P^{κ} is computed from Eq. (5.28). The jump J is a function of the jump of pressure and density gradients across the face with respect to its normal vector $\vec{n}$. Then, the largest value over all faces is determined and used in the definition of the viscosity coefficients.

With the definition of the viscosity coefficients μ and κ proposed in Eq. (5.29), the low-Mach asymptotic limit is ensured for isentropic flow and transonic flows with shocks will be correctly resolved. As the flow becomes locally supersonic, the viscosity coefficients μ and κ will be of the same order of magnitude which is consistent with the original definition of the entropy viscosity method [28, 29] recalled in Section 5.2.1.

5.4 Extension of the entropy viscosity technique Euler equations with variable area

Fluid flows in nozzles and in pipes of varying cross-sectional area can be modeled using the variable-area variant of the Euler equations, where the conservative variables are now multiplied by the area A. In addition, these equations differ from the standard Euler equations in that the momentum equation Eq. (5.30b) contains a non-conservative term proportional to the area gradient. For the purpose of this paper, the variable area is assumed to be a smooth function of space.

$$\partial_t \left(\rho A\right) + \vec{\nabla}\cdot\left(\rho\vec{u}A\right) = 0 \tag{5.30a}$$

$$\partial_t \left(\rho\vec{u}A\right) + \vec{\nabla}\cdot\left[A\left(\rho\vec{u}\otimes\vec{u} + P\mathbb{I}\right)\right] = P\vec{\nabla}A \tag{5.30b}$$

$$\partial_t \left(\rho E A\right) + \vec{\nabla}\cdot\left[\vec{u}A\left(\rho E + P\right)\right] = 0 \tag{5.30c}$$

The application of the entropy viscosity method to the Euler equations with variable area is not fundamentally different to its application to the standard Euler equations. However, we need to derive the associated dissipative terms and verify that the entropy minimum principle is still satisfied. The variable-area Euler equations with viscous regularization are given below; details of the derivation are provided in Appendix A.

$$\partial_t \left(\rho A\right) + \vec{\nabla}\cdot\left(\rho\vec{u}A\right) = \vec{\nabla}\cdot\left(A\kappa\vec{\nabla}\rho\right) \tag{5.31a}$$

$$\partial_t \left(\rho\vec{u}A\right) + \vec{\nabla}\cdot\left[A\left(\rho\vec{u}\otimes\vec{u} + P\mathbf{I}\right)\right] = P\vec{\nabla}A + \vec{\nabla}\cdot\left[A\left(\mu\rho\vec{\nabla}^s\vec{u} + \kappa\vec{u}\otimes\vec{\nabla}\rho\right)\right] \tag{5.31b}$$

$$\partial_t \left(\rho A E\right) + \vec{\nabla}\cdot\left[\vec{u}A\left(\rho E + P\right)\right] = \vec{\nabla}\cdot\left[A\left(\kappa\vec{\nabla}\left(\rho e\right) + \frac{1}{2}||\vec{u}||^2\kappa\vec{\nabla}\rho + \rho\mu\vec{u}\vec{\nabla}^s\vec{u}\right)\right] \tag{5.31c}$$

The dissipative terms are indeed very similar to the ones obtained for the standard Euler equations: each dissipative flux is simply multiplied by the variable area A in order to ensure conservation of the dissipative flux. When assuming a constant area, Eq. (5.2) are recovered.

A low-Mach asymptotic limit of the multi-D Euler equations with variable area on the same model as in Section 5.3.2 will lead to the divergent constraint $\vec{\nabla}\cdot(\vec{u}A) = 0$ that can be recast as $\vec{\nabla}\cdot\vec{u} = -\vec{u}\cdot\vec{\nabla}A/A$. The gradient of the area acts as a source term and will force the fluid to accelerate or decellerate, depending on its sign.

5.5 Entropy-viscosity method and source terms:

In this section, the 1-D Euler equations are considered with source terms such as wall friction, gravity and wall-heat source terms. The objective is to study how these source terms affect the entropy-viscosity method, and more precisely, the definition of the viscosity coefficients. Our approach consists of starting with the 1-D Euler equation with source terms in both the momentum and energy equation but without the viscous regularization (the dissipative terms were derived using the entropy minimum principle and should not be affected by the addition of source terms). Then, the entropy residual is derived in order to study how the source terms affect its sign.

- describe the source terms: friction, wall heat source, ...
- explain the method with examples: surface force, volumetric force, energy source/sink.
- derive the entropy residual
- explain how the def of the viscosity coeff. gets modified.
- advantage and disadvantage.

5.6 Boundary conditions

In order to detail the partial and temporal discretization used for this study, the system of equations Eq. (5.31) is considered under the following form for simplicity:

$$\partial_t U + \vec{\nabla} \cdot F\left(U\right) = S \tag{5.32}$$

where U is the vector solution, F is a conservative vector flux and S is a vector source that can contain the non-conservative term $P\vec{\nabla}A$. The system of equation given in Eq. (5.32) is discretized using a continuous Galerkin finite element method and high-order temporal integrators provided by the MOOSE framework. Because we cannot consider infinitely large domain, the computational domain needs to be truncated at some particular points. These particular points are referred to as boundaries and need to be treated with great care in order to preserve the mathematical and physical properties of the equations to solve. An error in the treatment of the boundary conditions can lead to inaccurate transient and steady-state numerical solutions. The conditions applied to the boundaries of the computational domain are called boundary conditions and correspond to a set of boundary values to supply and relations that will link the boundary values to the rest of the computational domain. When dealing with wave-dominated problem alike the multi-D Euler equations, the eigenvalues and the characteristic equations are of great help since they inform us on how the physical information travel through the computational domain. A great amount of work on the treatment of the inlet and outlet boundary conditions is available in the literature (REFS). The method proposed depend on the equation of state and the discretization method. For discontinuous schemes alike finite volume and discontinuous Galerkin method, providing some boundary values, a Riemann problem is often solved. A Riemann solver is derived from the study of the eigenvalues and characteristic equations, and therefore will preserve the balance of the system, but is strongly dependent on the equation of state used. In the case of continuous schemes, the same approach cannot be used since the variables are continuous at the interfaces. Instead a more direct approach is required and is the focus of this section.

This short paragraph aims at illustrating the reasoning behind the treatment of the boundary conditions. For academic purpose, let us consider a 1-D computational domain with a left inlet and right outlet boundaries. The 1-D characteristic equations and the eigenvalues for the multi-D Euler equations are recalled:

$$\begin{cases} dP - \rho c du = 0 \text{ along } \frac{dx}{dt} = u - c = \lambda_1 \\ d\rho + \text{ along } \frac{dx}{dt} = u = \lambda_2 \\ dP + \rho c du = 0 \text{ along } \frac{dx}{dt} = u + c = \lambda_3 \end{cases} \tag{5.33}$$

The 1-D characteristic equations can be derived by either performing matrix calculations or by combining the conservative form of the 1-D Euler equations. These two approach are equivalent and lead to the same form of the characteristic equations given in Eq. (5.33). The three eigenvalues λ_i in 1-D are also recalled in Eq. (5.33). We are focusing on the inlet boundary. The sign of the eigenvalues at the inlet given in Eq. (5.33) will depend on the flow type: either supersonic with a Mach number larger than one and all eigenvalues are positive, or subsonic with a Mach number smaller than one and λ_2 and λ_3 are positive but λ_1 is negative. More specifically, for a supersonic flow, none physical information exits the computational domain. In another hand, for a subsonic flow, waves are both exiting and entering the computational domain at the inlet. From a numerical point of view, it means that in the case of a supersonic flow, all of the quantities need to be supplied to the code. When dealing with a subsonic flow, only two variables will be supplied and the third one will be computed using relations derived from the characteristic equations. From this simple study of the sign of the eigenvalues, it is understood that a distinction between supersonic and subsonic needs to be made in order to properly evaluate the boundary values. A similar approach can be used for the treatment of the outlet boundary.

In this section, four types of boundary conditions will be described: static pressure, stagnation pressure, mass inflow and wall boundary conditions. For each of them, details regarding the theory in 1-D and the implementations for a temporal implicit solver described in Section 3.1 will be given. The characteristic equations and eigenvalues are independent of the equation of states. However, derivation of the relations to implement for the subsonic and supersonic boundary conditions will depend on the equation of state: the Stiffened Gas equation of state is considered here. The method used here is partly derived from the work of Berry et al. [49] that was applied to the two-phase flow seven equations model solved with a temporal explicit solver. It is also chosen to work with the primitive variables (density ρ^*, velocity u^* and pressure P^*) that are computed at the boundaries using relations that will depend on the boundary type. The upper script $*$ denotes the values on the element located at the boundary (node, curve and face in 1, 2 and 3-D, respectively). From a numerical point of view, the boundary terms are under the form of an integral over the surface $\int_{\partial\Omega} F(U^*)\vec{n}\phi$ as shown in EQUATION. For each type of boundary conditions, it is explained how to compute the primitive variables at the boundary: then the hyperbolic flux $F(U^*)$ is easily obtained.

5.6.1 Stagnation pressure boundary condition

The stagnation pressure boundary condition is used whenever a pipe is connected to a very large tank or reservoir. The tank is so large in comparison to the pipe that the flow coming from the tank is considered to be at quasi-steady-state. This boundary condition assumes a subsonic flow: two eigenvalues are positive and one is negative. Physically it means that information exits and enters the computational domain at the inlet boundary. In order to be consistent with the physic, two boundary values have to be provided and the third one needs to be obtained from the computational domain. The stagnation pressure and temperature are usually supplied for this type of boundary. A general definition for a stagnation pressure is the following: *the stagnation pressure is the static pressure a flow retain when brought to rest isentropically from a given fluid state.* For incompressible flow, the stagnation pressure is simply the sum of the static pressure and the kinetic energy. When brought to rest, all of the kinetic energy is converted into internal energy that is function of the static pressure. For compressible flow described by the Stiffened gas equation of state, the stagnation pressure and temperature are function of the Mach number M as follows:

$$P_0 = P_s \left(1 + \frac{\gamma - 1}{2} M^2\right) \tag{5.34}$$

$$T_0 = T_s \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma-1}{\gamma}} \tag{5.35}$$

where P_s and T_s denote the static pressure and temperature respectively. The stagnation relations given in Eq. (5.34) are derived by assuming an isentropic flow. From the stagnation pressure and temperature, the stagnation density and enthalpy can be computed using the equation of state. The sub script 0 is used to denote any stagnation variable.

We now need to find out how to compute the primitive variables in order to get the flux $F(U^*)$ at the boundary. For a temporal explicit scheme and a finite volume discretization method, Berry et al. [49] used the characteristic equations and the isentropic assumption along with the definition of the stagnation variables to derive relations that relate the boundary node values to the middle-point value of the first cell in. A non-linear equation, obtained from the integration of the characteristic equations over the first cell in, has to be solved to get the boundary velocity value. This typical finite volume approach is not easily adaptable to a continuous finite element method since the cell values are not constant. The method to compute and implement the boundary conditions detailed in this section, takes advantage of the non-linear solver that is used to solve each implicit time step. First, relations that link the pressure P^* and the density ρ^* to the velocity u^* are derived by using enthalpy and entropy conservation through the boundary. In order to derive analytical expression for the entropy, an equation of state is required. As mentioned earlier in this section, the Stiffened Gas equation of state is used to describe the fluid. Derivation of the analytical expression for the entropy S can be found in (REF) and lead to the following expression:

$$S(\rho, P) = C_v \log \frac{P + P_\infty}{\rho^\gamma}$$

where C_v is a constant heat capacity. Since the flow is assumed to be isentropic at the boundary, the following relation can be used:

$$S_0 = S^* \rightarrow \frac{P_0 + P_\infty}{\rho_0^\gamma} = \frac{P^* + P_\infty}{\rho^{*\gamma}} = K$$

which can be recast under the flooding form to obtain our first relation:

$$P^* = K\rho^{*\gamma} - P_\infty \tag{5.36}$$

where K is computed from the stagnation pressure and temperature, and using the equation of state to obtain the stagnation density. The second relation is obtained by using an enthalpy balance between the boundary and a far field location in the tank. The enthalpy is denoted by H and defined as $H = E + p/\rho$. Thus,

$$H_0 = H^* \rightarrow \bar{H} = \frac{\gamma(P_0 + P_\infty)}{\rho_0(\gamma - 1)} = \frac{\gamma(P^* + P_\infty)}{\rho^*(\gamma - 1)} + \frac{1}{2}u^{*2}$$

after splitting the total energy in the sum of the internal and kinetic energy and using the Stiffened Gas equation of state to express the internal energy as a function of the pressure and the density. The constant $\bar{H}$ is again computed from the stagnation variables. Using Eq. (5.36), the second relation for the density ρ^* only function of the velocity u^* is obtained:

$$\rho^* = \left(\frac{\gamma - 1}{K\gamma} (\bar{H} - \frac{u^{*2}}{2}) \right)^{\frac{1}{\gamma - 1}} \tag{5.37}$$

From Eq. (5.36) and Eq. (5.37), once a value for the velocity u^* is known, the pressure and the density values at the boundary can be computed. In [49], the velocity at the boundary is computed from a non-linear equation that is obtained by integrating the characteristic equations over the first cell in. The objective is to link the inside of the computational domain to the boundary in order to transmit the proper physical information entering and exiting the boundary. Our approach differs from [49] in a sense that it takes advantage of the non-linear solver used to solve for the numerical solution at each implicit time step. While iterating on the solution to converge the residual, a direct connection between the boundary values and the rest of the computational domain is ensured. Thus, it is believed that the non-linear function to compute the velocity used in [49] is not required and the connection between the boundary and the rest of the domain will be simply ensured by taking the latest value of the velocity u_k^* given by the non-linear solver, where k denotes the k^{th} iterate. Thus, given the last iterate of the velocity value at the boundary u_k^*, the corresponding pressure p_k^* and density ρ_k^* values can be computed from Eq. (5.36) and Eq. (5.37). During a non-linear solve, the velocity gets updated based on the inside values and so are the pressure and density. The boundary values are converged once the residual of the full computational domain is converged. From the star primitive values $U^* = \{\rho^*, u^*, P^*\}$, the flux $F(U^*)$ can be computed and then used to weakly impose the boundary in EQUATION. It is also possible to strongly impose the boundary conditions by computing the conservative variables from the star primitive values and using Dirichlet method (REF).

Remark. *For the stagnation pressure boundary condition, it can be also chosen to specify the stagnation density instead of the stagnation temperature. Using the equation of state, the third value can be easily obtained.*

5.6.2 Back pressure outlet boundary condition for subsonic flow

The back pressure boundary condition is applied when a flow exits a computational domain and has to match a specified back pressure at the exit. This type of boundary condition is only valid for subsonic flow (outlet boundary condition for supersonic flow is treated in Section 5.6.4). This boundary condition is often referred in the literature as a static pressure outlet boundary condition. We rather use this terminology for the static pressure inlet boundary.

Once again, the basis of the method relies on the sign of the eigenvalues. A subsonic flow at the outlet is considered: the eigenvalues λ_2 and λ_3 are positive and the eigenvalue λ_1 is negative. Therefore, only one variable needs to be specified in order to be consistent with the mathematical properties of the system of equations:

this variable is the back pressure P_b. For an explicit temporal discretization, alike in [49], the characteristic equations associated to the eigenvalues λ_2 and λ_3 are used to compute the values of the density and the velocity at the outlet: the characteristic equations are integrated over the last cell and a relation linking the star values to the first node-in values is obtained. In the case of an implicit temporal discretization, the non-linear solver substitutes itself to the characteristic equations. The values of the density ρ^* and the velocity u^* are updated through the iterations and , thus are related to the rest of the computational domain. Every time the pressure is required, it is set equal to the back pressure P_b. When considering the k^{th} iteration, the 1-D flux at the boundary looks like:

$$F(U_k^*) = \begin{cases} \rho_k^* u_k^* \\ \rho_k^* (u_k^*)^2 + P_b \\ u_k^* \left(\rho_k^* E_k^* + P_b\right) \end{cases} \tag{5.38}$$

where E_k^* is computed using the equation of state and the values ρ_k^*, u_k^* and P_b.

5.6.3 Mass flow inlet boundary condition

5.6.4 Outlet boundary condition for supersonic flow

A supersonic boundary condition is required when the flow becomes supersonic at the outlet. In that case, all of the eigenvalues are locally positive which means all of the physical information exits the computational domain. Then, the primitive values at the boundary are all taken equal to the last iterate ones and used to compute the flux at the boundary as follows:

$$F(U_k^*) = \begin{cases} \rho_k^* u_k^* \\ \rho_k^* (u_k^*)^2 + P_k^* \\ u_k^* \left(\rho_k^* E_k^* + P_k^* \right) \end{cases} \tag{5.39}$$

where E_k^* is computed using the equation of state and the values ρ_k^*, u_k^* and P_k^*. The outlet supersonic boundary condition is often paired to the back pressure outlet boundary condition described in Section 5.6.2. When implementing an outlet boundary condition a switch statement on the Mach number is required in order to account for flow going from subsonic to supersonic or the other way around. The only major difference between the two types of boundary condition is the treatment of the pressure.

5.6.5 Solver

A Free-Jacobian-Newton-Krylov (FJNK) method is used to solve for the solution at each time step. The jacobian matrix of the discretized equations was derived by hand, hard coded and used as a preconditioner. This method requires the partial derivative of the pressure with respect to the conservative variables to be known. The contribution of the artificial dissipative terms to the jacobian matrix is simplified by assuming constant viscosity coefficients as shown in Eq. (5.40) for the dissipative terms of the continuity equation:

$$\frac{\partial}{\partial U_i} \left(\kappa \vec{\nabla} \rho \vec{\nabla} \phi \right) = \kappa \frac{\partial}{\partial U_i} (\rho) \vec{\nabla} \phi \tag{5.40}$$

where U_i denotes the set of conservative variables.

5.7 1-D shock tubes:

1-D and 2-D Numerical solutions for the Euler equations with viscous regularization using the entropy viscosity method are presented here. Our results validate the chosen definitions for the viscosity coefficients in the low-Mach limit and verify that the new definitions resolve shocks appropriately.

The first set of 1-D simulations consist of liquid water and steam flowing in a convergent-divergent nozzle. This test is of interest for multiple reasons: (a) a steady state can be reached (some stabilization methods are known to have difficulties reaching a steady state, [13, 14]), (b) an analytical solution is available and a space-time convergence study can be performed, (c) it can be performed for liquid and gas phases, the gas phase simulation presents a shocks while the liquid-phase simulation has a significantly lower Mach number. Next, a 1-D shock tube test (in a straight pipe) taking from the Leblanc test-case suite [42] is performed. This test is known to be more challenging than Sod shock tubes and the fluid's Mach number varies spatially between 0 and 5. A convergence study is also performed in order to demonstrate convergence of the numerical solution to the exact solution. A slow moving shock is also investigated [47]. This test helps in assessing the ability of the method at damping the post-shock low frequency noise. Finally, a strong shock for a liquid phase is also investigated. need ref for this

The initial conditions (density in $kg.m^{-3}$, velocity in $m.s^{-1}$, pressure in Pa) for the last three 1-D shock tubes are given in Table 5.1.

2-D simulations are presented next. First, subsonic flows of around a 2-D cylinder [34] and over a Gaussian hump [18] are presented for various far-field Mach numbers (as low of 10^{-7}). Numerical results of a transonic flow in a compression corner are provided to illustrate the ability of the new viscosity definitions to handle supersonic

ρ_{left}	u_{left}	P_{left}	ρ_{right}	u_{right}	P_{right}
Leblanc shock tube (Section 5.7.3)					
1	0	$4\ 10^{-2}$	10^{-3}	0	$4\ 10^{-11}$
Strong shock for liquid phase (Section 5.7.4)					
1000	0	10^{9}	1000	0	10^{5}
Slow moving shock (Section 5.7.5)					
1	-0.81	1	3.86	-3.44	10.33

Table 5.1: Initial conditions for the 1-D shock tube tests.

flows. Convergence studies are performed when analytical solutions are available.

For each simulation, data relative to the boundary conditions, the Courant-Friedrichs-Lewy number (CFL), mesh and equation of state are provided. All of the numerical solution presented are obtained using BDF2 as temporal integrator and linear (1-D mesh), $\mathbb{P}_1$ (2-D triangular mesh), $\mathbb{Q}_1$ (2-D quadrangular mesh) finite elements. The integrals are numerically computed using a second-order Gauss quadrature rule. The steady-state solution is detected by monitoring the norm of the total residual (including all of the equations) and reached when the norm of the total residual is below 10^{-6}. The ideal gas [46] or stiffened gas equations of state [38] are used; a generic expression is given in Eq. (5.41).

$$P = (\gamma - 1)\rho(e - q) - \gamma P_\infty \tag{5.41}$$

where the parameters q and P_∞ are fluid-dependent and are given in Table 5.2. The ideal gas equation of state is recovered by setting $q = P_\infty = 0$ in Eq. (5.41). The entropy function for the stiffened gas equation of state is convex and given by

$$s = C_v \ln\left(\frac{P + P_\infty}{\rho^{\gamma-1}}\right),$$

Table 5.2: Stiffened Gas Equation of State parameters for steam and liquid water.

fluid	γ	C_v $(J.kg^{-1}.K^{-1})$	P_∞ (Pa)	q $(J.kg^{-1})$
liquid water (Section 5.7.1)	2.35	1816	10^9	$-1167\ 10^3$
steam (Section 5.7.2)	1.43	1040	0	$2030\ 10^3$
liquid water (Section 5.7.4)	4.4	1000	$6\ 10^8$	0

where C_v is the heat capacity at constant volume.

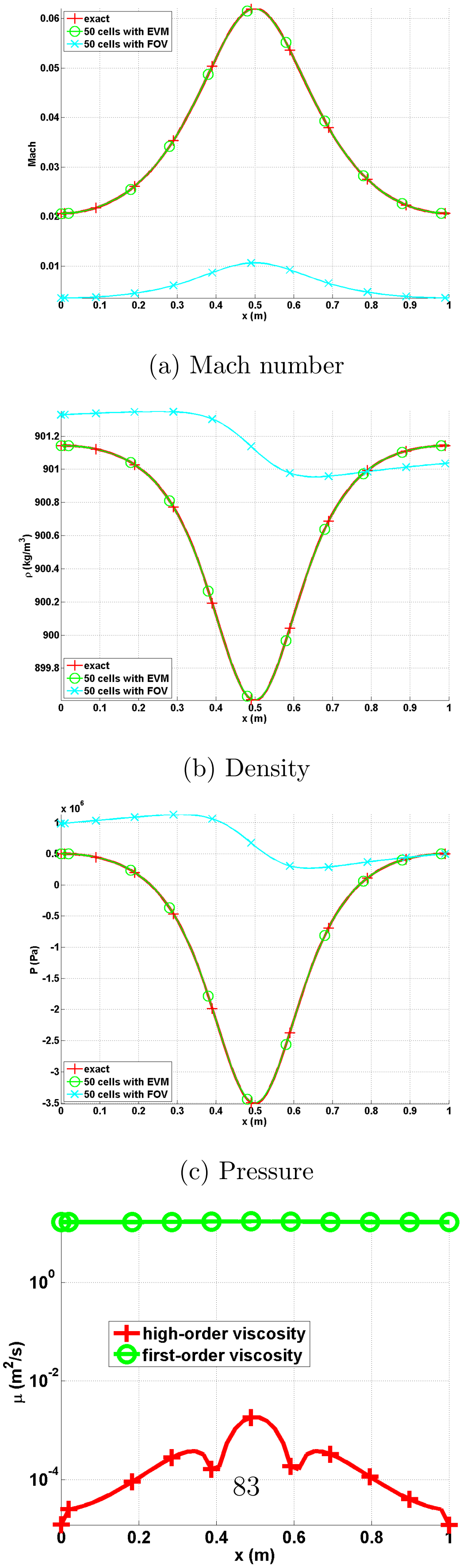
Finally, the convergence rates are computed using the following relation

$$rate_h = \ln\left(\frac{||U_{2h} - U_{\text{exact}}|| - ||U_h - U_{\text{exact}}||}{||U_h - U_{\text{exact}}||_h - ||U_{h/2} - U_{\text{exact}}||}\right) / \ln 2 \tag{5.42}$$

where $||\cdot||$ denotes either the L_1 or L_2 norms and h is the grid size.

5.7.1 Liquid water in a 1-D convergent-divergent nozzle

This simulation consists of liquid water flowing through a 1-D convergent-divergent nozzle. The variable-area Euler equations are solved, with $A(x) = 1+0.5\cos(2\pi x/L)$ where $L = 1m$ is the length of the nozzle. At the inlet, the stagnation pressure and temperature are set to $P_0 = 1MPa$ and $T_0 = 453K$, respectively. At the outlet, only the static pressure is specified: $P_s = 0.5MPa$. Initially, the liquid is at rest, the temperature is uniform and equal to the stagnation temperature and the pressure linearly decreases from the stagnation pressure inlet value to the static pressure outlet value. The stiffened gas equation of state is used to model the liquid water with the parameters provided in Table 5.2. Because of the low pressure difference between the inlet and the outlet, the smooth initial conditions, and the large value of P_∞, the flow remains subsonic and thus does not display any shock. A detailed derivation of the exact steady-state solution can be found in [37]. A uniform mesh of 50 cells was used to obtain the numerical solution and the time step size was computed using a CFL number of 750. Plots of the Mach number, density, and pressure are given at steady-state in Fig. 5.1 for the numerical and exact solutions. The viscosity coefficients are also graphed in Fig. 5.1d.



(a) Mach number

(b) Density

(c) Pressure

(d) Viscosity coefficients

In Fig. 5.1, the numerical solutions obtained using the first-order viscosity (FOV) and the entropy viscosity method (EVM) are plotted against the exact solution. The numerical solution obtained with the EVM and the exact solution overlap, even for fairly coarse mesh (50 cells). On the other hand, the numerical solution obtained with the FOV does not give the correct steady state: this is an illustration of the effect of ill-scaled dissipative terms. Note that the entropy viscosity coefficient is very small compared to the first-order one (Fig. 5.1d): (i) the numerical solution is smooth as shown in Fig. 5.1 and (ii) the flow is in a low-Mach regime and thus isentropic . A convergence study was performed using the exact solution as a reference: the L_1 and L_2 norms of the error and the corresponding convergence rates are computed at steady state on various uniform mesh from 4 to 256 cells. Spatial convergence results using linear finite elements are reported in Table 5.3 and Table 5.4 for the primitive variables: density, velocity and pressure.

Table 5.3: L_1 norm of the error for the liquid phase in a 1-D convergent-divergent nozzle at steady state.

cells	density	rate	pressure	rate	velocity	rate
4	$2.8037\ 10^{-1}$	–	$8.4705\ 10^{5}$	–	7.2737	–
8	$1.3343\ 10^{-1}$	0.495	$4.7893\ 10^{5}$	0.24	6.1493	0.0747
16	$2.9373\ 10^{-2}$	2.10	$1.0613\ 10^{5}$	2.09	1.2275	2.25
32	$5.1120\ 10^{-3}$	2.58	$1.8446\ 10^{4}$	2.58	$1.8943\ 10^{-1}$	2.78
64	$1.0558\ 10^{-3}$	2.31	$3.7938\ 10^{3}$	2.31	$3.7919\ 10^{-2}$	2.37
128	$2.3712\ 10^{-4}$	2.18	$8.4471\ 10^{2}$	2.19	$8.5517\ 10^{-3}$	2.17
256	$5.6058\ 10^{-5}$	2.08	$1.9839\ 10^{2}$	2.09	$2.0475\ 10^{-3}$	2.07
512	$1.3278\ 10^{-5}$	2.07	$4.6622\ 10^{1}$	2.08	$4.9516\ 10^{-4}$	2.06
1024	$3.1193\ 10^{-6}$	–	$1.1755\ 10^{1}$	–	$1.2379\ 10^{-4}$	–

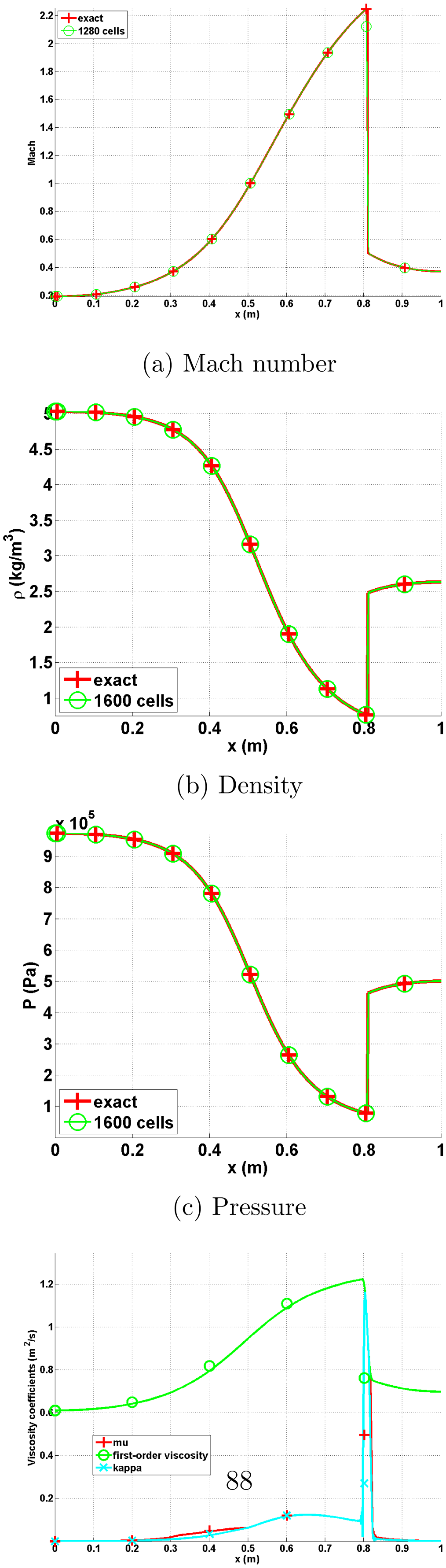
Table 5.4: L_2 norm of the error for the liquid phase in a 1-D convergent-divergent nozzle at steady state.

cells	density	rate	pressure	rate	velocity	rate
4	$3.106397\ 10^{-1}$	–	$5.254445\ 10^{5}$	–	3.288543	–
8	$7.491623\ 10^{-2}$	2.06	$1.636966\ 10^{5}$	1.62	1.823880	0.14
16	$2.079858\ 10^{-2}$	1.81	$4.627338\ 10^{4}$	1.77	$4.990605\ 10^{-1}$	1.83
32	$5.329627\ 10^{-3}$	1.96	$1.180287\ 10^{4}$	1.96	$1.261018\ 10^{-1}$	1.98
64	$1.341583\ 10^{-3}$	1.99	$2.967104\ 10^{3}$	1.99	$3.160914\ 10^{-2}$	1.99
128	$3.359766\ 10^{-4}$	1.99	$7.428087\ 10^{2}$	1.99	$7.907499\ 10^{-3}$	1.99
256	$8.403859\ 10^{-5}$	1.99	$1.857861\ 10^{2}$	2.01	$1.977292\ 10^{-3}$	2.00
512	$2.10075\ 10^{-5}$	–	$4.7024\ 10^{1}$	–	$4.9516\ 10^{-4}$	–

It is observed that the convergence rate for the L_1 and L_2 norm of the error is 2: the entropy viscosity method preserves the high-order accuracy when the numerical solution is smooth, and the new definition of the entropy viscosity coefficient behaves appropriately in the low-Mach limit.

5.7.2 Steam in a 1-D divergent-convergent nozzle

We use the same nozzle geometry, initial conditions and boundary conditions as in the previously example but replace liquid water with steam and use the steam parameters of the stiffened gas equation of state, Table 5.2. In this run, compressible effects will become dominant. The pressure difference between the inlet and outlet is large enough to accelerate the steam through the nozzle, leading to a formation of shock in the divergent portion of the nozzle. The behavior is different from what was observed for the liquid water phase in Section 5.7.1 because of the liquid to gas density ratio is about $1,000$. An exact solution at steady state is available for the gas phase [37]. The aim of this section is to show that when using the new definitions of the viscosity coefficients (Eq. (5.29)), the shock can be correctly resolved without spurious oscillations. The steady-state numerical solution, obtained using a uniform mesh with 1600 cells, is shown in Fig. 5.2. The CFL was set to 80 (a high CFL value can be used since the shock is stationary).



(a) Mach number

(b) Density

(c) Pressure

(d) Viscosity coefficients

The steady-state solution of the density, Mach number and pressure are given in Fig. 5.2. The steady-solution exhibits a shock around $x = 0.8m$ and matches the exact solution. In Fig. 5.2d, the first-order and entropy viscosity coefficients are plotted at steady-state (on a log scale): the entropy viscosity coefficient is peaked in the shock region around $x = 0.8m$ as expected where it saturates to the first-order viscosity coefficient. The graph also presents another peak at $x = 0.5m$ corresponding to the position of the sonic point for a 1-D convergent-divergent nozzle. This particular point is known to develop small instabilities that are detected when computing the jumps of the pressure and density gradients. Anywhere else, the entropy viscosity coefficient is small. In order to prove convergence of the numerical solution to the exact solution, a convergence study is performed. Because of the presence of a shock, second-order accuracy is not expected and the convergence rate of a numerical solution should be 1 and 1/2 when measured in the L_1 and L_2 norms, respectively (see Theorem 9.3 in [19]). Results are reported in Table 5.5 and Table 5.6 for the primitive variables: density, velocity and pressure. The convergence rates for the L_1 and L_2 norms of the error computed using Eq. (5.42) are in good agreement with the theoretical values.

