

# Numerical Modelling for Compressible Flows

MARC BARCELÓ ADROVER

*Cranfield University, January 2020*

## Abstract

The aim of this document is to evaluate the performance of different high-order discretization schemes (MUSCL, WENO) and flux approaches for the Approximate Riemann Solver (Lax-Friedrichs, Rusanov, HLL, HLLC) to predict the compressible fluid behaviour in three different benchmark problems for 1D structured and 2D unstructured meshes. The problems faced are: Forward-Facing Step, Double Mach Reflection and Cylindrical Explosion. The Riemann solver will rely on Godunov's theory which will be briefly explained in the following lines. Additionally, for the MUSCL schemes, the use of limiters will become necessary then the effect of Barth and Jesper and Venkatakrishnan limiters will be implemented and compared.

## 1. Introduction

It is not necessary to remark the importance of flows in our daily life. They appear in quite any scenario or subject that we can imagine, from the human body, to medicine, hydrodynamics, air, study of mass population, economy, and a large list of examples where they take place. However, the description of their behaviour through the analytically unsolvable and hostile partial differential equations known as the Navier-Stokes equations and conservation principles has led to the great development of Computational Fluid Dynamics, where numerical solutions to those equations can be obtained without sacrificing many important fluid traits through simplifications of the equations. Nevertheless, the schemes and procedures used in its computation may be properly chosen and developed as unphysical and artificial results may otherwise appear. Besides, the accuracy of the solution will be sometimes endangered, and it will suppose a compromise solution between expected accuracy and affordable computational cost.

Depending on the flow, some classifications can be made to clarify the scenario dealt and know the scope in which they move. If the velocity magnitude in a particular axis is taken as reference; subsonic, transonic and supersonic flows can be distinguished. Besides, a more global and used classification is most of times made: compressible or incompressible flows.

In this document the first ones will be explained, dealt, and simulated. For further information about incompressible flows, the reader is referred to: (Landau L.D. and Lifshitz E.M., 1987). A flow is considered to be compressible if its Mach Number  $M > 0.3$ , if the density is not constant and can be described as a function of the pressure or vice versa, and if the divergence of the velocity field is not 0. Thus, this group embodies a huge amount of flow problem examples and specially numerous in our reference field: Aerospace.

Compressible flows, most of times, are described with the Compressible Euler Equations. In them and throughout this document as well, the ideal fluid is many times assumed: viscous terms are neglected, an equation of state relates pressure with density, the fluid is isentropic with gamma-law behaviour ( $\gamma = 1.4$ ), and no energy or forces are considered except from the ones derived from gravity and pressure forces. The resulting simplification from the Navier-Stokes Equations will result in a set of hyperbolic-type partial differential equations.

$$U_t + F(U_x) = 0 \quad (1)$$

Where  $U_t$  represent the vector of conserved variables and  $F(U)_x$  the vector of fluxes. For 1-D Euler Equations, the resulting Continuity and Momentum equations will result in:

$$\left. \begin{aligned} \frac{d\rho}{dt} + \rho_0 \cdot \frac{\partial u}{\partial x} &= 0 \\ \frac{\partial u}{\partial t} + \frac{a^2}{\rho} \cdot \frac{\partial u}{\partial x} &= 0 \end{aligned} \right\} \quad (2)$$

Being, if this set of equations is rewritten as (1),  $U = \begin{bmatrix} \rho \\ u \end{bmatrix}$  and  $F(U_x) = \begin{bmatrix} 0 & \rho_0 \\ \frac{a^2}{\rho} & 0 \end{bmatrix} \cdot U_x$ .

The solution of this problem, when the flow environment is smooth, can be described with the semi-analytical Burger's equation by applying the Initial Value Problem and the proper boundary conditions, however, when there are discontinuities (see Riemann Problem) another approach is required to achieve reliable results. This is where the Godunov's method will stand.

In the following sections, a theoretical background of the Godunov method for 1-D and 2-D structured and unstructured grids will be outlined, along with the fluxes modelling, yet the types of waves present in compressible flows and limitations of finite difference elements will be also explained. Afterwards, a brief description of the benchmark problems faced will be provided and the results from our schemes (MUSCL, WENO) and flux approximations (Rusanov, Lax-Friedrich, HLL, HLLC) will accordingly be discussed and compared to reference publications. Finally, all the important outcome from this work will be properly reviewed in the conclusions section.

## 2. Theory Fundamentals

### 2.1 Godunov Method

Godunov achieved to obtain an exact solution of the conservative form set of non-linear hyperbolic PDEs (1) through a first order of spatial and time accuracy “upwind” scheme. It is important to remark that conservative form of equations is indispensable for treating flows where discontinuities are present, yet conservative and non-conservative forms are equally valid when smooth flow regions are treated. Serve as example of this fact the Burger's equation failure (Zanotti and Manca, 2010) in the computation of the correct shock speed if a discontinuity is present as an Initial Condition.

Using Finite Difference, the grid of discrete values of flow magnitudes over the adjacent nodes -or better, cells- will suppose a discontinuity of N-1 local Riemann problems, which are remarkably important in non-smooth regions. The solution of the Riemann problem always involves the method of characteristics in which the PDE is transformed into and the curves of constant solution (once applied the initial condition) are obtained. Those curves will limit the scope of the solution and will also describe the nature of the present waves in the flow. Then, this procedure must be applied to the conservative equation (1).

Godunov uses a piece-wise constant distribution of initial data along the cells as the following image reveals.

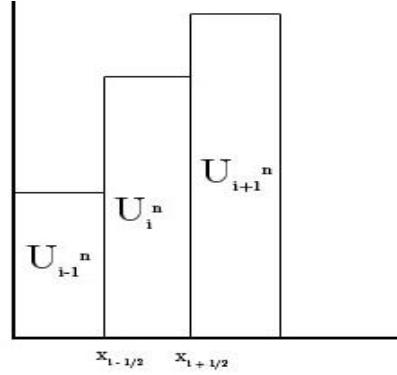


Figure 1. Godunov's piecewise constant construction of the current-time spatial domain

The definite integral of (1) must be applied into the discrete domain resulting in: Then, he solves the intercell fluxes and  $u$  values -let's say, whether  $U^*(x \pm \frac{1}{2})$  and  $F^*(x \pm \frac{1}{2})$  – with the integral form from (1)

$$F_{i \pm \frac{1}{2}} = \frac{1}{\Delta t} \cdot \int_{t^n}^{t^{n+1}} F \left[ U \left( x_{i \pm \frac{1}{2}}, t \right) \right] dt \quad (3)$$

$$U_i^n = \frac{1}{\Delta x} \cdot \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} U(x, t^n) dt \quad (4)$$

Leading to

$$U_i^{n+1} = U_i^n + \frac{\Delta t}{\Delta x} \cdot \left( F_{i-\frac{1}{2}} - F_{i+\frac{1}{2}} \right) \quad (5)$$

It can be seen that the numerical flux does not now depend on the whole Riemann solution, only on the flux value at  $x_{i-1/2}$  as the Riemann solution is the same along straight curves (characteristics) of  $x/t = \text{constant}$  values. Additionally, according to Jensen's inequality (Harten, Lax and van Leer, 1983) Godunov's scheme is entropy satisfying, thence all shocks performed are physically correct.

The time evolution from  $t^n$  to  $t^{n+1}$  by adding the term  $\Delta t$  assumes that the flow variables in over the spatial grid are Now it is time to apply the Riemann Problem solution, as the initial and boundary conditions -between adjacent cells- suppose a discrete discontinuity, then a N-1 set of local Riemann problems is dealt.

$$\begin{array}{ll} \text{Initial and} & \left. \begin{aligned} U(x, 0) &= U_x^0 \\ U(0, t) &= U_L(t) \end{aligned} \right\} \\ \text{Boundary} & \\ \text{Conditions} & \left. \begin{aligned} U(\Delta x, t) &= U_R(t) \end{aligned} \right\} \end{array} \quad (6)$$

The curve characteristics will be the eigenvalues from the matrix system of (2). Hence, according to (Toro, 2009) the solution of the 3 present waves including the solution (for 1-D Euler Eqs):

$$U_L = \alpha_1 \cdot e^{(1)} + \alpha_2 \cdot e^{(2)}$$

$$U_R = \beta_1 \cdot e^{(1)} + \beta_2 \cdot e^{(2)}$$

with  $\alpha_1 = \frac{a \cdot \rho_L - \rho_0 \cdot U_L}{2 \cdot a \cdot \rho_0}$ ,  $\alpha_2 = \frac{a \cdot \rho_L + \rho_0 \cdot U_L}{2 \cdot a \cdot \rho_0}$ ,  $\beta_1 = \frac{a \cdot \rho_R - \rho_0 \cdot U_R}{2 \cdot a \cdot \rho_0}$  and  $\beta_2 = \frac{a \cdot \rho_R + \rho_0 \cdot U_R}{2 \cdot a \cdot \rho_0}$ . Finally, the intermediate solution of the conserved variables is

$$U_* = \beta_1 \cdot \begin{bmatrix} \rho_0 \\ -a \end{bmatrix} + \alpha_2 \cdot \begin{bmatrix} \rho_0 \\ a \end{bmatrix}$$

Then, by applying the upwind scheme, the final solution for the conserved variables are

$$U(t, x) = \begin{bmatrix} \rho(t, x) \\ u(t, x) \end{bmatrix} = \begin{cases} U_L & 0 < t \leq -x/a \\ U_* & 0 \leq \left| \frac{x}{a} \right| < t \\ U_R & 0 < t \leq x/a \end{cases} \quad (7)$$

As an overall summary, Godunov's first order method in space and time accuracy uses piecewise constant distribution of data, thence a part of the value of  $U(x, t^n)$  is lost in the cell and reconstruction procedures may be required to enhance it maintain a desired order of spatial accuracy. For the time domain, according to (Zanotti and Manca, 2010) it is recommended to implement a conservative Runge-Kutta scheme. Besides, the Courant condition is restricted to be satisfied at any case.

The previous information gathers the procedure to be applied for a one-dimensional problem. When a 2-D structured grid is aimed, it must be considered that a new dimension in y coordinate is introduced and now the Conservative PDE (1) becomes:

$$U_t + F(U)_x + G(U)_y = 0 \quad (8)$$

Although the method remains the same, the vectors will introduce one term per flow vectorial variable and the reconstruction procedures will also enlarge by adding another spatial loop. For the 2-D unstructured grid, although (Toro, 2009) provides a long explanation, yet our reference document will be (Chen, Tang and Zhang, 2008), where it is briefly but elegantly explained. Imagine -for simplicity and as it is widely used- the scheme of a 2-D grid unstructured whose partial differential equation in conservative form is again (8). The presence of triangular-grid domain implicitly derives in an structured domain. As a reference, serve the following figure of a cell with its neighbouring triangles to easy the explanation of this case.

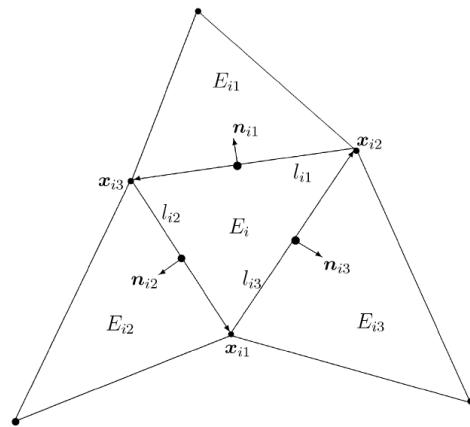


Figure 2. Schematic representation of cells in a triangular non-structured grid. Source: (Chen, Tang, Zhang; 2008)

Thus, attending to the notation, given a triangular unstructured grid of N elements such that any element  $E_i$  belongs to the domain  $\{E_1, \dots, E_n\}$ , with its vertexes referenced as  $x_{i,j}$ , the edges are named with  $l_{i,j}$  and  $n_{i,j}$  its normal unitary vector.

Besides, time domain will also be discretized as  $[0,t]$  domain with  $t^{n+1}=t^n+\Delta t$  construction or forward stepping. Nonetheless,  $\Delta t$  as it has been seen is restricted to values that satisfy the CFL condition to ensure the discretization stability.

Flows must be obtained by integration over the triangular domain such as it follows – based on (Chen, Tang and Zhang, 2008) deduction-:

$$|E_i| \cdot \frac{d}{dt}(\bar{U}_{E_i})(t) = - \oint_{\partial E_i} F_{n_i}(U) \cdot dS = - \sum_{j=1}^n \int_{I_{ij}} F_{n_{ij}}(U) \cdot dS_j$$

where  $F_{n_{ij}}$  is the total vector of fluxes,  $F_{n_{ij}} = F(U) \cdot n_{ij}^x + G(U) \cdot n_{ij}^y$ ; and  $\bar{U}_{E_i}$  is the total cell average of the conservative variable  $U$  inside the triangle  $E_i$ .

If the midpoint integration is conveniently used to approximate the previous integral and, according to (Chen, Tang and Zhang, 2008) “*substituting the exact solution U at the middle point of the edge by the approximate solution (based on the piecewise polynomial reconstructed by using cell averages in the Finite Element Volume)*” and finally replacing the vector  $F_{n_{ij}}$  by any two-point flux  $\widehat{F}_{n_{ij}}(U_{l_{ij}}^L, U_{l_{ij}}^R)$  that satisfies Lipschitz uniform continuity criteria, the previous conservative equation can be discretely modelled as

$$\frac{d}{dt}(\bar{U}_{E_i})(t) = - \frac{1}{|E_i|} \cdot \sum_{j=1}^n \widehat{F}_{n_{ij}}(U_{l_{ij}}^L, U_{l_{ij}}^R) \cdot |I_{ij}|$$

and the flux can be defined as the Godunov flux with the exact Riemann Solver (see (Godunov, 1959)) by:

$$\widehat{F}_{n_{ij}}(U_{l_{ij}}^L, U_{l_{ij}}^R) = F_{n_{ij}}\left(\omega^*(0, U_{l_{ij}}^*, U_{l_{ij}}^*)\right)$$

where  $\omega^*$  gathers the 1-D Riemann problem seen in previous lines. The further methodology to be applied has been already schematically explained for the 1-D case and is not subject of interest due to the word limitation of this large document. For further information, consider (Toro, 2009).

It must be said that the procedure explained here remains the basis of the Godunov exact solution, although, for efficiency but also to achieve higher orders of accuracy -remember Godunov's method is only 1<sup>st</sup> for space and time domain- Approximate Riemann Solvers are used and will be detailed in Appendix 1.

Depending on their procedure, they can be classified into “Approximate State Riemann Solvers” where the state  $U(x_{i \pm \frac{1}{2}}, t)$  is approximated and then used to evaluate the corresponding flux, or the “Approximate Flux Riemann Solvers”, where the approximation is directly performed in the modelling of the flux and therefore avoiding the tedious calculation of  $U(x_{i \pm \frac{1}{2}}, t)$  at each zone edge. Rusanov, HLL, HLLC, and Lax-Friedrichs fluxes are contained in this last classification and will be detailed in Appendix 1.

The applicability of the Godunov Method to solve other type of PDEs instead of hyperbolic system has not been fully developed nowadays. Thus is, as it is said in (Toro, 1999), a product of the fact that the current mathematical and computational –understanding– theory of

hyperbolic equations is largely more advanced than more complete and complex equations such as the Navier-Stokes for Fluid Dynamics.

## 2.2 Types of (Non-linear) Waves in Compressible Flows

Inside the compressible flow classification, fluid problems where the local velocity dramatically approaches the local sound speed or even overpasses it can be faced. Specially in transonic and supersonic environments, a wide variety of waves or fluid phenomena can take place and may cause discontinuities in the flow thermodynamic variables. Depending on its effects and particular traits, these present waves can be divided into contact waves, rarefaction, or shock waves. Here, the most important features of each one will be outline, nevertheless, for further information, see (Landau L.D. and Lifshitz E.M., 1987) .