Table 5.5: L_1 norm of the error for the vapor phase in a 1-D convergent-divergent nozzle at steady state.

cells	density	rate	pressure	rate	velocity	rate
5	$0.72562\ 10^{-1}$	–	$1.5657\ 10^{5}$	–	173.69	–
10	$0.4165\ 10^{-1}$	0.80088	$9.6741\ 10^{4}$	0.63425	120.69	0.52519
20	$0.20675\ 10^{-1}$	1.0104	$4.9193\ 10^{4}$	0.96971	72.149	0.74228
40	$0.093703\ 10^{-1}$	1.1417	$2.0103\ 10^{4}$	0.72728	34.716	1.0554
80	$0.047328\ 10^{-1}$	0.9854	$1.0208\ 10^{4}$	0.9777	16.082	1.1101
160	$0.023965\ 10^{-2}$	0.9817	$5.1969\ 10^{3}$	0.9739	7.9573	1.0150
320	$0.020768\ 10^{-2}$	0.9886	$2.5116\ 10^{3}$	1.0490	3.7812	1.0734
640	$0.0059715\ 10^{-2}$	1.0160	$1.2754\ 10^{3}$	0.9776	1.8353	1.0428

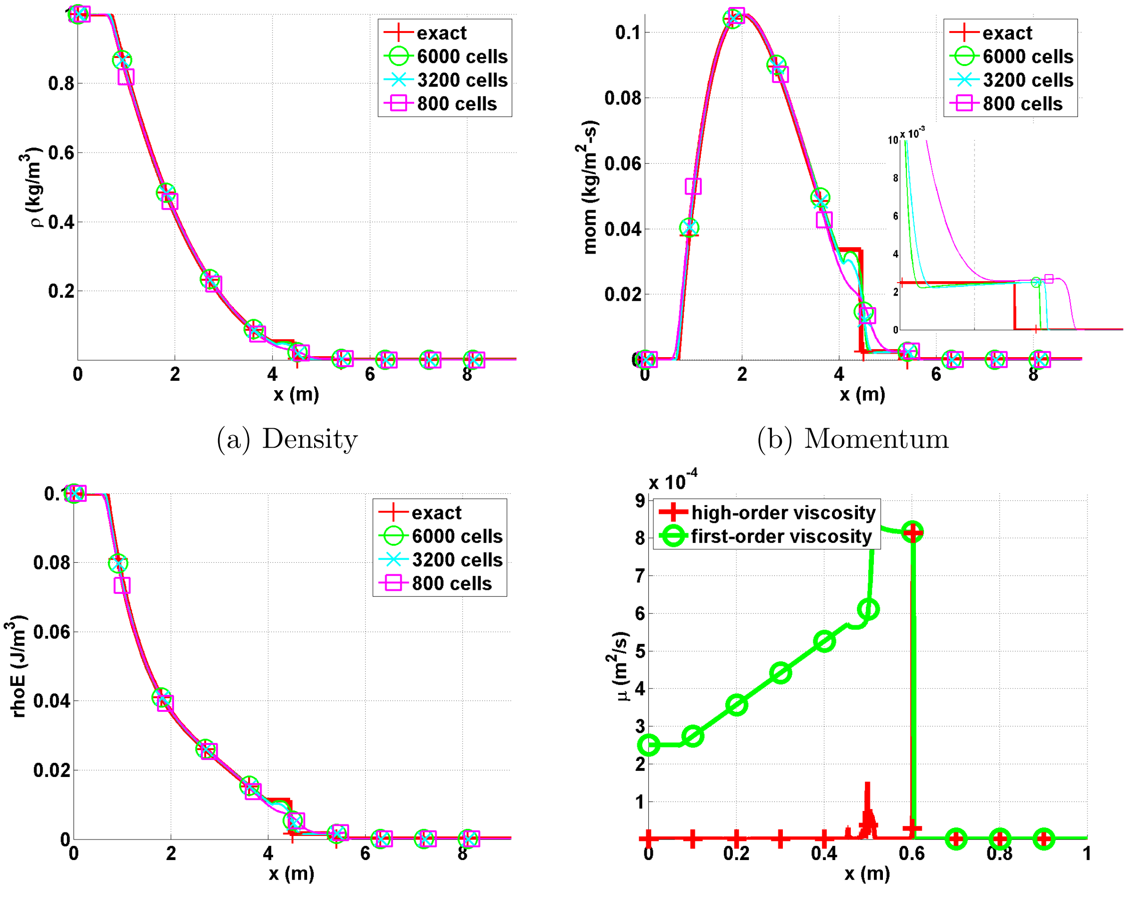
Table 5.6: L_2 norm of the error for the vapor phase in a 1-D convergent-divergent nozzle at steady state.

cells	density	rate	pressure	rate	velocity	rate
5	$9.7144\ 10^{-1}$	–	$2.0215\ 10^{5}$	–	236.94	–
10	$5.9718\ 10^{-1}$	0.70195	$1.3024\ 10^{5}$	0.63425	166.56	0.50854
20	$2.9503\ 10^{-1}$	1.0173	$6.6503\ 10^{4}$	0.96971	103.36	0.68831
40	$1.8193\ 10^{-1}$	0.69747	$4.0171\ 10^{4}$	0.72728	66.374	0.6390
80	$1.3366\ 10^{-1}$	0.44485	$2.3163\ 10^{4}$	0.43576	42.981	0.62692
160	$9.6638\ 10^{-2}$	0.46790	$1.7263\ 10^{4}$	0.42413	31.717	0.43844
320	$7.0896\ 10^{-2}$	0.44688	$1.2763\ 10^{4}$	0.43571	23.138	0.45499
640	$5.2191\ 10^{-2}$	0.44190	$9.4217\ 10^{3}$	0.43790	16.910	0.45238

5.7.3 Leblanc shock tube

The 1-D Leblanc shock tube is a Riemann problem designed to test the robustness and the accuracy of stabilization methods. The initial conditions are given in

Table 5.1. The ideal gas equation of state (with $\gamma = 5/3$) is used to compute the pressure. This test is computationally challenging because of the large pressure ratio at the shock interface. The computational domain consists of a 1-D straight pipe of length $L = 9m$ with the initial interface located at $x = 2m$. At $t = 0.s$, the interface is removed. The numerical solution is run until $t = 4s$ and the density, momentum and total energy profiles are given in Fig. 5.3, along with the exact solution. The viscosity coefficients are also plotted in Fig. 5.3d. These plots were run with three different uniform mesh of 800, 3200 and 6000 cells and a constant $CFL = 1$.



(a) Density

(b) Momentum

(c) Total energy

(d) Viscosity coefficients

Figure 5.3: Exact and Numerical solutions for the 1-D Leblanc shock tube at $t = 4s$.

The density, momentum and total energy profiles are provided in Fig. 5.3. In Fig. 5.3b, the shock region is zoomed in for better resolution: the shock is well resolved. We also observe that the shock position computed numerically converges to the exact position under mesh refinement. The contact wave at $x = 4.5m$ can be seen in Fig. 5.3b. The entropy viscosity coefficient profile is shown in Fig. 5.3d and behaves as expected: it saturates to the first-order viscosity in the shock region, thus preventing oscillations from forming. At the location of the contact wave, a smaller peak is observed that is due to the presence of the jumps in the definition of the entropy viscosity coefficient (Eq. (5.29)). The Mach number, not plotted, is of the order of 1.3 right before the shock and reaches a maximum value close to 5 in the contact region.

Once again, a convergence study is performed in order to prove convergence of the numerical solution to the exact solution. As in the previous example (vapor phase in the 1-D nozzle, Section 5.7.2), the expected convergence rates in the L_1 and L_2 norms are 1 and 1/2, respectively. The exact solution was obtained by running a 1-D Riemann solver and used as the reference solution to compute the L_1 and L_2-norms that are reported in Table 5.7 and Table 5.8 for the conservative variables: density, momentum and total energy. The convergence rates are again approaching the theoretical values.

5.7.4 1-D shock tube with a liquid phase

The purpose of this test is to investigate the ability of the entropy viscosity method to stabilize a strong shock with a small Mach number [1]: the Mach number in the shock region is of the order of 0.1. In this case, as explained in Section 5.3.2, the viscosity coefficients are required to have different order of magnitude in order to ensure the correct scaling of the dissipative terms. This test will allows us to validate

Table 5.7: L_1 norm of the error for the 1-D Leblanc test at $t = 4s$.

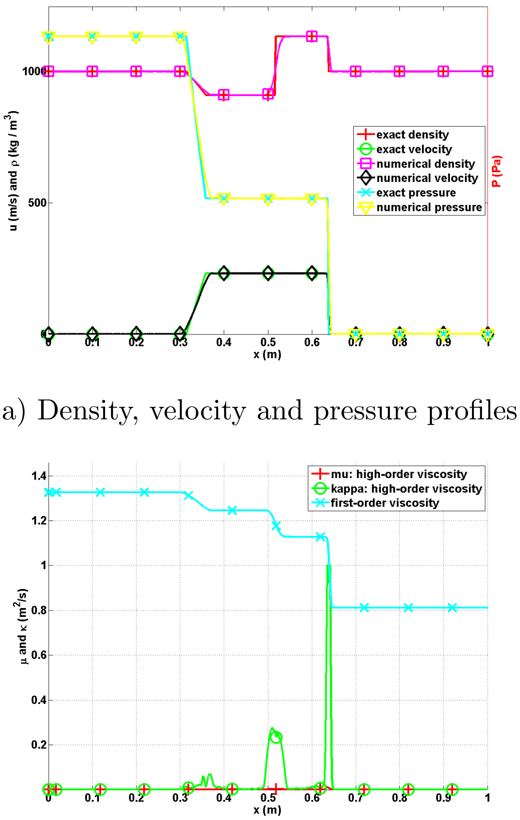
cells	density	rate	momentum	rate	total energy	rate
100	$1.0354722\ 10^{-2}$	–	$3.5471714\ 10^{-3}$	–	$1.4033046\ 10^{-3}$	–
200	$7.2680512\ 10^{-3}$	0.51064841	$2.5933119\ 10^{-3}$	0.45187331	$9.8611746\ 10^{-4}$	0.5089968
400	$5.0825628\ 10^{-3}$	0.51601245	$2.0668092\ 10^{-3}$	0.32739054	$7.7844421\ 10^{-4}$	0.34116585
800	$3.4025056\ 10^{-3}$	0.57895861	$1.4793838\ 10^{-3}$	0.48240884	$5.5702549\ 10^{-4}$	0.48285029
1600	$2.1649953\ 10^{-3}$	0.65223363	$9.7152832\ 10^{-4}$	0.6066684	$3.5720171\ 10^{-4}$	0.64100438
3200	$1.2465433\ 10^{-3}$	0.79643094	$5.5937409\ 10^{-4}$	0.79644263	$2.0491799\ 10^{-4}$	0.80169235
6400	$6.4476928\ 10^{-4}$	0.95107804	$3.0244198\ 10^{-4}$	0.88715502	$1.0914891\ 10^{-4}$	0.90874889
12800	$3.3950948\ 10^{-4}$	0.92533116	$1.5958118\ 10^{-4}$	0.9223679	$5.7909794\ 10^{-5}$	0.91441847

Table 5.8: L_2 norm of the error for the 1-D Leblanc test at $t = 4s$.

cells	density	rate	momentum	rate	total energy	rate
100	$5.7187851\ 10^{-3}$	–	$1.7767236\ 10^{-3}$	–	$7.6112265\ 10^{-4}$	–
200	$3.8995238\ 10^{-3}$	0.55241073	$1.4913161\ 10^{-3}$	0.25263314	$5.5497308\ 10^{-4}$	0.45571115
400	$2.8103526\ 10^{-3}$	0.4725468	$1.3305301\ 10^{-3}$	0.164585	$4.6063172\ 10^{-4}$	0.26880405
800	$2.1081933\ 10^{-3}$	0.41474398	$1.1398931\ 10^{-3}$	0.22310254	$3.7798953\ 10^{-4}$	0.28526749
1600	$1.5731052\ 10^{-3}$	0.42239201	$9.0394227\ 10^{-4}$	0.33459602	$2.9584646\ 10^{-4}$	0.35349763
3200	$1.0610667\ 10^{-3}$	0.56809979	$6.2735595\ 10^{-4}$	0.52694639	$2.054455\ 10^{-4}$	0.52609289
6400	$7.3309974\ 10^{-4}$	0.53343397	$4.4545754\ 10^{-4}$	0.49399631	$1.4670834\ 10^{-4}$	0.48580482
12800	$5.1020991\ 10^{-4}$	0.52291857	$3.1266758\ 10^{-4}$	0.5106583	$1.0299897\ 10^{-5}$	0.51032105

the approach presented in Section 5.3.2.

The stiffened gas equation of state is used to model a liquid flow with the parameters given in Table 5.2. The computational domain of length $L = 1m$ is uniformly discretized using 500 cells. The step initial conditions are given in Table 5.1. The simulation is run with a $CFL = 1$ until the final time $t_{\text{final}} = 7\ 10^{-5}s$. Results for pressure, density, velocity and the viscosity coefficients are given in Fig. 5.4 along with the exact solution for comparison purposes. The numerical solution is in good agreement with exact solution in Fig. 5.4a. The viscosity coefficients μ and κ are not equal in the shock because the Mach number is of order 0.1. The viscosity coefficient κ saturates to the first-order viscosity in the shock region around $x = 0.65m$ and is sufficient to stabilize the numerical scheme.



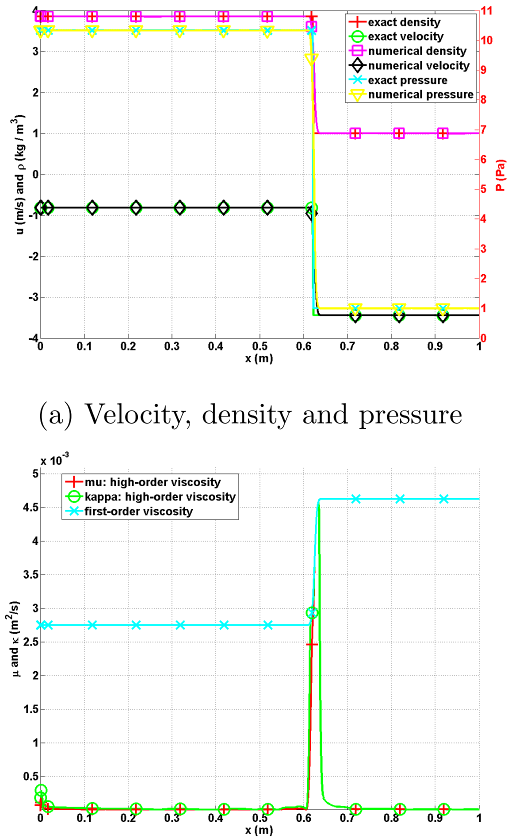
(a) Density, velocity and pressure profiles.

(b) Viscosity coefficients profile.

Figure 5.4: Numerical solution for the 1-D liquid shock tube at at $t_{\text{final}} = 7\ 10^{-5}s$.

5.7.5 1-D slow moving shock

Slow moving shocks are known to produce post-shock noise of low frequency that are not damped by some numerical dissipation methods [47]. The aim of this simulation is to test the ability of the entropy viscosity method to dampen the low frequency waves. The 1-D slow moving shock consists of a shock wave moving from left to right with the initial conditions given in Table 5.1. The ideal gas equation of state is used with a heat capacity ratio $\gamma = 1.4$. In order to make the shock travel a significant distance, the final time is taken equal to $t = 1.1s$. A pressure boundary condition is used at the left boundary to let the rarefaction and contact waves exit the domain. The numerical solution, obtained with 200 equally-spaced cells, is given in Fig. 5.5 and is compared to the exact solution obtained from a Riemann solver. We use a CFL of 1. With this CFL value, it takes about 50 time steps for the shock to traverse one cell. The numerical results are in good agreement with the exact solution and do not display any post-shock noise. The rarefaction and contact waves are not visible on Fig. 5.5a since they exited the computational domain through the left pressure boundary condition earlier. As explained in [52], Godunov's type method usually fails to resolve a slow moving shock because of the nature of the stabilization method: the method scales as the eigenvalue of the appropriate field. In the case of a slow moving shock, the dissipation added to the system is underestimated and leads to post-shock noise. In the case of the entropy viscosity method, the entropy residual detects the shock position and the viscosity coefficients saturate to the first-order viscosity values in the shock region. The main difference between a Godunov's type method and the entropy viscosity method relies in the definition of the first-order viscosity coefficients that are proportional to the *local maximum eigenvalue* $||\vec{u}|| + c$ and not to the eigenvalue of the characteristic field.



(a) Velocity, density and pressure

(b) Viscosity coefficients

Figure 5.5: Slow moving shock profiles at $t = 1.1s$.

5.7.6 Typical 1*-D shock tubes [55]*

5.8 2-D numerical results for supersonic flows:

This section focuses on demonstrating the ability of the entropy viscosity method, with the new definition of the viscosity coefficients derived in Section 5.3.2, to accurately resolve shocks occurring in transonic flows. Such tests were already performed in [57] with the former definition of the entropy viscosity method recalled in Section 5.2.1, and using a discontinuous Galerkin finite element discretization. Our objective here, is to show that the new definition of the viscosity coefficients is still capable of resolving shocks. The numerical tests presented in this section include: flow pass a forward facing step (REFS), a circular explosion (REFS), a steady-state flow over a double wedge and a steady-state flow in a compression corner. The last two tests will also allow us to evaluate the ability of the method to reach a steady-state. For each numerical results presented in this section, information relative to the equation of state and its parameters, the boundary conditions, the initial conditions, the mesh and the discretization order will be provided along with the numerical results. For clarity purpose we will refer to as Ω. Since only 2-D computational domain is considered, left, right, bottom and top boundaries are referred to as $\delta\Omega_1$, $\delta\Omega_2$, $\delta\Omega_3$ and $\delta\Omega_4$, respectively, with $\delta\Omega = (\delta\Omega_1, \delta\Omega_2, \delta\Omega_3, \delta\Omega_4)$.

5.8.1 Supersonic 2-D flow over a forward facing step:

This benchmark was first introduced by (NAME + REF) and later popularized by Woodward et al. (REF). It consists of a Mach 3 flow past a forward-facing step in a 2-D wind tunnel. The geometry is given in (FIGURE) and was discretized with an uniform mesh of $1.5 \times 10^(4)$ cells. A supersonic inlet boundary condition is used to set the flow conditions. A slip wall boundary condition is specified at the top and bottom wall following the method explained in (SECTION FOR BCS). The outflow, in $x = 4$ is free since the flow remains supersonic at the outlet boundary. The uniform initial conditions are given in Table 5.9 for the primitive variables. The Ideal gas equation of state is used with a adiabatic gas constant $\gamma = 1.4$.

Table 5.9: Initial conditions for a 2-D supersonic flow past a forward-facing step.

primitive variables	ρ	$\vec{u}$	P
value	1.4	$(3., 0.)$	1.

The numerical solution was obtained with a $\mathbb{Q}_1$ continuous Galerkin finite element method and the second-order temporal integrator $BDF2$. The solution was run until $t = 4.s$ with a CFL of 2. The density and viscosity coefficients profiles are plotted at different times during the transient, to witness how the entropy-viscosity method adapt to the solution itself.

5.8.2 2-D circular explosion:

We now consider a 2-D circular explosion (REFS) that is known to develop an unstable layer contact. The computational domain is a square of dimension $\Omega = (-1, 1)^2$ as shown in (FIGURE). The initial conditions are also given in (FIGURE) and consists of a pressure and density step located in the center of the computational domain. The values of the initial conditions are given in Table 5.10 in function of the radius $r^2 = x^2 + y^2$. The Ideal gas equation of state is sill used with the same parameters as in Section 5.8.1. Dirichlet boundary conditions are used to specify the values on the boundaries $\delta\Omega$ of the computational domain Ω, assuming that the simulation is stopped before the waves reach the boundaries.

Table 5.10: Initial conditions for a 2-D explosion.

primitive variables	ρ	$\vec{u}$	P
$r \in [0, 0.4]$	1	$(0,0)$	1
$r \geq 0.4$	0.125	$(0,0)$	0.1

Once again, the numerical solution was obtained with a $\mathbb{Q}_1$ continuous Galerkin finite element method and the second-order temporal integrator $BDF2$. The solution was run until $t = 0.25s$ with a CFL of 2. An uniform mesh was used made of 2000 elements. The density and the viscosity coefficients profiles are given in Fig. 5.6-Fig. 5.13. It was chosen to show the numerical solution at times $t = 0.314$, $t = 0.664$, $t = 1.551$ and $t = 4$ s to illustrate the ability of the entropy viscosity method to detect shocks and discontinuities during a transient, and add significant dissipation only in their close neighborhood.

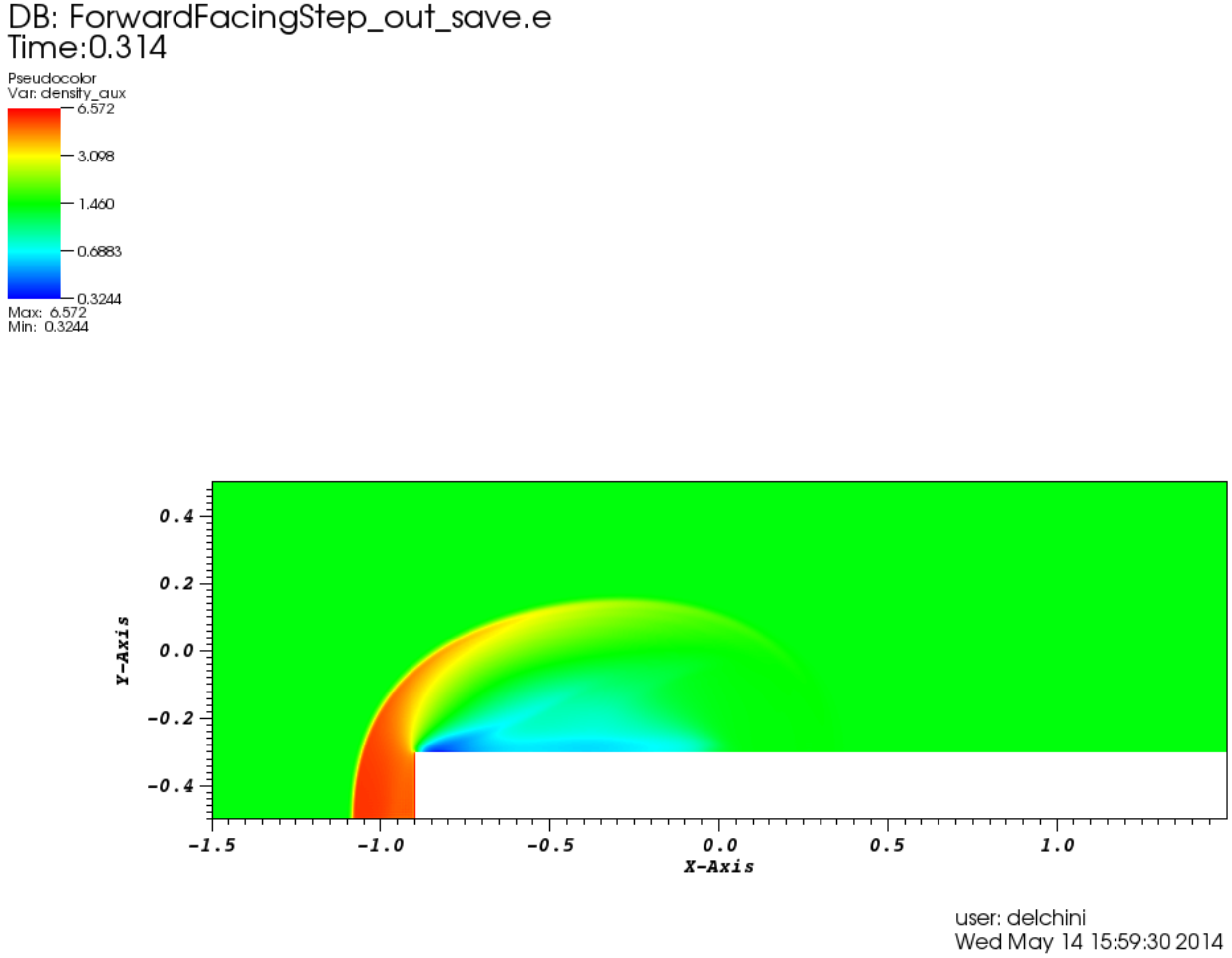


Figure 5.6: Density solution at $t = 0.314\ s$.

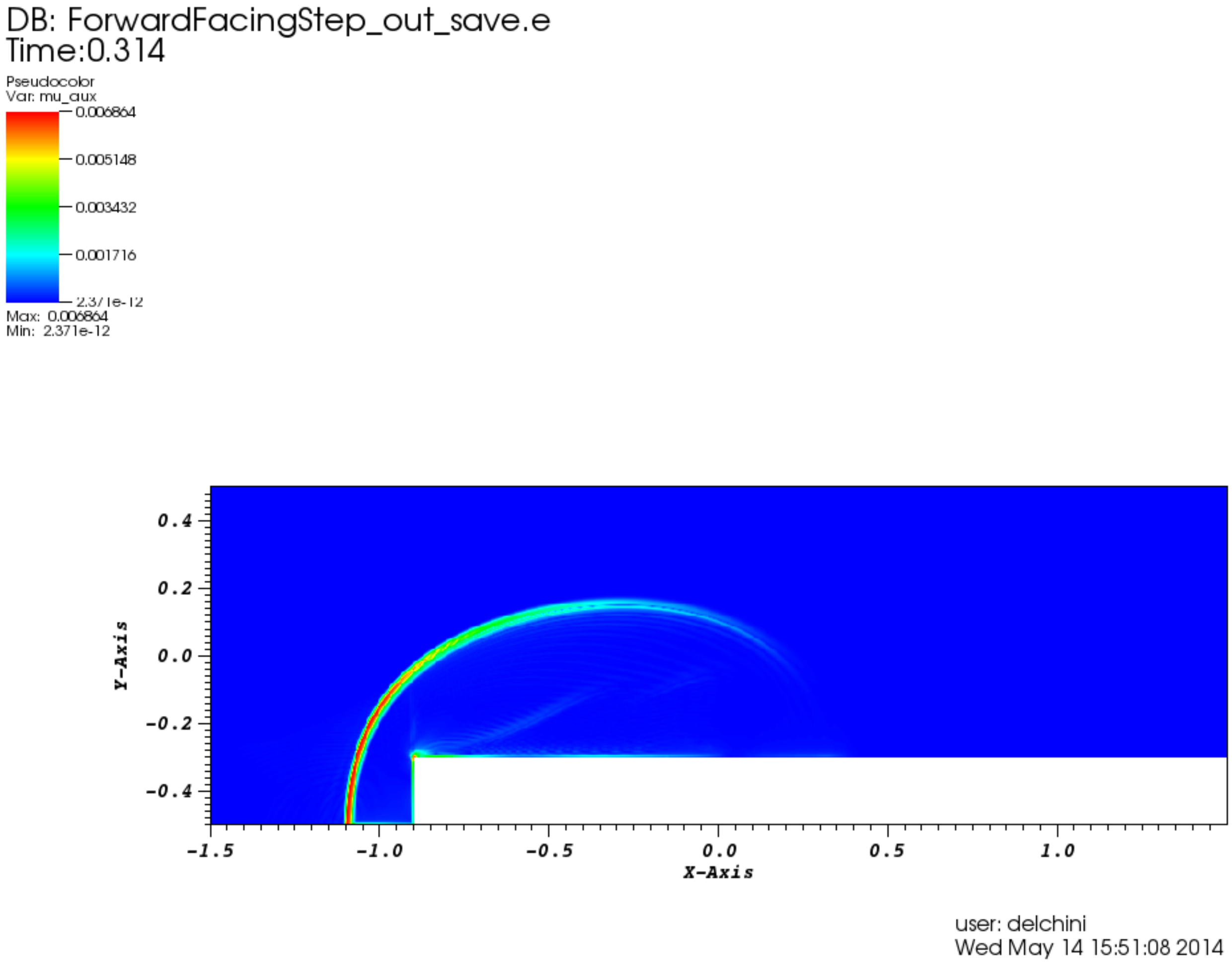


Figure 5.7: Viscosity coefficient solution at $t = 0.314$ s.

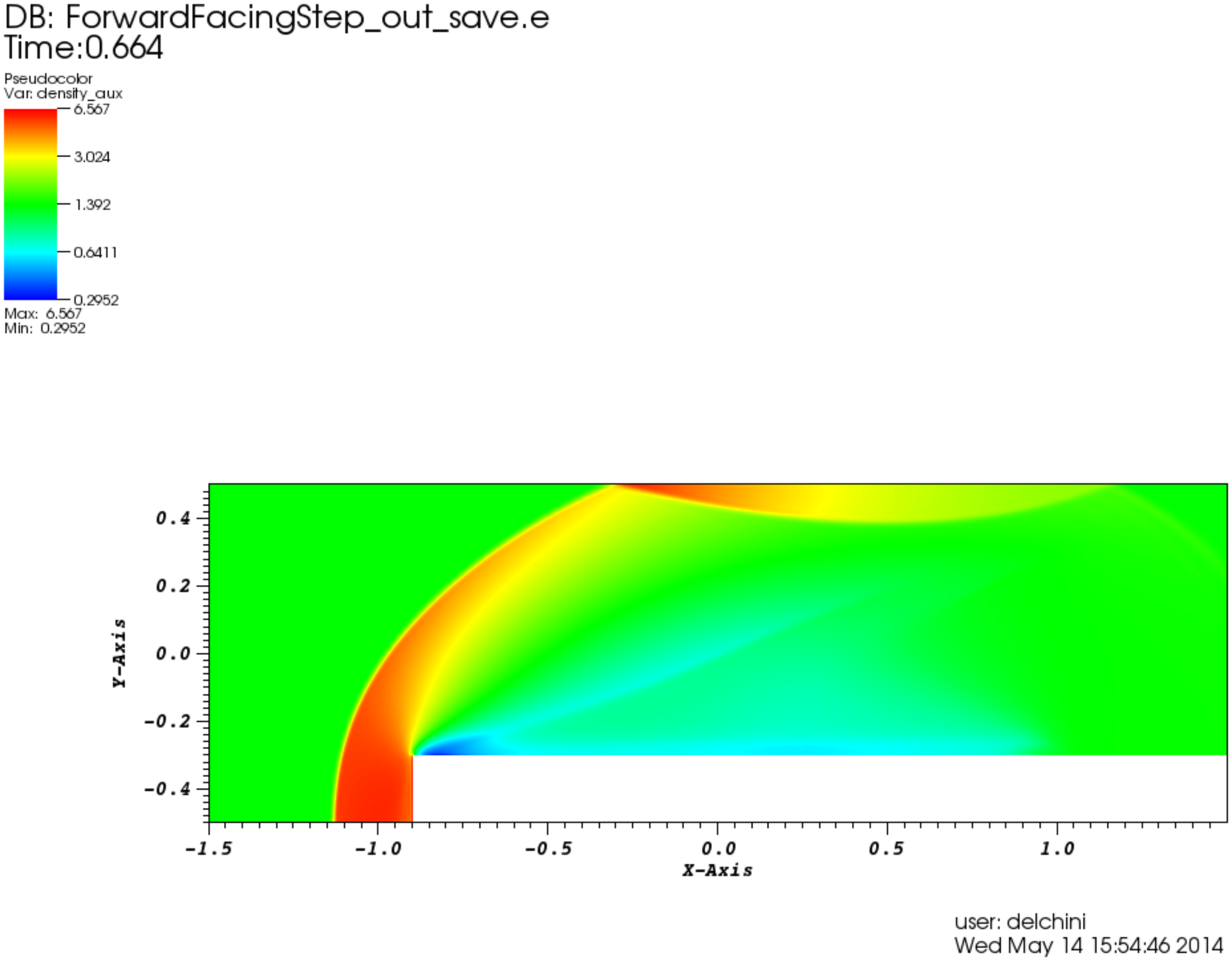


Figure 5.8: Density solution at $t = 0.664$ s.

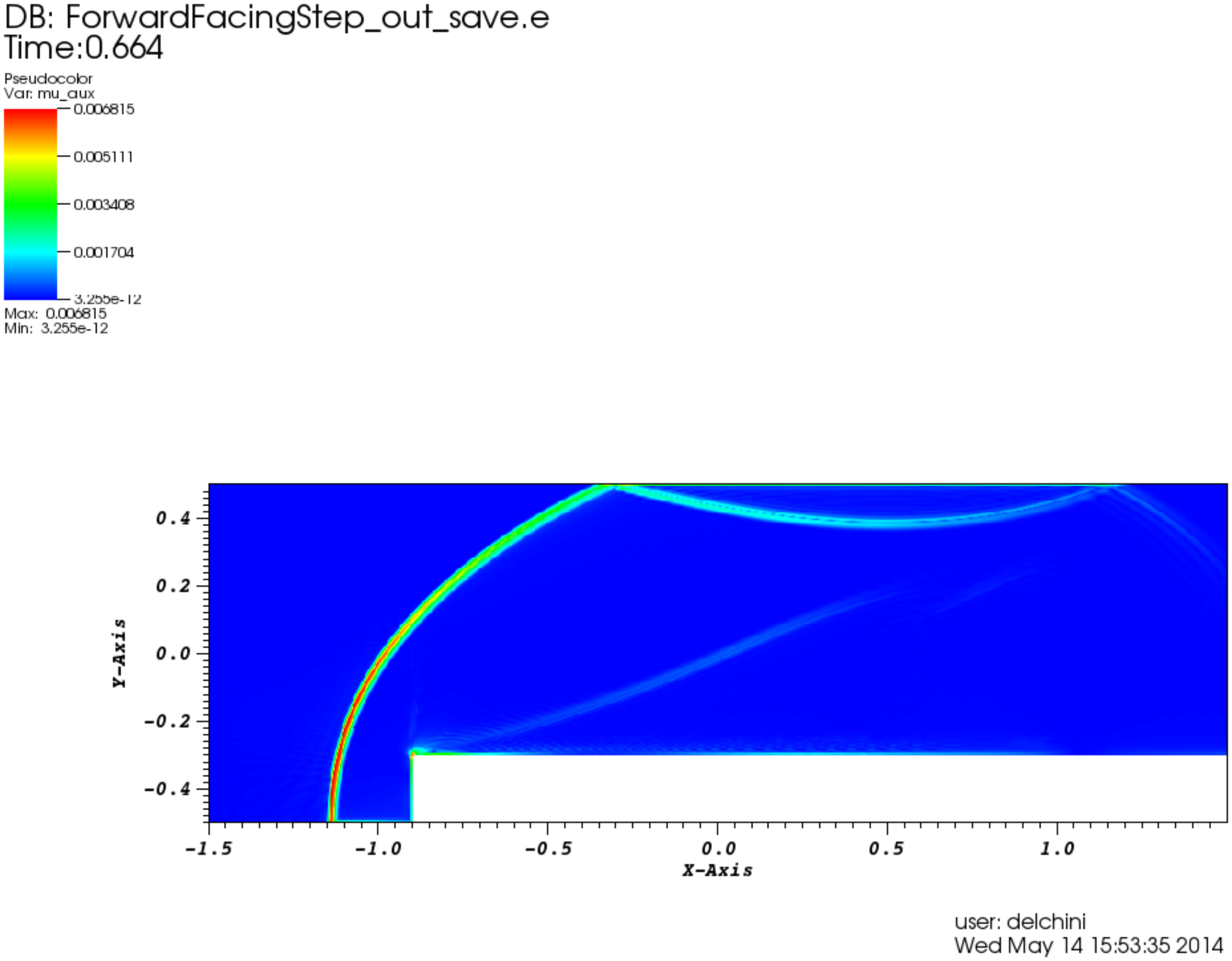


Figure 5.9: Viscosity coefficient solution at $t = 0.664$ s.

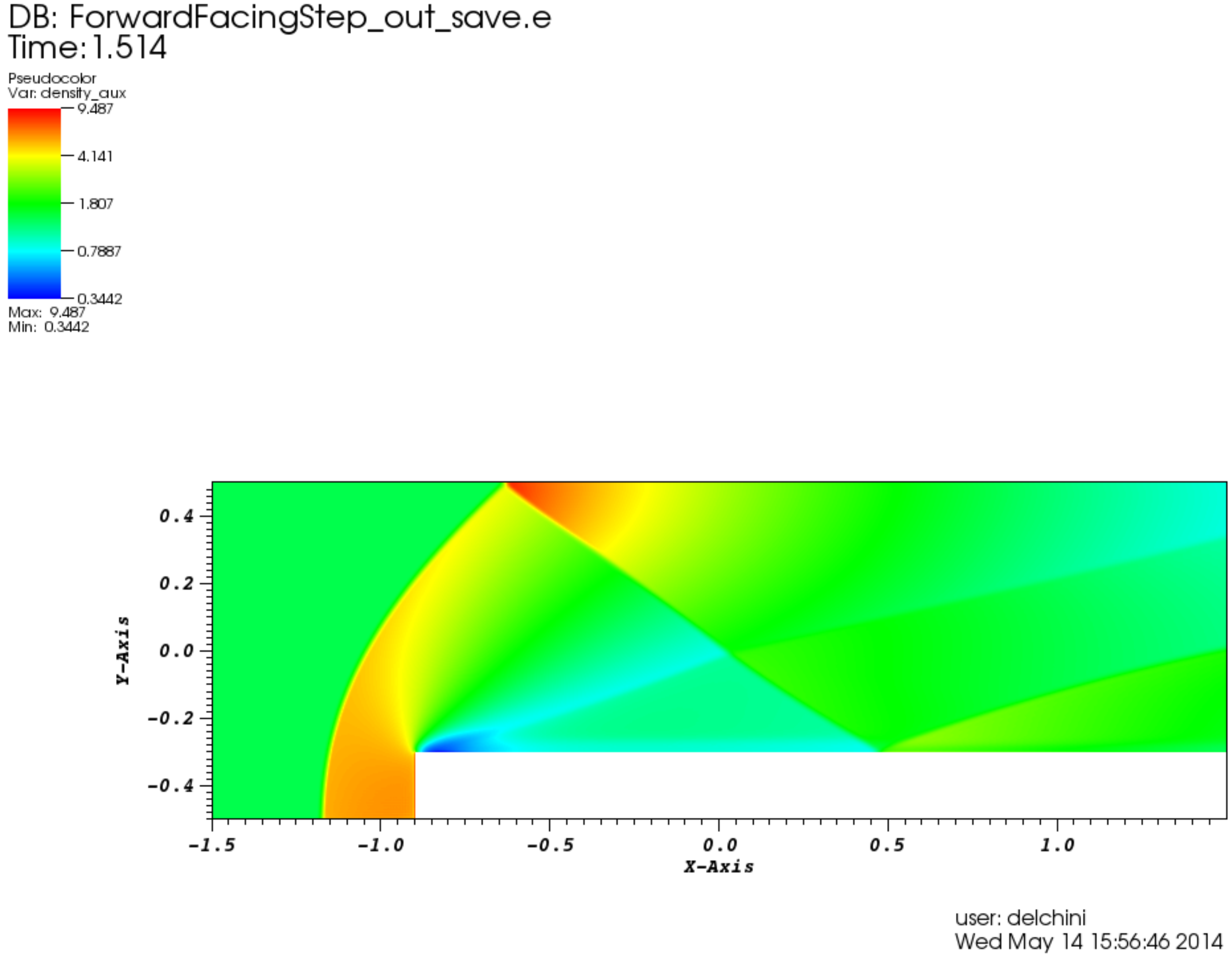


Figure 5.10: Density solution at $t = 1.514$ s.

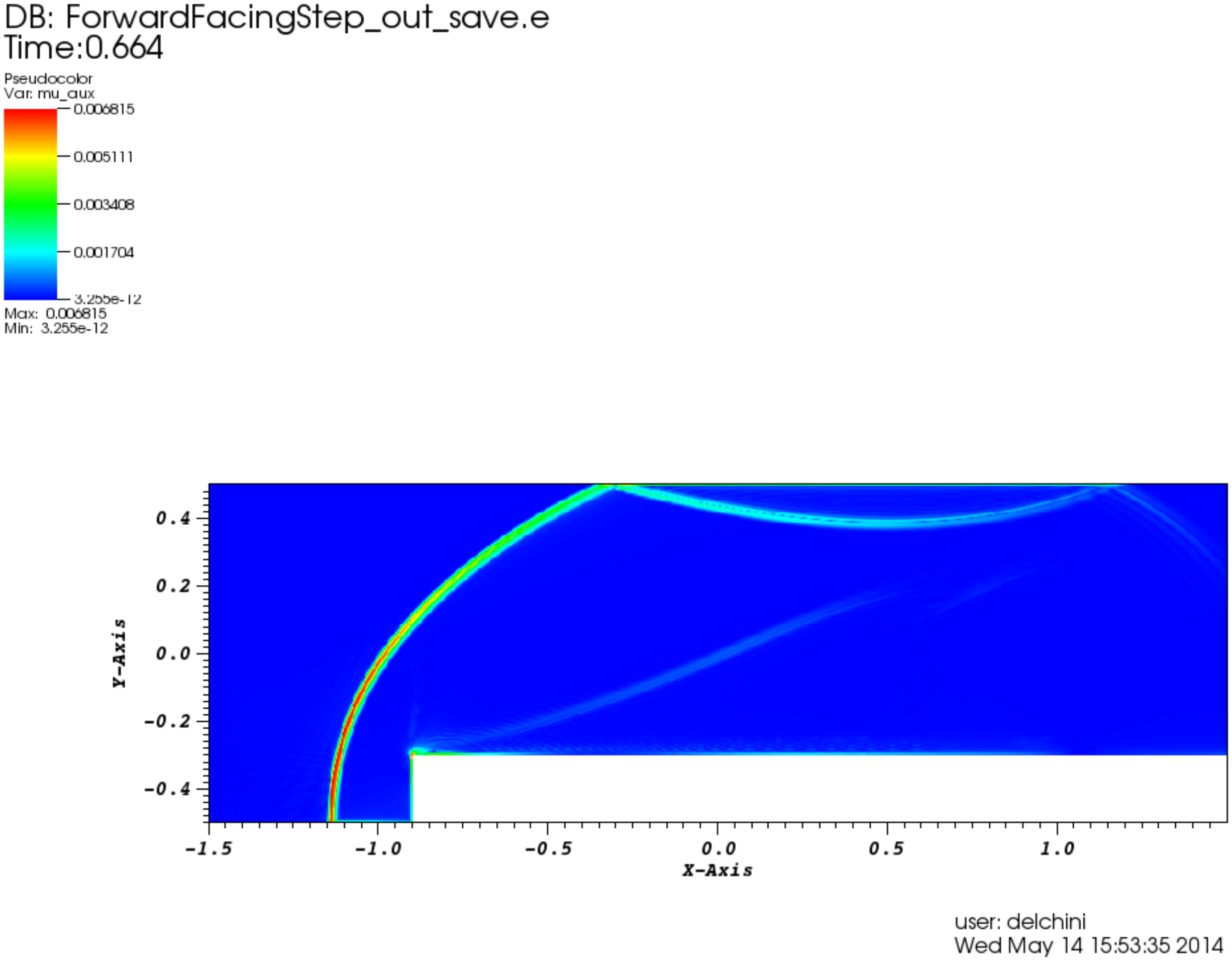


Figure 5.11: Viscosity coefficient solution at $t = 1.514$ s.

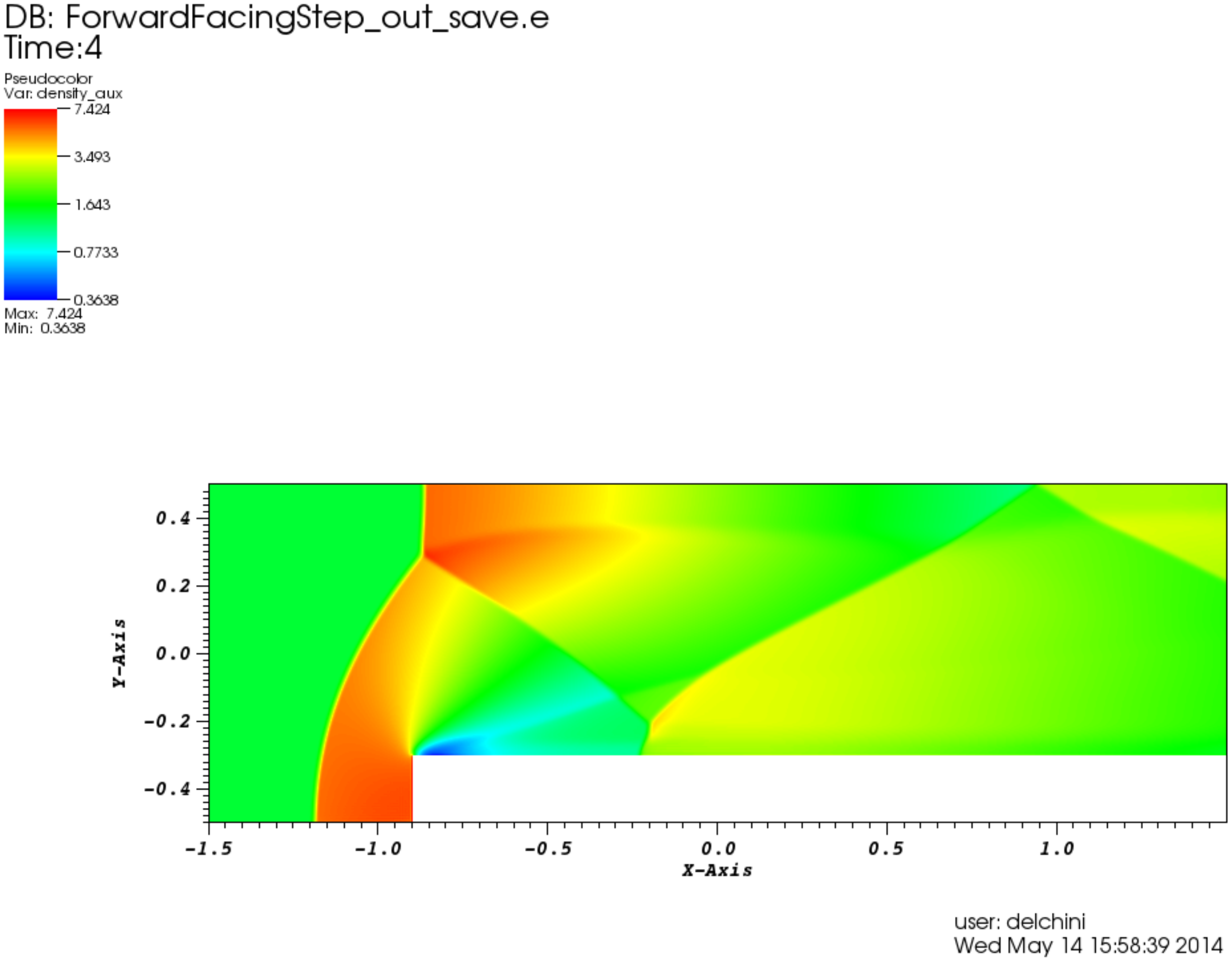


Figure 5.12: Density solution at $t = 4$ s.

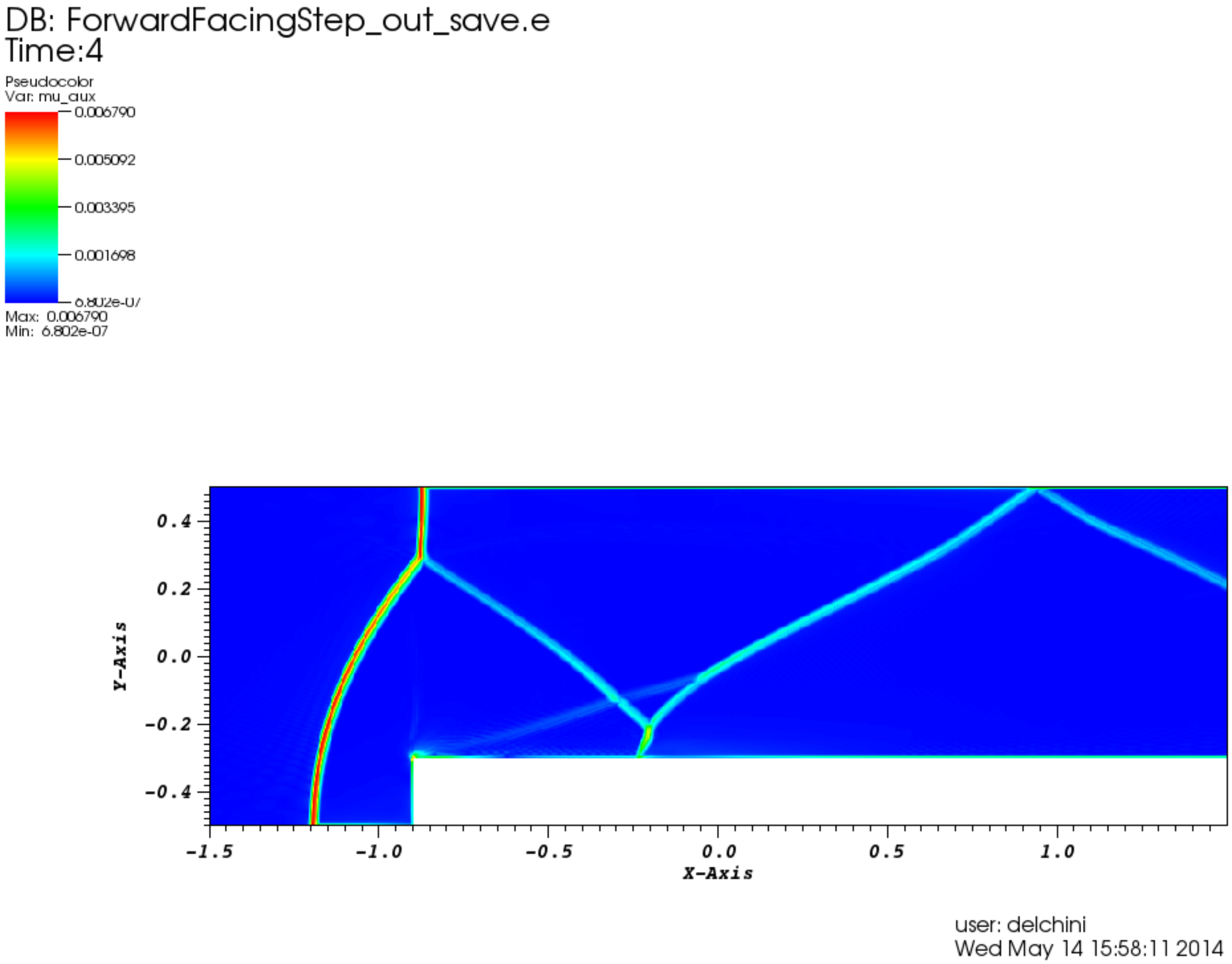


Figure 5.13: Viscosity coefficient solution at $t = 4$ s.

The numerical solution of the density at $t = 4$ s compares well to the ones obtained in (REFS), as least in a visual norm. The triple-point feature and the contact wave emerging from it are well resolved even if no Kelvin-Helmotz instability is observed (REF). It is also noticed that a significant amount of entropy is produced near the corner region. This is due to the corner singularity and this phenomenon is well documented (REFS). This artifact can be treated either by using special boundary condition to the corner since its normal vector is not defined, or by aggressively refining the mesh in the singularity region, or lastly, by modifying the geometry and use a round corner.

5.8.3 Riemann problem number 12:

Riemann problem number 12 (REF) is a popular 2-D benchmark that is known to develop contact and shock waves, as well as fine structures. The computational domain $\Omega = (0,1)^2$ is initialized with four square regions. In each region a different set of initial values is set as follows:

$$\begin{cases} \rho = 4/5, & \vec{u} = (0,0) \quad P = 1 \quad \text{for } x \in (0,0.5) \text{ and } y \in (0,0.5) \\ \rho = 1, & \vec{u} = (3/\sqrt{(}17),0) \quad P = 1 \quad \text{for } x \in (0,0.5) \text{ and } y \in (0.5,1) \\ \rho = 17/32, & \vec{u} = (0,3/\sqrt{17}) \quad P = 1 \quad \text{for } x \in (0.5,1) \text{ and } y \in (0,0.5) \\ \rho = 17/32, & \vec{u} = (0,0) \quad P = 2/5 \quad \text{for } x \in (0.5,1) \text{ and } y \in (0.5,1) \end{cases}$$

The same equation of state as in Section 5.8.1 is used. An uniform mesh of 400×400 elements was used along with a $\mathbb{Q}_1$ continuous Galerkin finite element method and the second-order temporal integrator $BDF2$. The simulation is run until $t = 0.25s$ with a CFL of 1. The density and viscosity profiles are given in (FIGS).

5.8.4 Supersonic flow in a compression corner

This is an example of a supersonic flow over a wedge of angle 15° where an oblique shock is generated at steady-state. The Mach number upstream of the shock is fixed to $M = 2.5$. The initial conditions are uniform: the pressure and temperature are set to $P = 101325\ Pa$ and $T = 300\ K$, respectively. The initial velocity is computed from the upstream Mach number and using the Ideal Gas equation of state with the same parameters as in Section 5.9.2. The code is run until steady-state. An analytical solution for this supersonic flow is available and give the downstream to upstream pressure, entropy and Mach number ratios [3]. The analytical and numerical ratios are given in see in Table 5.11, and are very close. The pressure and viscosity coefficient solution are given for different times in Fig. 5.14 - Fig. 5.19.

Table 5.11: Analytical solution for the supersonic flow on an edge eat 15° at $M = 2.5$.

	analytical downstream to upstream ratio	numerical downstream to upstream ratio
Pressure	2.47	2.467
Mach number	0.74	0.741
Entropy	1.03	1.026

The inlet is supersonic and therefore, the pressure, temperature and velocity are specified using Dirichlet boundary conditions. The outlet is also supersonic and none of the characteristics enter the domain through this boundary: the values will be computed by the implicit solver.

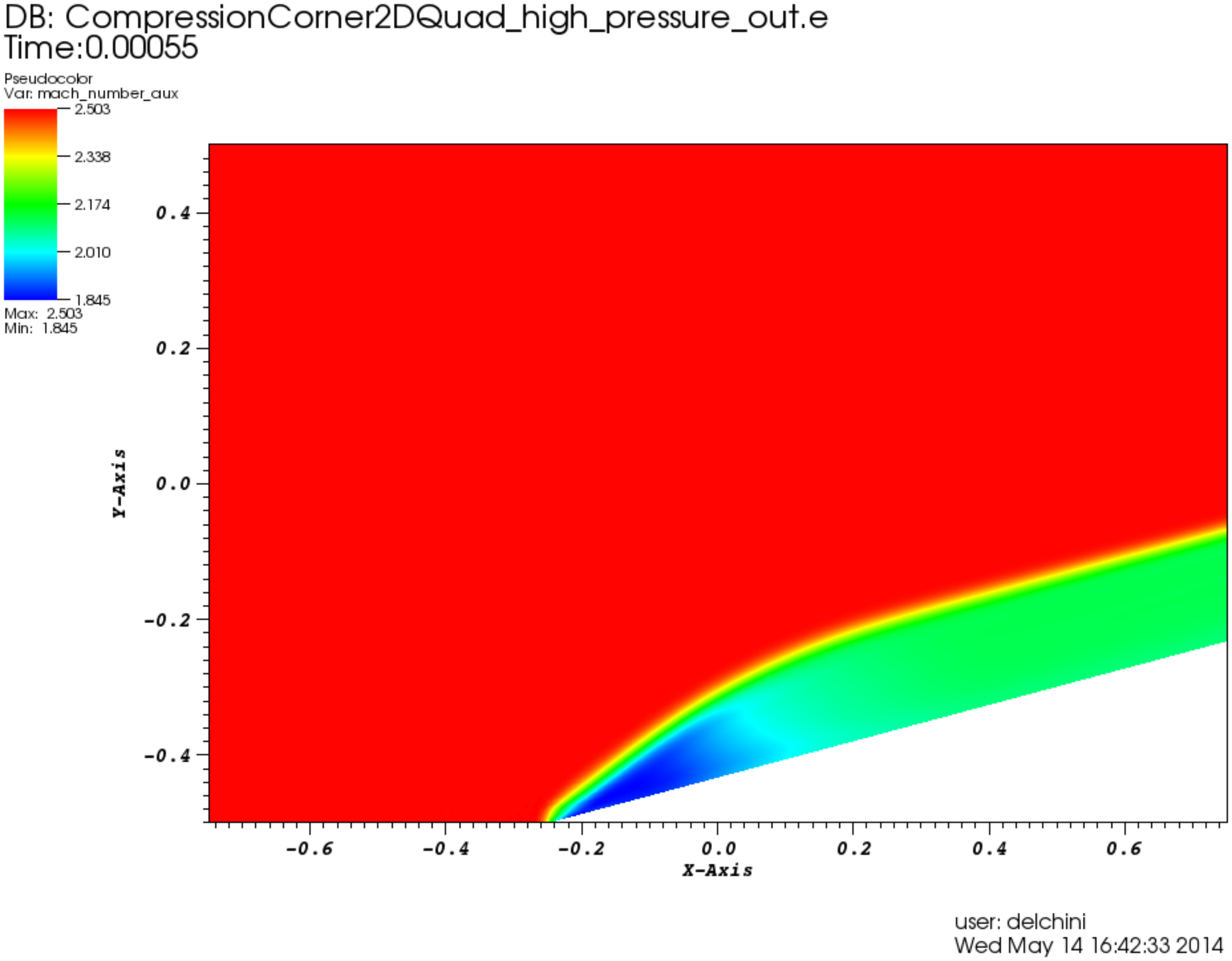


Figure 5.14: Pressure solution at $t = 5.5 \times 10^{-4}$.

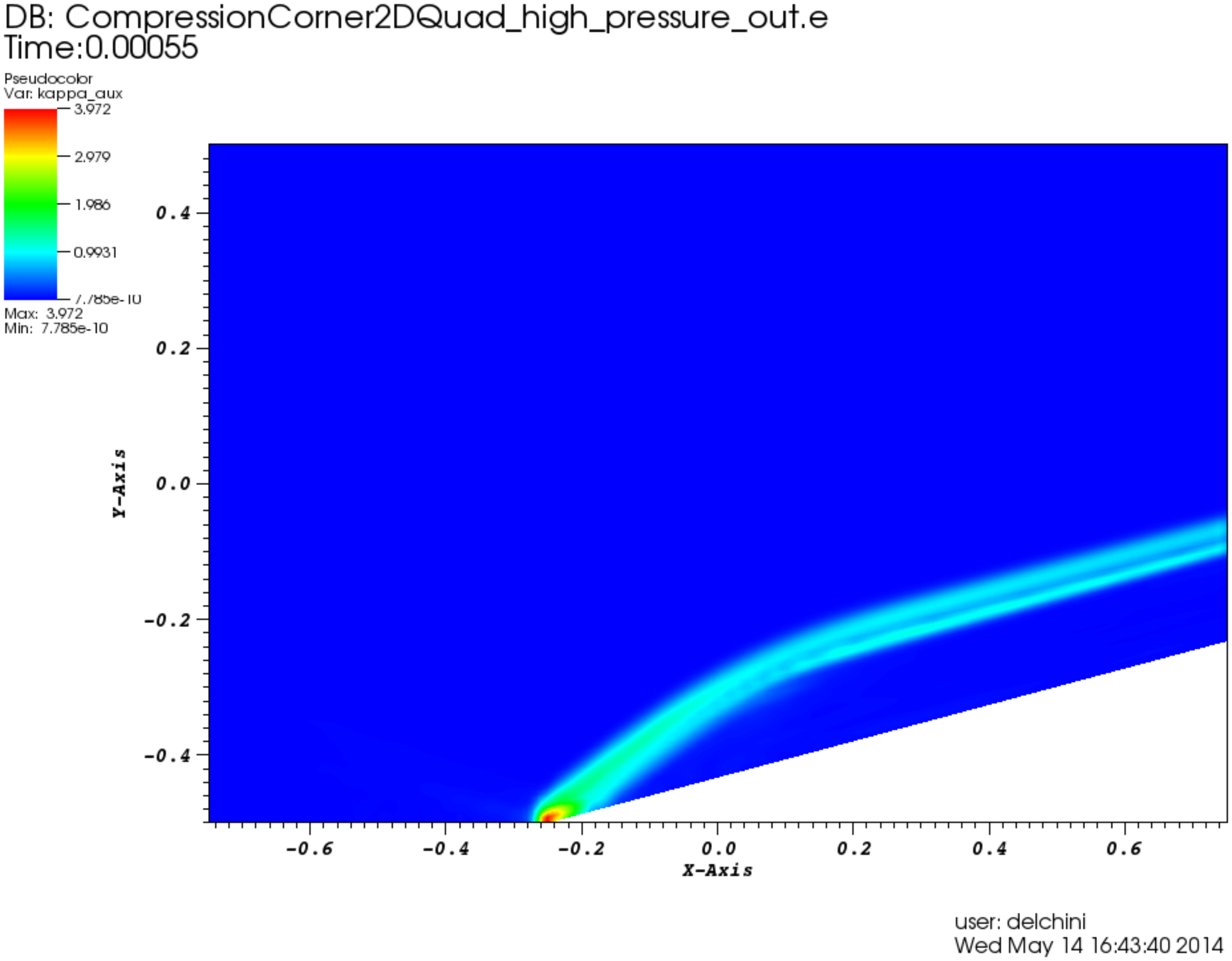


Figure 5.15: Viscosity coefficient at $t = 5.5 \times 10^{-4}$.

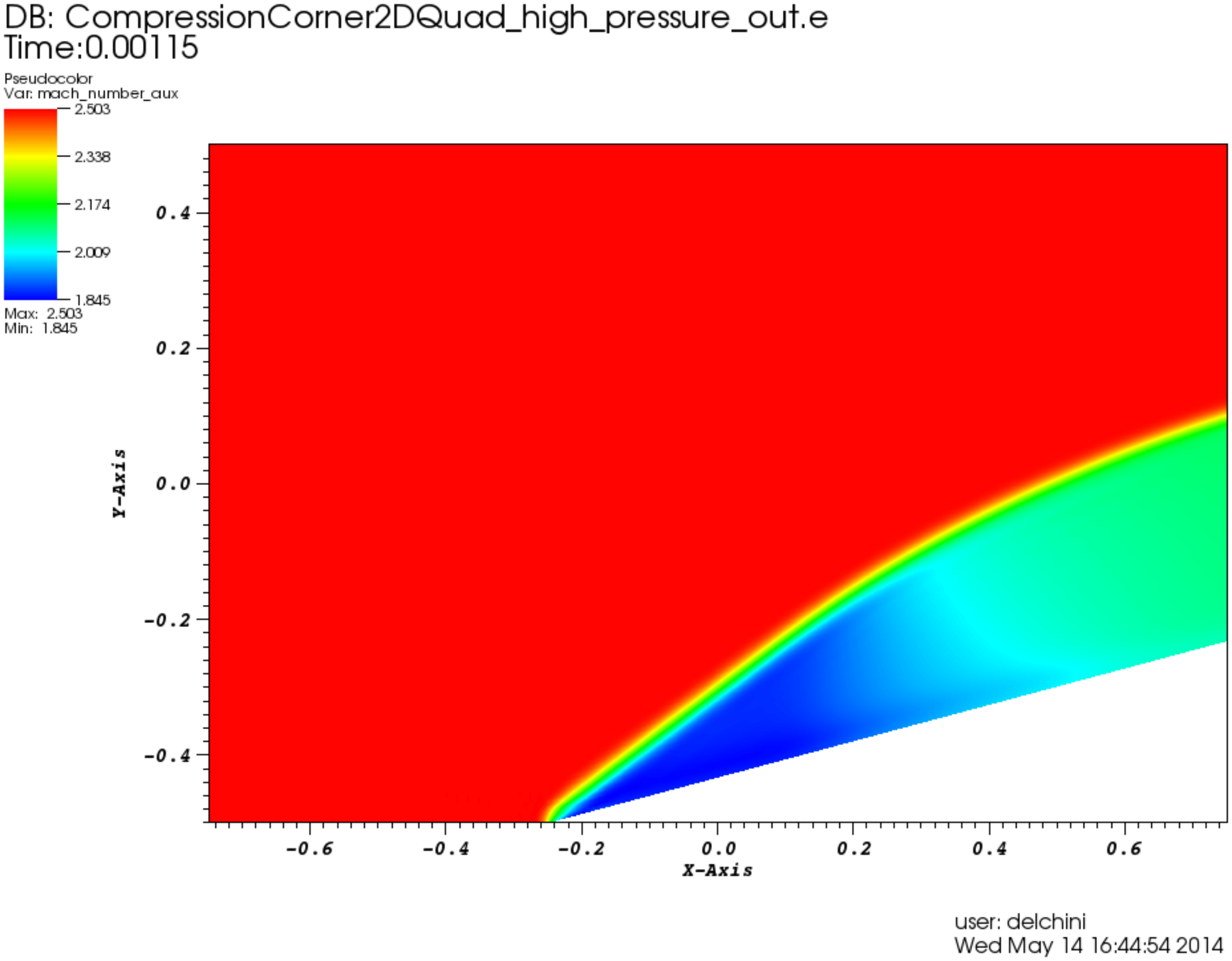


Figure 5.16: Pressure solution at $t = 1.15 \times 10^{-3}$.

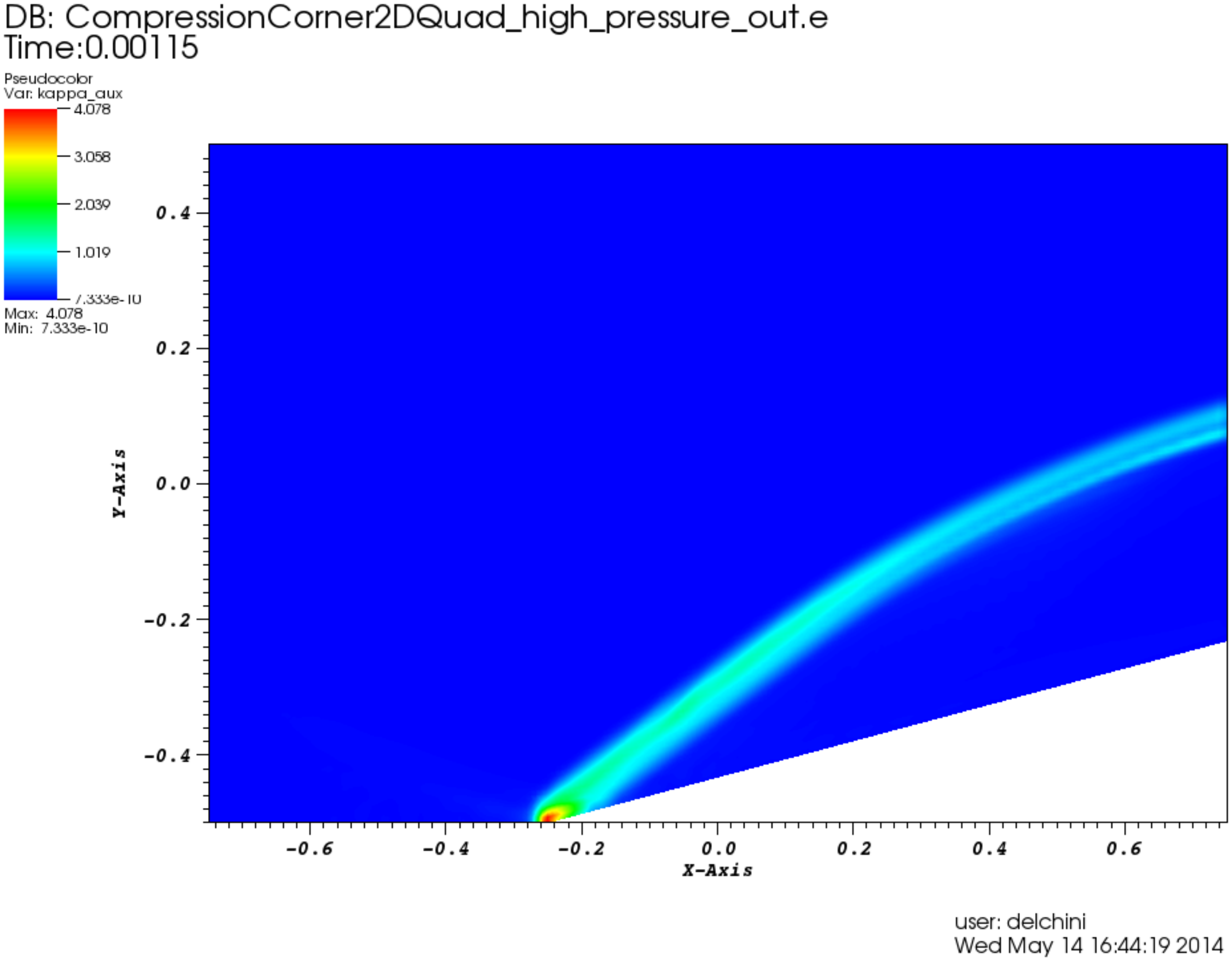


Figure 5.17: Viscosity coefficient at $t = 1.15 \times 10^{-3}$.