### 2.2.1 Contact waves / Contact discontinuities

Also called as simple discontinuities, the variables of density and temperature are not constant before and after the wave besides pressure and velocity do. The main case of this example is a multiphase fluid problem in the interface of different gases/flows, as for boundary conditions between fluids, velocity must remain equal at the infinitesimal interface. Differently to the rarefaction or the shock waves, they do not have any dependency on the local speed to appear. When the Riemann problem is about to be formulated containing this discontinuity, one should notice that Rankine-Hugoniot conditions apply, the Generalized Riemann Invariants remain constant and their characteristic curves of both side waves are parallel. Besides, this kind of wave, it includes the known case of “slip discontinuity”, where the tangential velocity of both gases differs significantly, as it happens in most nozzles or by the interaction between a shock wave and Prandtl-Meyer expansion in a particular geometry. As a summary, the mathematical formulation of the thermodynamic variables before and after the discontinuity are given by:

$$\rho_1 \neq \rho_2, \quad p_1 = p_2, \quad u_1 = u_2, \quad T_1 \neq T_2 \quad (9)$$

### 2.2.2 Rarefaction

Also known as “expansion zones” or Taylor waves, they tend to occur after a shock wave and cause the flow magnitudes of density and pressure decrease abruptly. In the contrary of shock waves, according to (Zhang, 2016), it travels directly in the opposite direction than the acceleration of particles. Even though shock waves are usually a steady continuity, rarefactions are clearly unsteady and spread throughout space and time, thence, their velocity is also function of time. They can be described with Hugoniot principle. More information about rarefaction waves can be sighted in (Landau L.D. and Lifshitz E.M., 1987), and (Henrie, Carpenter and Nicholas, 2016). When plots of the flow magnitudes are given, rarefaction phenomena can be detected when:  $\rho_1 > \rho_2, \quad p_1 > p_2 \quad (10)$

In longitudinal waves, the existence of a region of rarefaction implies a zone of compression, where  $U_1 < U_2$ .

### 2.2.3 Shock waves

They occur after  $M=1$  is achieved in the flow field, causing all the fluid variables to have different values than they had before the discontinuity. Usually, there's an increment of pressure and temperature after the shock, but normal velocity decreases. Their behaviour is strongly dependent on the velocity magnitude in a determined axis but also to the geometry of a solid body inherent inside the fluid flow. Thence, depending on the geometry and flow behaviour, shock waves can be classified into: Oblique/Normal, Strong/Very weak, and Adhered/Detached shock waves. Each one of them has its particular traits and equations -or better said, simplifications/approximations-although the expected increment in pressure and temperature after the shock and decreasing in normal velocity are expected at any case. Due to their nature, they are said to be a compressive phenomenon, and, as it has been said, they tend to be really steady—"fixed"- in time and space. Once the Riemann problem is formulated, the solution must satisfy Rankine-Hugoniot theorem and the entropy condition. Overall, most of times all the fluid variables have a discontinuity across the shock wave, then

$$\rho_1 \neq \rho_2, \quad p_1 \neq p_2, \quad u_1 \neq u_2, \quad T_1 \neq T_2 \quad (11)$$

For further information, please see (Landau L.D. and Lifshitz E.M., 1987)

## 2.3 Limitations of Finite Difference Schemes for Compressible Flows

The Finite Difference Methods outstands in comparison with Finite Volume Methods and Finite Difference method due to its simplicity to implement. The numerical schemes are easily constructed and implemented, as partial derivatives of PDE are substituted with approximations by series -usually Taylor-, and diminishing the error by the addition of higher order terms. Nonetheless, the more terms included, the more complexity in computational cost and denser results. Besides, it is implemented "dimension per dimension", thence the easiness to enhance the accuracy order dealing with a "decoupled" problem. In this sense, it is much simpler to construct and compute than FVM or FEM. The main problem is that its good performance is restricted to structured meshes and regular geometries. If not, the equations before series approximations must be transformed and it introduces a large collection of derived problems, such as cross-coupling of equations, mesh generations, issues in convergence, etc. Neither the most efficient techniques of mesh treatment such as local grid refinement-which is most of times required in non-smooth regions- or "adaptive mesh refinement" merge well with the developed schemes. Furthermore, FDM do not perform as good as expected when material discontinuities are present either.

This important issue is applied to the boundaries of the problem as well, are also heavily limited: they cannot be curvilinear neither move - they must remain steady. Although they do not perform great with conservative equations, currently, -according to <https://scicomp.stackexchange.com/questions/2357/disadvantages-of-common-discretization-schemes-for-cfd-simulations> - some low-order conservative finite difference methods are actually capable to deal with unstructured meshes, however, oscillations may be present. Additionally, some developed Finite Difference WENO schemes that try to overcome this problem, are not suitable for all the Riemann solvers-

### 3. Benchmark Problems' Characteristics

#### 3.1 The Forward-Facing Step

The Forward-Facing step is one of the most studied benchmark problems in Computational Fluid Dynamics, serving to contrast the accuracy of the numerical scheme developed. Thus, there's a wide amount of analytical and numerical data from research papers to serve as an accurate reference. One of the first documents to describe this problem is (Emery, 1968a), although the first computational simulation to implement the MUSCL scheme for it is seen in (van Leer, 1979a). Additionally, more recent and complete investigations are: (Iftekhar, 2016; Iftekhar and Agelin-Chaab, 2015; Scheit, Esmaeili and Becker, 2013). In this section exactly the same operation conditions and geometry of (Woodward and Colella, 1984) will be computed, thence it will be used as the main reference.

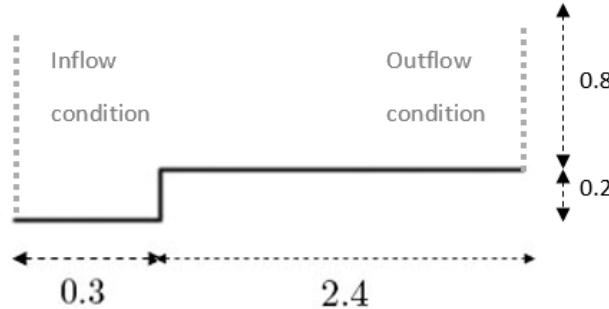


Figure 3. Schematic Representation of the Forward-Facing Step Geometry

The wind's tunnel geometry is 3x1 units of length. The 3<sup>rd</sup> dimension will not be considered as the step is supposed to be infinite in it. The configuration of the step is just as Figure 3 suggests. As initial conditions, the inflow current is uniform  $M=3$ . The exit of the wind tunnel will be considered as an outflow, thence the gradients of the flow magnitudes in it will be set to 0. Additionally, this boundary condition will not suppose any further effect on the flow region as the exit velocity is, in any case, supersonic. The solid walls will be properly described with reflective boundary conditions. Additionally, the ideal gas parameters of gamma, density and pressure will take the values of:  $\gamma = 1.4$ ,  $\rho = 1.4$  and  $p = 1.0$ .

#### 3.2 The Double Mach Reflection

The reference document for this scenario will be again (Woodward and Colella, 1984), who where actually the first ones to propose this case, nonetheless, it is also described in (Colella and Glaz, 1985; Kemm, 2016; Vevek, Zang and New, 2019).

A  $M=10$  inflow shock faces a  $30^\circ$  reflective wall in its propagation direction, creating primary and secondary Mach stems and slip lines as it ascends tangentially to the wall. Besides, the undisturbed air in front of the initial shock is characterized with  $\rho = 1.4$  and  $p = 1.0$ . The problem's schema in the initial stage will look like the following.

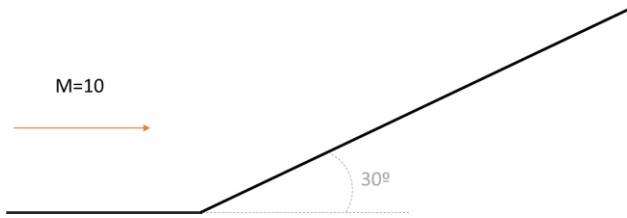


Figure 4. Double Mach Reflection Problem Geometry

### 3.3 Cylindrical Explosion

For this problem, no reference document is available to perform a comparison and validation of the experiment, nevertheless, reference data has been provided yet also the definition of the problem. However, in pages 602-612 of (Toro, 2009) this problem has been commented. In a determined domain over a grid of  $[-1,1] \times [-1,1]$  (problem will be dealt for 2-D unstructured mesh and 1-D structured grid), there's the presence of two different gases with uniform parameters for each that are separated by a cylinder of 0.4 radius. Thence, the formulation of the gas parameters according to the geometry of this problem is -for the 2-D case- :

$$(\rho, p) = \begin{cases} (1.0, 1.0) & \text{for } r \leq 0.4 \\ (\rho_0, p_0) & \text{for } r > 0.4 \end{cases}, \quad u=0, v=0, \quad r^2 = x^2 + y^2 ,$$

and therefore, for the 1-D case (x-axis):

$$(\rho, p) = \begin{cases} (1.0, 1.0) & \text{for } r \leq 0.4 \\ (\rho_0, p_0) & \text{for } r > 0.4 \end{cases}, \quad u=0, \quad r^2 = x^2 + y^2 .$$

## 4. Implementation of the Code

All the files (meshes, initialization files, solvers) have been given by Dr. Panagiotis Tsoutsanis thence they are not property or created by the author of this document. However, two Riemann solvers and two limiters that are required to compute in the assignment do not appear in the *solver.f90* file. It is then a duty of the student to implement both by using the same variables already used in the *solver.f90* and by using the literature reference of: (Michalak and Ollivier-Gooch, 2008; Venkatakrishnan, 1993, 1994; Zheng, 2013) for the limiters and (Harten, Lax and van Leer, 1983; Toro, 1992, 2009, 2019; Toro, Spruce and Speares, 1994; Zanotti and Manca, 2010) for both HLL and HLLC schemes.

The essential theory of the Godunov method, the Approximate Riemann Solvers and Limiters has already been explained in the sections of Introduction, Theory Fundamentals and Appendix 1. Therefore, in Appendix 2 will be attached the Missing Codes for the 1-D and 2-D solvers.

## 5. Results

### 5.1 One-dimensional Cylindrical Explosion

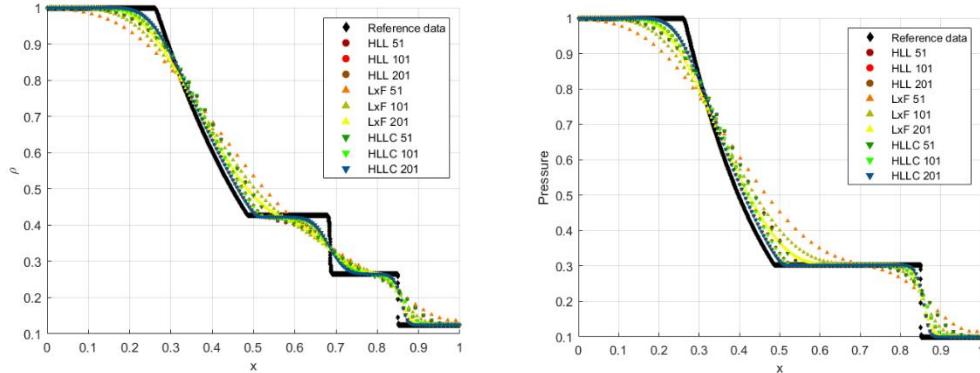
Once implemented the missing codes (HLL, HLLC) yet also the second order reconstructor based on the minmod limiter, the results in the 1-D Cylindrical Explosions will be discussed here. The base problem of the Cylindrical Explosion has been already explained in **Cylindrical Explosion** section, although a brief summary of it will be provided.

Initially, the problem considers that two gases of uniform but different parameters between themselves are separated by a cylinder of radius 0.4. An explosion is simulated, and different non-linear waves will be observed travelling within the solution evolving in time and space as time goes by.

Firstly, as the problem possesses a radial symmetry, only one side of the solution will be plotted. The CFL number is set to 0.9, which will limit the time step. We can expect identically the same behaviour in the other direction. In (Figure 1 ref), the first order solution for three different Riemann Solvers (Lax-Friedrichs, HLL, HLLC) will be compared to an analytical reference solution provided. Also, the effect of the resolution of the mesh (51, 101, 201 cells) will be shown since determinate resolutions may be not enough to describe and correctly predict the non-smooth fluid phenomena that normally takes place in compressible flows.

Figure 5 reveals, at first glance, two evident issues. The first is the late prediction of the fluid behaviour, thence there's a translation in the space domain that looks constant for all the fluid magnitudes, yet it is not too significant and a proportionality between them is sighted.

Afterwards, the first order of accuracy introduces an evident dispersion which is coherent with what is detailed in (Anderson, 1995). Thus, an increment of the order of accuracy seems necessary, although diffusion can also take place for even orders of accuracy.



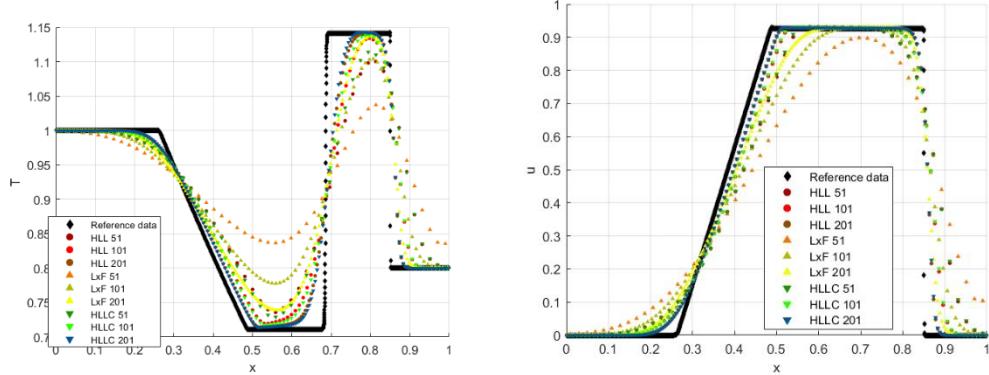


Figure 5. First-Order solution of fluid variables in the 1-D Explosion problem for  $t=0.2$ . Comparison between analytical solution with different Riemann Solver with different mesh resolutions.

The effect of the mesh and the Riemann solvers is also remarkable. As grid resolution is doubled, the higher accuracy is achieved. Grid convergence is therefore obtained. It is a consequence of the fact that solution is not able to evolve correctly in the spatial domain, thence the more spatial information, evidently -in this case- the better the results will be. It is specially critical for the “worst” Riemann solver analyzed, the Lax-Friedrich flux, whose results are unacceptable for the 51 cells mesh. It performs as the weakest solver for all the mesh resolutions.

Thus, HLLC and HLL results for the 201-cell mesh outstand as the most accurate ones, they barely are overlapped. However, as it can be seen, for the 101-cell mesh, the HLLC is clearly more accurate than HLL probably due to the introduction of the intermediate contact wave in its formulation. This fact has been explained in Flux Approximations: Riemann Solvers.

Remember, that this solution is the spatial distribution of flow variables for the precise time of  $t=0.2$  s. The discussion of results that follow this sentence will be referred to this exact time.

The jumps in the fluid magnitudes of density, pressure, temperature and velocity reveal the presence of non-linear waves that evolve in time and space. There are three jumps in density and temperature variables but only two in the pressure and velocity. It implies the presence of a contact wave located at  $x=0.7$ . The other two linear waves are firstly a shock wave in  $x=0.27$  and a rarefaction at approximately  $x=0.85$ . Thence, the correct sequence of effects sighted, from the center to the exterior, is an initial shock wave followed by a contact wave and then finished by a rarefaction.

If the file movie.dat is processed by Tecplot into a video (extension .mp4 or similar) the evolution in time and space of both three waves can be perfectly followed. A circular shock wave travels from the center to the outer region, similarly as the contact discontinuity does. On the other hand, the rarefaction respectively propagates from the exterior to the interior. It is coherent with has been explained in the section of **Types of (Non-linear) Waves in Compressible Flows**, but also with the information provided by (Toro, 2009), where a 2-D explosion test is analyzed. However, more information of it will be explained in the 2-D Explosion problem that will follow this section.

The results for the 1<sup>st</sup> order of accuracy demonstrated a clearly diffusive behaviour of the first order of accuracy. Thence, the need to compute higher orders of accuracy, in this case, the 2<sup>nd</sup> order.

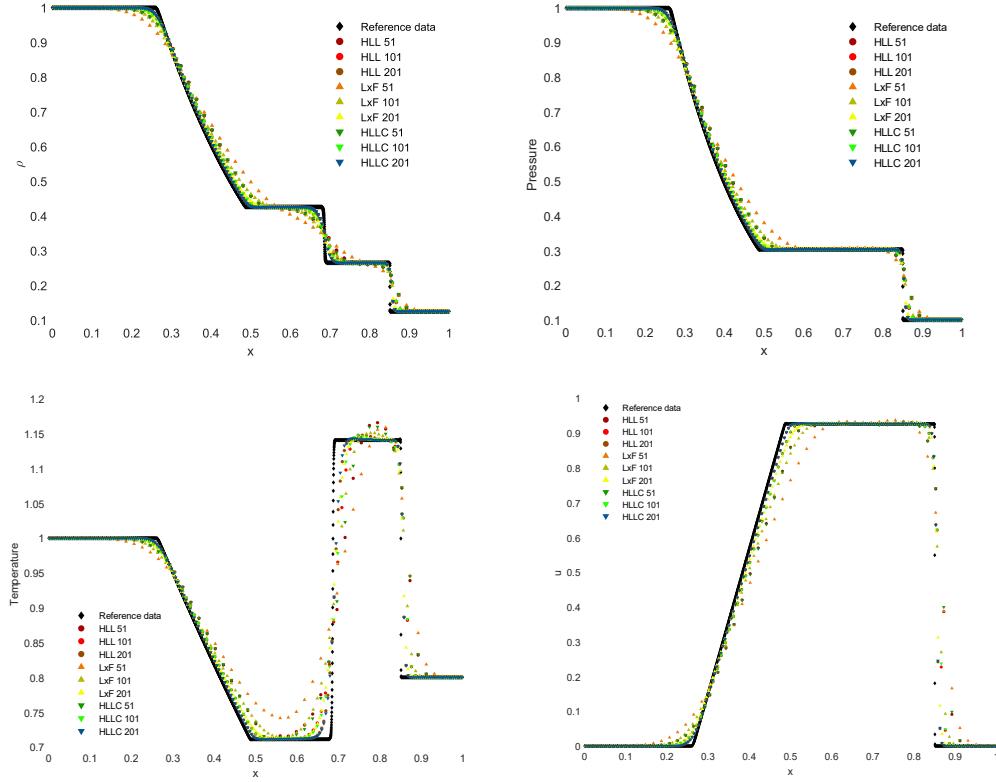


Figure 6. Second-Order solution of fluid variables in the 1-D Explosion problem for  $t=0.2$ . Comparison between analytical solution with different Riemann Solver with different mesh resolutions

The diffusion introduced by the first order scheme is not present anymore. Solution for all the Riemann solvers and mesh configurations remains more accurate to the reference data. The only weakness is the fact that – according to (Anderson, 1995) – although odd orders of accuracy introduce diffusion, even orders are prone to be dispersive. Thence, some oscillations could be seen in the slope of the solution. However, it is only evident in the plot of the Temperature at Figure2 ref.

Despite from this issue, all the other aspects are similar to the 1<sup>st</sup> order of accuracy solution. The best Riemann solver is again HLLC for the 201 mesh, solutions are more accurate as grid is refined. The worst Riemann solver, accordingly, is Lax-Friedrichs.

The same three waves at the exact three spatial points of  $x=0.27$ ,  $0.7$  and  $0.85$  are sighted: shock wave, contact wave and rarefaction respectively.

## 5.2 Two-dimensional Cylindrical Explosion

Once the results from the 1-D case have been analyzed, the same problem will be simulated for a 2-D unstructured grid with CFL number of 0.7. The same non-linear waves, however, are expected to take place as well as the phenomena already predicted. The principal change introduced in the results will be the presence of the Y dimension, and their relative fluid variables in this Y axis. Additionally, the Riemann solvers Rusanov and HLLC and the

Venkatakrishnan limiters have been implemented to the 2-D solver as it is detailed in Appendix 2. Solver Missing Code.

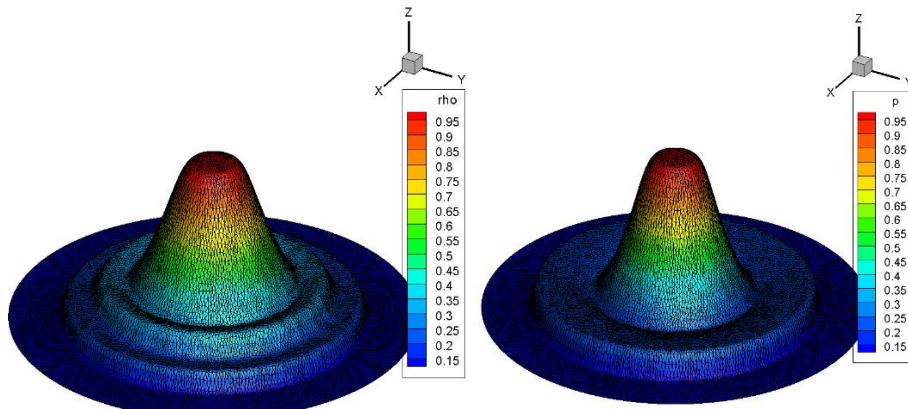
Nonetheless, for this problem, the procedure to obtain the desired plots have enlarged a little. The results that come from the executable *a.out* have been graphed in Tecplot and a radial slice from the whole two-dimensional domain has been created. Afterwards, this same slice is extracted and an output file of the fluid variables of this slice has been written. The coordinates of the fluid variables are given in cartesian form, thence, the radial coordinate must be computed through the transformation of  $r = \sqrt{x^2 + y^2}$ .

The representation of the variables can now be done in 3-D, which gives an accurate view of what is spatially happening to the fluid magnitudes. The third coordinate, in Z axis, will be accordingly the plotted fluid magnitude, as X and Y will be designed for the spatial evolution of it.

Remember that for the 2-D plots, radial and symmetric behaviour will be considered. The results will be shown for the time step  $t=0.2$  s., for second order of accuracy. For the 3-D representation of the fluid magnitudes, please see Figure 7, where density, pressure and velocity have been plotted. They demonstrate great similarity with the solution of the 2-D explosion problem shown in(Toro, 2009), although he has computed it for  $t=0.25$  and with a totally different numerical scheme (WAF, RCM), limiter (Van Leer) yet also the constants from the gases.

The three types of waves can be visually sighted: shock wave, contact wave and rarefaction in the case of the density, and accordingly, the shock wave and the rarefaction in the pressure graph.

In the movie file, where the evolution from the initial time to  $t=0.2$  for each fluid variable is shown, additional traits can be seen. Just like the 1-D Explosion Problem, as time goes by, the shockwave and the contact wave travel from the center to the outer region, in contrast, the rarefaction evolves inversely (from outside to inside). Besides, the shockwave and the contact wave become weaker proportionally to the time. According to (Toro, 2009), if the movie was allowed to last more time, a pattern of interaction of rarefaction-contact wave could be seen repeatedly. The rarefaction that travels radially inwards expands the flow by creating an inward travelling shock wave, then it implodes to the origin, reflects and travels radially outwards until it collides with the contact wave, and again the process is repeated but with less magnitude.



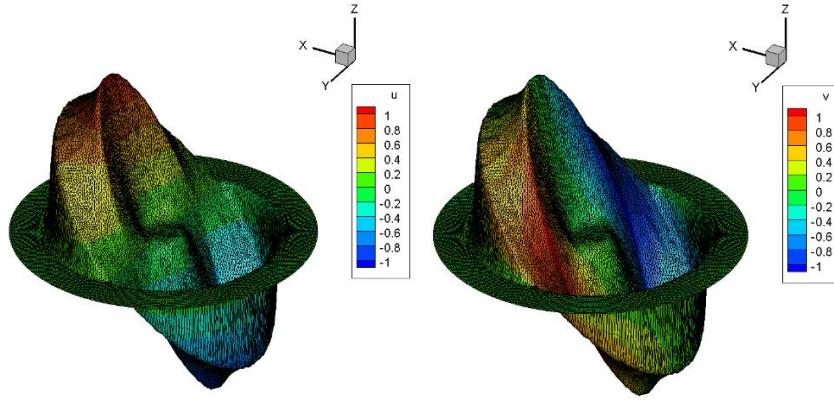


Figure 7. 3-D Plot of solution of fluid variables (density, pressure,  $u, v$ ) in the 2-D Explosion problem for  $t=0.2$ , HLLC solver with 2<sup>nd</sup> order MUSCL scheme, Barth and Jespersen Limiter.

The two-dimensional plots may reveal further information such as the effect of the Riemann Solver and the limiter on the solution, hence the fluid magnitudes of density, pressure and velocity (magnitude) will be represented in Figure 8.

Figure 8 confirms the initial hypothesis: there's -for  $t=0.2$ - a shockwave in  $r=0.2$ , followed by a contact wave at approximately  $r=0.58$  and ends with a rarefaction wave in  $r=0.72$ . The overall slope of both Rusanov and HLLC remain very close to the analytical solution, although the HLLC outstands as the more accurate. The explanation for this issue has been detailed in Appendix 1 where all the Riemann Solvers are detailed. Nonetheless, given the fact that the Rusanov flux is much simpler than the HLLC, its results are shown to be positive.

It must be said that the velocity plot represents is magnitude, given by the squared root of the summation  $u$  and  $v$  components each powered to two. In it, as it was expected, the effects of the shockwave and the rarefaction introduce their two corresponding and noticeable discontinuities.

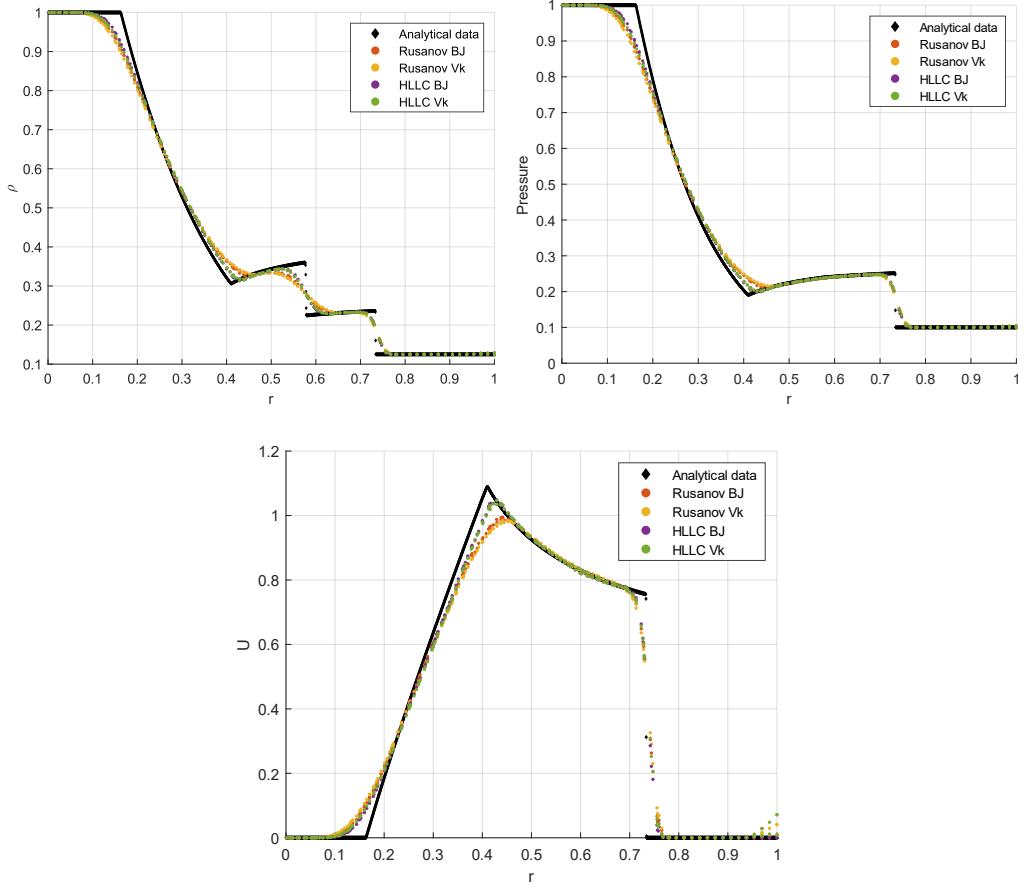


Figure 8. 2-D Plot of solution of fluid variables (density, pressure, velocity) in the 2-D Explosion problem for  $t=0.2$ . Comparison between different Riemann solvers implemented with Barth and Jespersen and Venkatakrishnan slope limiters.

Here, regarding the choice of the flux limiter, for this case, although the Venkatakrishnan seems to be the most accurate at some points, the Barth and Jespersen has provided the most satisfactory results. In order to summarize but also to support all the information explained, the norms for each Riemann solver with its corresponding limiter have been calculated in Table 1 in reference to the analytical data provided.

Table 1. Norm Computations for the different Riemann solvers and limiters.

	Norm 1	Norm 2	Norm 3
Rusanov BJ	0.0155	6.1076e-04	0.0896
Rusanov VK	0.0177	7.664e-04	0.0991
HLLC BJ	0.0111	3.6147e-04	0.0767
HLLC VK	0.0125	4.3849e-04	0.0825

In order to compare each of the Riemann solvers with the Analytical Reference solution, it is necessary to interpolate points such that the terms of the analytical solutions are equally distributed as the ones from the simulation. This procedure is done in Matlab.

As it can be seen, the errors remain relatively low thence the MUSCL 2<sup>nd</sup> order numerical scheme along with the Riemann Solvers and the limiters have demonstrated an accurate performance. Furthermore, the fact that Norm 2 (quadratic error) is much lower than

Norm 1 demonstrates the fact that most of errors are not uniformly distributed: they come from certain points.

Table 2. Computational cost for the 2-D Cylindrical Explosion Problem

	Total Time to run (in seconds)
Rusanov BJ	29.41
Rusanov VK	30.32
HLLC BJ	30.97
HLLC VK	33.94

Additionally, it is interesting to evaluate the computational cost (in time) as Table 2 displays. The difference is so small that it is clearly recommended, if accuracy is aimed, to compute the HLLC flux with the Barth and Jespersen limiter. In the same manner, the Venkatakrishnan Limiter seems to imply a larger computational cost than the Barth and Jespersen, although -for this benchmark problem- its results have been worse. Specially considering the information of both limiters detailed in Appendix 1, the less strict monotonicity preservation of the Venkatakrishnan limiter has shown slight oscillations that do not take place in the Barth and Jespersen limiter for both HLLC and Rusanov fluxes. These can be noted in the plot of stream velocity, just near the edges as the following figure reveals.