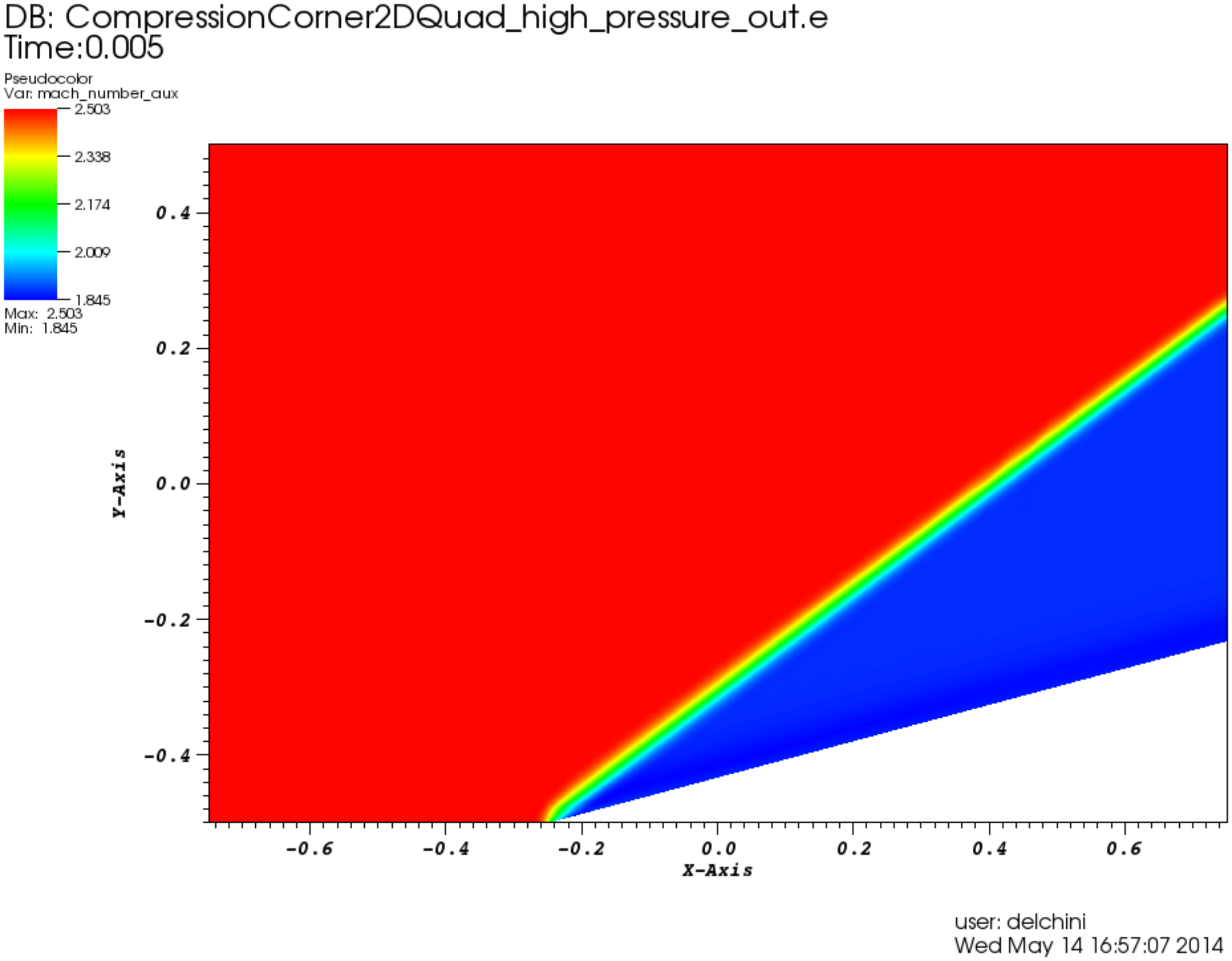


Figure 5.18: Pressure solution at steady-state.

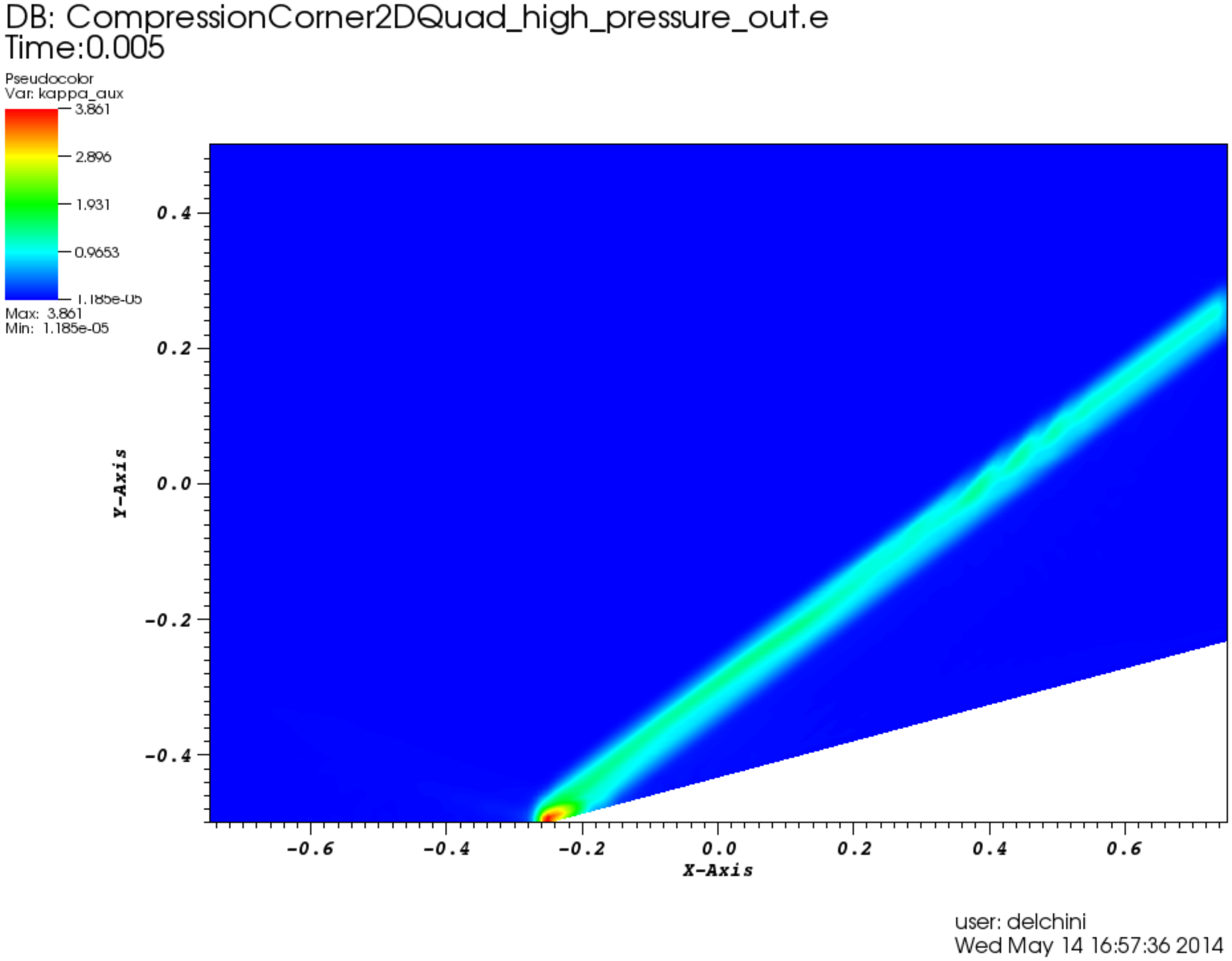


Figure 5.19: Viscosity coefficient at steady-state.

From the above figures, it is observed that the solution is formed of two regions of constant state. During the transient, the shock moves from the bottom wall to its steady-state solution. The same variations are observed in viscosity coefficient solution. At steady-state, the viscosity coefficient is large in the shock region and small anywhere else and thus, behaves as expected. At the corner of the edge at $x = -0.25\ m$, the viscosity coefficient is peaked because of the treatment of the wall boundary condition: at this particular node, the normal is not well defined and can cause numerical errors.

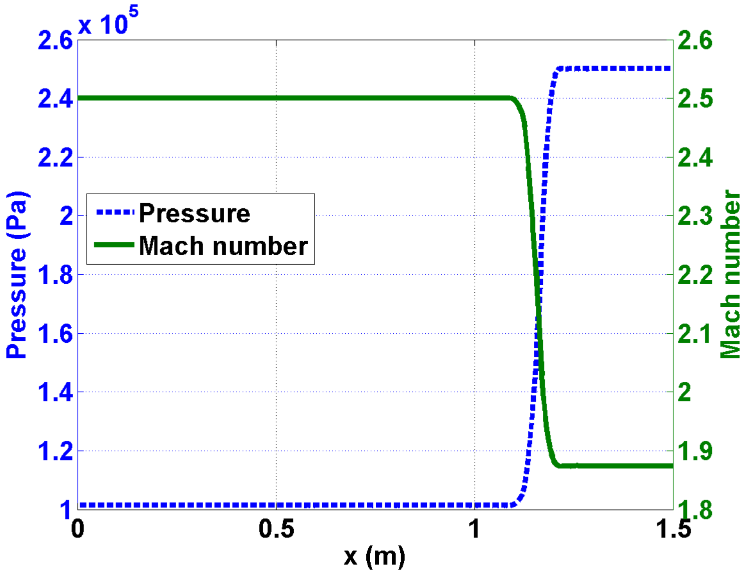


Figure 5.20: Pressure and Mach number profiles at steady-state

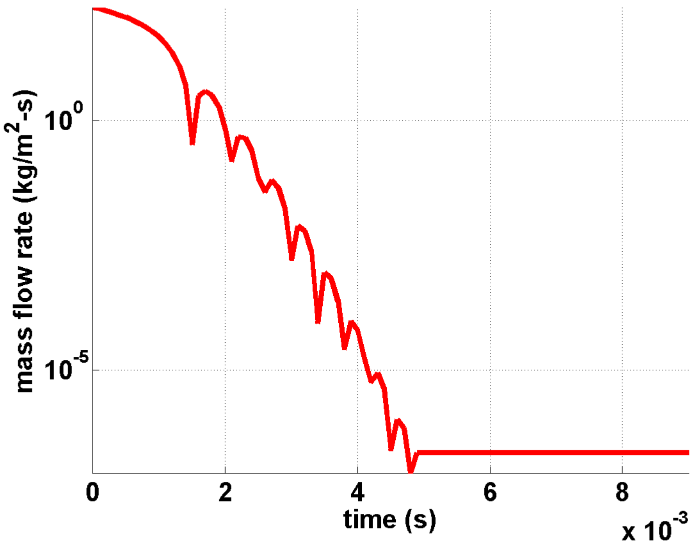


Figure 5.21: Difference between inlet and outlet mass flow rates as a function of time.

The 1-D plots of the pressure and the mach number at $y = 0$, are also given in Fig. 5.20: the shock does not show any spurious oscillations and is well resolved. Finally, the difference between the inlet and outlet mass flow rates is plotted in Fig. 5.21 and show that a steady-state is reached.

Overall, the numerical solution does not show any oscillations, match the analytical solution, and the shock is well resolved.

5.8.5 Supersonic flow over a 5° *double-wedge obstruction:*

The last of the 2-D supersonic example that is proposed to study is a Mach2, 5 flow over a double-wedge obstruction located on the lower wall. The interesting feature of this test is that a steady-state is reached. The geometry used is shown in (FIG) and was discretized with 4000 $\mathbb{Q}_1$ elements. The double wedge extends on the bottom boundary from $x = 1$ to $x = 5$ m. The top wall is located at $y = 5$ m. A supersonic inlet boundary condition was set at the inlet by specifying the pressure, $P = 101,325$

Pa, the temperature, $T = 300\ K$ and the vector velocity $\vec{u} = (868.032, 0)\ m \cdot s^{-1}$. The wall-boundary and supersonic outlet boundary conditions were implemented following the method described in (SECTION BCS). The second-order temporal integrator $BDF2$ was used with a CFL of 5 to reach the steady-state that was detected by monitoring the norm of the total residual. The Ideal gas equation of state was used with an adiabatic constant $\gamma = 1.4$ and a volumetric heat capacity $C_v = 716.7\ J \cdot K^{-1}$ (air properties). The Mach number and viscosity coefficients profiles at steady-state are given in Fig. 5.22 and Fig. 5.23, respectively.

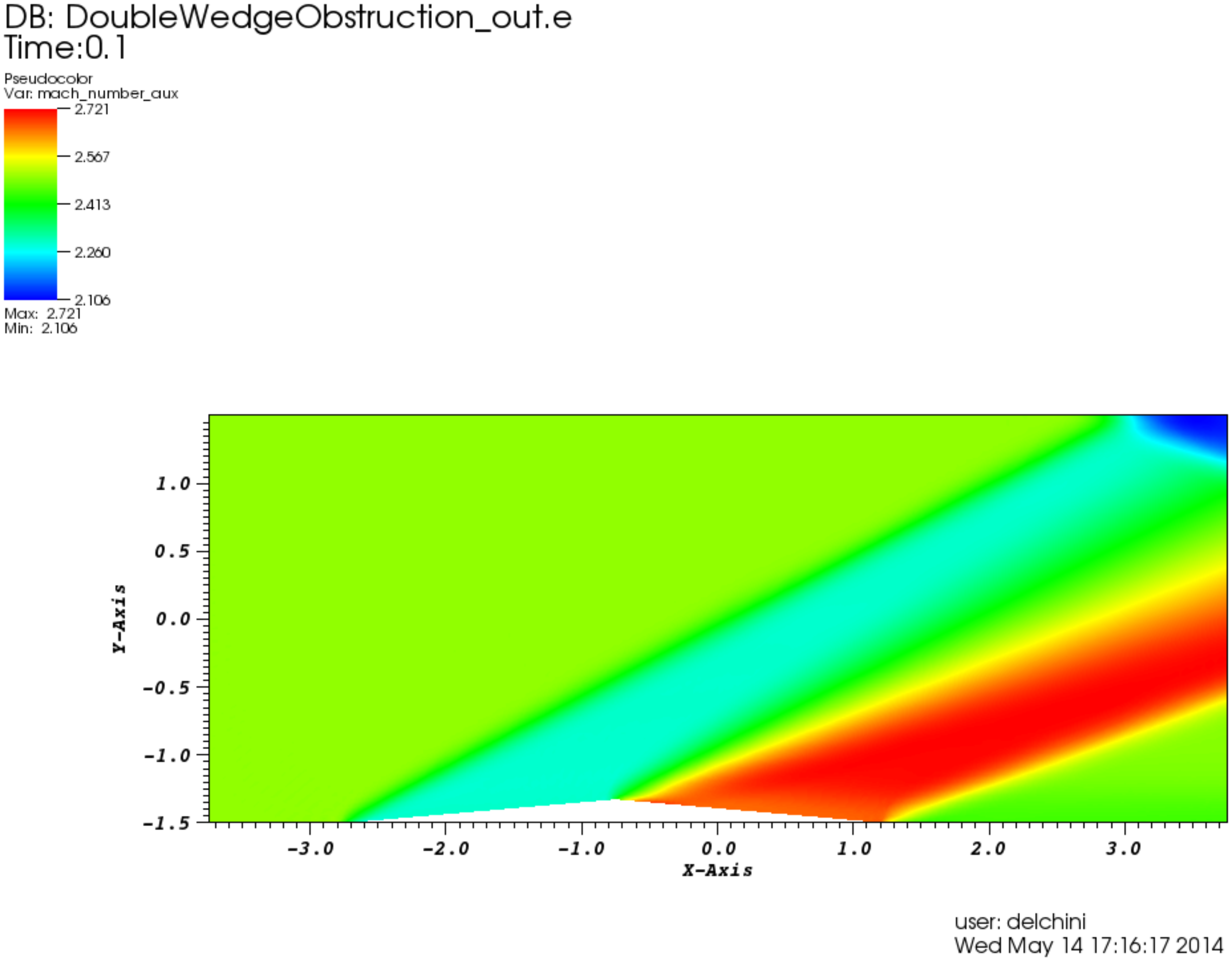


Figure 5.22: Pressure solution at steady-state.

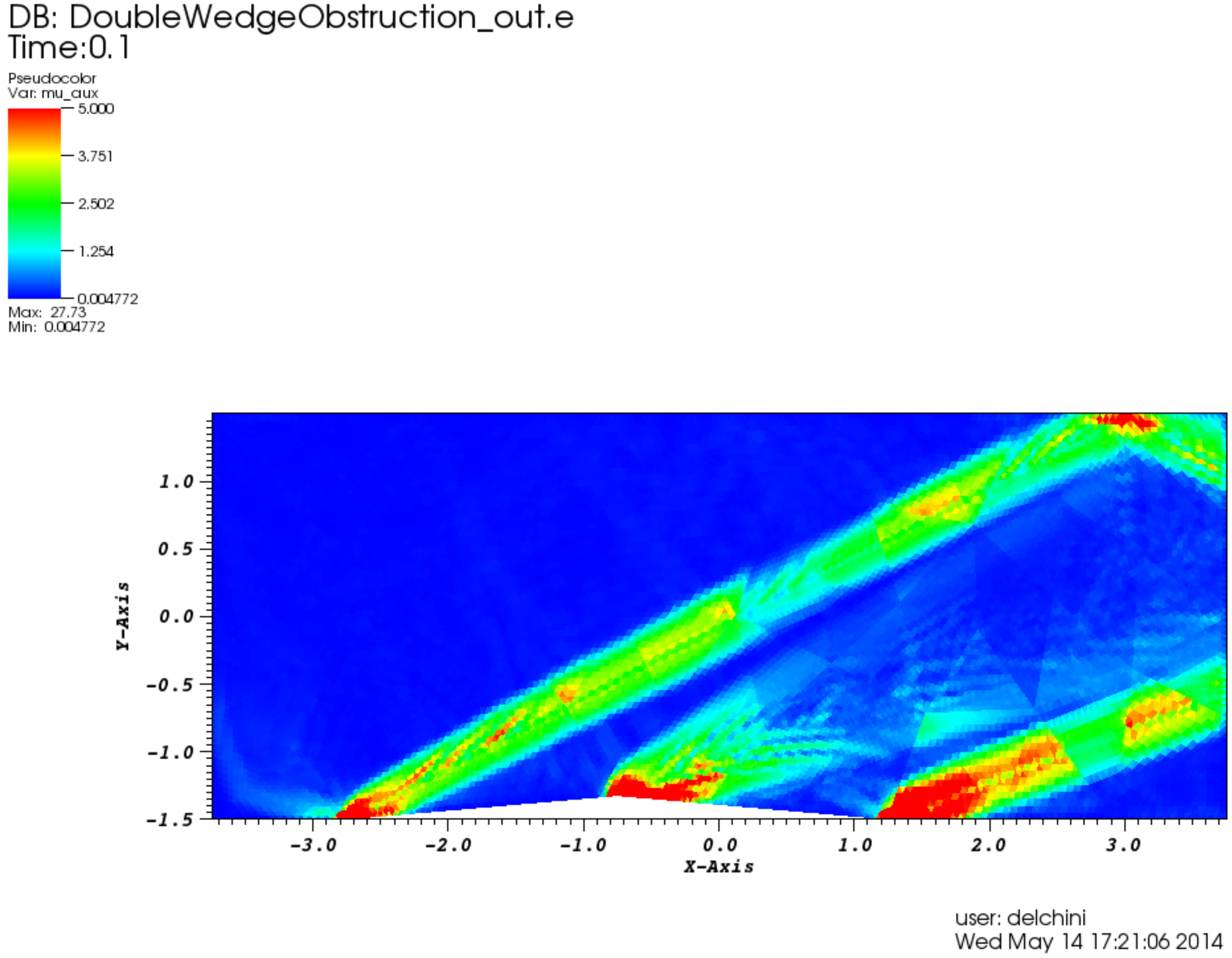


Figure 5.23: Viscosity coefficient at steady-state.

The stay-state solution consists of a two shocks that form because of the interaction of the flow with the double wedge. The first wedge generates a shock that reflects on the top wall and then exits the computational domain: the interaction of the shock with the wall close to the outlet boundary requires a robust implentation of the boundary conditions and the stabilization method. The second shock is generated by the trailing wedge. In between the two shock regions, an expansion fan is formed.

5.9 2-D numerical results for subsonic flows:

5.9.1 Subsonic flow over a 2-D cylinder

Fluid flow over a 2-D cylinder is often used as a benchmark case to test numerical schemes in the low-Mach regime [24, 56, 34]. For this test, an analytical solution is available in the incompressible limit or low-Mach limit and is often referred to as the potential flow solution. The main features of the potential flow are the following:

- The solution is symmetric: the iso-Mach contour lines are used to asses the symmetry of the numerical solution;
- The velocity at the top of the cylinder is twice the incoming velocity set at the inlet;
- The pressure fluctuations are proportional to the square of inlet Mach number, i.e.,

  $$\delta P = \frac{\max(P(\vec{r})) - \min(P(\vec{r}))}{\max(P(\vec{r}))} \propto M_\infty^2 \tag{5.43}$$

  where δP and M_∞ denote the pressure fluctuations and the inlet Mach number, respectively.

The computational domain consists of a 1×1 square with a circular hole of radius 0.05 in its center. A $\mathbb{P}_1$ triangular mesh with 4008 triangular elements was used to discretize the geometry. The ideal gas equation of state, with $\gamma = 1.4$ is used. At the inlet, a subsonic stagnation boundary condition is used: the stagnation pressure and temperature are computed using the following relations:

$$\begin{cases} P_0 = P\left(1 + \frac{\gamma-1}{2}M^2\right)^{\frac{\gamma-1}{\gamma}} \\ T_0 = T\left(1 + \frac{\gamma-1}{2}M^2\right) \end{cases} \tag{5.44}$$

An static pressure boundary condition is used for the outlet boundary and the following static pressure $P_s = 101,325\ Pa$ is set. The implementation of the pressure boundary conditions is based of [49]. A solid wall boundary condition is set for the top and bottom walls of the computational domain. The simulations are run until a steady state is reached with a CFL of 40. The steady state is considered reached when the residual norm (for all equations) is less than 10^{-12}.

Several simulations are performed, with inlet Mach numbers M_{inlet} ranging from 10^{-3} to 10^{-7}, and are shown from Fig. 5.24 through Fig. 5.28. The iso-Mach contour lines are drawn using 30 equally-spaced intervals 2 10^{-10} to M_{inlet} and allow us to assess the symmetry of the numerical solution.

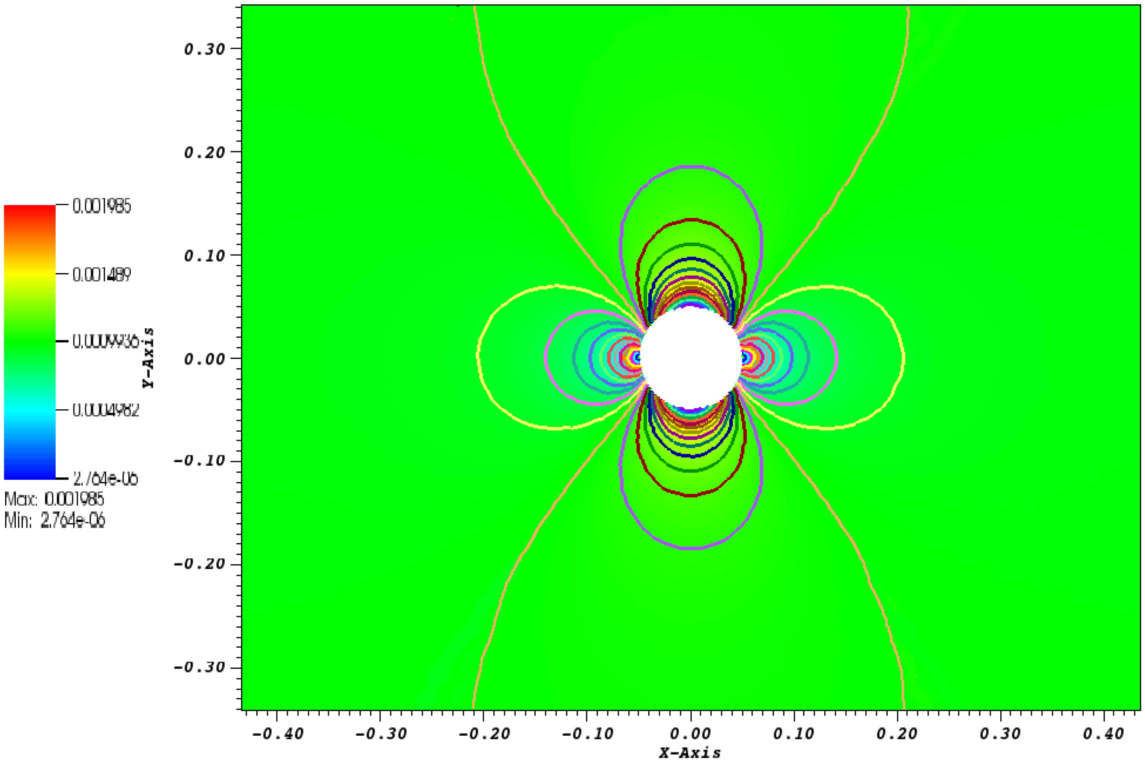


Figure 5.24: $M_{\text{inlet}} = 10^{-3}$

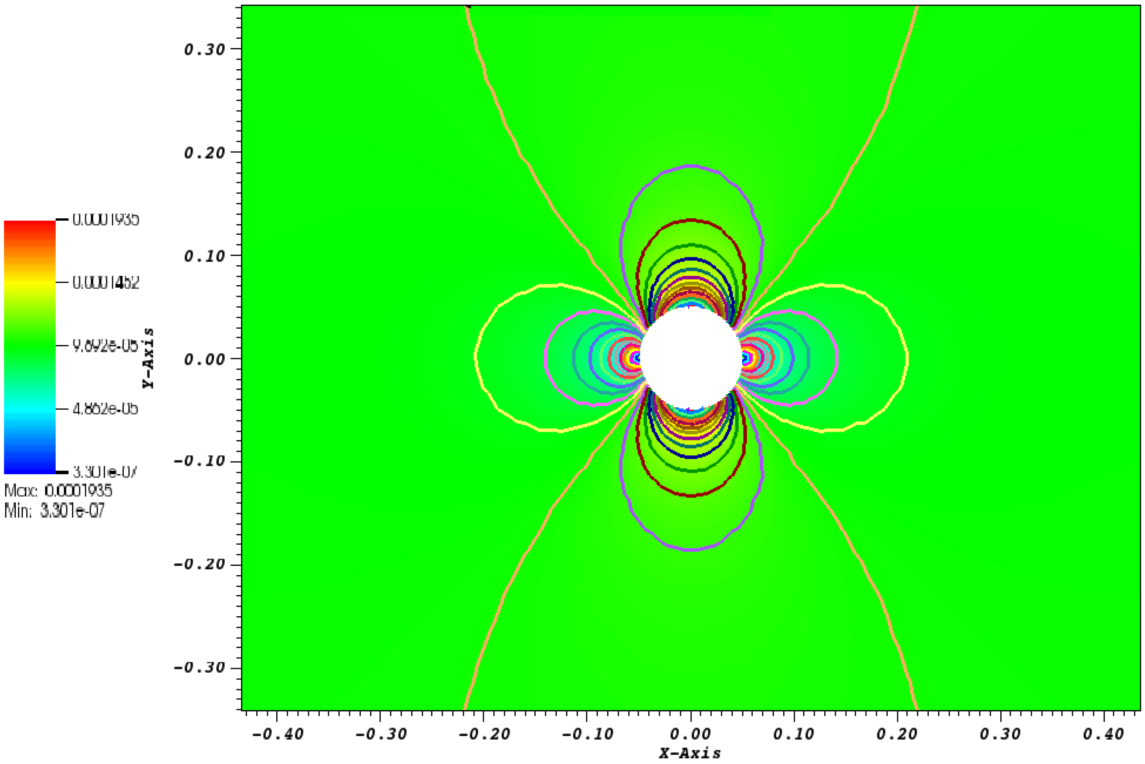


Figure 5.25: $M_{\text{inlet}} = 10^{-4}$

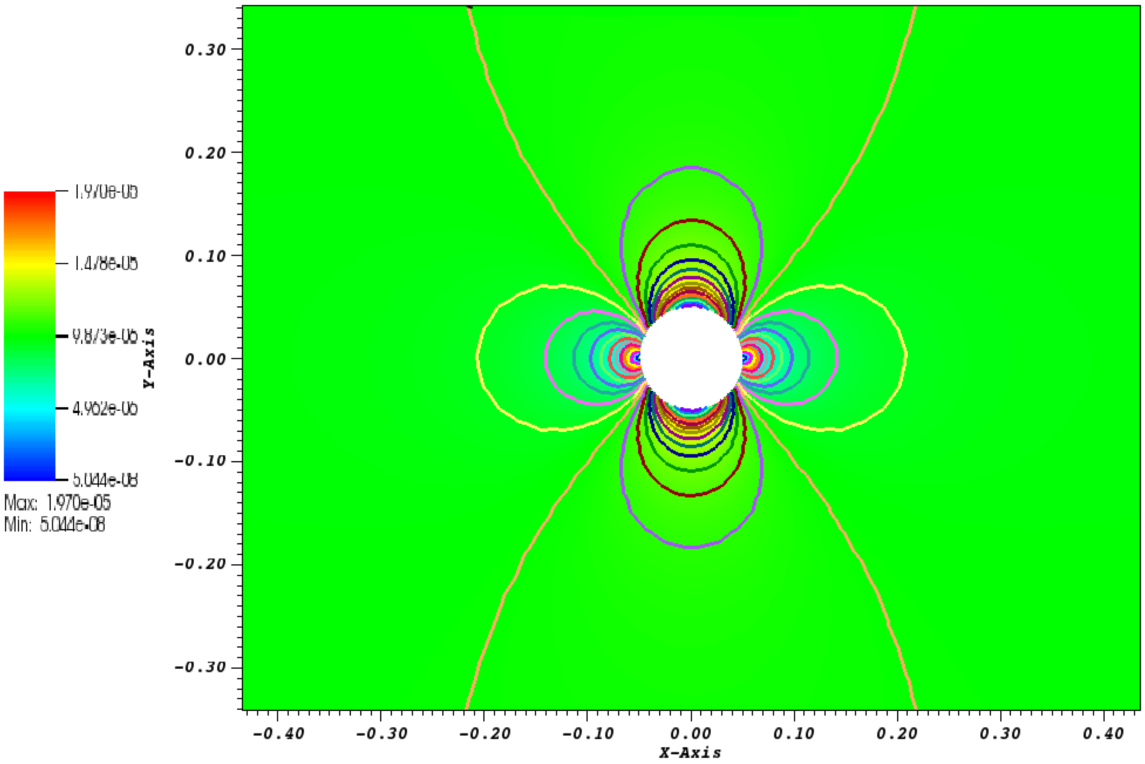


Figure 5.26: $M_{\text{inlet}} = 10^{-5}$

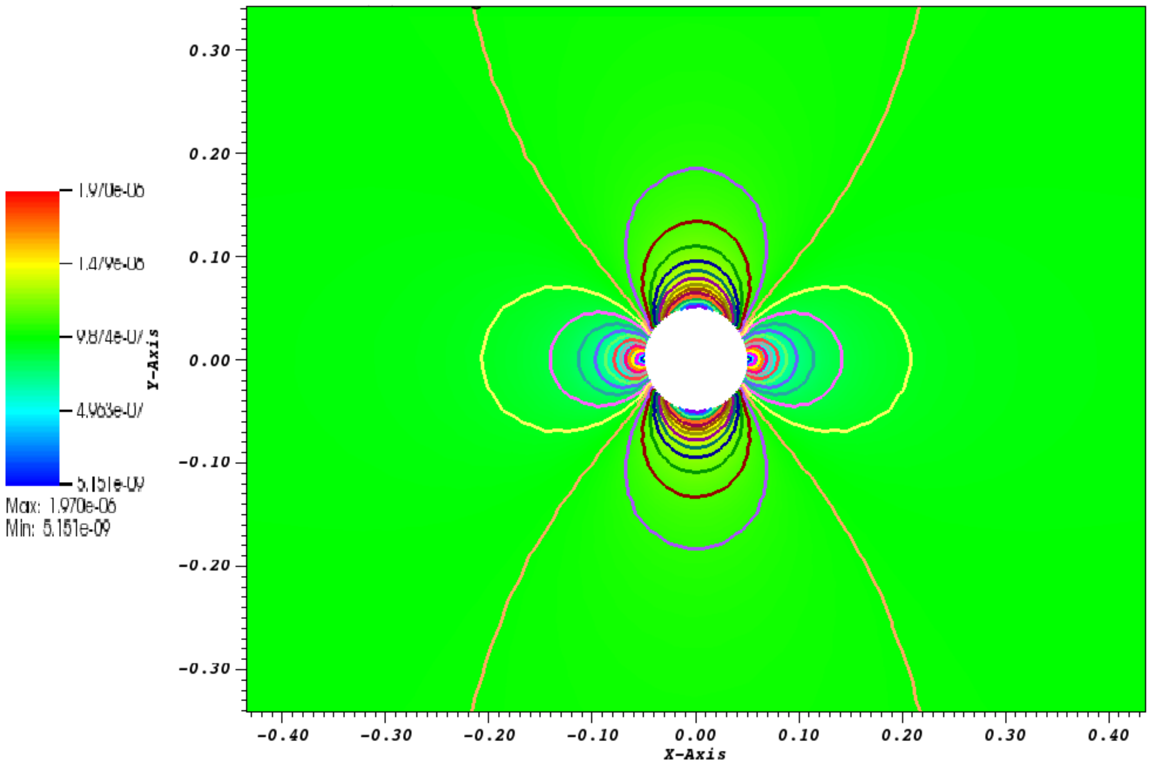


Figure 5.27: $M_{\text{inlet}} = 10^{-6}$

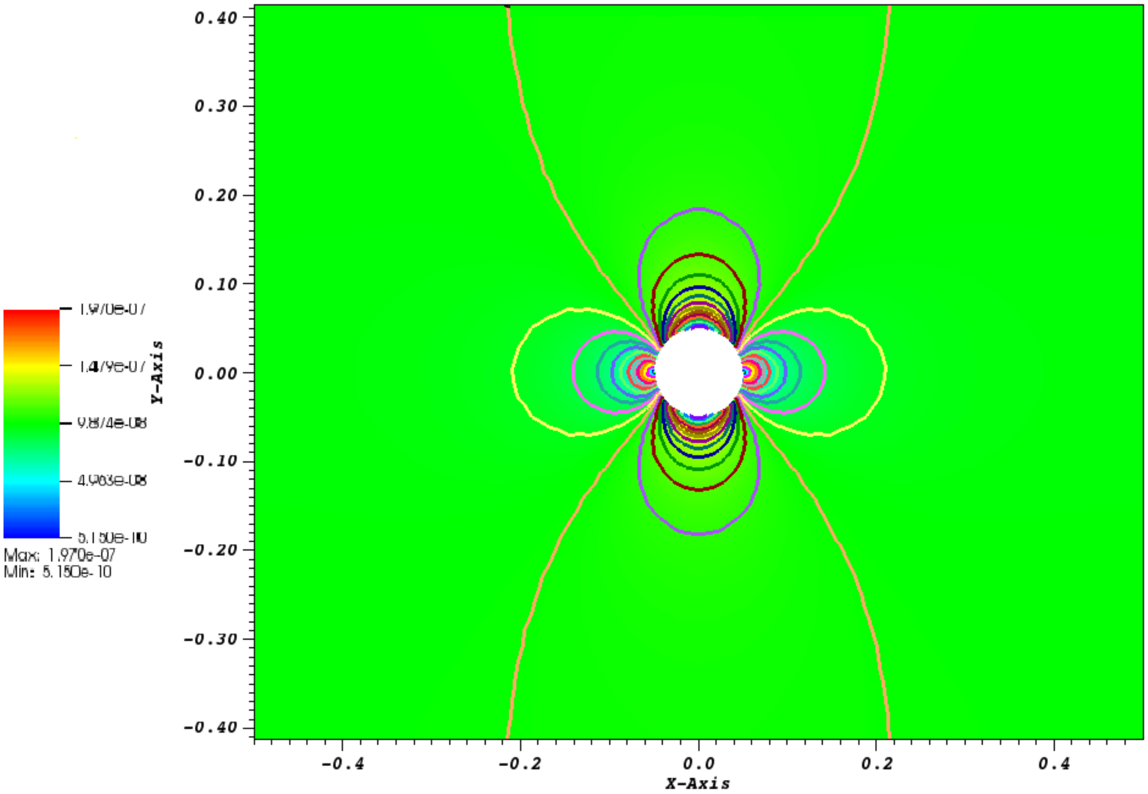


Figure 5.28: $M_{\text{inlet}} = 10^{-7}$

The velocity at the top of the cylinder and at the inlet are given for different Mach-number values (ranging from 10^{-3} to 10^{-7}) in Table 5.12. The ratio of the inlet velocity to the velocity at the top of cylinder is also computed and is very close to the theoretical value of 2 that is expected in the incompressible limit.