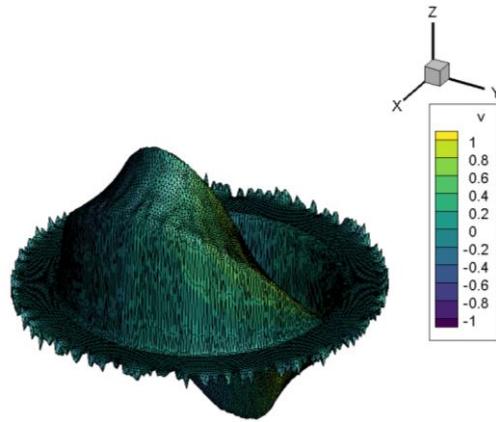


Figure 9. Slight oscillations of the Venkatakrishnan limiter for MUSCL scheme, HLLC and Rusanov Riemann solvers,  $t=0.2$

### 5.3 Forward Facing Step

This two-dimensional problem was early introduced by (Emery, 1968b), although the first publication to face it with the basic MUSCL scheme is (van Leer, 1979b). Besides, it is also faced in (Woodward and Colella, 1984) -which has been our reference publication for the whole section of results- with the same conditions that will be applied here, serving as a benchmark problem for the Riemann solvers and slope limiters implemented. Thence all the results obtained will be compared to their data. Besides, an interesting lecture is the PhD Thesis from (Pearson, 2012) which deals uniquely but in depth with this problem yet also the investigation of (Pearson, Goulart and Ganapathisubramani, 2011) where the Forward Facing Step -from now on, FFS- is evaluated in terms of turbulence.

The features and initial conditions of this problem have been explained in The Forward-Facing Step section.

Once the simulation is run, due to the supersonic flow nature and to the geometry, in the upper corner of the step a fan of rarefaction waves is produced and consequently causing this point to be singular. This fact is specially critical, since if it is not properly dealt computationally, the solution will be jeopardized by the introduction of significant numerical errors. Actually, that is what (Woodward and Colella, 1984) found out: no anterior publications apart from theirs considered this fact and subsequently unreliable results took place in the neighbourhood of the singular point- they state that a non-physical boundary region is originated which interacts with shocks and therefore the flow in the tunnel is fully compromised. As a solution, they proposed a procedure to minimize this issue: to introduce a particular yet artificial boundary condition next to the corner of the step that resets the fluid magnitudes and corrects velocities so as to satisfy the entropy and energy conservation.

Another problem -more physical than numerical- they noticed is a weak Kelvin-Helmhotz instability caused by the contact shock seen in the upper wall, specially visible at  $t=2.5$  and  $t=3$  of their simulation. It is suggested that it may be caused as a consequence of the presence of tiny oscillations in the values of entropy just behind the Mach shock. The oscillations are clearly a numerical effect but at the same time enlarged by the physical problem.

Once acknowledged all those problems, the simulation of the same benchmark problem is run for a mesh of 50705 elements for the Rusanov and HLLC Riemann problems with a 2<sup>nd</sup> order MUSCL scheme with Barth and Jespersen and Venkatakrishnan slope limiters for a CFL number of 0.7.

The results of the 2-D density distribution are collected in Figure 10, where the first image is the results from (Woodward and Colella, 1984). They obtained this result by applying a PPM -which showed the most accurate results in their investigation- scheme for a uniform grid of 19200 elements. The overall 2-D plots look very similar: there's the presence of 4 shock waves, a rarefaction fan in the corner of the step yet also a weak contact wave for the (Woodward and Colella, 1984) and HLLC Riemann solvers.

The geometry of the problem and the multiple reflections of the shockwaves leading the creation of them, show that, there's a singular moment where they reach such intensity that the  $\delta$  parameter of the shock wave is not sufficient to handle it and a Mach stem -Normal shock wave- is created between the two first shock waves. This wave, as it is expected, is

perpendicular to the upper surface and is correctly predicted in all the graphics. However, this phenomenon seems to happen again in the simulations for the HLLC and Rusanov fluxes -see the second middle point between waves- but did not appear in the results of (Woodward and Colella, 1984). Fluid variables seem to be less dissipated in our simulations. This fact could be caused by the different Riemann solver, the difference on the grid refinement or even due to a different treatment of the boundaries -see the special treatment of the boundaries done by (Woodward and Colella, 1984). Anyway, as they have discussed, the accuracy of this 2-D problem is not highly certain and relies in subjective criteria.

What can be noticed is the fact that Rusanov Riemann Solver -for both limiters- has not been able to predict the weak contact discontinuities sighted in (Woodward and Colella, 1984) and the HLLC solutions: one near to the upper corner, another even weaker near the step's late surface. They have been remarkably smeared within the flow. This fact shows the difficulty of this Riemann Solver to deal with weak contact discontinuities, which was actually one of the aims of the HLLC solver, to introduce their effect within the formulation and therefore solution of the problem.

The most useful resource to understand the flow of this problem is the movie or animation created for the density magnitude from  $t=0$  to  $t=4.0$ . It is a vivid proof of the information discussed here. This file will be submitted to blackboard along with other plots and animations for the different benchmark problems.

Furthermore, the pressure distribution over the problem will also be presented in Figure 10's next page, and will confirm the phenomena described previously. More illustrations, such as the U and V values of the velocity components will be uploaded to blackboard yet they do not provide further information than the already described.

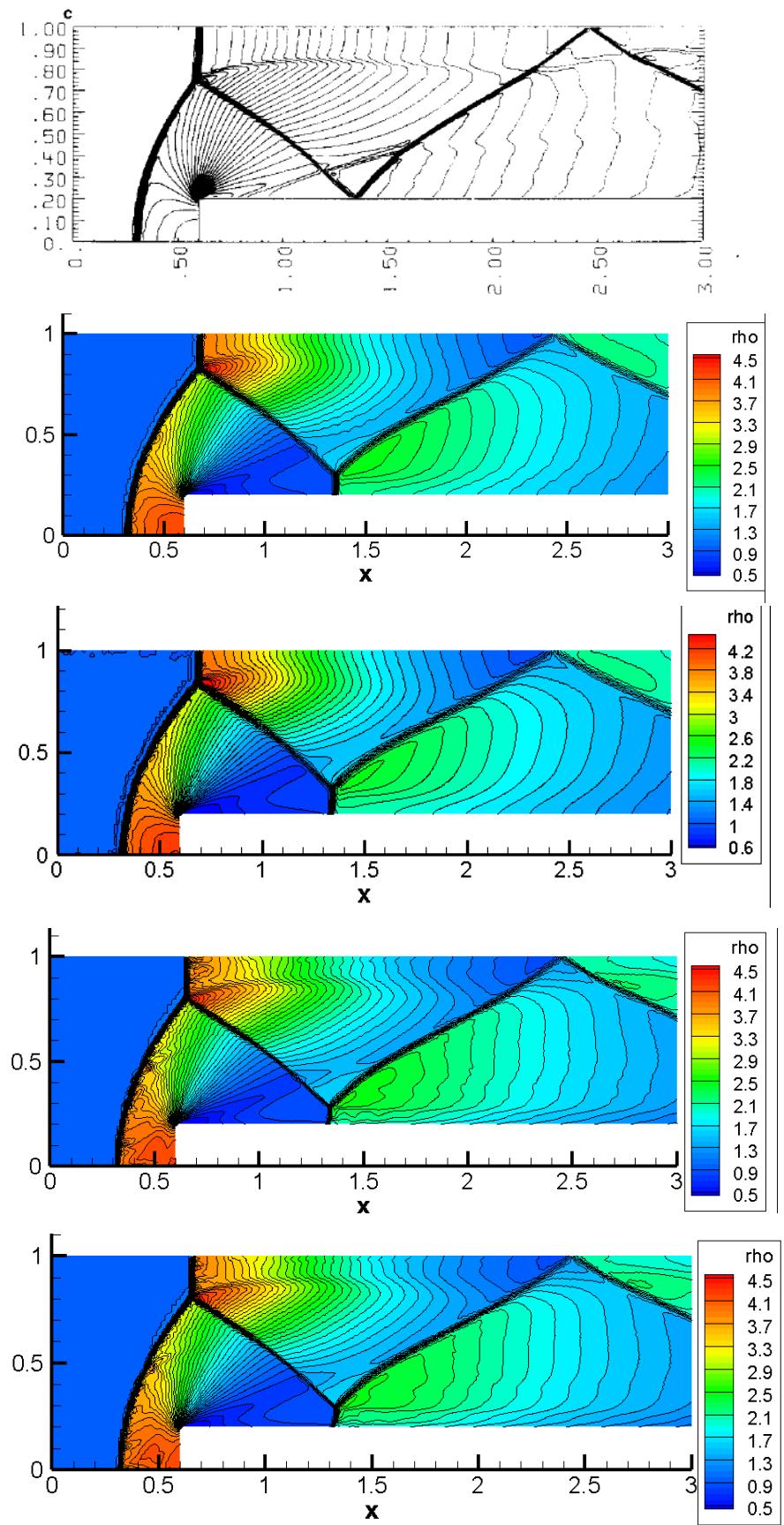


Figure 10. 2-D density distribution for the FFS problem ( $t=0.4$ ). From upside to the last graph: Results from (Woodward, Colella, 1984), Rusanov with Barth and Jespersen Limiter, Rusanov with Venkatakrishnan, HLLC Barth and Jespersen limiter, HLLC Venkatakrishnan limiter

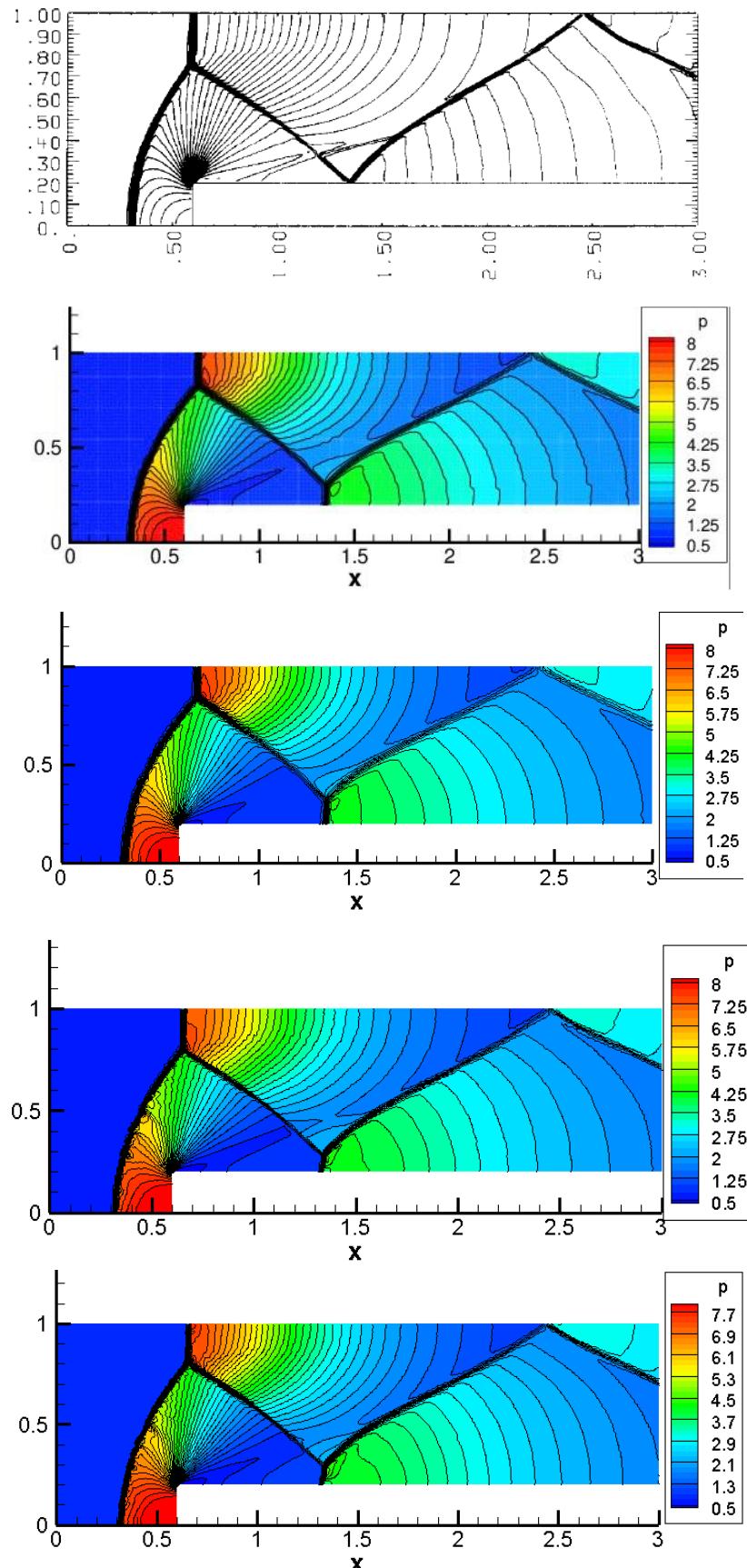


Figure 11. 2-D pressure distribution for the FFS problem ( $t=0.4$ ). From upside to the last graph: Results from (Woodward, Colella, 1984), Rusanov with Barth and Jespersen Limiter, Rusanov with Venkatakrishnan, HLLC Barth and Jespersen limiter, HLLC Venkatakrishnan

The only remaining interesting trait to remark for this problem is the computational cost comparison for the Riemann Solvers with the different limiters, visually attached in Table 3.

Table 3. Computational cost for the 2-D Forward Facing Step

Total Time to run (in hour:min:second format )	
Rusanov BJ	1:30:06
Rusanov VK	1:34:01
HLLC BJ	1:31:49
HLLC VK	1:35:01

The computational effort required to run it, forced the author of this document compute it with the HPC-Crescent. As it can be seen, no major difference in time has been noticed between both Riemann solvers. For this simulation, HLLC takes approximately 1 minute more than the Rusanov Flux, and Venkatakrishnan limiter adds, as an average, 4 minutes more to the ones expended in the Barth and Jespersen Limiter. For this reason, and regarding the results, the recommended choice is either the HLLC Riemann solver with the Barth and Jespersen limiter or the same with the Venkatakrishnan. Their differences, though, have been commented in Appendix 1.

#### 5.4 Double Mach Reflection – performed with the finer mesh

Finally, the last benchmark problem will also follow the assumptions from (Woodward and Colella, 1984), and the geometry and conditions of the problem have been already commented in The Double Mach Reflection section. Nevertheless, several publications have dealt with the same problem yet different numerical schemes such as (Colella and Glaz, 1985) using polytropic, Hansen and Gilmore Equation of States; (Kemm, 2016) where a discussion of a wide spectra of the artificial numerical phenomena that takes place within the solution as a consequence of initial and boundary conditions is detailed, but more recently, (Vevek, Zang and New, 2019) has dealt with the same problem implementing a 7<sup>th</sup>-order WENO scheme and described two procedures for setting up the initial and boundary conditions so as to overcome the non-physical errors significantly.

Similarly as it happened for the FFS problem, (Woodward and Colella, 1984) found out the requirement to implement especial boundary conditions in order to attach the shockwave to the wall. Besides, the left boundary region of the flow will introduce the post-shock flow whilst the right boundary will represent the outflow condition thence the gradients of all fluid variables will be set to 0.

The distribution of density for the (Woodward and Colella, 1984) PPM, which will be taken as reference, in conjunction with the results from simulation with 2<sup>nd</sup> order MUSCL schemes with the Riemann Solvers of Rusanov and HLLC with the Barth and Jespersen and Venkatakrishnan limiters will be collected in Figure 12. The pressure distribution will be equally plotted in Figure 13.

As the name indicates, a double Mach reflection takes place when flow faces the wall. Two Mach stems are formed, yet two contact waves, two reflected shock waves and the incident shock wave take place as well. Similarly to (Vevek, Zang and New, 2019), an schematic image

of the waves' structures formed within the simulation will be attached and properly detailed. Note that this image, such as the simulation plots, will be referred to the reflective wall's local axis.

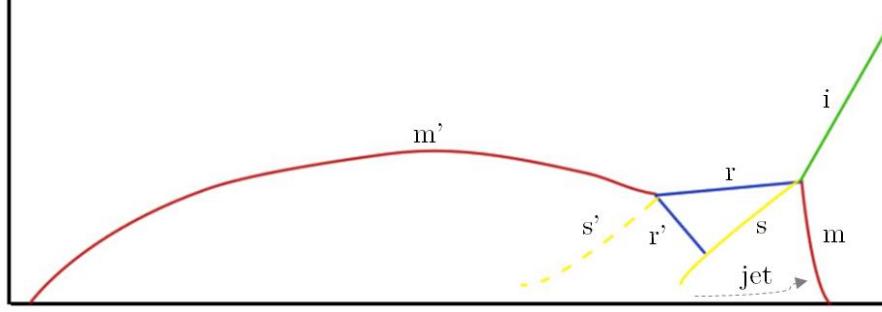


Figure 12. Schematic representation of the non-linear waves in the DMR problem.

Where  $m$  and  $m'$  refer to the primary and secondary Mach stem,  $r$  and  $r'$  represent the primary and secondary reflected shock waves,  $s$  and  $s'$  will identify the primary and secondary contact or slip waves, and finally  $i$  describes the location for the incident shock wave and an approximate trajectory of the jet stream produced is also represented.

As it has been remarked in (Woodward and Colella, 1984) and in (Vevek, Zang and New, 2019), the second contact discontinuity or slip line is actually so weak that it may be mainly noticed in the plot of the velocity components rather than in density. Thence, in Figure 12 note that it has been represented with a discontinuous line. We advance that in our density plots it will not be perceived. Besides, the second Mach stems is also weak and totally vanishes when it arrives to the first contact discontinuity originated by the first reflection. According to (Woodward and Colella, 1984) this last case is specially hard to numerically predict accurately.

The production of a jet that travels parallel and practically attached to the reflective wall is caused as a pressure gradient: it directs the denser fluid to the right along the wall's surface.

All this fluid behaviour will be seen in the following figures, where density and pressure will be plotted to compare the different slope limiters (Barth and Jespersen, Venkatakrishnan) and Riemann Solvers (Rusanov, HLLC).

As it can be perceived in Figure 12, there's a difference in the density values predicted by both Riemann Solvers. Density seems to acquire higher values in the vicinity of the secondary Mach stem for the HLLC solver, but also slight numerical artifacts can be noted in the external region where the upside flow has constant value of approximately  $\rho=8$ . Furthermore, what most solvers and schemes predicted a single primary mach stem (denoted as  $m$  in Figure 12), HLLC seems to capture two slightly different stems which may be influenced by a possible overprediction of the jet that takes place before. However, as it is said in (Vevek, Zang and New, 2019), we do not have a strict procedure to determine the influence of numerical artifacts and inaccuracies except for simple comparison with high fidelity simulations and also reference of analytical experiments from (Ben-Dor, 2007). Finally, the secondary contact wave or slip line that can be intuited in the first two reference images, has not been captured in our simulations.

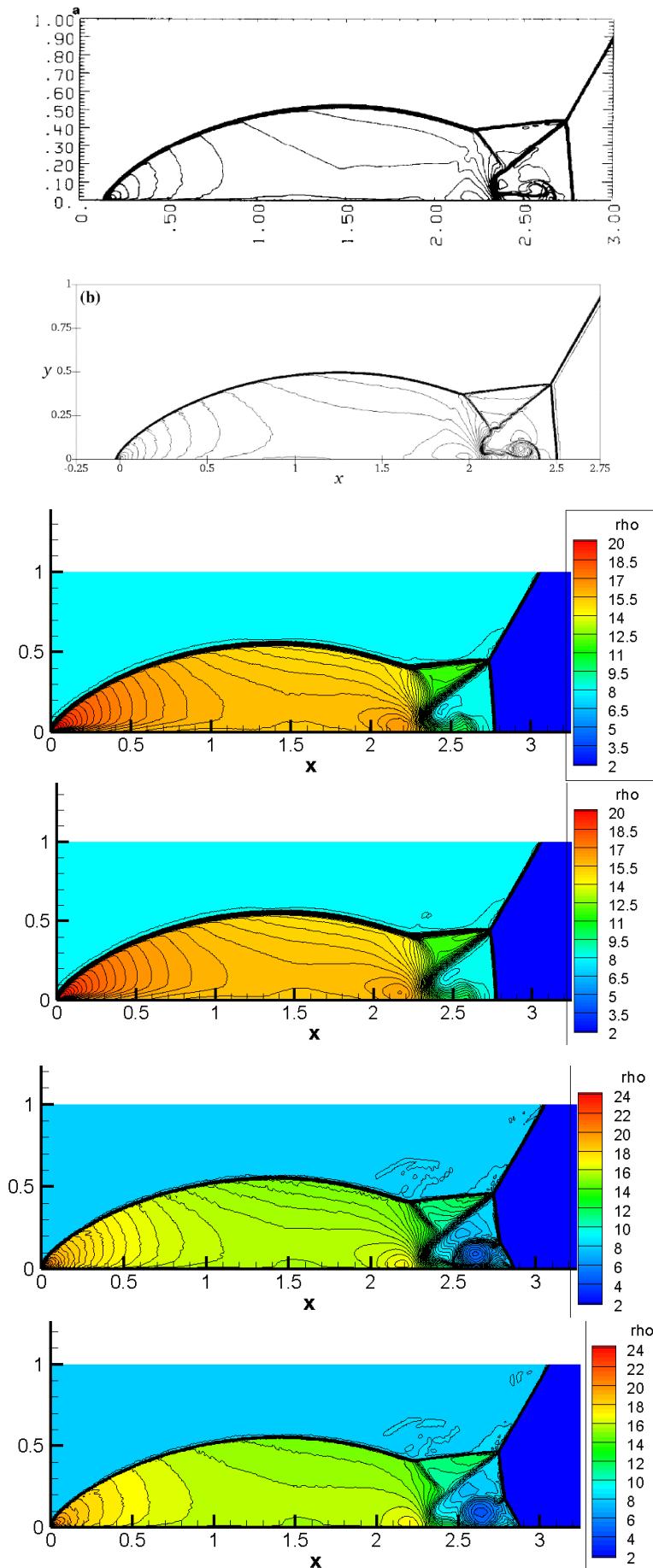


Figure 13. 2-D density distribution for the DMR problem ( $t=0.2$ ). From upside to the last graph:  
 Results from (Woodward, Colella, 1984), (Vevek, Zang and New, 2019) Rusanov with Barth and  
 Jespersen Limiter, Rusanov with Venkatakrishnan, HLLC Barth and Jespersen limiter and  
 HLLC Venkatakrishnan

Finally, the last important feature that can be sighted is the difference between both limiters: although they do not change the minimum and maximum values of the density, the Venkatakrishnan slope limiter seems to smoothen the flow more than Barth-Jespersen does.

Similar results can be noticed from Figure 14, where the pressure distribution is evaluated in the same manner. As it was expected, due to their features, the contact or slip discontinuities do not have any effect on the pressure.

Again, the HLLC Riemann solver predicts higher pressure extrema than the Rusanov flux by a considerable amount, an approximately 8%. Nonetheless, it captures better the difference of pressures that provoke the appearance of the already mentioned jet. In all the plots, no evident numerical artifacts are perceived.

Additionally, as they remain important for the study of the weak contact discontinuities, reference distributions for the  $u$ , and  $v$  components of (Woodward and Colella, 1984) and the ones obtained by our simulations -specially the HLLC flux with Barth and Jespersen Limiter – which is considered to be a strong solution, are presented in the following image:

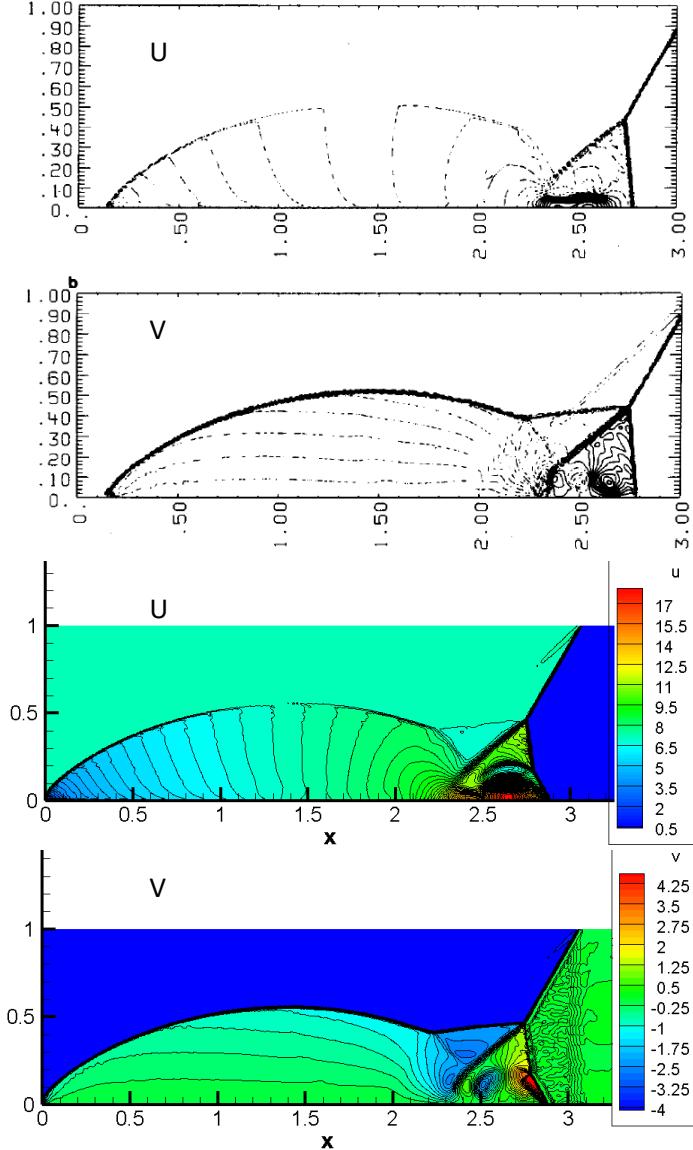


Figure 14. U and V velocity components comparison between(Woodward and Colella, 1984) and HLLC Barth and Jespersen

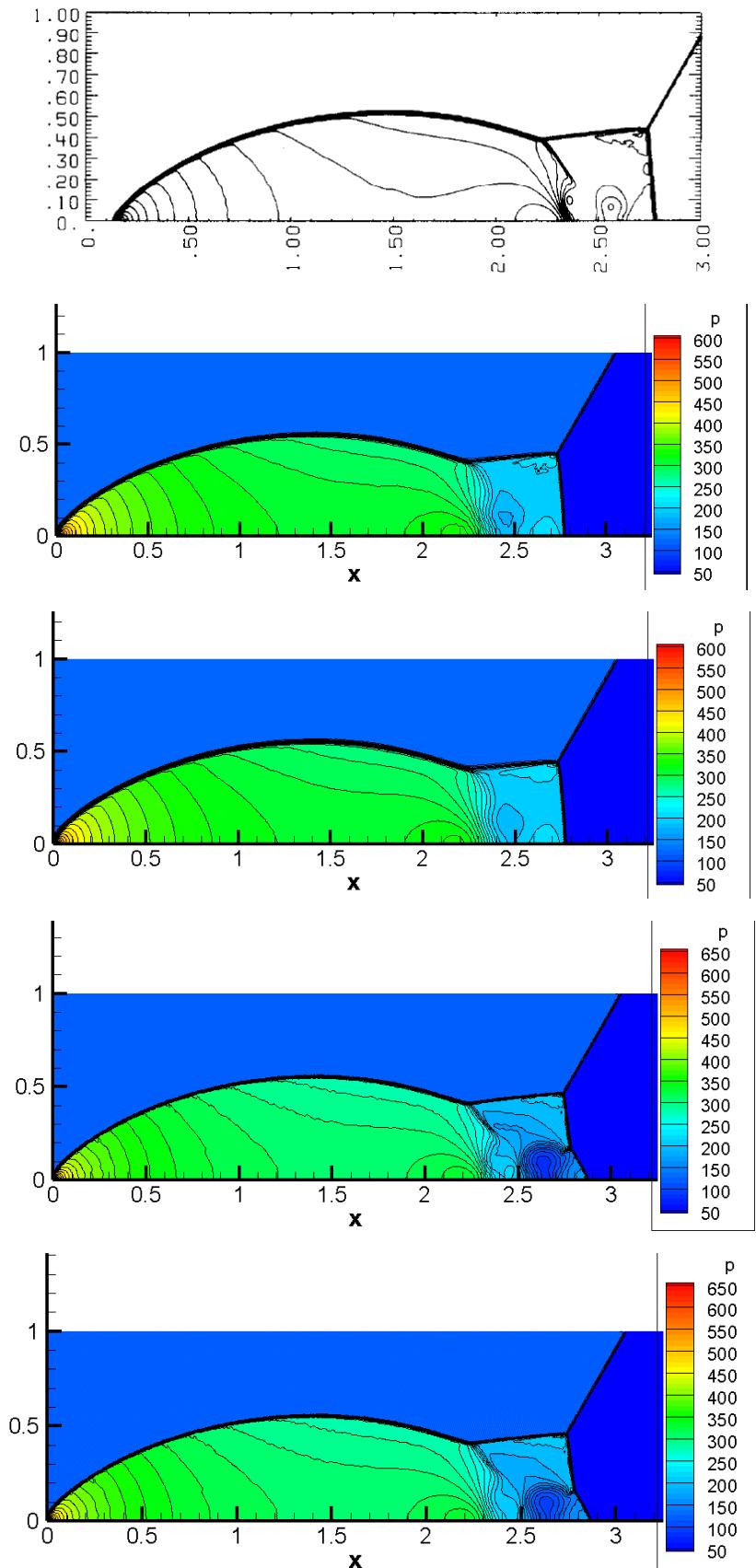


Figure 15. 2-D density distribution for the DMR problem ( $t=0.2$ ). From upside to the last graph: Results from (Woodward, Colella, 1984), Rusanov with Barth and Jespersen Limiter, Rusanov with Venkatakrishnan, HLLC Barth and Jespersen and HLLC with Venkatakrishnan limiters.

The velocity contours give an estimate idea of the slip discontinuities: the primary is really evident but the secondary can be only grasped. Additionally, in the U,V plots of the HLLC Riemann solver but also in the V plot of (Woodward and Colella, 1984) a tiny non-physical effect can be noted in the upper region of the incident shock wave. Also, the jet is better seen in this graphic.

As it has been said, all the benchmark problems analyzed have been under a MUSCL scheme of 2<sup>nd</sup> order of accuracy. Now, in order to determine the strengths and weakness of the schemes chosen, a 5<sup>th</sup> order MUSCL scheme will be compared to a 5<sup>th</sup> order WENO scheme -for spatial accuracy-. The basic details of both schemes have been explained in Appendix 1. Besides, both solutions will be contrasted with the results from (Vevek, Zang and New, 2019) where a 7<sup>th</sup> order WENO scheme is implemented.

After submitting the WENO 5<sup>th</sup> order scheme for 2 days to Crescent and collapsing it (even when the executable *a.out* had “debug options” disabled) as the computation required more than 158 GB of RAM according to Dr. Michael Knaggs, we obtained the results for time =0.1229 s after 48 h of computation (and approximately 12000 time steps) . However, as it was predictable, results from this simulation have not been satisfactory. Thence, the results for the same time cannot be contrasted for MUSCL 5<sup>th</sup> order scheme and fine mesh.