Table 5.12: Velocity ratio for different Mach numbers.

Mach number	inlet velocity	velocity at the top of the cylinder	ratio
10^{-3}	$2.348\ 10^{-3}$	$1.176\ 10^{-3}$	1.99
10^{-4}	$2.285\ 10^{-4}$	$1.145\ 10^{-4}$	1.99
10^{-5}	$2.283\ 10^{-5}$	$1.144\ 10^{-5}$	1.99
10^{-6}	$2.283\ 10^{-6}$	$1.144\ 10^{-6}$	1.99
10^{-7}	$2.283\ 10^{-7}$	$1.144\ 10^{-7}$	1.99

In Fig. 5.29, the fluctuations in pressure and velocity are plotted as a function of the Mach number (on a log-log scale). The fluctuations are expected to be of the order of M^2 and M for the pressure and velocity, respectively. It is known that some stabilization methods, e.g., [24, 56, 34], can produce pressure fluctuations with the wrong Mach-number order. Here, entropy viscosity method yields the correct order in the low-Mach limit. For ease of comparison, the reference lines with slope values of 1 and 2 are also plotted.

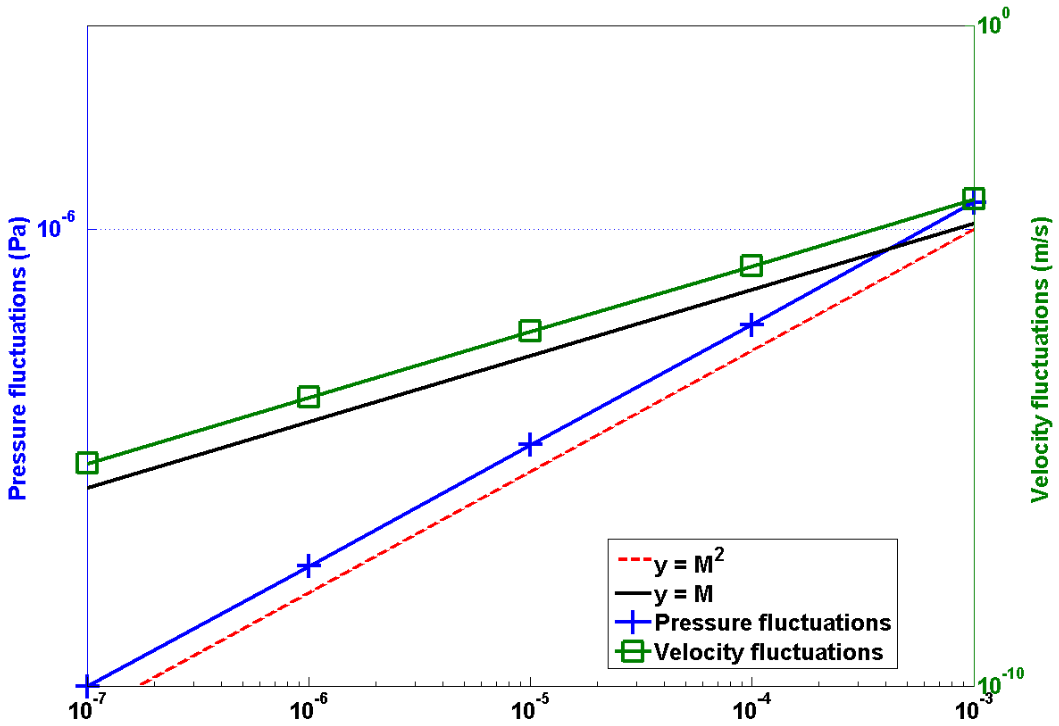


Figure 5.29: Log-log plot of the pressure and velocity fluctuations as a function of the far-field Mach number.

5.9.2 Subsonic flow over a 2-D hump

This is a another example of an internal flow configuration. It consists of a channel of height $L = 1\ m$ and length $3L$, with a circular bump of length L and thickness $0.1L$. The bump is located on the bottom wall at a distance L from the inlet. The system is initialized with an uniform pressure $P = 101,325\ Pa$ and temperature $T = 300\ K$. The initial velocity is computed from the inlet Mach number, the pressure, the temperature and the ideal gas equation (with $\gamma = 1.4$). Here, $C_v = 717\ J/kg - K$. At the inlet, a subsonic stagnation boundary condition is used and the stagnation pressure and temperature are computed using Eq. (5.44).

The static pressure $P_s = 101,325\ Pa$ is set at the subsonic outlet. The results are shown in Fig. 5.30, Fig. 5.31, Fig. 5.32 and Fig. 5.33 for the inlet Mach numbers $M_\infty = 0.7$, $M_\infty = 0.01$, $M_\infty = 10^{-4}$ and $M_\infty = 10^{-7}$, respectively. It is expected that, within the low Mach number range, the solution does not depend on the Mach number and is identical to the solution obtained with an incompressible flow code. On the other hand, for a flow at $M = 0.7$, the compressible effects become more important and shock can form. An uniform grid of 3352 Q_1 elements was used to obtain the numerical solution for Mach numbers below $M_\infty = 0.01$. A once-refined mesh was employed for the $M_\infty = 0.7$ simulation in order to better resolve the shock. A CFL of 20 was employed and the simulations were run until steady state.

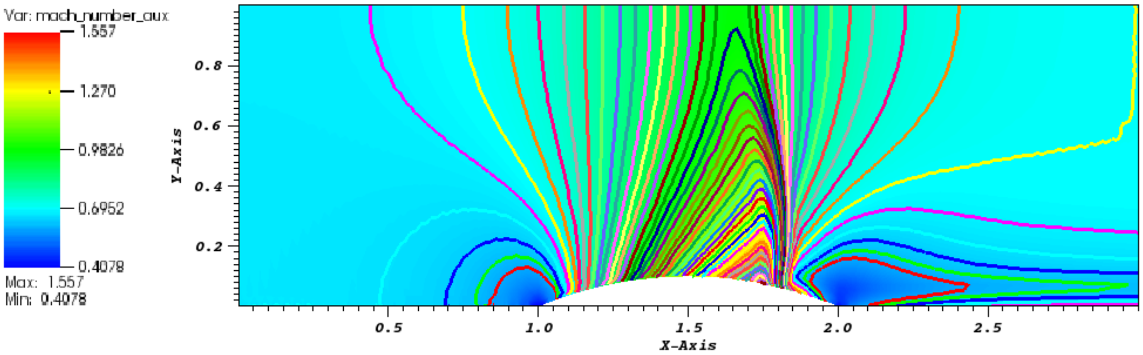


Figure 5.30: Mach 0.7

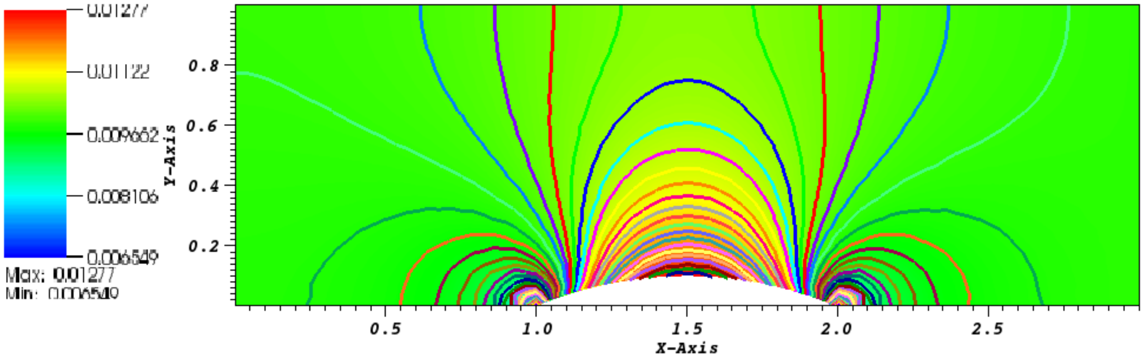


Figure 5.31: Mach 10^{-2}

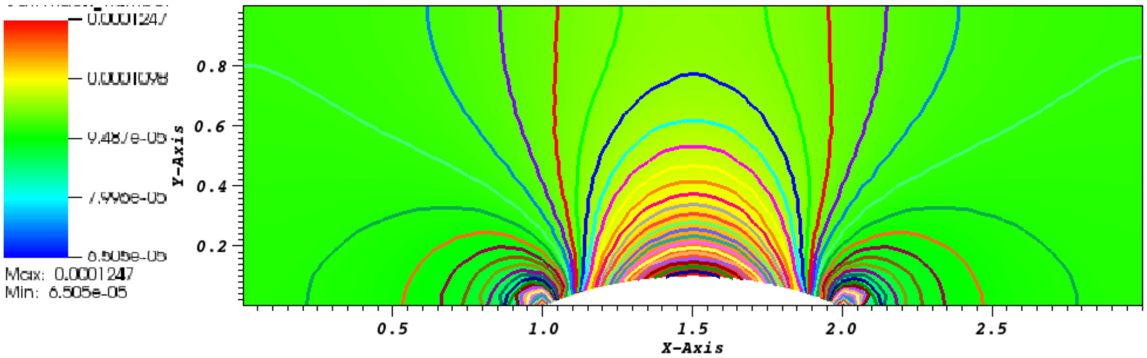


Figure 5.32: Mach 10^{-5}

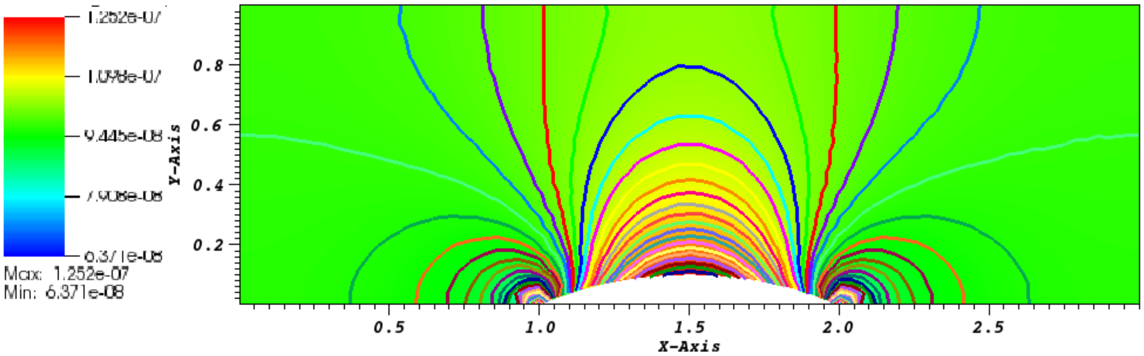


Figure 5.33: Mach 10^{-7}

The results showed in Fig. 5.31, Fig. 5.32 and Fig. 5.33 correspond to the low-Mach regime. The iso-Mach lines are drawn ranging from the minimum and the maximum values (provided in each legend) using 50 equally-spaced intervals. The steady-state solution is symmetric and does not depend on the value of the inlet Mach number, as expected in the incompressible limit.

In Fig. 5.30, the steady-state numerical solution develops a shock: the compressibility effect are no longer negligible. The iso-Mach lines are also plotted with 50 intervals and range from 0.4 to 1.6. The shock is well resolved and does not display any instabilities or spurious oscillations.
The results presented from Fig. 5.31 through Fig. 5.33 were obtained with the new definitions of the viscosity coefficients and illustrate the ability of the entropy viscosity method to correctly simulate several types of flows (subsonic and transonic flows) without tuning parameters.

5.10 Conclusions

A new version of the entropy viscosity method valid for a wide range of Mach number and applied to the multi-D Euler equations with variable area was derived and presented. The definition of the viscosity coefficient is now consistent with the low Mach asymptotic limit, does not require an analytical expression of the entropy function, and thus, could be used with any equation of state having a convex entropy. Tests were performed with the Ideal and Stiffened Gas equation of states. In 1-D, convergence of the numerical solution (either smooth or with shocks) to the exact solution was demonstrated by computing the convergence rates of the L1 and L2 norms of the error for flows in convergence-divergent nozzle and a straight pipe. 2-D simulations were also performed for both subsonic and supersonic flows, and various geometries: the entropy viscosity method behaves well for a wide range of Mach number. The numerical results obtained for a flow over a circular bump (subsonic and transonic flows) illustrates the capabilities of the method to adapt to the flow type.

As future work, the entropy viscosity method will be extended to the 1-D seven equations model [49]. This two-phase flow system of equations is a good candidate for two reasons: it is unconditionally hyperbolic and degenerates to the multi-D Euler equations when one phase disappears.

6. APPLICATION OF THE ENTROPY VISCOSITY METHOD TO THE SEVEN EQUATIONS MODEL.

6.1 Description of the seven equation model:

Because it is not economical to solve the entire two-phase flow field with highly resolved three-dimensional computational fluid dynamics for an entire light water reactor coolant system, it is necessary to construct a one-dimensional model for flow in pipes, nozzles, and other components. The one-dimensional model is constructed to allow the representation of continuously variable cross-sectional area.

Consider flow through a duct with local cross-sectional area $A = A(x,t)$. Actually, most of the time we consider local cross-sectional area to depend upon position coordinate x only, for which a time rate of change of cross-sectional area is not necessary because for this case $\frac{\partial A}{\partial t} = 0$. However, $A(x,t)$ is left inside the time derivative terms for generality and possible future use. The seven-equation two-phase system model can be stated as balances of mass, momentum, and total energy, along with volume fraction evolution as

$$\frac{\partial \left(\alpha\rho\right)_{liq} A}{\partial t} + \frac{\partial \left(\alpha\rho u\right)_{liq} A}{\partial x} = -\Gamma A_{int} A \tag{6.1}$$

$$\begin{aligned}\frac{\partial \left(\alpha\rho u\right)_{liq} A}{\partial t} + \frac{\partial \alpha_{liq} A \left(\rho u^2 + p\right)_{liq}}{\partial x} &= p_{int} A \frac{\partial \alpha_{liq}}{\partial x} + p_{liq}\alpha_{liq}\frac{\partial A}{\partial x} \\ &+ A\lambda(u_{vap} - u_{liq}) \\ &- \Gamma A_{int} u_{int} A \\ &- F_{\text{wall friction},liq} - F_{\text{friction},vap} \\ &+ \left(\alpha\rho\right)_{liq} A \vec{g}\cdot\vec{n}_{axis}\end{aligned} \tag{6.2}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{liq}A}{\partial t}+\frac{\partial\alpha_{liq}u_{liq}A\left(\rho E+p\right)_{liq}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{liq}}{\partial x}-\bar{p}_{int}A\mu(p_{liq}-p_{vap})\\&+\bar{u}_{int}A\lambda(u_{vap}-u_{liq})\\&+\Gamma A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{liq,int}\right)A\\&+Q_{int,liq}+Q_{\text{wall},liq}\qquad(6.3)\\\frac{\partial\alpha_{liq}A}{\partial t}+u_{int}A\frac{\partial\alpha_{liq}}{\partial x}&=A\mu(p_{liq}-p_{vap})-\frac{\Gamma A_{int}A}{\rho_{int}}\qquad(6.4)\end{aligned}$$

for the liquid phase, and

$$\begin{aligned}\frac{\partial\left(\alpha\rho\right)_{vap}A}{\partial t}+\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial x}&=\Gamma A_{int}A\qquad(6.5)\\\frac{\partial\left(\alpha\rho u\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}A\left(\rho u^2+p\right)_{vap}}{\partial x}&=p_{int}A\frac{\partial\alpha_{vap}}{\partial x}+p_{vap}\alpha_{vap}\frac{\partial A}{\partial x}\\&+A\lambda(u_{liq}-u_{vap})\\&+\Gamma A_{int}u_{int}A\\&-F_{\text{wall friction},vap}-F_{\text{friction},liq}\\&+\left(\alpha\rho\right)_{vap}A\vec{g}\cdot\vec{n}_{axis}\qquad(6.6)\end{aligned}$$

$$\begin{aligned}\frac{\partial\left(\alpha\rho E\right)_{vap}A}{\partial t}+\frac{\partial\alpha_{vap}u_{vap}A\left(\rho E+p\right)_{vap}}{\partial x}&=p_{int}u_{int}A\frac{\partial\alpha_{vap}}{\partial x}-\bar{p}_{int}A\mu(p_{vap}-p_{liq})\\&+\bar{u}_{int}A\lambda(u_{liq}-u_{vap})\\&-\Gamma A_{int}\left(\frac{p_{int}}{\rho_{int}}-H_{vap,int}\right)A\\&+Q_{int,vap}+Q_{\text{wall},vap}\qquad(6.7)\\\frac{\partial\alpha_{vap}A}{\partial t}+u_{int}A\frac{\partial\alpha_{vap}}{\partial x}&=A\mu(p_{vap}-p_{liq})+\frac{\Gamma A_{int}A}{\rho_{int}}\qquad(6.8)\end{aligned}$$

for the vapor phase. It is noted that for two-phase flow, either of the differential relations (6.4) or (6.8) may be replaced with the algebraic relation

$$\alpha_{vap} = 1 - \alpha_{liq} \tag{6.9}$$

throughout, reducing the total number of equations to be solved to seven.

In equations (6.1)–(6.8), Γ is the net mass transfer per unit interfacial area from the liquid to the vapor phase and A_{int} is the interfacial area per unit volume of mixture. Also, $H_{liq,int}$ and $H_{vap,int}$ are the liquid and gas total enthalpies at the interface, respectively. The nomenclature has also been modified so that now u_{int} and $\bar{u}_{int}$ are, respectively, the interfacial velocity and average interfacial velocity; and p_{int} and $\bar{p}_{int}$ are, respectively, the interfacial pressure and average interfacial pressure. In the momentum balance equations $\vec{n}_{axis}$ is the unit vector directly along the axis of the duct, which is also the ± flow direction. Of course $F_{\text{wall friction},k}$ is the frictional force due to the wall acting on phase k and $F_{\text{friction},k'}$ is the frictional force acting on phase k due to the presence of the other phase k'. Similarly, $Q_{int,k}$ is the direct heat transfer from the interface to phase k and $Q_{\text{wall},k}$ is the direct heat transfer from the wall to phase k.

Equation system (6.1)–(6.8) is the basic system solved with RELAP-7. The system was implemented within the MOOSE computational framework following a series of logically-complete steps [7] designed to confidently allow physically- and mathematically-meaningful benchmark testing at each step of increased complexity. This 7-equation two-phase model allows both phases to be compressible.

6.1.1 Seven-Equation Two-Phase Flow Constitutive Models

Without additional closure equations the balance relations derived above are generic, i.e. they apply to all materials (fluids). They must made to apply to the unique material (fluid) being considered – material specific. Also, though averaging the microlevel balance equations led to the "simplified" or perhaps more tractable model above, this simplification (averaging) led to a loss of information, and some additional relations must also be specified to supply (or restore) at least some information that was lost in this process[1]. Collectively, any additional relations, or sub-models, that must be specified to render mathematical closure (allowing a solution to be obtainable) to the generic balance equations are known as "constitutive relations".

Because the 7-equation two-phase model's most unique features are reflected in the presence of a volume fraction evolution equation, interfacial pressure and velocity, and mechanical relaxation terms involving pressure and velocity relaxation, it is natural to begin with their constitutive relations. Constitutive ideas associated with the volume fraction evolution equation were discussed previously for pedagogical reasons. Thermodynamical relaxation will be discussed subsequently, followed by other closures.

6.1.1.1 Interface Pressure and Velocity, Mechanical Relaxation Coefficients

In the continuous limit of small mesh spacing and time steps along with employment of the Godunov weak wave limit, the finite closure relations converge [8, 10] to

$$p_{int} = \bar{p}_{int} + \frac{Z_{liq}Z_{vap}}{Z_{liq}+Z_{vap}}\mathrm{sgn}\left(\frac{\partial \alpha_{liq}}{\partial x}\right)(u_{vap}-u_{liq}) \tag{6.10}$$

$$\bar{p}_{int} = \frac{Z_{vap}p_{liq}+Z_{liq}p_{vap}}{Z_{liq}+Z_{vap}} \tag{6.11}$$

[1] The process of averaging the balance equations produced a system with more unknowns than equations; thus postulates or empirical correlations are required to resolve this deficiency.

$$u_{int} = \bar{u}_{int} + \mathrm{sgn}\left(\frac{\partial \alpha_{liq}}{\partial x}\right)\frac{p_{vap} - p_{liq}}{Z_{liq} + Z_{vap}} \tag{6.12}$$

$$\bar{u}_{int} = \frac{Z_{liq}u_{liq} + Z_{vap}u_{vap}}{Z_{liq} + Z_{vap}} \tag{6.13}$$

$$\lambda = \frac{1}{2}\mu Z_{liq} Z_{vap} \tag{6.14}$$

$$\mu = \frac{A_{int}}{Z_{liq} + Z_{vap}} \tag{6.15}$$

where λ is the velocity relaxation coefficient function, μ is the pressure relaxation coefficient function, $Z_k = \rho_k c_k$, $(k = liq, vap)$, is the phasic acoustic impedance and A_{int} is the specific interfacial area (i.e. the interfacial surface area per unit volume of two-phase mixture) which must be specified from some type of flow regime map or function. The DEM model for two-phase flow of water and its vapor in a one dimensional duct of spatially varying cross-section was derived and demonstrated with these closures by Berry et al. [49].

Remark (1): From this specification of λ and μ it is clear that special coupling is rendered. To relax the 7-equation model to the ill-posed classical 6-equation model, the pressures should be relaxed toward a single pressure for both phases. This is accomplished by specifying the pressure relaxation coefficient to be very large, i.e. letting it approach infinity. But if the pressure relaxation coefficient goes to infinity, so does the velocity relaxation rate also approach infinity. This then relaxes the 7-equation model not to the classical 6-equation model, but to the mechanical equilibrium 5-equation model of Kapila. This reduced 5-equation model is also hyperbolic and well-posed. The 5-equation model provides a very useful starting point for constructing multi-dimensional interface resolving methods which dynamically captures evolving, and even spontaneously generating, interfaces [53]. Thus the 7-

equation model of RELAP-7 can be relaxed locally to couple seamlessly with such a multi-dimensional, interface resolving code.

Remark (2): Numerically, the mechanical relaxation coefficients μ (pressure) and λ (velocity) can be relaxed independently to yield solutions to useful, reduced models (as explained previously). It is noted, however, that relaxation of pressure only by making μ large without relaxing velocity will indeed give ill-posed and unstable numerical solutions, just as the classical 6-equation two-phase model does, with sufficiently fine spatial resolution, as confirmed in [49, 25].

Remark (3): Even though the implementation of the 7-equation two-phase model within RELAP-7 (or any other code for that matter) does not use the generalized approach of DEM, the interfacial pressure and velocity closures as well as the pressure and velocity relaxation coefficients of Equations (6.10) to (6.15) are utilized.

6.1.1.2 Wall and Interphase Friction

A simple wall friction model results from making the same assumptions as for single-phase duct flow with the exception that the duct wall area over which the shear stress acts is reduced by the fraction of the wall area which the phase occupies. Thus

$$F_{\text{wall friction},k} = \frac{f_k}{2d_h}\rho_k u_k \left|u_k\right| \alpha_k A \tag{6.16}$$

for phases $k = (liq, vap)$, where f_k is the wall friction factor associated with phase k. The hydraulic diameter d_h depends on the shape of the cross section, and the position x in the pipe.

The friction force acting between the two phases due to their relative motion is also given in analogy to that of single-phase duct flow:

$$F_{\text{friction},k'} = f_{k,k'}\frac{1}{2}\rho_k(u_k - u_{int})\left|u_k - u_{int}\right| A_{int} A \tag{6.17}$$

for $k = (liq, vap)$, $k' = (vap, liq)$, with $f_{k,k'}$ denoting the friction factor acting upon phase k due to the (relative) motion of the other phase k'.

The frictional pressure drop in each phase will be different in general due the different velocities of the two phases. However, because of the tendency toward pressure equilibrium between the phases an effective pressure drop will be realized.

6.1.1.3 Wall and Interface Direct Heat Transfer

Without wall boiling, a simple model for the direct, convective heat transfer from the wall to fluid phase k will be the same as that of a single-phase except the duct wall area over which this heat transfer can occur is weighted by the wetted fraction of the phase. That is,

$$Q_{\text{wall},k} = H_{w,k} a_w \left(T_k - T_{\text{wall}}\right) \alpha_k A \tag{6.18}$$

for phase $k = (liq, vap)$, where $H_{w,k}$ is the wall convective wall heat transfer coefficient associated with phase k. Similarly, the direct heat transfer from/to the interface to/from the phase k, which will also be used to determine the mass transfer between the phases, is

$$Q_{int,k} = h_{T,k} \left(T_{int} - T_k\right) A_{int} A \tag{6.19}$$

with $h_{T,k}$ denoting the convective heat transfer coefficient between the interface and phase k. The phasic bulk temperature T_k is determined from the respective phase's equation of state.

6.1.1.4 Interphase Mass Transfer

For a vapor to be formed from the liquid phase (vaporization) energy must be added to the liquid to produce vapor at nucleation sites; whether the liquid is heated directly or decompressed below its saturation pressure. A liquid to vapor phase change may occur based on two main mechanisms. The first is related to vaporization induced by external heating or heat transfer in a nearly constant pressure environment which is called heterogeneous boiling, or simply boiling. This heat input can occur through a solid/liquid interface with the solid typically hotter than the liquid, or through a liquid/gas interface with the gas being hotter than the liquid.

To examine the mass flow rate between phases, local mechanisms of the vaporization (condensation) process are considered between the liquid phase and its associated vapor in the presence of temperature gradients. The mechanisms of interest here are dominated by heat diffusion at the interface. The pertinent local equations to consider are the mass and energy equations. As a vaporization front propagates slowly (on the order of 1 mm/s to 1 m/s) compared to acoustic waves present in the medium (which propagate with speeds of the order 1 km/s), acoustic propagation results in quasi-isobaric pressure evolution through vaporization fronts. The momentum equation is therefore not needed – because the quasi-isobaric assumption (neglecting the pressure and kinetic energy variations in the total energy equation) is made. A simple expression for the interphase mass flow rate is obtained

$$\begin{aligned}\Gamma = \Gamma_{vap} &= \frac{h_{T,liq}\left(T_{liq}-T_{int}\right)+h_{T,vap}\left(T_{vap}-T_{int}\right)}{h_{vap,int}-h_{liq,int}} \\ &= \frac{h_{T,liq}\left(T_{liq}-T_{int}\right)+h_{T,vap}\left(T_{vap}-T_{int}\right)}{L_v\left(T_{int}\right)}\end{aligned} \tag{6.20}$$

where $L_v\left(T_{int}\right) = h_{vap,int} - h_{liq,int}$ represents the latent heat of vaporization. The interface temperature is determined by the saturation constraint $T_{int} = T_{sat}(p)$ with the appropriate pressure $p = \bar{p}_{int}$ determined above, the interphase mass flow rate is thus determined. The lower graphic of Figure 6.1, schematically shows the p-T state space in the vicinity of the saturation line (shown for the case with $T_{liq} < T_{vap}$).