Once seen the movie of the whole simulation (WENO 5<sup>th</sup> order, HLLC) from t=0 to t=0.1229 which was the last time step achieved for the 48h simulations, a really strange phenomenon is sighted: before the shock structure reaches the end -as it was supposed- an explosion of “infinite” values take place in the outflow and expands towards the inflow collapsing all the density values and turning it to infinite values. This fact explains the fact of why it has been too slow: a huge mesh of values close to infinite for 4 fluid variables evolving in time and space requires too much memory and computational cost. We guess that is caused by a non-proper right boundary condition applied (outflow) such that an accumulation of propagated errors -maybe as a result of the special boundary conditions to be applied- take place and burst into the outflow as the shock structures evolve and push them forward.

Our next proposal was to then compare the results for both schemes using Rusanov Riemann solver and mesh 2, since mesh 1 is too coarse. It may be inconsistent to compare a 5<sup>th</sup> order for a coarse mesh once 2<sup>nd</sup> order results have been obtained for a finer mesh and they look more accurate as it will be seen soon. Nevertheless, for the WENO 5<sup>th</sup> order scheme the same problem is faced again: propagation, at a certain time previous to 0.2, of “infinite” values from the outflow to the inflow. The error, as expected, did not come from the Riemann solver. It has to be related to the procedure of the WENO scheme. Results from MUSCL have been successful for each Riemann solvers and order of accuracy.

The video for both simulations (5<sup>th</sup> order WENO for HLLC and Rusanov Riemann Solvers) will be provided as a proof and in order to check if some undetected error is made.

Thus, the simulations have been run using Rusanov and HLLC Riemann Solvers with 5<sup>th</sup> order MUSCL and WENO schemes for mesh 2 -the finer-, and finally for HLLC Riemann Solver for the coarser mesh -mesh 1- with the same schemes yet results strongly suggest that a refinement is absolutely required. As said, they are less accurate than the ones from 2<sup>nd</sup> order fine mesh displayed above.

The only available option is then -although it is inconsistent to evaluate higher order scheme with significantly coarser mesh- compare both 5<sup>th</sup> order schemes from mesh 1. First, the density distribution will be graphed in Figure 16.

As Figure 16 reveals, the 5<sup>th</sup> order WENO scheme is not able to correctly propagate the information of the fluid variable over time and space probably as a result of the coarse mesh. A wide variety of numerical artifacts can be detected, which is totally undesirable. Although the 5<sup>th</sup> order MUSCL did display the overall behavior in a more smooth and accurate manner, its results still are worse than the previously obtained by a 2<sup>nd</sup> order MUSCL scheme but in a finer mesh. As a result, as it is already advised, a further refinement should be done in the mesh. However, it would imply a considerably higher computational effort specially for the WENO scheme which would not make much difference if it is ran in Crescent. Our suggestion for this case would be the hard task of parallelizing efficiently the full code.

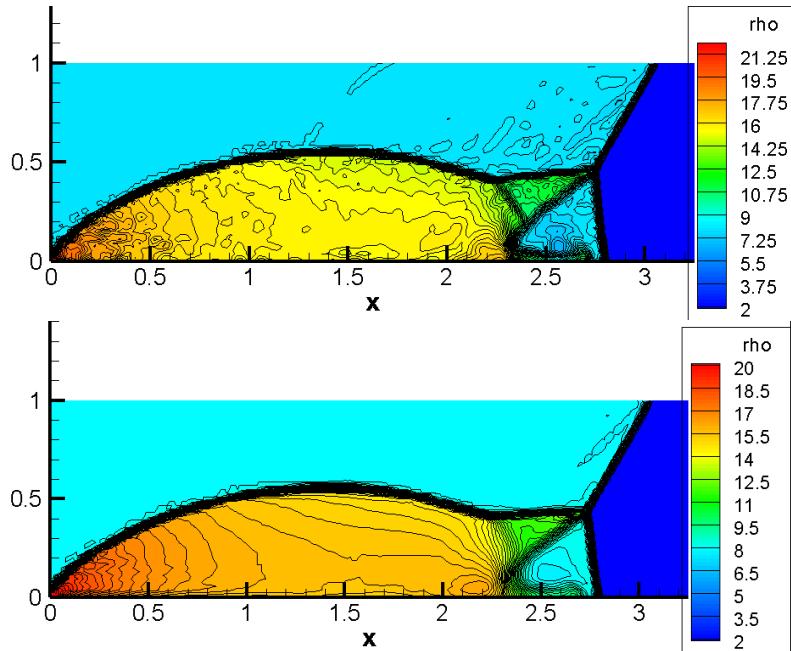


Figure 16. 2-D density distribution for the DMR problem ( $t=0.2$ ). Upper image: 5<sup>th</sup> order WENO, Lower Image: 5<sup>th</sup> order MUSCL scheme with Barth and Jespersen Limiter

Actually, although it has demonstrated worse overall performance, which is not expected at all as it is explained in Appendix 1: High Resolution Shock Capturing Methods, the WENO scheme did capture with greater intensity the discontinuities (including the jet) and even starts to display the secondary contact discontinuity with more presence than in the 5<sup>th</sup> order MUSCL scheme, where it is barely appreciated. The plot of the pressure distribution of Figure 17 will not add much further information than the one already commented, as the conclusions deducted remained the same.

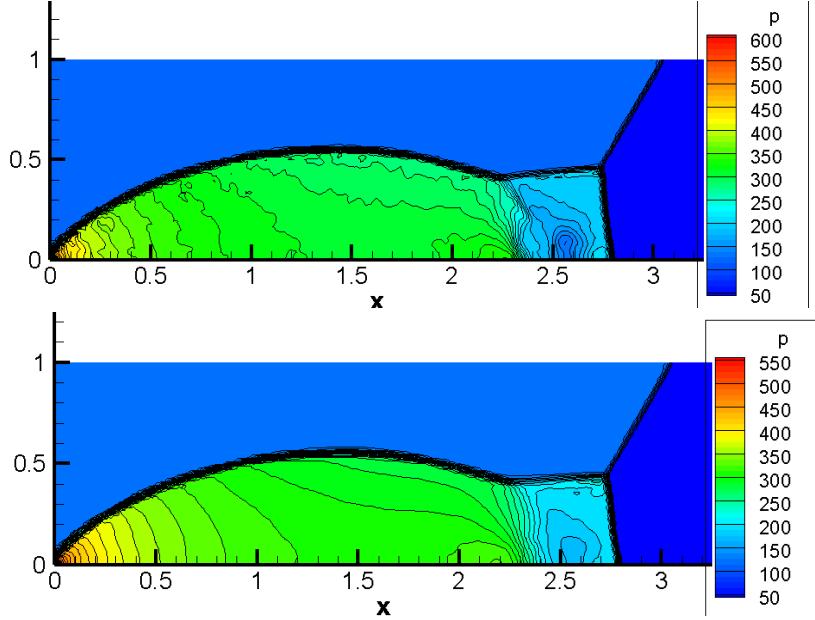


Figure 17. 2-D pressure distribution for the DMR problem ( $t=0.2$ ). Upper image: 5<sup>th</sup> order WENO, Lower Image: 5<sup>th</sup> order MUSCL scheme with Barth and Jespersen Limiter

The fact that for this case the WENO scheme performed worse than the MUSCL scheme was not expected at all and may be caused as a mesh sensitivity when representing the stencil. Besides, a proper setup of the boundary conditions as the last ones proposed by (Vevek, Zang and New, 2019) should make a difference.

The advantages of WENO schemes have been detailed in Appendix 1, as they are usually more accurate than MUSCL schemes -which need a limiter- due to their stencil methodology for obtaining the solution. They specially standout in cases where complex geometries and strong discontinuities structures take place, where MUSCL schemes struggle. However, as (Mastorakis and Edisson, 2014) say, MUSCL schemes are recommended when quick and general results are aimed for a simulation.

Results from a 7<sup>th</sup> order WENO scheme for this problem have been obtained by (Vevek, Zang and New, 2019) with a mesh of 272000 elements and HLL Riemann solver, whilst ours has been of 16400 cells for the coarse and 152272 for the fine mesh, where the WENO 5<sup>th</sup> order quite collapsed the Crescent computer and did not finish the simulation even after 48 h. Their results will be shown in Figure 18. The first image a slight numerical artifact is detected next to the upper right corner, just next to the incident shock wave. This is due to the conventional setup of the problem, proposed by (Woodward, Colella, 1984). As a result of their investigation, two more setups that reduce this problem have been developed, tested and detailed. As it can be seen, with the first of their two setups this issue is no longer seen. Then, as a further enhancement of our code, it could be positive to test their proposals for treating conveniently the boundaries.

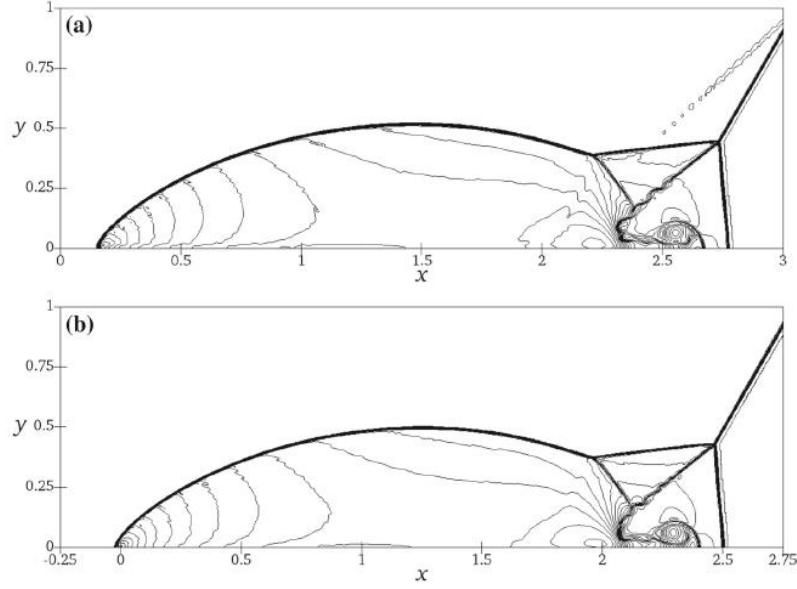


Figure 18. 2-D distribution of density for WENO 7th order scheme in DMR problem. (a): Conventional Setup, (b): First setup suggested. Source: (Vevek, Zang and New, 2019)

The only remaining aspect to evaluate is the computational cost of each scheme, limiter and Riemann solver for the simulations presented for the Double Mach Reflection benchmark problem. They have been collected in

Table 4. Computational cost for the DMR problem of different schemes, Riemann solvers and limiters

	Total time to run (in hours:minutes:seconds)
2 <sup>nd</sup> Order MUSCL - Rusanov Barth and Jespersen	02:10:08
2 <sup>nd</sup> Order MUSCL - Rusanov Venkatakrishnan	03:01:12
2 <sup>nd</sup> Order MUSCL - HLLC Barth and Jespersen	03:08:19
2 <sup>nd</sup> Order MUSCL - HLLC Venkatakrishnan	03:21:14
5 <sup>th</sup> Order MUSCL – HLLC Barth and Jespersen*	00:05:06
5 <sup>th</sup> Order WENO – HLLC*	00:14:06

Note that \* refers to coarse mesh instead of fine mesh.

Table 4 demonstrates the great influence of the mesh resolution for the computational time: mesh 1 is composed by 16400 cells and the fine 152272 cells. Given the results, the most effective simulation has been the Rusanov flux with the Barth and Jespersen Limiter. Even though it is not the best solver -their results are not the most accurate, being the HLLC the most reliable in most of cases- their difference in computational cost is so huge (at least 30% less than the others) that if higher precision is aimed, it would seem logical to compute higher order MUSCL schemes with the same Riemann solver and flow limiter.

## 6. Conclusions

In this document, four Riemann Solvers (Lax and Friedrich, Rusanov, HLL, HLLC) and three slope limiters (minmod, Barth and Jespersen, Venkatakrishnan) have been implemented to a FORTRAN solver code and evaluated in three different benchmark problems (Cylindrical Explosion, Forward Facing Step and Double Mach Reflection) on 2-D unstructured but also the cylindrical explosion for 1-D structured meshes.

In order to understand the solver suitable for compressible flows, an introduction of the Godunov's method to solve hyperbolic PDEs where the Riemann problem is faced is given, providing special attention of how it may be implemented for 1-D and 2-D structured and unstructured meshes. Furthermore, in Appendix 1, an introduction regarding high order schemes (Monotonicity, TVD, MUSCL and WENO schemes, slope limiters) but also the Riemann solvers implemented (Lax and Friedrichs, Rusanov, HLL, HLLC) is clearly detailed.

Then, a brief description of the 4 possible non-linear waves present at compressible flows (rarefaction, compression, contact discontinuity and shockwave) is given in order to clearly perceive them in the plots of the results. Special attention is given to clarify how fluid variables vary across each wave.

As the problem can be approached through Finite Element Methods, Finite Volume Methods or Finite Difference Methods, the Finite Difference Approaches will be reviewed remarking their simplicity and easiness to implement but also their bad performance to face curvilinear geometries, unstructured meshes, non-steady boundaries and also the few WENO schemes available to be implemented within this approach with different Riemann solvers is conveniently remarked as its main drawbacks.

The results have been compared to reference solutions from (Woodward and Colella, 1984) and other similar publications except for the Cylindrical Explosion, in 1-D and 2-D, which is analyzed regarding an analytical result given by Dr. Panagiotis Tsoutsanis. They have been compared and analyzed in terms of accuracy, dissipation/dispersion (according to the order of the scheme), computational cost and overall performance considering all these factors.

For the 1-D problem, a grid convergence is appreciated as three meshes (51, 101 and 201 cells) have been used. The diffusion appreciated in first order of MUSCL scheme even for the finer demonstrate the need of a higher order of spatial accuracy, thence 2<sup>nd</sup> order spatial scheme is also evaluated, and by this procedure results have improved to a great extent. The performance of the Riemann solvers has been analyzed and equally for both orders of accuracy the best solutions were obtained by HLLC, followed by HLL and consequently Lax and Friedrich performed the worst results.

For the 2-D problems (Cylindrical Explosion, DMR, FFS) a similar methodology has been applied, yet here only two Riemann solvers have been compared (Rusanov, HLLC) with the same MUSCL scheme of 2<sup>nd</sup> order with the slope limiters of Barth and Jespersen and Venkatakrishnan. For all the cases, the HLLC solver provided more accurate results without incurring in a higher cost of computation (simulations took approximately 1 minute more for the same limiter implemented, except for the DMR problem). The only issue noted is for the DMR, an additional Mach stem seems to be created that is not documented in other literature

review and could be caused artificially as an overprediction of the jet's nature or a non-appropriate treatment of the boundary conditions. Additionally, as it has been detailed in Appendix 1, Barth and Jespersen provides better results than Venkatakrishnan possibly because of the stricter monotonicity restraint, although it is not differentiable. Slight oscillations with the Venkatakrishnan limiter have been perceived particularly in the velocity plots of the 2-D Explosion case for both Riemann Solvers (Rusanov, HLLC). Furthermore, even Venkatakrishnan was supposed to emphasize on convergence of results, the reality we have faced was that it required more time (averagedly 4 minutes more). However, it could be caused by a non-efficient implementation compared to the Barth and Jespersen.

The non-linear waves, according to the literature reviewed, have been correctly identified in all the simulations and described in the text and through spatial plots. Besides, their evolution in time is sighted through an animation outputted with the Tecplot software, which has been the program used to carry out all the post processing of the simulation.

Additionally, a 5<sup>th</sup> order scheme comparison of the DMR problem with MUSCL and WENO schemes has been done to identify their differences, which have been theoretically commented in Appendix 1. As a result, for the finer mesh of DMR. As it has been proved, the WENO results for the finer mesh have not been achieved because of a propagation of infinite-values as the movie file demonstrates, yet the MUSCL performed accurately within a relatively short time (approximately 1h and 44 minutes). Nevertheless, they have been obtained for the mesh 1, which is the coarsest of the two provided. The WENO scheme has been able to predict with higher intensity the shock structures; however, the general flow is less smooth than the one obtained with the MUSCL scheme. These results may confront the theoretical results expected as it has been commented in Appendix 1. A further comparison with a 7<sup>th</sup> order WENO scheme with HLL Riemann solver for the same problem performed by (Vevek, Zang and New, 2019) is carried out, as they propose two different setups in order to largely reduce the numerical artifacts that take place as a result of inappropriate boundary treatment.

This document, as it is believed by the author, details satisfactorily the basis of how Compressible Flows must be approached computationally, a brief summary of Godunov's theory, high-order schemes and Riemann solvers; especially when discontinuities are faced within the flow field. The codes implemented seem to be working adequately. The list of requirements for this assignment is thought to be fulfilled. However, there are a lot of possibilities as future work to improve it or to extend it to a wider field. Recent setups for boundary conditions could be tested for the same benchmark problems, as well as more problems could be run in order to detail the performance more completely. The WENO scheme issue should be fixed, probably with one of those setups. Even physical real problems of engineering could be evaluated. Besides, other Riemann Solvers and limiters for MUSCL Schemes, could also be implemented and then contrasted.

## 7. References

- Anderson, J.D. (1995) *Computational fluid dynamics : the basics with applications*. McGraw-Hill.
- Barth, T. and Jespersen, D. (1989) “*The design and application of upwind schemes on unstructured meshes,*” American Institute of Aeronautics and Astronautics (AIAA).
- Ben-Dor, G. (2007) *Shock Wave Reflection Phenomena*. Berlin, Heidelberg: Springer Berlin Heidelberg, Shock Wave and High Pressure Phenomena.
- Chen, G., Tang, H. and Zhang, P. (2008) “Second-order accurate Godunov scheme for multicomponent flows on moving triangular meshes,” *Journal of Scientific Computing*, 34(1), pp. 64–86.
- Colella, P. and Glaz, H.M. (1985) *Efficient Solution Algorithms for the Riemann Problem for Real Gases*.
- Davis, S.F. (1988) *SIMPLIFIED SECOND-ORDER GODUNOV-TYPE METHODS*.
- Emery, A.F. (1968a) “An evaluation of several differencing methods for inviscid fluid flow problems,” *Journal of Computational Physics*, 2(3), pp. 306–331.
- Godunov, S. (1959) “*A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics*”
- Harten, A., Engquist, B. and Osher, S. (1987) *Uniformly High Order Accurate Essentially Non-oscillatory Schemes, I II*.
- Harten, A., Lax, P.D. and van Leer, B. (1983) *ON UPSTREAM DIFFERENCING AND GODUNOV-TYPE SCHEMES FOR HYPERBOLIC CONSERVATION LAWS\**.
- Henrie, M., Carpenter, P. and Nicholas, R.E. (2016) “Statistical Processing and Leak Detection,” in *Pipeline Leak Detection Handbook.* , pp. 91–114.
- Iftekhar, H. (2016) *Experimental and Numerical Studies of Flows over Forward Facing Steps in Pressure Gradients*.
- Iftekhar, H. and Agelin-Chaab, M. (2015) “*Reynolds number effect on forward facing step in a pressure gradient Thermal analysis and management View project Renewable Energy System View project*”
- Kemm, F. (2016) “On the proper setup of the double Mach reflection as a test case for the resolution of gas dynamics codes,” *Computers and Fluids*, 132 Elsevier Ltd, pp. 72–75.
- Landau L.D. and Lifshitz E.M. (1987) “Fluid Mechanics,” *Institute of Physical Problems, USSR Academy of Sciences*, 6
- van Leer, B. (1979) “Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method,” *Journal of Computational Physics*, 32(1), pp. 101–136.
- Mastorakis, N. and Edisson, S. (2014) *Comparison Among MUSCL, ENO, and WENO Procedures as Applied to Reentry Flows in 2D – Short Report*.

- Michalak, K. and Ollivier-Gooch, C. (2008) *Limiters for Unstructured Higher-Order Accurate Solutions of the Euler Equations.*
- Pearson, D.S. (2012) *Characterisation and estimation of the flow over a forward-facing step.*
- Pearson, D.S., Goullart, P.J. and Ganapathisubramani, B. (2011) “Investigation of turbulent separation in a forward-facing step flow,” *Journal of Physics: Conference Series*. Institute of Physics Publishing, Vol.318.
- Scheit, C., Esmaeili, A. and Becker, S. (2013) “Direct numerical simulation of flow over a forward-facing step - Flow structure and aeroacoustic source regions,” *International Journal of Heat and Fluid Flow*, 43, pp. 184–193.
- Shu, C.-W. (2009) “High Order Weighted Essentially Nonoscillatory Schemes for Convection Dominated Problems,” *SIAM Review*, 51(1), pp. 82–126.
- Toro, E.F. (2009) *Riemann Solvers and Numerical Methods for Fluid Dynamics Third Edition.*
- Toro, E.F. (1992) *The weighted average flux method applied to the Euler equations.*
- Toro, E.F. (2019) The HLLC Riemann solver *Shock Waves*. Springer,
- Toro, E.F., Spruce, M. and Speares, W. (1994) *Restoration of the contact surface in the HLL-Riemann solver*. Springer-Verlag.
- Venkatakrishnan, V. (1993) “*On the Accuracy of Limiters and Convergence to Steady State Solutions*”
- Venkatakrishnan, V. (1994) “*Convergence to Steady State Solutions of the Euler Equations on Unstructured Grids with Limiters*”
- Vevek, U.S., Zang, B. and New, T.H. (2019) “On Alternative Setups of the Double Mach Reflection Problem,” *Journal of Scientific Computing*, 78(2) Springer New York LLC, pp. 1291–1303.
- Woodward, P. and Colella, P. (1984) *The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks.*
- Zanotti, O. and Manca, G.M. (2010) “A very short introduction to Godunov methods,” *COMPSTAR School on Computational Astrophysics, 8-13/02/10, Caen, France*
- Zhang, Z.-X. (2016) “Shock Waves,” in *Rock Fracture and Blasting* , pp. 39–66.
- Zheng, H.W. (2013) “A new limiter for hybrid grid,” *Procedia Engineering*. Elsevier Ltd, Vol.67, pp. 430–437.

## 8. Appendix 1: High Resolution Shock Capturing Methods

### 8.1 Principles of Higher Order Schemes

The Godunov's exact method provides first order of spatial and time accuracy which may not be sufficient for capturing complex fluid phenomena or to perform real engineering computations. Thus, the approximate Riemann Solvers and high order schemes have been developed afterwards, some of them basing its core in Godunov's approach.

By experience, all higher order linear schemes developed are prone to present artificial ("spurious") oscillations in its results -serve, as an example, the central scheme- when large gradients of fluid variables take place. Nevertheless, it is proven that Monotonous schemes do not suffer from this behaviour. The only drawback from them is the fact that, as Godunov demonstrated, they are only 1<sup>st</sup> order accurate as well. Thence the importance of developing non-linear schemes.

The current methodology relies in a merge of Godunov-type methods – which perform greatly from conservation equations for treating discontinuities- with other numerical schemes of higher order methods for dealing with smooth regions of the flow. Thence, the higher order of accuracy is obtained in the solution. Examples of it are the MUSCL and WENO schemes that will be reviewed later.

One of these methods is the TVD (Total Variation Diminishing), which imposes that no spurious oscillations must be produced in time evolution. In another manner, according to (Manca), the mathematical formulation of it is that “the total variation in each time is uniformly bounded by the total variation of the initial data”. This principle is inherited within the formula of

$$TV(Q^n) = \sum_{-\infty}^{\infty} |Q_i^n - Q_{i-1}^n| \rightarrow TV(Q^{n+1}) \leq TV(Q^n). \quad (11)$$

However, TVD usually do not provide higher order of accuracy than 2<sup>nd</sup> due to its restrictive condition, thence for achieving 3<sup>rd</sup> or higher order of accuracy this condition must be given up. As ((Toro, 2009) discusses, we must allow some Time Variation proportional to some power of the step size. From this point of view, the ENO (Essentially Non-Oscillatory) schemes are created. In the other hand, other higher order schemes such as the MUSCL schemes (Monotonous Upwind Schemes for Conservation Laws) will base their functioning to monotonicity properties – “monotone schemes do not produce spurious oscillations”- the reconstruction procedure and the adjustment of Godunov's piece-wise constant data into a linear function so that higher order of spatial accuracy is obtained.

The MUSCL scheme was introduced by (van Leer, 1979) with the idea to substitute the constant piecewise approximation suggested by Godunov by a linear function that is dependent on a reconstruction, in which the values are obtained through cell-average of the immediately previous time step. The monotonicity principle of the solution by using this scheme relies in the implementation of non-linear slope limiters (such as minmod, Barth and Jespersen, Venkatakrishnan) as they limit the amplitude of the gradients that take place within the solution and thence no local extrema is created, which is one of the basic requirements of monotonicity. In each cell, the reconstruction of the left and right states are computed so as to then obtain the variables' fluxes at the intercells. In other words, an

extrapolation between adjacent cell averages will give us the state variables at the intercells. The fluxes calculated can be then introduced to a Riemann solver to acquire an averaged solution and advance it in time. Extensions of MUSCL schemes have been introduced such as the PPM of (Woodward, Colella, 1984).

On the other hand, ENO (Essentially Non-Oscillatory) schemes have been developed quite recently – firstly introduced by (Harten, Engquist and Osher, 1987)- and rely on the technique of using an adaptive stencil. An algorithm that is applied to each mesh cell establishes which region of the neighbouring cell flow is the smoothest, and is later used -as (Mastorakis and Edisson, 2014) say- to build a high order accurate, conservative interpolation to calculate the flow variables at the intercells. The spurious oscillations that are present at the discontinuities are largely minimize by imposing the priority of using data from the smoothest regions. The typical configuration is then to use 3 stencils englobing different neighbouring cells-but can be more- and then select the one that gives smoothest results.

A variation of this idea is produced with the introduction of WENO -Weighted Essentially Non-Oscillatory Schemes- where the x -let's say, 3- stencils are at the same time used and the solution is obtained by a weighted average of the them, giving more power to the smoother. Nevertheless, if any of the stencil is found to have discontinuities, their corresponding weight is set to nearly 0 and also the total sum of weights must be equal to 1. By this procedure, as remarked in(Shu, 2009) in the best-case scenario, an improvement of the ENO procedure is perceived as from a  $k$ th order of accuracy of ENO a  $(2k-1)$ th order of accuracy WENO can be obtained. Furthermore, it overcomes the deficiencies of ENO schemes such as the freely adaptive stencil that is not required for smooth regions but also the fact that this stencil, according to (Mastorakis and Edisson, 2014), “could change even by a round-off perturbation near zeroes of the solution and its derivatives”. ENO schemes are not efficient in vectoral supercomputers either as they face their limitation: a vast amount of logical statements is required in the ENO procedure to select the best stencil.

We have commented the need of slope limiters for MUSCL schemes to inherit the monotonicity properties in order to avoid oscillations when discontinuities are faced. In 8.3 the Venkatakrishnan and Barth and Jespersen limiters will be discussed. But, before, in the following section the Riemann solvers will be commented to discuss how fluxes at the intercell are calculated and therefore the Riemann problem is approximately solved.

## 8.2 Flux Approximations: Riemann Solvers

The fluxes from (1), for unstructured grids and Approximate Flux Riemann solvers must be obtained within the finite volume (in the intercell of reference and adjacent “triangles” or another geometry of unitary cells) defined with a numerical approach that will update the flux at every iteration in time and in space. Here, a brief description of Lax-Friedrich flux, Rusanov Flux, HLL flux and HLLC flux will be provided.

### 8.2.1 Lax-Friedrichs Flux

It is the simplest scheme for computing the Riemann Problem’s fluxes of the ones described here, but also the most dissipative and less accurate. Its main advantage is the easiness to

implement. The upwind principle is imposed within each cell edge, thence the Riemann problem is here faced again with the discontinuity of the conserved variables  $\mathbf{U} = \begin{cases} \mathbf{U}_L \\ \mathbf{U}_R \end{cases}$ , and therefore  $F(U_x) = F(U_L(t), U_R(t))$ . Besides, to satisfy that there's no interaction between the neighbouring local Riemann Problems, the maximum wave speeds are set to a maximum value of  $S_L = -\frac{\Delta x}{\Delta t}, S_R = +\frac{\Delta x}{\Delta t}$ . Thence when the flux is computed -by applying the Rankine-Hugoniot conditions-, the Lax-Friedrichs Flux approach is obtained by the expression

$$F^{\text{Lax-Friedrichs}} = \frac{(F_L + F_R)}{2} - \frac{\Delta x}{\Delta t} * \frac{U_R - U_L}{2} \quad (12)$$

### 8.2.2 Rusanov Flux

It improves the previous Lax-Friedrich scheme, which -according to (Harten, Lax and van Leer, 1983)- experience demonstrates that is really diffusive when shocks are dealt. The solution of the Riemann Problem is achieved through the approximation -in reference of local (i) and vicinity triangles (i+1,i-1) - of estimating (different to Lax-Friedrichs) the maximum wave speeds left and right and then averaging the flux such that  $S_L = |u_L| + a_L, S_R = |u_R| + a_R$ , being  $a$  the local sound speed, then

$$F^{\text{RUSANOV}} = \frac{(F_L + F_R)}{2} - \max(|S_L|, |S_R|) \cdot \frac{U_R - U_L}{2}. \quad (13)$$

### 8.2.3 HLL Flux

Introduced by (Zanotti and Manca, 2010), it gives an intermediate solution in the flow with the subsequent benefits that will be derived in the following line. It departs from the idea that there are two waves dividing the Riemann Problem into 3 constant states instead of Roe's large system of n conservative laws. Thence, the initial hypothesis relies in the fact that, after the decay of the initial discontinuity of the Riemann problems, it will only derive in the propagation of two different waves moving towards opposite directions with respective velocities of  $S_L$  and  $S_R$ , and between them an intermediate state is created. Therefore, the formulation of the Riemann problem leads to the following scheme

$$\mathbf{U} = \begin{cases} \mathbf{U}_L & \frac{x}{t} < S_L \\ \mathbf{U}^{\text{HLL}} & S_L < \frac{x}{t} < S_R \\ \mathbf{U}_R & S_R < \frac{x}{t} \end{cases}, \quad (14)$$

being the intermediate velocity state  $\mathbf{U}^{\text{HLL}} = (S_R \cdot \mathbf{U}_R - S_L \cdot \mathbf{U}_L + F_L \cdot F_R) / (S_R - S_L)$ . (15)

The terms of the wave speeds can be approached through different expressions, such as the ones seen in Lax and Friedrichs and Rusanov Fluxes. The easiest one, according to(Davis, 1988) is (Toro, Spruce and Speares, 1994) which proposed  $S_L = u_L - a_L$  and the same for  $S_R = u_R + a_L$ . In our code, depending on defined parameters "PL" and "PR", it will be computed through the anterior formula or by the multiplication of a term to the local speed sound. Once applied the Rankine Hugoniot conditions to both waves left and right, the remaining flux will be computed as it is shown with the posterior substitution oh  $\mathbf{U}^{\text{HLL}}$ :

$$F^{\text{HLL}} = F_L + S_L \cdot (U^{\text{HLL}} - U_L)$$

$$F^{\text{HLL}} = F_R + S_R \cdot (U^{\text{HLL}} - U_R)$$

obtaining:

$$F_{HLL} = \frac{S_R \cdot F_L - S_L \cdot F_R + S_L \cdot S_R \cdot (U_R - U_L)}{S_R - S_L} \quad (16)$$

This Riemann Solver, although is quite simple when it is computed as it is here, has shown excellent results when critical sonic rarefactions are predicted, however, it also introduces non-physical blur at contact discontinuities since middle waves are not considered in the solution. Besides,  $S_L$  and  $S_R$  must be numerically approached and depending on the formula developed, an effect on the solution may be caused.