To better illustrate the model for vaporization or condensation, Figure 6.2 shows

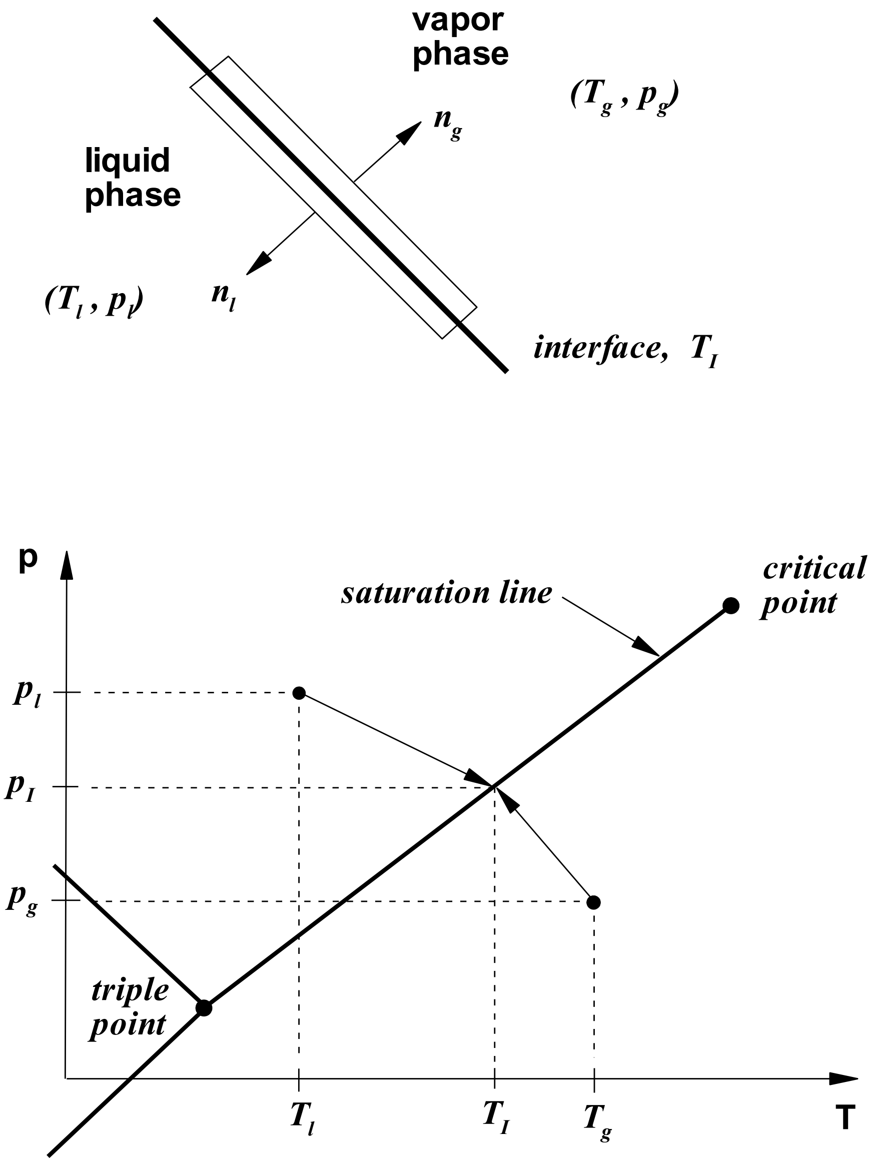
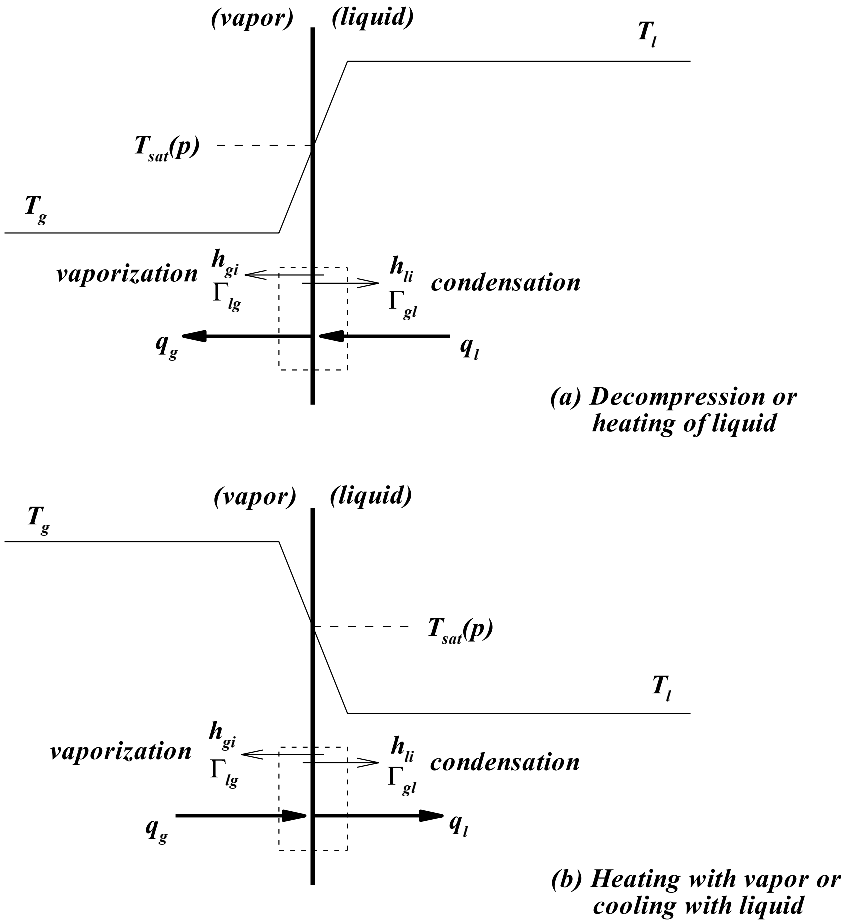


Figure 6.1: Interface control volume (top); T-p state space around saturation line, $T_{liq} < T_{vap}$, (bottom).



(a) Decompression or heating of liquid

(b) Heating with vapor or cooling with liquid

Figure 6.2: Vaporization and condensation at a liquid-vapor interface (after Moody [44]).

pure liquid and pure vapor regions separated by an interface. Representative temperature profiles are shown for heat transfer from vapor to liquid or liquid to vapor. As discussed by Moody [44], either vaporization or condensation can occur for both temperature profiles. The interphase mass transfer is determined by the net interfacial heat transfer: if net heat transfer is toward the interface, vapor will form; conversely, if net heat transfer is away from the interface, liquid will condense. Figure 6.2 shows heat transfer rates q_{vap} and q_{liq} from the vapor and liquid sides of the interface. For bidirectional phase change (vaporization and condensation), mass transfer based on heat balance at the interface is adopted. When vaporization occurs, vapor is assumed to form at a saturated interface temperature $T_{int} = T_{sat}(\bar{p}_{int})$. If condensation occurs, liquid is assumed to form also at a saturated interface temperature $T_{int} = T_{sat}(\bar{p}_{int})$. The interfacial total enthalpies correspond to the saturated values in order that the interphase mass transfer rate and conservation of total energy be compatible:

$$H_{k,int} = h_{k,int} + \frac{1}{2}u_{int}^2 \tag{6.21}$$

for phase $k = (liq, vap)$, where $h_{k,int}$ is the phase k specific enthalpy evaluated at the interface condition. Phasic specific enthalpy depends upon the equation of state used and will be discussed with the equations of state. The interfacial density corresponds to the liquid saturated density $\rho_{int} = \rho_{liq,sat}(p_{int})$.

6.1.1.5 Stiffened Gas Equation of State for Two-phase Flows

With the 7-equation two-phase model each phase is compressible and behaves with its own convex equation of state (EOS). For initial development purposes it was decided to use a simple form capable of capturing the essential physics. For this purpose the stiffened gas equation of state (SGEOS) [38] was selected (as it was also for single phase)

$$p(\rho, e) = (\gamma - 1)\rho(e - q) - \gamma p_\infty \tag{6.22}$$

where p, ρ, e, and q are the pressure, density, internal energy, and the binding energy of the fluid considered. The parameters γ, q, and p_∞ are the constants (coefficients) of each fluid. The first term on the right hand side is a repulsive effect that is present for any state (gas, liquid, or solid), and is due to molecular vibrations. The second term on the right represents the attractive molecular effect that guarantees the cohesion of matter in the liquid or solid phases. The parameters used in this SGEOS are determined by using a reference curve, usually in the $\left(p, \frac{1}{\rho}\right)$ plane.

To extend this equation of state for two phases, LeMetayer [38] uses the saturation curves as this reference curve to determine the stiffened gas parameters for liquid and vapor phases. The SGEOS is the simplest prototype that contains the main physical properties of pure fluids, repulsive and attractive molecular effects, thereby facilitating the handling of the essential physics and thermodynamics with a simple analytical formulation. Thus each fluid has its own thermodynamics. For each phase the thermodynamic state is determined by the SGEOS:

$$e(p,\rho) = \frac{p+\gamma p_\infty}{(\gamma-1)\rho} + q \tag{6.23}$$

$$\rho(p,T) = \frac{p+p_\infty}{(\gamma-1)c_v T} \tag{6.24}$$

$$h(T) = \gamma c_v T + q \tag{6.25}$$

$$g(p,T) = (\gamma c_v - q')\,T - c_v T \ln \frac{T^\gamma}{(p+p_\infty)^{\gamma-1}} + q \tag{6.26}$$

where T, h, and g are the temperature, enthalpy, and Gibbs free enthalpy, respectively, of the phase considered. In addition to the three material constants mentioned above, two additional material constants have been introduced, the constant volume specific heat c_v and the parameter q'. The method to determine these parameters in liquid-vapor systems, and in particular the coupling of liquid and vapor parameters, is given in [38]. The values for water and its vapor from that reference are given in Table 2. These parameter values appear to yield reasonable approximations over a temperature range from 298 to 473K. For higher temperature range the parameters can easily be refit.

Unlike van der Waals type modeling where mass transfer is a thermodynamic path, with the 7-equation two-phase model the mass transfer modeling, which produces a relaxation toward thermodynamic equilibrium, is achieved by a kinetic process. Thus the 7-equation model preserves hyperbolicity during mass transfer. From equation (6.25) it is readily seen that the phase k specific enthalpy evaluated at the interface condition from equation (6.21) is

$$h_{k,int} = c_{p,k} T_{int} + q_k \tag{6.27}$$

because $c_{p,k} = \gamma_k c_{v,k}$.

The bulk interphase mass transfer from the liquid phase to the vapor phase Γ is due to their difference in Gibb's free energy. At saturated conditions the Gibb's energies of the two-phases are equal. It is necessary to determine the saturation temperature $T_{sat}(p)$ for given pressure $p = \bar{p}_{int}$ and the heat of vaporization $L_v\left(T_{sat}(\bar{p}_{int})\right)$ at this saturation temperature with the SGEOS for each phase. For this calculation the procedure of [38] is adopted. This procedure for the determination of SGEOS parameters can be made very accurate provided the two reference states are picked sufficiently close to represent the experimental saturation curves as locally quasi-linear. Restrictions occur near the critical point, but away from this point wide ranges of temperatures and pressures can be considered. At thermodynamic equilibrium at the interface, the two phasic Gibbs free enthalpies must be equal, $g_{vap} = g_{liq}$, so the use of equation (6.26) yields

$$\ln\left(p + p_{\infty,vap}\right) = A + \frac{B}{T} + C\ln(T) + D\ln\left(p + p_{\infty,liq}\right) \tag{6.28}$$

where

$$A = \frac{c_{p,liq} - c_{p,vap} + q'_{vap} - q'_{liq}}{c_{p,vap} - c_{v,vap}} \tag{6.29}$$

$$B = \frac{q_{liq} - q_{vap}}{c_{p,vap} - c_{v,vap}} \tag{6.30}$$

$$C = \frac{c_{p,vap} - c_{p,liq}}{c_{p,vap} - c_{v,vap}} \tag{6.31}$$

$$D = \frac{c_{p,liq} - c_{v,liq}}{c_{p,vap} - c_{v,vap}} \ . \tag{6.32}$$

Relation (6.28) is nonlinear, but can used to compute the theoretical curve $T_{sat}(p)$. A simple Newton iterative numerical procedure is used. With $T_{sat}(p)$ determined, the heat of vaporization is calculated as

$$\begin{aligned} L_v\left(T_{int}\right) &= h_{vap,int} - h_{liq,int} \\ &= h_{k,int} \\ &= \left(\gamma_{vap} c_{v,vap} T + q_{vap}\right) - \left(\gamma_{liq} c_{v,liq} T + q_{liq}\right) \ . \end{aligned} \tag{6.33}$$

6.2 Derivation of the dissipative terms:

6.3 The viscosity coefficients:

6.4 Numerical results:

7. APPLICATION OF THE ENTROPY VISCOSITY METHOD TO THE 1-D GREY RADIATION-HYDRODYNAMIC EQUATIONS.

7.1 Introduction

Solving the radiation hydrodynamic equations is a challenging task for multiple reasons. First, the characteristic time scales between the radiation and hydrodynamics are different by several orders of magnitude which often requires the radiation part to be solved implicitly to ensure stability. Second, as with any wave-dominated problems, high resolution schemes are needed to accurately resolve shocks. Third, achieving high-order accuracy is challenging but some recent developments provided high-order accuracy results both in time and space when discretizing either the Euler equations [26, 28, 29, 27] or the radiation equation independently from each other.

Significant effort has been put into developing Riemann solvers for both the radiation and hydrodynamic equations. Balsara [6] developed a Riemann solver for the radiation-hydrodynamic equations by considering the frozen approximation that decouples the two physics components. However, such an approach may be questionable in the equilibrium diffusion limit. In this case, the coupling terms drive the physics and have to be accounted for. A *generalized Riemann solver* that accounts exactly for the relaxation terms was developed in [6]. Another approach assumes the strong equilibrium diffusion limit in which radiation diffusion is negligible and the radiation simply advects at the material velocity [17]. In this limit, the radiation hydrodynamics equation can be expressed in the form of the Euler equations with a radiation-modified equation of state (REOS) . Any solution technique for the Euler equations may be applied to these equations. Thus, one may develop approximate Riemann solvers for these equations and applied them in a general context.

Edwards and al. [30] proposed a two-stage semi-implicit IMEX scheme to solve the coupled radiation-hydrodynamic equations. They applied a Trapezoidal/BDF2 temporal discretization scheme to the nonlinear grey radiation diffusion. The radiation and hydrodynamic equations are solved implicitly and explicitly, respectively. A Riemann solver along with a flux limiter is used to resolve shocks and other waves. Their results show good agreement with semi-analytical solutions.

In this article we propose to solve the 1-D radiation-hydrodynamics equations by using *the entropy viscosity method.* This technique, developed by Guermond et al. for hyperbolic systems of equations [28, 29], consists in adding appropriate dissipative terms to the governing equations. The viscosity coefficient of these terms is modulated by the local entropy production. These dissipative terms are devised to stabilize the numerical scheme and to remove the non-physical oscillations appearing at the shock locations. Generally speaking, entropy is produced at shocks [55]. Thus, by setting the viscosity coefficient proportional to the entropy production, shocks can be detected and tracked and an adequate amount of viscosity is added locally to stabilize the numerical scheme. The entropy production is computed on the fly, by analyzing the entropy residual. This residual is strongly peaked in shocks and small elsewhere. The entropy viscosity method was shown to achieve high-order accuracy away from the shock regions, was successfully applied to non-linear hyperbolic equations using various discretization methods (finite volume, continuous and discontinuous finite elements, spectral method) and yielded high-order accuracy on non-uniform meshes and complex geometries [29, 57]. Because of the similarity between Euler equations and the radiation hydrodynamic equations, it is conjectured that the entropy viscosity method may be a good candidate for resolving shocks occurring in radiation-hydrodynamic phenomena.

The 1-D grey radiation-hydrodynamic (GRH) equations are recalled in Eq. (7.1):

$$\begin{cases} \partial_t\left(\rho\right)+\partial_x\left(\rho u\right)=0 \\ \partial_t\left(\rho u\right)+\partial_x\left(\rho u^2+P+\frac{\epsilon}{3}\right)=0 \\ \partial_t\left(\rho E\right)+\partial_x\left[u\left(\rho E+P\right)\right]=-\frac{u}{3}\partial_x\epsilon-\sigma_a c\left(aT^4-\epsilon\right) \\ \partial_t\epsilon+\frac{4}{3}\partial_x\left(u\epsilon\right)=\frac{u}{3}\partial_x\epsilon+\partial_x\left(\frac{c}{3\sigma_t}\partial_x\epsilon\right)+\sigma_a c\left(aT^4-\epsilon\right) \end{cases}, \tag{7.1}$$

where ρ, u, E, ϵ, P and T are the material density, material velocity, material specific total energy, radiation energy density, material pressure and temperature, respectively. The total and absorption cross sections, σ_t and σ_a, are either constant or density- and temperature-dependent. The variables a and c are the Boltzman constant and the speed of light, respectively. Lastly, the symbols ∂_t and ∂_x denote the temporal and spatial partial derivatives, respectively. The material temperature and pressure are computed with the Ideal Gas equation of state (IGEOS):

$$\begin{cases} P=(\gamma-1)C_v\rho T \\ e=C_vT \end{cases}, \tag{7.2}$$

where e is the specific internal energy and is obtained from the expression $e = E-0.5u^2$. The heat capacity C_v and the heat ratio coefficient γ are assumed constant.

The objective of this paper is to extend the entropy-based viscosity method to the 1-D grey radiation-hydrodynamic equations. The approach followed in this paper is similar to those of [6, 50]: an infinite opacity is assumed and the relaxation terms are ignored in order to make Eq. (7.1) hyperbolic. Then, an entropy equation is derived and used to obtain the functional forms of the viscous stabilization terms. Definitions for the viscosity coefficients are provided.

This paper is organized as follows. In Section 7.2, the entropy viscosity method is extended to the grey radiation-hydrodynamic equations; details regarding the derivation of the adequate dissipative terms and definitions for the new viscosity coefficients are provided. Spatial and temporal discretization schemes are discussed in Section 7.3 along with the solution algorithm employed to solve the discretized equations. Numerical results are presented in Section 7.4 where the second-order accuracy of the scheme is demonstrated in both the equilibrium diffusion and streaming limits, using the method of manufactured solutions applied to the GRH equations. Then, several numerical test cases, taken from the published literature, are provided; in these simulations, the Mach number varies from 1.05 to 50 [48]. Conclusions are presented in Section 7.5.

7.2 The entropy-based viscosity method applied to the 1-D Radiation-Hydrodynamic equations

In this section, we extend the entropy viscosity method [28, 29, 57] to the 1-D radiation-hydrodynamic equations in a staged process. First, the reader is guided through the main steps that lead to the derivation of the dissipative terms, using the entropy minimum principle [54]. Then, a definition for the entropy viscosity coefficient based upon the entropy production is given.

We recall that the entropy viscosity method was developed for hyperbolic system of equations. However, the radiation hydrodynamic equations are not strictly hyperbolic but several numerical techniques are based on the study of their hyperbolic parts [6, 50]. Thus, following the same rationale, the system of equations given in Eq. (7.1) is made hyperbolic by assuming an infinite opacity (the frozen approximation) and by ignoring the relaxation terms. These two assumptions yield the following system of equations:

$$\begin{cases} \partial_t \left(\rho\right) + \partial_x \left(\rho u\right) = 0 \\ \partial_t \left(\rho u\right) + \partial_x \left(\rho u^2 + P + \frac{\epsilon}{3}\right) = 0 \\ \partial_t \left(\rho E\right) + \partial_x \left[u \left(\rho E + P\right)\right] = -\frac{u}{3}\partial_x \epsilon \\ \partial_t \epsilon + \frac{4}{3}\partial_x \left(u\epsilon\right) = \frac{u}{3}\partial_x \epsilon \end{cases} . \tag{7.3}$$

The jacobian matrix of the hyperbolic terms can be computed to derive the eigenvalues:

$$\lambda_1 = u - c_m,\ \lambda_{2,3} = u \text{ and } \lambda_4 = u + c_m, \tag{7.4}$$

where c_m is the radiation-modified material speed of sound and is defined as follows:

$$c_m^2 = \underbrace{P_\rho + \frac{P}{\rho^2}P_e}_{c^2_{Euler}} + \frac{4\epsilon}{9\rho} \tag{7.5}$$

with P_x the standard shorthand notation for $\partial_x P$, and c^2_{Euler} denotes the definition of the speed of sound when considering only the 1-D Euler equations. The above hyperbolic system of equations can be recast in a conservative form. This allows us to assume the existence of an entropy function s [36] that depends upon the internal energy e, the density ρ, and the radiation energy density ϵ. Following some algebra given in Appendix D, an equation satisfied by the entropy s is obtained:

$$\rho \frac{ds}{dt} = \rho \left(\partial_t s + u \partial_x s\right) = 0, \tag{7.6}$$

where $\frac{d\cdot}{dt}$ denotes the total or material derivative. Eq. (7.6) is often referred as the entropy residual and is used to prove the entropy minimum principle, $\frac{ds}{dt} \geq 0$, [54].

When adding dissipative terms to each equation of Eq. (7.3) as required in the entropy viscosity method, the entropy residual equation is modified and some additional terms will appear in the right-hand side of Eq. (7.6). The sign of these extra terms needs to be studied for the entropy minimum principle to hold. As such, the entropy minimum principle is invoked to guide in the derivation of appropriate expressions for each of the dissipative terms. Obtaining the final expression of the dissipative terms is a lengthy process and only the final result along with the key assumptions are stated here. The reader is referred to Appendix D for the details of the derivation. The system of equations with the dissipative terms is as follows:

$$\begin{cases} \partial_t \left(\rho\right) + \partial_x \left(\rho u\right) = \partial_x \left(\kappa \partial_x \rho\right) \\ \partial_t \left(\rho u\right) + \partial_x \left(\rho u^2 + P + \frac{\epsilon}{3}\right) = \partial_x \left(\kappa \partial_x \rho u\right) \\ \partial_t \left(\rho E\right) + \partial_x \left[u \left(\rho E + P\right)\right] + \frac{u}{3}\partial_x \epsilon = \partial_x \left(\kappa \partial_x (\rho E)\right) \\ \partial_t \epsilon + \frac{4}{3}\partial_x \left(u\epsilon\right) - \frac{u}{3}\partial_x \epsilon = \partial_x \left(\kappa \partial_x \epsilon\right) \end{cases}, \tag{7.7}$$

where κ is a locally defined positive viscosity coefficient. It was assumed that the following conditions hold:

$$\begin{cases} P\frac{\partial s}{\partial e} + \rho^2 \frac{\partial s}{\partial \rho} + \frac{4}{3}\rho\epsilon\frac{\partial s}{\partial \epsilon} = 0 \\ s(\rho, e, \epsilon) = \hat{s}(\rho, e) + \frac{\rho_0}{\rho}\tilde{s}(\epsilon) \end{cases} \tag{7.8}$$

where $-\tilde{s}$ is convex with respect to the radiation energy density ϵ and $-\hat{s}$ is convex with respect to the internal energy e and the specific volume ρ^{-1}. The constant ρ_0 is of order one and appears only for dimensionality purposes.

Once the dissipative terms are obtained, it remains to define the local viscosity coefficient $\kappa(x,t)$. We require the following to hold in the prescription for κ:

- Since the entropy residual is a measure of the entropy production that occurs in shock regions, it is natural to define a viscosity coefficient proportional to the entropy residual. This will enable shock detection and tracking and will also provide a measure of the viscosity required to stabilize the scheme. This viscosity coefficient is referred to as the *entropy viscosity coefficient* or *second-order viscosity coefficient* and is denoted by $\kappa_e(x,t)$.

- An upper bound on κ is to be set since entropy production can be very large in shocks. For explicit time integration, the maximum value of the viscosity coefficient is related to the Courant-Friedrichs-Lewy number (CFL). The upper bound on κ is defined by analogy to the standard upwind (Godunov) scheme that is known to efficiently smooth out oscillations (but is only first-order accurate). With implicit temporal integrators, the same reasoning is used even if the CFL number may not need to be strictly respected. This upper bound will be referred to as the *first-order viscosity*, denoted by $\kappa_{max}(x,t)$.

- The viscosity coefficient κ that is actually used in the dissipative terms of Eq. (7.3) is defined as follows: $\kappa(x,t) = \min(\kappa_e(x,t), \kappa_{max}(x,t))$. With such a definition, the viscosity added to the system of equations will saturate to the first order viscosity in the shock regions. Elsewhere, the entropy production and thus the viscosity coefficient κ are expected to be small.

Next, we define the local first- and second-order viscosity coefficients $\kappa_{max}(x,t)$ and $\kappa_e(x,t)$, respectively. Following the work of Zingan et al. [57], the first-order viscosity definition is based on the local largest eigenvalue that is known to be $|u|+c_m$ in 1-D:

$$\kappa_{max} = \frac{h}{2}\left(|u| + c_m\right) \tag{7.9}$$

where h is the local grid size. This definition is derived based on the upwind scheme and a simple derivation can be found in [28] in the case of a scalar hyperbolic equation. Through the definition of the radiation-modified speed of sound c_m, both the material and radiation properties are accounted for in the definition of the first-order viscosity coefficient.

The definition of the second order viscosity coefficient $\kappa_e(x,t)$ is based upon the entropy residual (Eq. (7.6)) recast as a function of pressure P, density ρ and radiation energy density ϵ:

$$\tilde{D}_e(x,t) = \frac{s_e}{P_e} \underbrace{\left(\frac{dP}{dt} - c^2_{Euler} \frac{d\rho}{dt} \right)}_{\hat{D}_e(x,t)} \tag{7.10}$$

The term s_e is the inverse of the material temperature (Appendix D) and P_e is computed from the IGEOS. These two terms are positive so that the sign of the entropy residual $\tilde{D}_e(x,t)$ can be determined by simply inspecting the terms inside the parentheses, denoted by $\hat{D}_e(x,t)$. Such an expression is easier to compute than the one given in Eq. (7.6) which required an analytical expression for the entropy function. In addition to the entropy residual, inter-element jumps in the pressure and density gradients, J, are also accounted for. The objective is to be able to also detect discontinuities that are not shocks, such as contact waves (there is no entropy production in a contact wave), in order to stabilize them as well.

Thus, the entropy viscosity coefficient $\kappa_e(x,t)$ is set to be proportional to $\hat{D}_e(x,t)$ and J with the following form:

$$\kappa_e(x,t) = h^2 \frac{\max(|\hat{D}_e(x,t)|, J)}{n_P} \tag{7.11}$$

where $J = \max_i(J(x_i,t))$, and $J(x_i,t)$ is the jump of a given quantity at cell interface x_i, and n_P is a normalization function (of the same units as pressure) that has to be chosen so that the viscosity coefficient κ has units of m^2/s. The following definition for the normalization function has been chosen: $n_P = \rho c_m^2$. Thus, the final definition for the viscosity coefficient κ is the following:

$$\kappa_e(x,t) = h^2 \frac{\max(|\hat{D}_e(x,t)|, J)}{\rho c_m^2} \tag{7.12}$$

The jump J in the definition of $\kappa(x,t)$ is piecewise-constant. Its definition is discretization-dependent and defined as follows for Continuous Galerkin FEM:

$$\begin{cases} J_P(x_i,t) = |u|[[\partial_x P]] \\ J_\rho(x_i,t) = c_m^2 |u|[[\partial_x \rho]] \\ J(x_i,t) = \max(J_\rho(x_i,t), J_P(x_i,t)) \end{cases} \tag{7.13}$$

The symbol $[[\cdot]]$ denotes the jump at the cell interface.

The entropy viscosity method is now well defined for the hyperbolic system given in Eq. (7.3) and will be used to solve for the grey radiation-hydrodynamic equations given in Eq. (7.1). However, one may question how the relaxation source terms, $\sigma_a c(aT^4 - \epsilon)$ and the physical diffusion term, $\partial_x(D\partial_x\epsilon)$, may affect the entropy viscosity method. When applying the entropy viscosity method, the radiation energy density equation will now contain a diffusive term and a numerical dissipative term with a vanishing viscosity coefficient κ. As long as the diffusive coefficient $D = \frac{c}{3\sigma_t}$ is larger than the viscosity coefficient κ, the numerical dissipative term should not be required. A way to ensure consistency and prevent the formation of oscillations in the frozen limit is to merge the two second-order derivative terms into one as follows:

$$\partial_x \left(\frac{c}{3\sigma_t} \partial_x \epsilon \right) + \partial_x \left(\kappa \partial_x \epsilon \right) \Longrightarrow \partial_x \left[\max\left(\frac{c}{3\sigma_t},\, \kappa \right) \partial_x \epsilon \right] \tag{7.14}$$

Thus, as long as the artificial viscosity coefficient κ is locally smaller that the physical diffusive coefficient $D = \frac{c}{3\sigma_t}$, no artificial viscosity is required to ensure stability of the numerical scheme. As the diffusive coefficient D goes to zero, shocks can form in the radiation energy density profile and will require a certain amount of viscosity in order to prevent oscillations from appearing.

The effect of the relaxation source terms onto the entropy viscosity method can become problematic in the equilibrium diffusion limit ($\sigma_a c \rightarrow \infty$): the relaxation source terms behave as dissipative terms and make the system parabolic [27]. In [31], a study on the impact of various artificial viscosity methods onto hyperbolic systems with relaxation terms was carried out. It was shown that high-order viscosity coefficients are more suitable since they do not alter the physical solution as much as first-order viscosity terms (upwind scheme). A manufactured solution is employed in Section 7.4.1 to test the convergence of the numerical solution in the equilibrium-diffusion limit. The normalization factor has to be larger than h in order to conserve high-order accuracy.

The reader will notice that, except for the definition of the jumps, the whole method is independent of the spatial discretization employed. The technique could be used with discontinuous Galerkin finite element or finite volume methods. In both cases, an adequate definition of the jump terms can be found in [57].

7.3 Numerical scheme and solution technique

The 1-D radiation-hydrodynamic equations Eq. (7.1) are discretized with the continuous Galerkin finite element method (CGFEM) under the MOOSE framework [16]. To obtain a weak form, the following generic form of Eq. (7.1) is considered:

$$\partial_t U + \partial_x F\left(U\right) = S + \partial_x H\left(U\right) \tag{7.15}$$

where U is the solution vector, F is a conservative vector flux, S is a vector containing the relaxation source terms and non-conservative terms, and H is the artificial viscosity dissipative flux:

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho E \\ \epsilon \end{bmatrix}, \ F(U) = \begin{bmatrix} \rho u \\ \rho u^2 + P + \frac{\epsilon}{3} \\ u\left(\rho E + P\right) \\ \frac{4}{3}u\epsilon \end{bmatrix}, \ S = \begin{bmatrix} 0 \\ 0 \\ -\frac{u}{3}\partial_x\epsilon - \sigma_a c\left(aT^4 - \epsilon\right) \\ \frac{u}{3}\partial_x\epsilon + \partial_x\left(\frac{c}{3\sigma_t}\partial_x\epsilon\right) + \sigma_a c\left(aT^4 - \epsilon\right) \end{bmatrix},$$

$$\text{and } H(U) = \begin{bmatrix} \kappa\partial_x\rho \\ \kappa\partial_x(\rho u) \\ \kappa\partial_x\left(\rho e\right) + \frac{u^2}{2}\kappa\partial_x\rho + \rho u\mu\partial_x u \end{bmatrix}$$

In order to apply the continuous finite element method, Eq. (7.15) is multiplied by a test function ϕ, integrated by parts over the discrete mesh Ω bounded by $\partial\Omega$, to obtain a weak solution:

$$\sum_e \int_e \partial_t U\phi - \sum_e \int_e F(U)\partial_x\phi + \int_{\partial\Omega} F(U)\mathbf{n}\phi - \sum_e \int_e S\phi + \sum_e \int_e H(U)\partial_x\phi - \int_{\partial\Omega} H\left(U\right)\mathbf{n}\phi = 0 \tag{7.16}$$

where e represents the cells of Ω and $\mathbf{n}$ is the outward normal vector to the boundary of the computational domain.

The integrals over the elements e are evaluated using third-order Gauss quadrature rules. The time-dependent term will be evaluated using the implicit second-order temporal integrator BDF2 [27]. Only linear test functions are considered in this paper. The integral on $\partial\Omega$ requires computation of the flux $F(U)$ and $H(U)$ on the boundary. It is chosen to treat the boundary for each physics component independently and details will be given in Section 7.4.2 on how to compute $F(U)$. The viscosity dissipative flux $H(U)$ is zero on the boundaries since the viscosity coefficients κ and μ are set to zero [28, 29, 57].

The Jacobian-Free Newton Krylov method [33] is employed to solve the system for each time step: Newton's and Krylov methods are used for the outer nonlinear and inner linear solves, respectively. The full Jacobian matrix is used as a preconditioner and is computed by finite difference: this is one of the options available in MOOSE and is reasonably efficient for 1-D simulations.

The entropy residual expression is not integrated over the cell volume as it is usually done in the Galerkin finite element method. The variable values and their gradients are available at quadrature points and at different times, and, thus, can be used to evaluate the entropy residual.

7.4 Numerical results

In this section, numerical results using the entropy viscosity method are presented for the dimensional 1-D grey radiation-hydrodynamic equations. First, second-order accuracy of the method is demonstrated using the method of manufactured solution (MMS). Then, results for some standard radiation-hydrodynamic test cases are given.

7.4.1 Space/time accuracy

The same manufactured solution as in [30] is used in order to test both the diffusive and streaming limit solutions in a slab of thickness $L = 2\pi\ cm$. The manufactured solutions are composed of trigonometric functions. Periodic boundary conditions are used for all of the variables. The L_2 norm of the error between the numerical and exact solutions are computed for density, momentum, total material energy, and radiation energy density. For each new simulation, the time step is divided by two and the number of spatial degrees of freedom is doubled. With such settings, the error is expected to decrease by a factor 4 if second-order convergence is achieved.

The first manufactured solution is designed to test the equilibrium-diffusion limit. In that case, the radiation energy is in equilibrium with the material temperature and the opacity is large which means that the radiation mean-free path is not resolved but the variation of the solution is resolved. The following exact solution was used:

$$\begin{cases} \rho = \sin(x-t) + 2 \\ u = \cos(x-t) + 2 \\ T = \frac{0.5\gamma(\cos(x-t)+2)}{\sin(x-t)+2} \\ \epsilon = aT^4 \end{cases} . \tag{7.17}$$

The cross sections σ_a and σ_t are assumed constant and set to the same value 1000 cm^{-1}. The simulation is run until $t = 3\ sh$ ($1\ sh = 10^{-8}\ sec$). The L$_2$ error norm along with its ratio between consecutive simulations are given in Table 7.1 for the equilibrium diffusion limit case.

Table 7.1: L_2 norms of the error for for the equilibrium diffusion limit case using a manufactured solution.

# of cells	time step size (sh)	ρ	**ratio**	$\rho\mathbf{E}$	**ratio**
20	10^{-1}	0.590766	NA	1.333774	NA
40	$5\ 10^{-1}$	0.290626	2.03	0.478819	2.79
80	$2.5\ 10^{-2}$	0.0959801	3.021	0.154119	3.11
160	$1.25\ 10^{-2}$	0.02593738	3.70	0.0405175	3.80
320	$6.25\ 10^{-3}$	$6.471444\ 10^{-3}$	4.00	$9.90446\ 10^{-3}$	4.09
640	$3.125\ 10^{-3}$	$1.584158\ 10^{-3}$	4.01	$2.44727\ 10^{-3}$	4.04

# of cells	time step size (sh)	ϵ	**ratio**	$\rho\mathbf{u}$	**ratio**
20	10^{-1}	0.00650085	NA	0.910998	NA
40	$5\ 10^{-1}$	0.00124983	5.20	0.4090946	2.23
80	$2.5\ 10^{-2}$	0.000262797	4.76	0.125943	3.25
160	$1.25\ 10^{-2}$	$6.17726\ 10^{-5}$	4.25	$3.381042\ 10^{-3}$	3.72
320	$6.25\ 10^{-3}$	$1.509184\ 10^{-5}$	4.09	$8.373657\ 10^{-3}$	4.04
640	$3.125\ 10^{-3}$	$3.72548\ 10^{-6}$	4.05	$2.070538\ 10^{-3}$	4.04

The second manufactured solution is used to test the method in the streaming limit: the radiation streaming dominates the absorption/re-emission term and evolves at a fast time scale. The exact solution used is as follows :

$$\begin{cases} \rho = \sin(x-t)+2 \\ u = (\sin(x-t)+2)^{-1} \\ T = 0.5\gamma \\ \epsilon = \sin(x-1000t)+2 \end{cases} \tag{7.18}$$

For this manufactured solution, the cross sections are still assumed constant and set to the same value 1 cm^{-1}. The final time is $t_{final} = 3\ sh$. Once again, the L_2 error norm is given in Table 7.2 for the density, momentum, material total energy and radiation energy density.

Table 7.2: L_2 norms of the error for for the streaming limit case using a manufactured solution.

# of cells	time step size (sh)	ρ	**ratio**	$\rho\mathbf{E}$	**ratio**
20	10^{-1}	$1.4373\ 10^{-2}$	NA	$5.88521\ 10^{-1}$	NA
40	$5.\ 10^{-2}$	$3.760208\ 10^{-3}$	3.82	$1.4244\ 10^{-1}$	4.13
80	$2.5\ 10^{-2}$	$9.91724\ 10^{-4}$	3.79	$3.2047\ 10^{-2}$	4.44
160	$1.25\ 10^{-2}$	$2.4455\ 10^{-4}$	4.06	$7.4886\ 10^{-3}$	4.28
320	$6.25\ 10^{-3}$	$6.280715\ 10^{-5}$	3.89	$1.82327\ 10^{-3}$	4.11
640	$3.125\ 10^{-3}$	$1.57920\ 10^{-5}$	3.98	$4.50463\ 10^{-4}$	4.05
1280	$1.5625\ 10^{-4}$	$3.96096\ 10^{-6}$	3.99	$1.12061\ 10^{-4}$	4.02

# of cells	time step size (sh)	ϵ	**ratio**	$\rho\mathbf{u}$	**ratio**
20	10^{-1}	$3.82001\ 10^{-1}$	NA	$2.354671\ 10^{-3}$	NA
40	$5.\ 10^{-2}$	$1.21500\ 10^{-1}$	3.14	$6.138814\ 10^{-4}$	3.84
80	$2.5\ 10^{-2}$	$3.27966\ 10^{-2}$	3.70	$1.74974\ 10^{-4}$	3.51
160	$1.25\ 10^{-2}$	$8.38153\ 10^{-3}$	3.91	$3.61297\ 10^{-5}$	4.84
320	$6.25\ 10^{-3}$	$2.10925\ 10^{-3}$	3.97	$9.03866\ 10^{-6}$	3.99
640	$3.125\ 10^{-3}$	$5.28472\ 10^{-4}$	3.99	$2.25649\ 10^{-6}$	4.01
1280	$1.5625\ 10^{-4}$	$1.322268\ 10^{-4}$	3.99	$5.69984\ 10^{-7}$	3.95

For both manufactured solutions the error is divided by four as the time step and the spatial mesh are reduced by a factor two. Thus, we conclude that GRH equations can be numerically solved with second-order accuracy using the entropy viscosity method when the exact solution is smooth.

7.4.2 Radiation shock simulations

The purpose of this section is to show that the entropy-based viscosity method (Section 7.2) can accurately resolve shocks occurring in radiation-hydrodynamic simulations. Multiple test cases are considered, with Mach numbers of 1.05, 1.2, 2, 5 and 50 [48]. All of the simulations are run with 500 spatial cells and with a Courant-Friedrichs-Lewy (CFL number) of 10 until steady-state (even if the scheme employed here is fully implicit, a CFL number can still be computed and is a good reference for comparison against semi-implicit or fully explicit codes). Linear Lagrange polynomials and the second-order temporal integrator BDF2 are once again used. For clarity, the initial conditions for each test case will be recalled in a table and plots of the density, $\rho(x)$, the radiation temperature, $\theta(x)$, and material temperature, $T(x)$, at steady-state will be given as well as those of the viscosity coefficients, $\kappa(x)$ and $\kappa_{max}(x)$. The computational domain consists of a 1-D slab of thickness L. The initial discontinuity between the left and right states is located at x_0 and will be specified for all test cases. For all of the test cases presented in this paper, the cross sections σ_a and σ_t are assumed constant and set to 853.144 cm^{-1} and 390.711 cm^{-1}, respectively, if not otherwise specified. The heat capacity at constant specific volume is set to $C_v = 0.12348\ jerks/(g - keV)$.

For the Mach 2 simulation, results will also be shown when employing only the first-order viscosity ($\kappa(x,t) = \kappa_{max}(x,t)$) in order to show the benefits of using a high-order viscosity coefficient.

The inlet and outlet boundary conditions (BCs) are given next. The Euler equations and radiation equation are considered independently since the latter one is parabolic. At the inlet, the flow is supersonic and, therefore, no physical information exits the system. Thus, Dirichlet boundary condition can be used. At the outlet, the flow become subsonic which requires a particular treatment. Following the work from [49], a static boundary condition is implemented. Only the back pressure is provided and the other variables are computed using the characteristic equations. For the radiation equation, vacuum boundary conditions are used at both inlet and outlet.

7.4.2.1 An equilibrium diffusion test

For this test, the inlet Mach number is set to 1.05. The radiation field and material are in equilibrium. The initial conditions are given in Table 7.3. The computational domain is of size $L = 0.08$ *cm* and the initial step is at $x_0 = 0.015$ *cm*. The numerical solutions at steady state are given in Fig. 7.1, Fig. 7.2 and Fig. 7.3.

Table 7.3: Initial conditions for Mach 1.05.

	left	right
ρ (g/cm^3)	1.	1.0749588
u (cm/sh)	0.1228902	0.1144127
T (keV)	0.1	0.1049454
ϵ $(jerks/cm^3)$	1.372 10^{-6}	1.6642117 10^{-6}

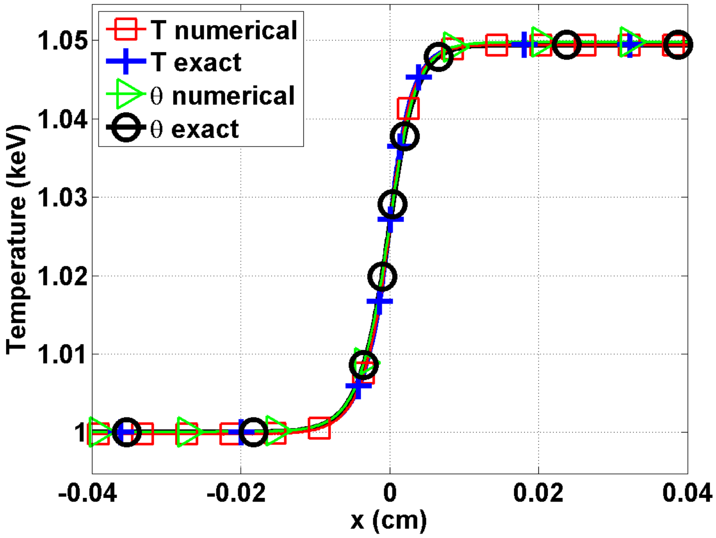


Figure 7.1: Material and radiation temperature profiles at steady state for Mach 1.05 test.

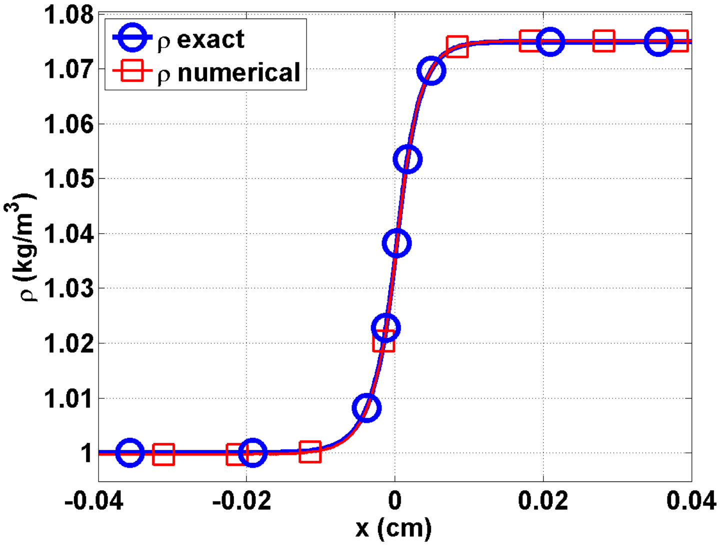


Figure 7.2: Material density profile at steady state for Mach 1.05 test.

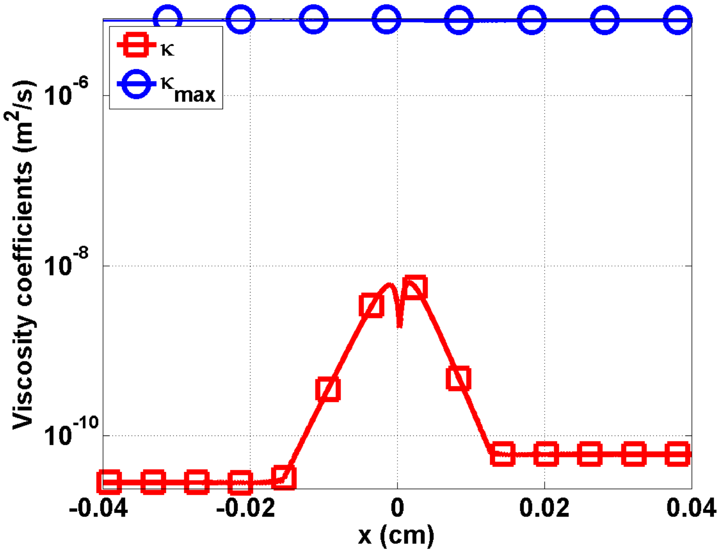


Figure 7.3: First-order viscosity κ_{max} and second-order viscosity κ profiles at steady state for Mach 1.05 test (logarithm scale).

The energy transfer between the material and radiation fields is not large enough to form a shock in the material. Thus, all of the material variables are smooth (Fig. 7.1 and Fig. 7.2) as well as the radiation temperature θ. Because of the smoothness of the solution, the viscosity coefficient κ is three order of magnitude smaller than the first-order viscosity coefficient κ_{max} (Fig. 7.3).

7.4.2.2 A 1.2 *Mach hydrodynamic shock*

In this test, the material experiences a shock and the radiation energy density remains smooth. The initial conditions, corresponding to a Mach number of 1.2 at the inlet, are as follows:

Table 7.4: Initial conditions for Mach 1.2.

	left	right
$\rho\ (g/cm^3)$	1.	1.0749588
$u\ (cm/sh)$	0.1405588	0.1083456
$T\ (keV)$	0.1	0.1194751
$\epsilon\ (jerks/cm^3)$	1.372 10^{-6}	2.7955320 10^{-6}

The slab thickness is set to $L = 0.045\ cm$ and the initial step was located at $x_0 = 0\ cm$.

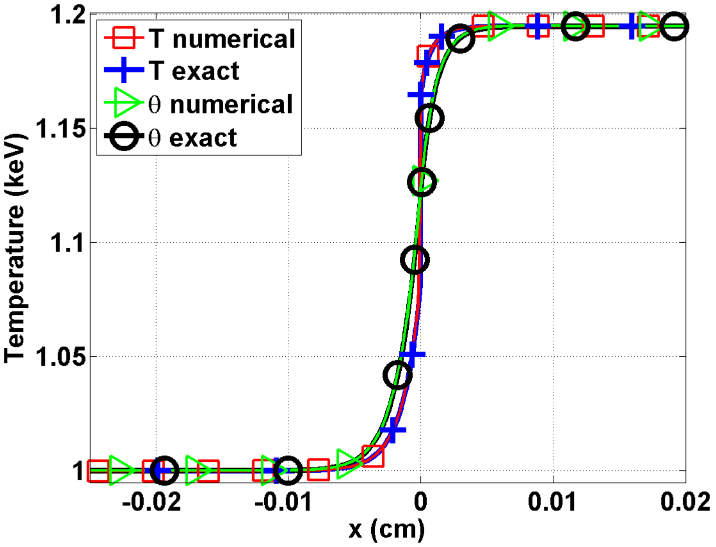


Figure 7.4: Material and radiation temperature profiles at steady state for Mach 1.2 test.

The radiation and material temperatures have two different behaviors (Fig. 7.4):

the later experiences an embedded hydrodynamic shock, whereas the radiation temperature is smooth because of the diffusion term. The material temperature profile does not show any pre- and post-shock oscillations. In Fig. 7.5, the material density profile has a shock as well. The viscosity coefficient (Fig. 7.6) is peaked in the shock as expected but does not saturate to the first-order viscosity. It is conjectured that the diffusion term in the radiation equation brings extra stability to the system. Overall, the numerical solution behaves as expected in the shock and the entropy-based viscosity method seems to efficiently stabilize the numerical scheme.

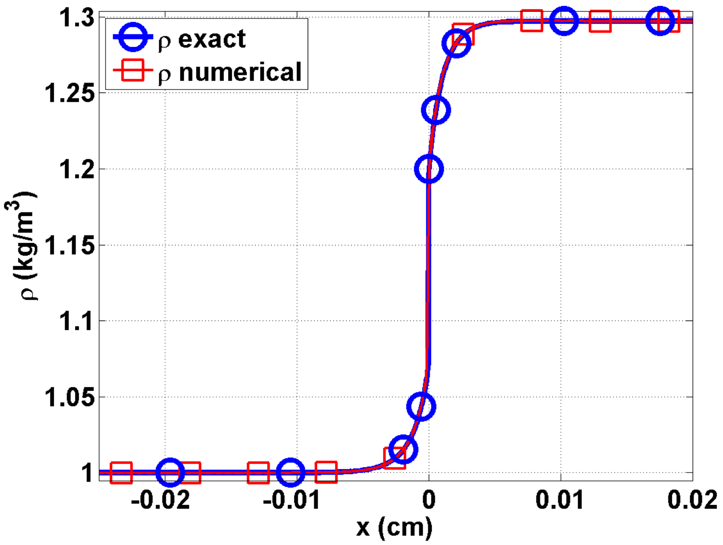


Figure 7.5: Material density profile at steady state for Mach 1.2 test.

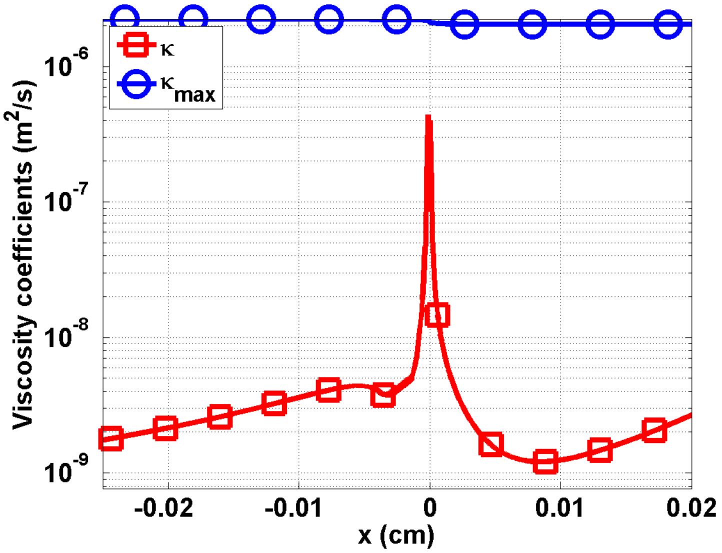


Figure 7.6: First-order viscosity κ_{max} and second-order viscosity κ profiles at steady state for Mach 1.2 test (logarithm scale).

7.4.2.3 A Mach 2 shock

The Mach 2 shock test has two features: a hydrodynamic shock and a Zeldovich spike, which make it interesting for testing the robustness of the entropy-based viscosity method. The initial conditions are specified in Table 7.5 for a slab of length $L = 0.04\ cm$ with $x_0 = 0.\ cm$.

Table 7.5: Initial conditions for Mach 2.

	left	right
ρ (g/cm^3)	1.	1.0749588
u (cm/sh)	0.1405588	0.1083456
T (keV)	0.1	0.1194751
ϵ $(jerks/cm^3)$	1.372 10^{-6}	2.7955320 10^{-6}

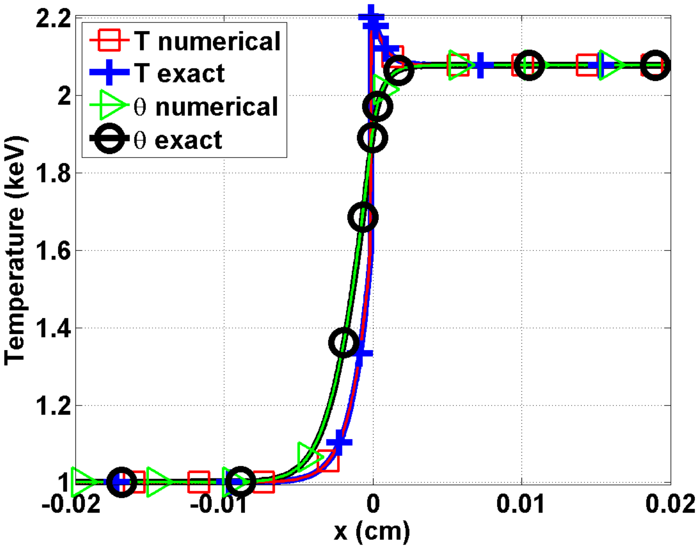


Figure 7.7: Material and radiation temperature profiles at steady-state for Mach 2 test.

Once again, the radiation temperature profile is smooth and the material temperature experiences an embedded hydrodynamic shock and a peak as shown in

Fig. 7.7. In Fig. 7.8, the shock is well resolved. The viscosity coefficient profile is given in Fig. 7.9 and is peaked, once again, in the shock region.

For comparison purpose, the same simulation was run with the first-order viscosity only, i.e., κ was set equal to κ_{max} for the whole domain in order to see the advantage of using a second-order viscosity coefficient. The results are given in Fig. 7.10 for the material density and temperature. Numerical solutions with first- and second-order viscosity coefficients are graphed. The radiation temperature profile (not shown here) is not affected much by the first-order viscosity and the curves are coincident. This is expected because of the way the artificial viscosity term is treated in the radiation equation (Section 7.2). However, on the same figure, the shock and peak in the material temperature profile are smoothed out: the shock is not as sharp and the peak amplitude is reduced because of the larger amount of viscosity added to the system. This test shows the benefits of using a high-order viscosity coefficient in order to avoid over-dissipation.

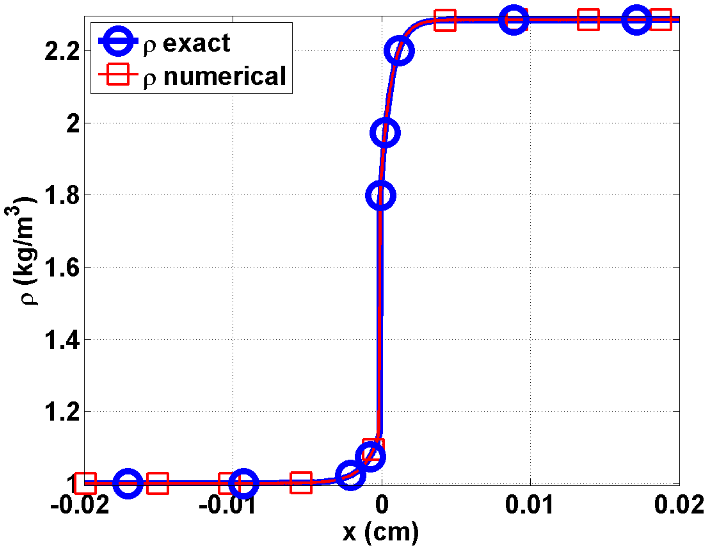


Figure 7.8: Material density profile at steady-state for Mach 2 test.

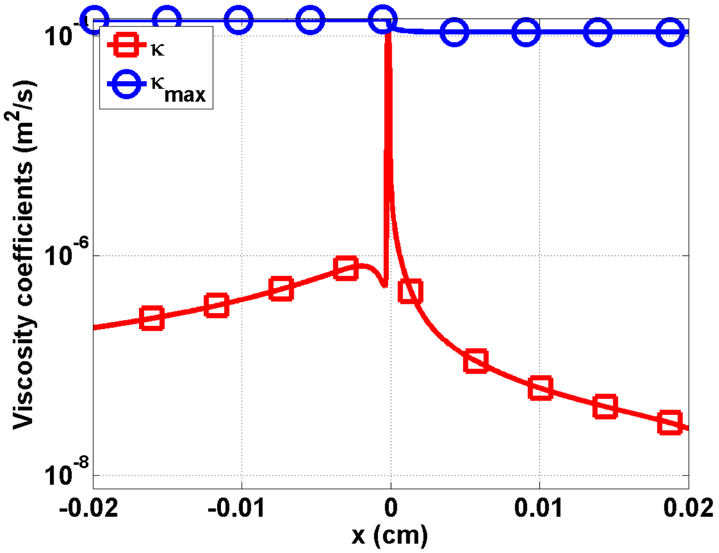


Figure 7.9: First-order viscosity κ_{max} and second-order viscosity κ profiles at steady state for Mach 2 test.

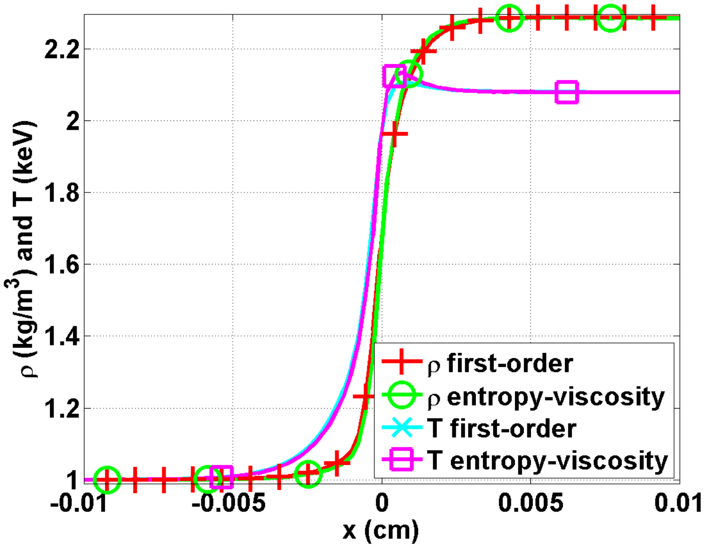


Figure 7.10: Comprarison between the material density and temperature profiles run with the high-order and first-order viscosity coefficients.

7.4.2.4 Mach 5 *shock*

A Mach 5 test is run with the initial conditions of Table 7.6 on a computational domain of length $L = 0.05\ cm$ ($x_0 = 0\ cm$). Steady-state results are shown in Fig. 7.11, Fig. 7.13, and Fig. 7.14 for the material and radiation temperatures, the density and the viscosity coefficients, respectively.

Table 7.6: Initial conditions for Mach 5.

	left	right
$\rho\ (g/cm^3)$	1.	1.0749588
$u\ (cm/sh)$	0.1405588	0.1083456
$T\ (keV)$	0.1	0.1194751
$\epsilon\ (jerks/cm^3)$	1.372 10^{-6}	2.7955320 10^{-6}

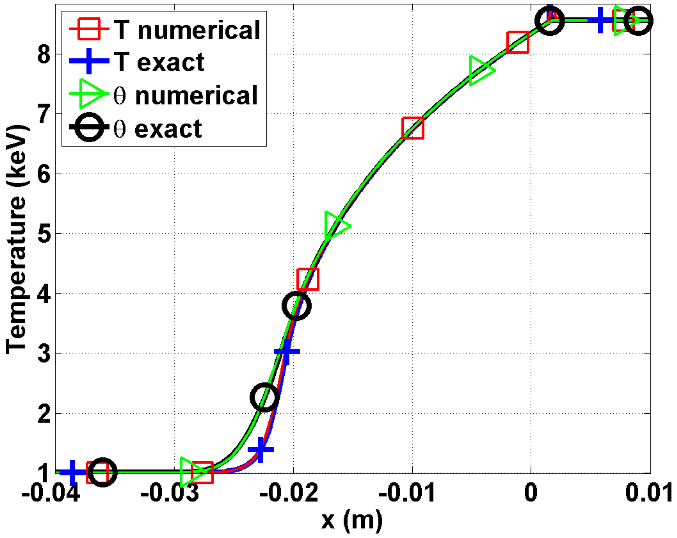


Figure 7.11: Material and radiation temperature profiles at steady state for Mach 5 test. Zoom at the location go the peak using different mesh resolutions.

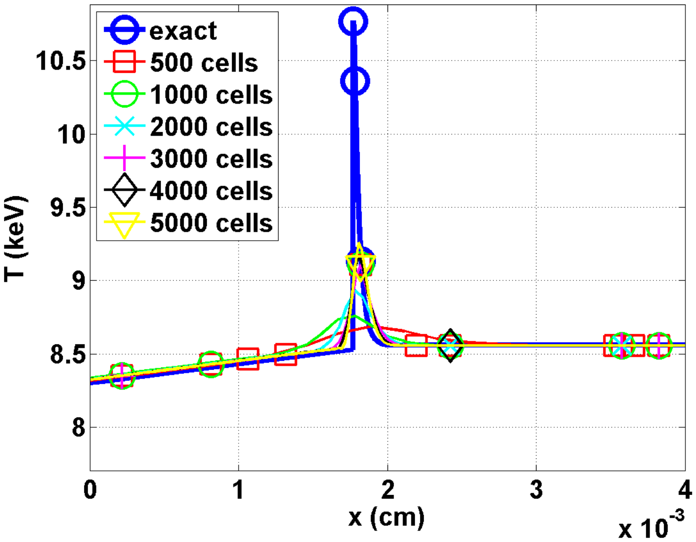


Figure 7.12: Material temperature profiles at steady state for the Mach 5 test in the neighborhood spike.

In Fig. 7.11, the radiation temperature profile is smooth. The material temperature no longer exhibits an embedded hydrodynamic shock but shows a Zeldovich spike. The mesh with 500 elements is not fine enough to correctly resolve the Zeldovich spike. In Fig. 7.12, the Zeldovich spike region is plotted for different mesh resolutions, using from 500 to 5000 elements: the peak is better resolved when using large numbers of elements and its position seems to be independent of the mesh size when appropriately refined. The density profile, Fig. 7.13, shows a shock located at the same position as the Zeldovich spike of the material temperature profile. The viscosity coefficient κ is also peaked in the shock region, as expected. The material and radiation variables do not present any numerical oscillations.

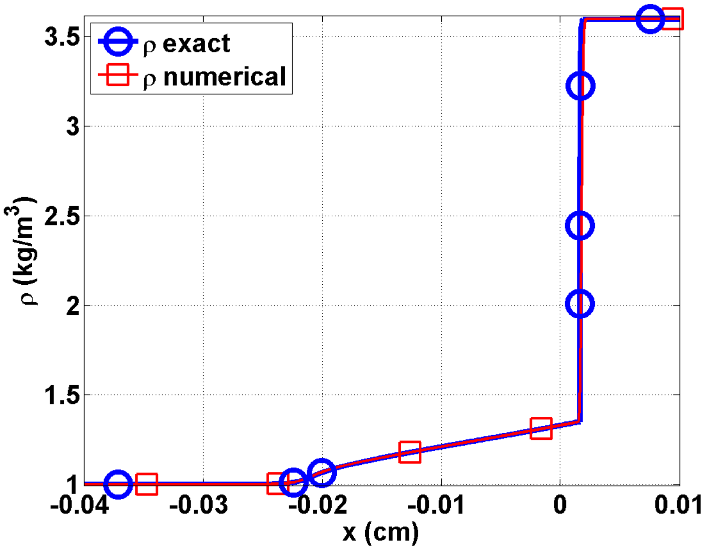


Figure 7.13: Material density profile at steady state for Mach 5 test.

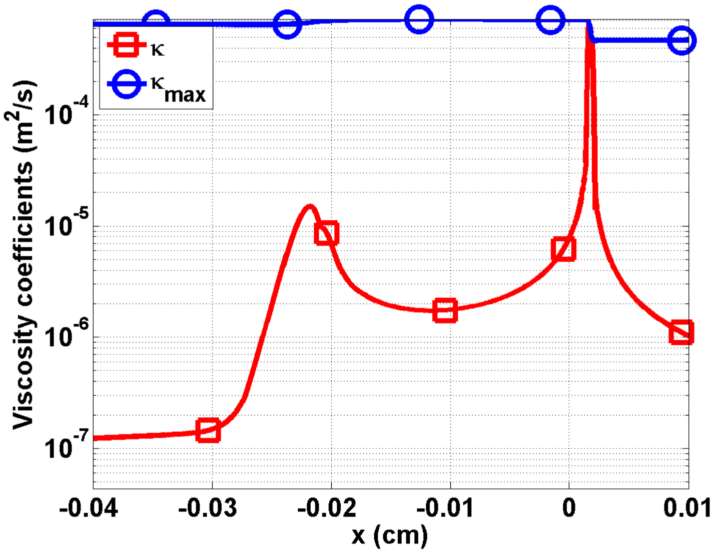


Figure 7.14: First-order viscosity κ_{max} and second-order viscosity κ profiles at steady state for Mach 5 test.

7.4.2.5 Mach 50 *shock*

The Mach 50 test is known to be challenging. The initial conditions are given in Table 7.7. The computational domain is of length $L = 0.2\ cm$. Results are once again given at steady state.

Table 7.7: Initial conditions for Mach 50.

	left	right
$\rho\ (g/cm^3)$	1.	6.5189217
$u\ (cm/sh)$	585.6620	89.84031
$T\ (keV)$	1.0	85.51552
$\epsilon\ (jerks/cm^3)$	$1.372\ 10^{-2}$	$7.33726\ 10^{5}$

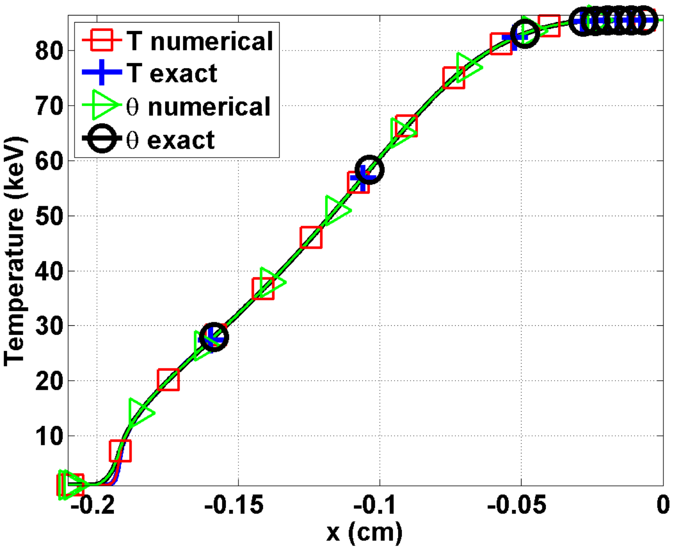


Figure 7.15: Material and radiation temperature profiles at steady state for Mach 50 test.

At Mach 50, there is no embedded hydrodynamic shock forming as shown in Fig. 7.15. The density profile is smooth as shown in Fig. 7.16. In Fig. 7.15, the material and radiation temperatures overlap on all of the computational domain except for a small region located between $x = -0.2$ and $x = -0.18$ cm. In this particular region, the viscosity coefficient saturates to the first-order viscosity (see Fig. 7.17) because of the inflection point in the material temperature profile. The artificial dissipative terms correctly stabilize the material temperature profile without altering the physical solution: the radiation temperature is expected to increase ahead of the material temperature.

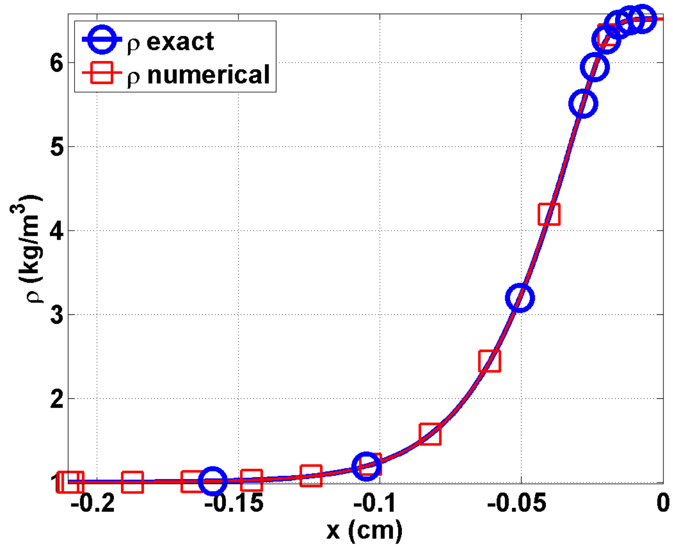


Figure 7.16: Material density profile at steady-state for Mach 50 test.

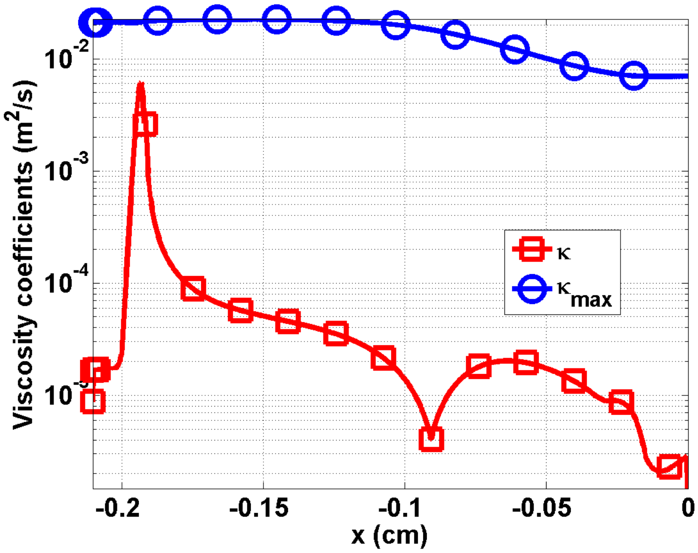


Figure 7.17: First-order viscosity κ_{max} and second-order viscosity κ profiles at steady-state for Mach 50 test.