#### 8.2.4 HLLC Flux

This new Riemann Solver has been developed by (Toro, 2009), further explained in the reference book for the Module of “Numerical Modelling of Compressible Flows” (Toro, 2019) and then recently detailed in (Harten, Lax and van Leer, 1983); supposes an extension of the HLL solver formulation of (Toro, Spruce and Speares, 1994). The need for this extension comes from the fact that HLL gathers an incomplete solution as contact discontinuities and shear waves are not included in the solution. To achieve this, an intermediate contact wave (noted by the subscript \* of speed  $S_*$ ) is restored in between the two waves of HLL. Thence, now three waves are present separating the solution into 4 constant regions when the Riemann problem is dealt.

$$U^{HLLC} = \begin{cases} U_L & \frac{x}{t} < S_L \\ U_{*L} & S_L \leq \frac{x}{t} \leq S_* \\ U_{*R} & S_* \leq \frac{x}{t} \leq S_R \\ U_R & \frac{x}{t} > S_R \end{cases} \quad (17), \text{ then } F^{HLLC} = \begin{cases} F_L & 0 < S_L \\ F_{*L} & S_L \leq 0 \leq S_* \\ F_{*R} & S_* \leq 0 \leq S_R \\ F_R & 0 > S_R \end{cases} \quad (18)$$

According to (Toro, Spruce and Speares, 1994),  $F_{*L}$  and  $F_{*R}$  are obtained with the following formula,

$$F_{*R} = F_R + S_R \cdot (U_{*R} - U_R), \quad F_{*L} = F_L + S_L \cdot (U_{*L} - U_L)$$

Where the star terms are still unknowns.

$F(U) = u \cdot U + p \cdot D$ , being  $D = [0, 1, 0, 0, u]^T$  and resulting in:

$$S_* = \frac{p_r - p_l + \rho_l \cdot u_l \cdot (S_l - u_l) - \rho_r \cdot u_r \cdot (S_r - u_r)}{\rho_l \cdot (S_l - u_l) - \rho_r \cdot (S_r - u_r)}$$

so that fluxes now can be calculated using:

$F_{*K} = F_K + S_K \cdot (U_{*K} - U_K)$  for  $K=Left$  and  $Right$  where the intermediate states are given by -for the 2D case-:

$$\mathbf{U}_{*K} = \rho_K \left( \frac{S_K - u_K}{S_K - S_*} \right) \left[ \begin{array}{c} 1 \\ S_* \\ v_K \\ w_K \\ \frac{E_K}{\rho_K} + (S_* - u_K) \left[ S_* + \frac{\rho_K}{\rho_K(S_K - u_K)} \right] \end{array} \right]$$

Once all the variables are known, the flux is computed according to (18).

### 8.3 Slope Limiters for the MUSCL scheme.

#### 8.3.1 Barth and Jespersen Limiter

The first slope limiter for unstructured grids was developed by (Barth and Jespersen, 1989). They aim to acquire a function whose value is between (0,1) such that is the largest possible but multiplied to the gradients in the reconstruction procedure does not vulnerable the monotonicity principle of creating new extrema. Once calculated the velocity in each of the vertices of the cell, noted by  $u_i$ , the Barth and Jespersen Limiter can be obtained with the following logical expression:

$$\varphi^{\text{Barth and Jespersen}} = \begin{cases} \min\left(1, \frac{u^{\max.} - u_A}{u_i - u_a}\right) & u_i - u_a > 0 \\ \min\left(1, \frac{u^{\min} - u_A}{u_i - u_a}\right) & u_i - u_a < 0 \\ 1 & u_i = u_a \end{cases}$$

#### 8.3.2 Venkatakrishnan Limiter

(Venkatakrishnan, 1993) is, as well as Barth and Jespersen, motivated to acquire “monotonous” results in 2-D unstructured meshes. He proposes, relying on the theory of Barth and Jespersen, to give the min function another treatment to acquire differentiability and also to facilitate convergence by loosing a little bit of the monotonicity behaviour, as he realized that there seems to be a proportionality between convergence and monotonicity. As (Zheng, 2013) points out, this approach does not satisfy strict monotonicity and therefore little oscillations may be noticed near strong discontinuities. The value of the limiter is given by:

$$\varphi^{\text{Venkatakrishnan}} = \frac{1}{\Delta_-} \cdot \left[ \frac{(\Delta_-^2 + \varepsilon^2) \cdot \Delta_- + 2 \cdot \Delta_-^2 \cdot \Delta_+}{\Delta_+^2 + 2 \cdot \Delta_-^2 + \Delta_+ \cdot \Delta_- + \varepsilon^2} \right]$$

where, similarly to Barth and Jespersen formulation, the variables have a value of

$$\Delta_- = u_i - u_a, \quad \Delta_+ = \begin{cases} u^{\max.} - u_A & u_i - u_a > 0 \\ u^{\min} - u_A & u_i - u_a < 0 \end{cases}, \text{ and } \varepsilon^2 = (k \cdot \Delta x)^3 \text{ where } k \text{ is normally set to } 0.3 \text{ and } \Delta x \text{ is referred to the mesh resolution.}$$

According to (Zheng, 2013), both limiters (Barth and Jespersen and Venkatakrishnan) have shown to be dissipative thence they may not satisfy the theoretical accuracy with fixed stencil. Besides, (Michalak and Ollivier-Gooch, 2008) analyzes both for higher order schemes - particularly 3<sup>rd</sup> and 4<sup>th</sup> - in terms of monotonicity, accuracy, and performance in uniform regions where extrema may be changed in transient solutions.

Regarding monotonicity, they describe that additional parameters can be introduced to “satisfy” the monotonicity for high order schemes, but it low-order schemes are faced within the same code,  $\varphi$  may be insufficient in both limiters to satisfy the monotonicity condition.

The issues when considering accuracy are mainly the following: for high order schemes Venkatakrishnan’s limiter has revealed bad performance even in smooth regions, whilst Barth-Jespersen limiter satisfies the accuracy restraint. However, the fact that it is not differentiable will difficult and compromise the time to obtain steady-state solutions.

Finally, for uniform regions, (Michalak and Ollivier-Gooch, 2008) propose a condition to erase the contribution of the limiter for the zones where is beneficial to suppress their effect. Besides, they propose a methodology to maintain the differentiability of the limiter.

For all these reasons, it will be preferable to use Barth and Jespersen when oscillations are noted in the Venkatakrishnan procedure or higher order spatial accuracy is aimed.

## 9. Appendix 2. Solver Missing Code

### 9.1 1-D Solver

The Rusanov, HLL, HLLC Riemann solvers and the reconstruction based on minmod slope limiter are not yet implemented in the file euler.f90. In order to do so, the following codes must be written in its corresponding subroutine/section:

· Second Order Reconstruction with minmod limiter. (Subroutine RECONSTRUCTION)

Case (2)

```
Do f=1,3
    r=minmod(U1D(f,0)-U1D(f,-1), U1D(f,1)-U1D(f,0))
    CDL(f)=U1D(f,0)+0.5*r
    r=minmod(U1D(f,1)-U1D(f,0), U1D(f,2)-U1D(f,1))
    CDR(f)=U1D(f,1)-0.5*r
Enddo
```

And the minmod function is defined as:

```
Real function minmod(x,y)
  Real x,y
  minmod = 0.5*(sign(1.0,x)+sign(1.0,y))*min(abs(x),abs(y))
End function
```

· Rusanov Riemann Solver.

```
Subroutine Rusanov(CDL,CDR,Flux)
  Real FL(3), FR(3), CDL(3), CDR(3), Flux(3), S, a_L, a_R

  CALL FLUEVAL(CDL,FL)
  CALL FLUEVAL(CDR,FR)
  a_L= ComputeSoundSpeed(CDL)
  a_R= ComputeSoundSpeed(CDR)
  S = max(abs(CDL(2)/CDL(1))+a_L,abs(CDR(2)/CDR(1))+_ar)

  Flux = 0.5*(FL+FR)-0.5*S*(CDR-CDL)
End subroutine
```

· HLL Riemann Solver.

```
Subroutine HLL(CDL,CDR,Flux)
  Real FL(3), FR(3), CDL(3), CDR(3), Flux(3), S_L, S_R
!Calculation of variables + Wave approximations
  CALL FLUEVAL(CDL,FL)
  CALL FLUEVAL(CDR,FR)
  S_L = (CDL(2)/CDL(1))-ComputeSoundSpeed(CDL)
  S_R = (CDR(2)/CDR(1))+ComputeSoundSpeed(CDR)
!Imposal of Flux Scheme
  if (S_L.GT.0) then
```

```

        Flux = FL
elseif (S_L.LE.0.AND.S_R.GE.0) then
    Flux = (S_R*FL-S_L*FR+S_L*S_R*(CDR-CDL))/(S_R-S_L)
elseif (S_R.LE.0) then
    Flux = FR
endif

End subroutine

```

· HLLC Riemann Solver.

```

Subroutine HLLC(CDL,CDR,Flux)
    Real CDL(3),CDR(3),Flux(3), FL(3), FR(3), U_int_L(3),
U_int_R(3)
    Real S_L, S_R, U_L, U_R, p_L ,p_R,density_L, density_R,
E_L, E_R, S_int
    !Calculation of variables + Wave approximations
    CALL FLUEVAL(CDL,FL)
    CALL FLUEVAL(CDR,FR)
    U_L = CDL(2)/CDL(1)
    U_R = CDR(2)/CDR(1)
    p_L = (gm-1)*(CDL(3) - 0.5*CDL(2)*uL)
    p_R = (gm-1)*(CDR(3) - 0.5*CDR(2)*uR)
    S_L = uL-ComputeSoundSpeed(CDL)
    S_R = uR+ComputeSoundSpeed(CDR)
    density_R = CDR(1)
    density_L = CDL(1)
    E_L = CDL(3)
    E_R = CDR(3)
    S_int = (p_R-p_L+density_L*u_L*(S_L-U_L)-density_R*U_R*(S_R-
U_R))/(density_L*(S_L-U_L)-density_R*(S_R-U_R))
    ! U intermediate (star) calculation, left and right
    U_int_L = (/ density_L*((S_L-U_L)/(S_L-S_int)), density_L*((S_L-
U_L)/(S_L-S_int))*S_int, density_L*((S_L-U_L)/(S_L-
S_int))*(E_L/density_L)+(S_int-U_L)*(S_int+(P_L/((S_L-U_L)))) /)
    U_int_L = (/ density_R*((S_R-U_R)/(S_R-S_int)), density_R*((S_R-
U_R)/(S_R-S_int))*S_int, density_R*((S_R-U_R)/(S_R-
S_int))*(E_R/density_R)+(S_int-U_R)*(S_int+(P_R/((S_R-U_R)))) /)
    !Imposal of Flux Scheme
    if (S_L.GE.0) then
        Flux = FL
    elseif (S_L.LE.0.AND.S_int.GE.0) then
        Flux = FL+S_L*(U_int_L-CDL)
    elseif (S_int.LE.0.AND.S_R.GE.0) then
        Flux = FR+S_R*(U_int_R-CDR)
    elseif (S_R.LE.0) then
        Flux = FR
    endif

End subroutine

```

## 9.2 2-D Solver

The HLLC Riemann solver and the reconstruction based on Venkatakrishnan slope limiter are not either implemented in the file solver.f90 for the 2-D unstructured grid solver. The HLL code will be really similar to the 1-D with the introduction of the v component of velocity, thence, all the matrixes and vectors involved will enlarge by 1 term or component. A proposal of programming and implementation of both could be done with the following code:

```

. 2nd order reconstruction with Venkatakrishnan Limiter (Subroutine
ComputeReconstructionBarth_i)

IF (LIMITER.EQ.2) THEN !VENKATATKRISHNAN
    D2=(Vert(L)-Va); KAPPA_Ven=10.0D0
;DELTA_X=MeshElements(i)%rad
    EPSI_ZERO=1E-16
    DMIN=D2
    KAPPA_VEN=0.5
    EPSI2=(KAPPA_VEN*MeshElements(i)%rad)**3

    IF (abs(d2).le. 1e-6) THEN
        psi(L) = 1.0D0
    ELSE
        IF (D2.GT.0.0D0) THEN
            DPLUS=u_max-va

PSI=((DPLUS**2+EPSI2)*DMIN+2*DPLUS*DMIN**2)/(DMIN*(DPLUS**2+s*DMIN**2+
DMIN*DMIN+EPSI2))

        END IF
        IF (D2.LT.0.0D0) THEN
            DPLUS=u_min-va

PSI=((DPLUS**2+EPSI2)*DMIN+2*DPLUS*DMIN**2)/(DMIN*(DPLUS**2+s*DMIN**2+
DMIN*DMIN+EPSI2))

        END IF
    END IF
END IF

```

### . HLLC Flux

```

SUBROUTINE HLLC_FLUX(CDL,CDR,FHLLC)

INTEGER K
Real(8), DIMENSION(4) :: FHLLC(4)
Real(8) ENEL, ENER, SL, SM, SR, S_Star
Real(8), DIMENSION(4) :: FDL, FDR, FSL, FSR
Real(8), DIMENSION(4) :: CSL, CSR, CDL, CDR, PVL, PVR, VectorL,
VectorR, U_star_l, U_star_r
REAL(8) DL, UL, VL, PL, DR, UR, VR, PR, CL, CR, EL, ER
    DL = PVL(1)
    UL = PVL(2)

```

```

VL = PVL(3)
PL = PVL(4)
CL = SQRT(Gamma*PL/DL)
DR = PVR(1)
UR = PVR(2)
VR = PVR(3)
PR = PVR(4)
CR = SQRT(Gamma*PR/DR)
PVL = csvtopv(CDL)
PVR = csvtopv(CDR)
CALL ESTIME(PVL,PVR,SL,SR,SM)
FDL = FLUEVAL(CDL)
FDR = FLUEVAL(CDR)
S_star=(PR-PL+DL*UL*(SL-UL)-DR*UR*(SR-UR)) / (DL*(SL-UL)-DR*(SR-
UR))
VectorR = (/ DL*((SL-UL)/(SL-S_star))*1., DL*((SL-UL)/(SL-
S_star))*S_star, DL*((SL-UL)/(SL-S_star))*VR, DL*((SL-UL)/(SL-
S_star))*((ER/DR)+(S_star-UR)*(S_star+(PR/(DR*(SR-UR)))) ) )
VectorL = (/ DR*((SR-UR)/(SL-S_star))*1., DR*((SR-UR)/(SL-
S_star))*S_star, DR*((SR-UR)/(SL-S_star))*VL, DR*((SR-UR)/(SL-
S_star))*((EL/DL)+(S_star-UL)*(S_star+(PL/(DL*(SL-UL)))) ) )
U_star_l=VectorL
U_star_r=VectorR
IF (SL>0.) then
    FHLLC=FDL
ELSEIF(SL.LE.0. .AND. S_star.GE.0.) THEN
    FHLLC=FDL+SL*(U_star_l-UL)
ELSEIF(S_star<0 .AND. SR.GE.0) THEN
    FHLLC=FDR+SR*(U_star_r-UR)
ELSEIF(SR.LE.0) THEN
    FHLLC=FDR
ENDIF
END SUBROUTINE

```