7.5 Conclusions

In this paper, we have shown that the entropy-based viscosity method is a valid candidate for solving the 1-D radiation-hydrodynamic equations. A theoretical derivation is given for the derivation of the dissipative terms that are consistent with the entropy minimum principle. The viscosity coefficient κ is defined proportional to the entropy residual that measures the local entropy production allowing detection of shocks. Through the manufactured solution method, it is demonstrated, firstly, that second-order accuracy is achieved when the solution is smooth, and secondly, that the artificial dissipative terms do not affect the physical solution in the equilibrium-diffusion limit.

The entropy-based numerical scheme also behaves well in the tests performed for Mach numbers ranging from 1.05 to 50. The main features such as the embedded hydrodynamic shock and the Zeldovich spike are resolved accurately without spurious oscillations. The viscosity coefficient is peaked in the shock region only and behaves as expected. All of these results were obtained by using an unique definition of the viscosity coefficient that is computed on the fly. The addition of dissipative terms to the set of equations requires more computational work but is rather simple to implement.

As future work, extension to the multi-D equations could be considered: all of the derivations presented in this paper hold. The definition of the viscosity coefficients κ and κ_{max} do not need to be modified. It would be interesting to model the radiation equation with an S_n transport approximation and apply the entropy based artificial viscosity to the resultant radiation-hydrodynamics equations. Given the advective nature of the S_n equations, dissipation would need to be added to these equations.

fix spacing in bibliography, if any...

REFERENCES

[1] R Abgrall and R Saurel. Discrete equations for physical and numerical compressible multiphase mixtures. *Journal of Computational Physics*, 186:361–396, 2003.

[2] T. Alazard. A minicourse on the low mach number limit. *Discrete and Continuous Dynamical Systems - Series S*, 1:365–404, 2008.

[3] J Anderson. *Modern Compressible Flow: With Historical Perspective*. McGraw-Hill Science Engineering, 1982.

[4] U. M. Ascher and L. R. Petzold. *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. SIAM, Philadelphia, 1998.

[5] Perthane B. and Shu C-W. On positivity preserving finite volume schemes for euler equations. *Numer. Math.*, 73:119–130, 1996.

[6] Dinshaw S. Balsara. An analysis of the hyperbolic nature of the equations of radiation hydrodynamics. *J. Quant. Spectrosc. Radiat. Transfer*, 61:617–627, 1999.

[7] R. A. Berry. A logical progression of steps for implementation and testing of the 7-equation, two-phase model into a computational framework. In *Int. Conf. on Mathematics and Computational Methods Applied to Nuclear Science and Engineering*, LaGrange Park, IL, 2013. American Nuclear Society.

[8] R. A. Berry, R. Saurel, F. Petitpas, E. Daniel, O. LeMetayer, S. Gavrilyuk, N. Dovetta, and R. C. Martineau. Progress in the development of compressible, multiphase flow modeling capability for nuclear reactor flow applications. Technical Report INL/EXT-08-15002, Idaho National Laboratory, U. S. A., 2008.

[9] J. C. Butcher. *Numerical Methods for Ordinary Differential Equations.* John Wiley & Sons, New York, 2003.

[10] A. Chinnayya, E. Daniel, and R. Saurel. Modelling detonation waves in heterogeneous energetic materials. *J. of Computational Physics*, 196:490–538, 2004.

[11] I Christov and B Popov. New non-oscillatory central schemes on unstructured triangulations for hyperbolic systems of conservation laws. *J Comput Phys*, 227:5736–5757, 2008.

[12] B Cockburn, C Johnson, C Shu, and E Tadmor. Advanced numerical approximation of nonlinear hyperbolic equations. *Lecture Notes in Mathematics*, 1697:97109, 1998.

[13] B Cockburn, C Johnson, C Shu, and E Tadmor. Advanced numerical approximation of nonlinear hyperbolic equations. *Lecture Notes in Mathematics*, 1697:97109, 1998.

[14] B Cockburn, G Karniadakis, and C Shu. Discontinuous galerkin methods: theory, computation and applications. *Lecture Notes in Computer Science and Engineering*, 11:97109, 2000.

[15] B Cockburn and C Shu. The local discontinuous galerkin method for time-dependent convection-diffusion systems. *Numer. Anal.*, 35:2440–2463, 1998.

[16] G. Hansen D. Gaston, C. Newsman and D. Lebrun-Grandie. A parallel computational framework for coupled systems of nonlinear equations. *Nucl. Eng. Design*, 239:1768–1778, 2009.

[17] Woodward P. R. Dai W. Numerical simulations for radiation hydrodynamics. i. diffusion limit. *Journal of Comput Phys*, 142:182–207, 1998.

[18] D L Darmofal and K Siu. A robust multigrid algorithm for the euler equations with local preconditioning and semi-coarsening. *Journal of Computational Physics*, 151:728756, 1999.

[19] R A DeVore and G G Lorentz. *Constructive Approximation.* Springer-Verlag, 1991.

[20] J Donea and A Huerta. *Finite Element Methods for Flow Problems.* Oxford University Press, 2003.

[21] E. Godlewski and P.-A. Raviart. *Numerical approximations of hyperbolic systems of conservation laws.* Springer, New-York, 1996.

[22] J L Guermond and R Pasquetti. Entropy-based nonlinear viscosity for fourrier approximations of conservation laws. In *C.R. Math. Acad. Sci.*, volume 326, pages 801–806, Paris, 2008.

[23] J-L. Guermond and B. Popov. Viscous regularization of the euler equations and entropy principles. *under review*, 2012.

[24] H Guillard and C Viozat. On the behavior of upwind schemes in the low mach number limit. *Computers & Fluids*, 28:63–86, 1999.

[25] J. M. Herrard and O. Hurisse. A simple method to compute standard two-fluid models. *Int. J. of Computational Fluid Dynamics*, 19:475–482, 2005.

[26] Van Rosendale J Hussaini MY, van Leer B. Upwind and high-resolution schemes. *Berlin: Springer*, 12(1):1, 1997.

[27] Leveque R. J. *Numerical methods for conservation laws.* Basel: Birhauser, Reading, Massachusetts, 1990.

[28] B. Popov J-L. Guermond, R. Pasquetti. Entropy viscosity method for nonlinear conservation laws. *Journal Comput. Phys.*, 230:4248–4267, 2011.

[29] R. Pasquetti J-L. Guermond. Entropy viscosity method for high-order approximations of conservation laws. *Lecture Notes in Computational Science and Engineering*, 76:411–418, 2011.

[30] Robert B. Lowrie Jarrod D. Edwards, Jim E. Morel. Second-order discretization in space and time for radiation hydrodynamics. In *International Conference on Mathematics and Computational Methods Applied to Nuclear Science & Engineering (M&C 2013)*, Sun Valley, Idaho, USA, 2013.

[31] Shi Jin and C. David Levermore. Numerical schemes for hyperbolic conservation laws with stiff relaxation terms. *Journal of Computational Physics*, 126:449–467, 1996.

[32] D. A. Knoll and D. E. Keyes. Jacobian-free Newton-Krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193(2):357–397, January 2004.

[33] Keyes D. E. Knoll D. A. Jacobian-free newton-krylov methods: a survey of approaches and applications. *Journal of Computational Physics*, 193:357–397, 2004.

[34] J S Wong D L, Darmofal, and J Peraire. The solution of the compressible euler equations at low mach numbers using a stabilized finite element algorithm. *Comput. Methods Appl. Mech. Engrg.*, 190:5719–5737, 2001.

[35] A Lapidus. A detached shock calculation by second order finite differences. *J. Comput. Phys.*, 2:154–177, 1967.

[36] P. Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Comm. Pure Appl. Math.*, 7:159–193, 1954.

[37] S LeMartelot, B Nkonga, and R Saurel. Liquid and liquid-gas flows at all speeds: Reference solutions and numerical schemes. *Research report*, 7935:97109, 2012.

[38] O LeMetayer, J Massoni, and R Saurel. Elaborating equation of state for a liquid and its vapor for two-phase flow models. *International Journal of Thermal Science*, 43:265–276, 2004.

[39] X-S Li and C-W Gu. An all-speed roe-type scheme and its asymptotic analysis of low mach number behavior. *Journal of Computational Physics*, 227:5144–5159, 2008.

[40] R Lohner. *Applied CFD Techniques: an Introduction based on Finite Element Methods.* 2^{nd} Edition Wiley, 2003.

[41] R Lohner, K Morgan, and J Peraire. A simple extension to multidimensional problems of the artificial viscosity due to lapidus. *Commun. Numer. Methods Eng.*, 1(14):141–147, 1985.

[42] R Loubere. Validation test case suite for compressible hydrodynamics computation. *Theoritical Division T-7 Los Alamos National Laboratory.*

[43] Hittinger J. Lowrie R., Morel J. Coupling radiation and hydrodynamics. *Journal Quant. Spectrosc. Radiat. Transfer*, 61:432–500, 1999.

[44] F. J. Moody. *Introduction to Unsteady Thermofluid Mechanics.* Wiley, New Jersey, 1990.

[45] B Muller. Low-mach number asymptotes of the navier-stokes equations. *Journal of Engineering Mathematics*, 34:97109, 1998.

[46] P Perrot. *A to Z of Thermodynamics.* Oxford University Press, 1998.

[47] James J Quirk. A contribution to the great riemann solver debate. *NASA Contractor Report 191409*, ICASE Report No.:92–64, 1992.

[48] J. D. Edwards R. B. Lowrie. Radiative shock solutions with grey non equilibrium diffusion. *Journal of Comput Phys*, 18:129–143, 2008.

[49] O. LeMetayer R. Berry, R. Saurel. The discrete equation method (dem) for fully compressible, two-phase flows in ducts of spatially varying cross-section. *Nuclear Engineering and Design*, 240:3797–3818, 2010.

[50] J.E. Morel R.B. Lowrie. Issues with high-resolution godunov methods for radiation hydrodynamics. *Journal of Quantitative Spectroscopy & Radiative Transfer*, 69:475–489, 2001.

[51] J Reisner, J Serencsa, and Shkoller S. A space-time smooth artificial viscosity method for nonlinear conservation laws. *Journal of Computational Physics*, 253:912–933, 2013.

[52] Thomas W Roberts. The behavior of flux difference splitting schemes near slowly moving shock waves. *Journal of Computational Physics*, 90:141–160, 1990.

[53] R. Saurel, F. Petitpas, and R. A. Berry. Simple and efficient relaxation methods for interfaces separating compressible fluids, cavitating flows and shocks in multiphase mixtures. *J. of Computational Physics*, 228:1678–1712, 2009.

[54] E. Tadmor. A minimum entropy principle in the gas dynamics equations. *Appl. Numer. Math.*, 2:211–219, 1986.

[55] E.F. Toro. *Riemann Solvers and numerical methods for fluid dynamics.* 2^{nd} Edition, Springer, 1999.

[56] E Turkel. Preconditioned techniques in computational fluid dynamics. *Annu. Rev. Fluid Mech.*, 31:385–416, 1999.

[57] J. Morel V. Zingan, J-L. Guermond and B. Popov. Implementation of the entropy viscosity method with the discontinuous galerkin method. *Journal of Comput Phys*, 253:479–490, 2013.

[58] J von Neumann and R D Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *J Appl Phys*, 190:232–237, 1950.

APPENDIX A

DERIVATION OF THE DISSIPATIVE TERMS FOR THE EULER EQUATIONS WITH VARIABLE AREA USING THE ENTROPY MINIMUM PRINCIPLE

Euler equations (without viscous regularization) with variable area are recalled here

$$\partial_t (\rho A) + \vec{\nabla} \cdot (\rho \vec{u} A) = 0 \tag{A.1a}$$

$$\partial_t (\rho \vec{u} A) + \vec{\nabla} \cdot [A (\rho \vec{u} \otimes \vec{u} + P \mathbf{I})] = P \vec{\nabla} A \tag{A.1b}$$

$$\partial_t (\rho E A) + \vec{\nabla} \cdot [\vec{u} A (\rho E + P)] = 0 \, . \tag{A.1c}$$

The specific entropy is a function of the density ρ and the internal energy e, i.e., $s(e, \rho)$, the above system of equations satisfies the minimum entropy principle [55] is this really shown in Toro? I think it is and will double check tomorrow

$$A\rho \left(\partial_t s + \vec{u} \cdot \vec{\nabla} \cdot s \right) \geq 0 \, . \tag{A.2}$$

The entropy function s satisfies the second law of thermodynamics, $Tds = de - \frac{P}{\rho^2} d\rho$, which implies $s_e := T^{-1}$ and $s_\rho := -PT^{-1}\rho^{-2}$. One can show that [23]

$$s_e = T^{-1} \geq 0 \text{ and } P s_e + \rho^2 s_\rho \tag{A.3}$$

In order to apply the entropy viscosity method to the variable-area Euler equations, dissipative terms need to be added to each equation in Eq. (A.1). The functional forms of these terms need to be such that the entropy residual derived with these terms present also satisfies the minimum entropy principle. To prove the minimum entropy principle, the extra terms appearing in the entropy residual are either recast as conservative terms or shown to be positive. The rest of this appendix presents this demonstration. Following [23], we first write the variable-area equation with dissipative terms.

$$\partial_t\left(\rho A\right) + \vec{\nabla}\cdot\left(\rho\vec{u}A\right) = \vec{\nabla}\cdot f \tag{A.4a}$$

$$\partial_t\left(\rho\vec{u}A\right) + \vec{\nabla}\cdot\left[A\left(\rho\vec{u}\otimes\vec{u} + P\mathbf{I}\right)\right] = P\vec{\nabla}A + \vec{\nabla}\cdot g \tag{A.4b}$$

$$\partial_t\left(\rho E A\right) + \vec{\nabla}\cdot\left[\vec{u}A\left(\rho E + P\right)\right] = \vec{\nabla}\cdot\left(h + \vec{u}\cdot g\right). \tag{A.4c}$$

where f, g and h are dissipative fluxes to be determined. Starting from the modified system of equations given in Eq. (A.4), the entropy residual is derived again. The derivation requires the following steps : express the governing laws in terms of primitive variables $(\rho, \vec{u}, e)$, multiply the continuity equation by ρs_ρ and the internal energy equation by s_e, and invoke multivariate chain rule, e.g., $\partial s/\partial x = s_e\partial e/\partial x + s_\rho\partial\rho/\partial x$. These steps are similar to the ones form the standard Euler equations [23]. Some of the lengthy algebra is omitted here. The above steps yield:

$$A\rho\left(\partial_t s + \vec{u}\cdot\vec{\nabla}s\right) = s_e\left[\vec{\nabla}\cdot h + g : \vec{\nabla}u + \left(\frac{u^2}{2} - e\right)\vec{\nabla}\cdot f\right] + \rho s_\rho\vec{\nabla}\cdot f \tag{A.5}$$

The next step consists of choosing a definition for each of the dissipative terms so that the left hand-side is positive. The right hand-side of Eq. (A.5) can be simplified using the following relations, $g = A\mu\vec{\nabla}^s\vec{u} + f\otimes\vec{u}$ and $h = \tilde{h} - 0.5||\vec{u}||^2 f$, which yields:

$$A\rho\left(\partial_t s + \vec{u}\cdot\vec{\nabla}\cdot s\right) = s_e\left[\vec{\nabla}\cdot\tilde{h} - e\vec{\nabla}\cdot f\right] + \rho s_\rho\vec{\nabla}\cdot f + As_e\mu\vec{\nabla}\vec{u}^s : \vec{\nabla}\vec{u}$$

The right hand-side is now integrated by parts:

$$\begin{aligned} A\rho\left(\partial_t s + \vec{u}\cdot\vec{\nabla}\cdot s\right) &= \vec{\nabla}\cdot\left[s_e\tilde{h} - s_e e f + \rho s_\rho f\right] - \\ \vec{\nabla}\cdot\tilde{h}\vec{\nabla}s_e &+ f\cdot\vec{\nabla}(e s_e) - f\cdot\vec{\nabla}(\rho s_\rho) + A s_e \mu \vec{\nabla}^s\vec{u} : \vec{\nabla}\vec{u} \end{aligned}$$

where $\vec{\nabla}^s$ is the symmetric gradient. The term $A s_e\mu\vec{\nabla}^s\vec{u} : \vec{\nabla}\vec{u}$ is positive and thus, does not need any further modification. It remains to treat the other terms of the right hand-side that we now call rhs:

$$rhs = \vec{\nabla}\cdot\left[s_e\tilde{h} - s_e e f + \rho s_\rho f\right] - \tilde{h}\cdot\vec{\nabla}s_e + f\cdot\vec{\nabla}(e s_e) - f\cdot\vec{\nabla}(\rho s_\rho)$$

The first term of rhs is a conservative term. By choosing carefully a definition for $\tilde{h}$ and f, the conservative term can be expressed as a function of the entropy s. It is also required to include the variable area in the choice of the dissipative terms so that when assuming constant area, the regular multi-D Euler equations are recovered. The following definitions for $\tilde{h}$ and f are chosen:

$$\tilde{h} = A\kappa\vec{\nabla}(\rho e) \text{ and } f = A\kappa\vec{\nabla}\rho,$$

which yields, using the chain rule:

$$rhs = \vec{\nabla}\cdot(\rho A\kappa\vec{\nabla}s) - A\kappa\underbrace{\left[\vec{\nabla}(\rho e)\vec{\nabla}s_e - \vec{\nabla}\rho\vec{\nabla}(e s_e) + \vec{\nabla}\rho\vec{\nabla}(\rho s_\rho)\right]}_{\mathbf{Q}}$$

It remains to treat the term **Q** that can be recast under a quadratic form, following the work done in [23]:

$$\begin{aligned}
\mathbf{Q} &= X^t \Sigma X \\
\text{with } X &= \begin{bmatrix} \vec{\nabla}\rho \\ \vec{\nabla} e \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} \partial_\rho(\rho^2 \partial_\rho s) & \partial_{\rho,e} s \\ \partial_{\rho,e} s & \partial_{e,e} s \end{bmatrix}
\end{aligned}$$

The matrix Σ is symmetric and identical to the matrix obtained in [23]. The sign of the quadratic form can be simply determined by studying the positiveness of the matrix Σ. In this particular case, it is required to prove that the matrix is negative definite: the quadratic form is in the right hand-side and is preceded of a negative sign. According to [23], the convexity of the opposite of the entropy function s with respect to the internal energy e and the specific volume $1/\rho$ is sufficient to ensure that the matrix Σ is negative definite.

Thus, the right hand-side of the entropy residual Eq. (A.5), are now either recast as conservative terms, or known to be positive. Following the work done by [23], the entropy minimum principle holds.

APPENDIX B

DERIVATION OF THE ENTROPY RESIDUAL AS A FUNCTION OF DENSITY, PRESSURE AND SPEED OF SOUND

The entropy residual is as follows:

$$R_e(\vec{r}, t) = \partial_t s(\vec{r}, t) + \vec{u} \cdot \vec{\nabla} \cdot s(\vec{r}, t),$$

where all variables were defined previously. This form of the entropy residual is not suitable for the low-Mach limit as explained in Section 5.2.1. In this appendix, we recast the entropy residual $R_e(\vec{r}, t)$ as a function of the primitive variables (pressure, velocity and density) and the speed of sound. The first step of this derivation is to use the chain rule, recalling that the entropy is a function of the internal energy e and the density ρ, yielding

$$R_e(\vec{r}, t) = s_e \frac{\mathrm{D}e}{\mathrm{D}t} + s_\rho \frac{\mathrm{D}\rho}{\mathrm{D}t},$$

where s_e denotes the partial derivative of s with respect to the variable e. We recall that $\frac{\mathrm{D}}{\mathrm{D}t}$ denotes the material derivative. Since the internal energy e is a function of pressure P and density ρ (through the equation of state), we use again the chain rule to re-express the previous equation as a function of of the material derivatives in P

and ρ:

$$\begin{aligned}
R_e(\vec{r},t) &= s_e e_P \frac{\mathrm{D}P}{\mathrm{D}t} + (s_e e_\rho + s_\rho)\frac{\mathrm{D}\rho}{\mathrm{D}t} \\
&= s_e e_P \left(\frac{\mathrm{D}P}{\mathrm{D}t} + \frac{1}{s_e e_P}(s_e e_\rho + s_\rho)\frac{\mathrm{D}\rho}{\mathrm{D}t}\right) \\
&= s_e e_P \left(\frac{\mathrm{D}P}{\mathrm{D}t} + (\frac{e_\rho}{e_P} + \frac{s_\rho}{s_e e_P})\frac{\mathrm{D}\rho}{\mathrm{D}t}\right).
\end{aligned}$$

We are now close to the final result (see Eq. (5.9)). To prove that the term multiplying the material derivative of the density is indeed equal to the square of the speed of sound, we recall that the speed of sound is defined as the partial derivative of pressure with respect to density at constant entropy, which can be recast as a function of the entropy as follows (see Appendix A.2 of [23]):

$$c^2 := \left.\frac{\partial P}{\partial \rho}\right|_{s=cst} = P_\rho - \frac{s_\rho}{s_e} P_e \,.$$

Using the following relations (see Appendix A.1 of [23])

$$P_e = \frac{1}{e_P} \text{ and } P_\rho = -\frac{e_\rho}{e_P}\,,$$

Eq. (5.9) is obtained and recalled below for completeness:

$$R_e(\vec{r},t) := \partial_t s + \vec{u}\cdot\vec{\nabla} s = \frac{\mathrm{D}s}{\mathrm{D}t} = \frac{s_e}{P_e}\left(\underbrace{\frac{\mathrm{D}P}{\mathrm{D}t} - c^2\frac{\mathrm{D}\rho}{\mathrm{D}t}}_{\widetilde{R_e}(\vec{r},t)}\right) \,.$$

APPENDIX C

ENTROPY RESIDUAL WITH AN ISENTROPIC EQUATIONS OF STATE:

This appendix aims at showing that the entropy residual is null when assuming an isentropic flow.

The entropy residual as a function of the pressure, the density, the velocity and the speed of sound is recalled here:

$$\widetilde{R_e} = \frac{dP}{dt} - c^2 \frac{d\rho}{dt} \tag{C.1}$$

Assuming an isentropic flow, the pressure is only a function of the density as follows: $P = f(\rho)$ or $\rho = f^{-1}(P)$. Using the definition of the speed of sound $c^2 = \left.\frac{\partial P}{\partial \rho}\right)_s$ and the above form the equation of state, the following relation is derived:

$$c^2 = \left.\frac{\partial P}{\partial \rho}\right)_s = \frac{dP}{d\rho} = \frac{df(\rho)}{d\rho} \tag{C.2}$$

Using the chain rule, the entropy residual of Eq. (C.1) can be recast as a function of the density, the velocity and the speed of sound, and proven equal to zero:

$$\begin{aligned}
\widetilde{R_e} &= \frac{df(\rho)}{d\rho}\frac{d\rho}{dt} - c^2 \frac{d\rho}{dt} \\
\widetilde{R_e} &= c^2 \frac{d\rho}{dt} - c^2 \frac{d\rho}{dt} \\
\widetilde{R_e} &= 0
\end{aligned}$$

APPENDIX D

PROOF OF THE ENTROPY MINIMUM PRINCIPLE FOR THE RADIATION-HYDRODYNAMIC EQUATIONS WITH DISSIPATIVE TERMS:

In this appendix, a demonstration of the entropy minimum principle for the system of equations Eq. (7.7) is given. This proof, inspired by [23], details the steps that lead to the derivation of the dissipative terms for the multi-D Euler equations by using the entropy minimum principle.

We start with the hyperbolic system given in Eq. (7.3) and add dissipative terms to each equation as follows:

$$\begin{cases} \frac{d\rho}{dt} + \rho\partial_x u = \partial_x f \\ \partial_t(\rho u) + \partial_x\left(\rho u^2 + P + \frac{\epsilon}{3}\right) = \partial_x g \\ \partial_t(\rho E) + \partial_x\left[u\left(\rho E + P\right)\right] = \partial_x\left(h + ug\right) \\ \partial_t\epsilon + u\partial_x\epsilon + \frac{4}{3}\epsilon\partial_x u = \partial_x l \end{cases} \tag{D.1}$$

where f, g, h and l are dissipative terms to be determined. Eq. (D.1) is then recast as a function of the primitive variables (ρ, u, e, ϵ) to yield:

$$\begin{cases} \frac{d\rho}{dt} + \rho\partial_x u = \partial_x f \\ \rho\frac{du}{dt} + \partial_x\left(P + \frac{\epsilon}{3}\right) = \partial_x g - u\partial_x f \\ \rho\frac{de}{dt} + P\partial_x u = \partial_x h + g\partial_x u + \left(0.5u^2 - e\right)\partial_x f \\ \frac{d\epsilon}{dt} + \frac{4}{3}\epsilon\partial_x u = \partial_x l \end{cases} \tag{D.2}$$

The right-hand side of the internal energy equation can be simplified by choosing the dissipative terms g and h as follows: $h = \tilde{h} - 0.5u^2 f$ and $g = \rho\mu\partial_x u + uf$ where

$\mu \geq 0$ is a dissipative coefficient. Using these definitions, the system of equation given in Eq. (D.2) becomes:

$$\begin{cases} \frac{d\rho}{dt} + \rho \partial_x u = \partial_x f \\ \rho \frac{du}{dt} + \partial_x \left(P + \frac{\epsilon}{3} \right) = \partial_x g - u \partial_x f \\ \rho \frac{de}{dt} + P \partial_x u = \rho \mu (\partial_x u)^2 + \partial_x \tilde{h} - e \partial_x f \\ \frac{d\epsilon}{dt} + \frac{4}{3} \epsilon \partial_x u = \partial_x l \end{cases} \tag{D.3}$$

This system of equation admits an entropy function s that depends on density ρ, internal energy e and radiation energy density ϵ. In order to prove the entropy minimum principle, a conservation statement satisfied by the entropy is needed. This equation which is referred to as an entropy residual $D_e(x,t)$, can be obtained by a combination of the equations given in Eq. (D.3). This process is motivated by the following (chain rule)

$$\partial_\alpha s = \partial_\rho s \partial_\alpha \rho + \partial_e s \partial_\alpha e + \partial_\epsilon s \partial_\alpha \epsilon, \tag{D.4}$$

which holds for any independent variable $\alpha = x, t$. It is also required to define the dissipative terms $\tilde{h}$, f and l. The following definitions are chosen:

$$\begin{cases} f & = \kappa \partial_x \rho \\ \tilde{h} & = \kappa \partial_x (\rho e) \\ l & = \kappa \partial_x \epsilon \end{cases} \tag{D.5}$$

where κ is another positive dissipative coefficient.

Thus, using the continuity, the internal energy and the radiation equations of Eq. (D.3) and using Eq. (D.4) along with the definition of the dissipative terms, a conservation statement satisfied by the entropy s is obtained:

$$\frac{ds}{dt} + \underbrace{\left(P\partial_e s + \rho^2\partial_\rho s + \frac{4}{3}\rho\epsilon\partial_\epsilon s\right)\partial_x u}_{\text{(a)}} = \partial_x\left(\rho\kappa\partial_x s\right) + \kappa\partial_e s\partial_x s - \rho\kappa\underbrace{XAX^t}_{\text{(b)}} + \underbrace{s_e\rho\mu(\partial_x u)^2}_{\text{(c)}} \tag{D.6}$$

where X is a row vector defined as $X = (\rho, e, \epsilon)$ and A is the 3x3 symmetric matrix:

$$A = \begin{bmatrix} \partial_\rho\left(\rho^2\partial_\rho s\right) & \partial_{\rho,e}s & \partial_\rho\left(\rho\partial_\epsilon s\right) \\ \partial_{\rho,e}s & \partial_{e,e}s & \partial_{e,\epsilon}s \\ \partial_\rho\left(\rho\partial_\epsilon s\right) & \partial_{e,\epsilon}s & \partial_{\epsilon,\epsilon}s \end{bmatrix} \tag{D.7}$$

In order to show that an entropy minimum principle holds, the signs of the terms (a), (b) and (c) in Eq. (D.6) need to be studied.

Regarding (a), it is assumed that $P\partial_e s + \rho^2\partial_\rho s + \frac{4}{3}\rho\epsilon\partial_\epsilon s = 0$. The motivation for this is two-fold: First, in order to have a negative sign for the term (a), it would require $P\partial_e s + \rho^2\partial_\rho s + \frac{4}{3}\rho\epsilon\partial_\epsilon s$ to have a sign of opposite to that $\partial_x u$. The thermodynamic variables cannot be a function of the material velocity or its derivative under a non-relativistic assumption. Such a statement would not be true when dealing with relativistic equations of state. Second, a similar assumption was made in [23] for multi-D Euler equations (without the radiation energy): $P\partial_e s + \rho^2\partial_\rho s = 0$.

The term (b), XAX^t, is a quadratic form and its sign is determined by simply looking at the positiveness of the matrix A [21]. Here we need to prove that A is negative-definite which is equivalent to showing the three following inequalities:

$$\begin{cases} A_1 \geq 0 \\ A_2 \leq 0 \\ A_3 = A \geq 0 \end{cases} \tag{D.8}$$

where A_k is the k^{th} order leading principle minor. Determining the sign of the last inequality that corresponds to the determinant of the 3 by 3 matrix A can be difficult and needs to be simplified. Zeroing out the off-diagonal entries of the last row or column would simplify the expression for the determinant of A. This can be achieved by assuming $\partial_\rho(\rho\partial_\epsilon s)$ and $\partial_{e,\epsilon}s$ are zero, which requires the following form for the entropy function:

$$s(\rho, e, \epsilon) = \tilde{s}(\rho, e) + \frac{\rho_0}{\rho}\hat{s}(\epsilon). \tag{D.9}$$

where $\tilde{s}$ and $\hat{s}$ are two functions whose properties will be provided later. The constant ρ_0 is used for a dimensionality purpose. Next, using the expression of the entropy given in Eq. (D.9), matrix A becomes:

$$A = \begin{bmatrix} \partial_\rho\left(\rho^2\partial_\rho\tilde{s}\right) & \partial_{\rho,e}\tilde{s} & 0 \\ \partial_{\rho,e}\tilde{s} & \partial_{e,e}\tilde{s} & 0 \\ 0 & 0 & \rho^{-1}\partial_{\epsilon,\epsilon}\hat{s} \end{bmatrix}$$

Proving that the matrix A is negative-definite is now straightforward by inspecting the sign of the leading principal minors:

$$\begin{cases} A_1 = \partial_\rho\left(\rho^2\partial_\rho\tilde{s}\right) \leq 0 \\ A_2 = \partial_\rho\left(\rho^2\partial_\rho\tilde{s}\right)\partial_{e,e}\tilde{s} - \left(\partial_{\rho,e}\tilde{s}\right)^2 \geq 0 \\ A_3 = \rho^{-1}\partial_{\epsilon,\epsilon}\hat{s}A_2 \leq 0 \end{cases} \tag{D.10}$$

This is easily achieved when assuming that the functions $-\tilde{s}$ and $-\hat{s}$ are convex. Thus, the sign of (b) is now determined.

Finally, it remains to determine the sign of the term $(c) = \partial_e s\rho\mu(\partial_x u)^2$. The density ρ and the viscosity coefficient μ are both positive: the latest proof for positivity of the density can be found in [23]. Then, only the sign of $\partial_e s$ remains unknown but it can be determined by studying (a). It was assumed earlier in this appendix that $P\partial_e s + \rho^2 \partial_\rho s + \frac{4}{3}\rho\epsilon\partial_\epsilon s = 0$. This equation is now recast and split into two equations using Eq. (D.9). Separation of variables yields:

$$P\partial_e \tilde{s} + \rho^2 \partial_\rho \tilde{s} = \alpha \text{ and } \hat{s} - \frac{4\epsilon}{3}\partial_\epsilon \hat{s} = \alpha$$

where α is a constant to determine. If one sets $\alpha = 0$, then the two physics are decoupled, which allows us to reconnect to the result derived in [23] for the multi-D Euler equations: $P\partial_e \tilde{s} + \rho^2 \partial_\rho \tilde{s} = 0$. Then, following [23], definitions for $\partial_e \tilde{s}$ and $\partial_\rho \tilde{s}$ are obtained:

$$\begin{cases} \partial_e s = \partial_e \tilde{s} = T^{-1} \\ \partial_\rho \tilde{s} = -\frac{P}{\rho^2}\partial_e \tilde{s} \end{cases}$$

where T is the material temperature which ensures positivity of $\partial_e s$. Thus, (c) is positive.

From the above results, the entropy minimum principle follows, so that the sign of the entropy residual is known:

$$\boxed{\partial_t s + u\partial_x s \geq 0} \tag{D.11}$$

Remark. *By assuming $\alpha = 0$, an expression for the $\hat{s}$ can be derived by solving the ODE, $\hat{s} - \frac{4\epsilon}{3}\partial_\epsilon \hat{s} = 0$, which yields: $\hat{s}(\epsilon) = \beta \exp\left(\frac{4\epsilon^2}{3}\right)$, where β is a constant. The sign of β is determined by using the condition, $\partial_{\epsilon,\epsilon}\hat{s} \leq 0$, derived above, so that $\beta \leq 0$.*

Remark. *The viscous regularization derived in this appendix, has two viscosity coefficients: μ and κ. For the purpose of this paper, these coefficients are set equal. Under this assumption, the above viscous regularization is equivalent to the parabolic* *regularization of [5].*