Explicit and Implicit TVD High Resolution Schemes in 2D

EDISSON SÁVIO DE GÓES MACIEL
IEA – Aeronautical Engineering Division
ITA – Aeronautical Technological Institute
Praça Mal. do Ar Eduardo Gomes, 50 – Vila das Acácias – São José dos Campos – SP – 12228-900
BRAZIL
edissonsavio@yahoo.com.br http://www.edissonsavio.eng.br

Abstract: - The present work compares the TVD schemes of Roe, of Van Leer, of Yee,Warming and Harten, of Harten, of Yee and Kutler and of Hughson and Beran applied to the solution of an aeronautical problem. Only the Van Leer scheme is a flux vector splitting one. The others are of flux difference splitting type. The Roe and Van Leer schemes reach second order accuracy and TVD properties by the use of a MUSCL approach, which employs five different types of nonlinear limiters, that assures TVD properties, being them: Van Leer limiter, Van Albada limiter, minmod limiter, Super Bee limiter and β-limiter. The other schemes are based on the Harten's ideas of the construction of a modified flux function to obtain second order accuracy and TVD characteristics. The implicit schemes employ an ADI ("Alternating Direction Implicit") approximate factorization to solve implicitly the Euler equations, whereas in the explicit case a time splitting method is used. Explicit and implicit results are compared trying to emphasize the advantages and disadvantages of each formulation. The Euler equations in conservative form, employing a finite volume formulation and a structured spatial discretization, are solved in two-dimensions. The steady state physical problem of the supersonic flow along a compression corner is studied. A spatially variable time step procedure is employed aiming to accelerate the convergence of the numerical schemes to the steady state condition. This technique has proved an excellent behavior in terms of convergence gains, as shown in Maciel. The results have demonstrated that the most accurate solutions are provided by the Roe TVD scheme in its Super Bee variant.

Key-Words: - Roe scheme, Van Leer scheme, Yee, Warming and Harten scheme; Harten scheme; Yee and Kutler scheme; Hughson and Beran scheme; Explicit and implicit formulations; TVD formulation; Euler and Navier-Stokes equations.

1 Introduction

High resolution upwind schemes have been developed since 1959, aiming to improve the generated solution quality, yielding more accurate solutions and more robust codes. The high resolution upwind schemes can be of flux vector splitting type or flux difference splitting type. In the former case, more robust algorithms are yielded, while in the latter case, more accuracy is obtained. Several studies were reported involving high resolution algorithms in the international literature, as for example:

[1] method, whose author presented a work that emphasized that several numerical schemes to the solution of the hyperbolic conservation equations were based on exploring the information obtained in the solution of a sequence of Riemann problems. It was verified that in the existent schemes the major part of these information was degraded and that only certain solution aspects were solved. It was demonstrated that the information could be preserved by the construction of a matrix with a certain "U property". After the construction of this matrix, its eigenvalues could be considered as wave velocities of the Riemann problem and the U_L-U_R projections over the matrix's eigenvectors would be the jumps which occur between intermediate stages. This scheme was originally first order accurate.

[2] method, whose author suggested an upwind scheme based on the flux vector splitting concept. This scheme considered the fact that the convective flux vector components could be written as flow Mach number polynomial functions, as main characteristic. Such polynomials presented the particularity of having the minor possible degree and the scheme had to satisfy seven basic properties to form such polynomials. This scheme was also originally developed in its first order accurate version.

[3] implemented a high resolution second order explicit method based on Harten's ideas. The method had the following properties: (a) the scheme was developed in conservation form to ensure that the limit was a weak solution; (b) the scheme satisfied a proper entropy inequality to ensure that the limit solution would have only physically relevant discontinuities; and (c) the scheme was

designed such that the numerical dissipation produced highly accurate weak solutions. The method was applied to the solution of a quasi-one-dimensional nozzle problem and to the two-dimensional shock reflection problem, yielding good results. An implicit implementation was also investigated to one- and two-dimensional cases.

[4] developed a class of new finite difference schemes, explicit and with second order of spatial accuracy to calculation of weak solutions of the hyperbolic conservation laws. These schemes highly non-linear were obtained by the application of a first order non-oscillatory scheme to an appropriated modified flux function. The so derived second order schemes reached high resolution, while preserved the robustness property of the original non-oscillatory scheme.

[5] presented a work which extended the [4] scheme to a generalized coordinate system, in two-dimensions. The method called "TVD scheme" by the authors was tested to the physical problem of a moving shock impinging a cylinder. The numerical results were compared with the [6] scheme, presenting good results.

[7] proposed an explicit, second order accurate in space, TVD scheme to solve the Euler equations in axis-symmetrical form, applied to the studies of the supersonic flow around a sphere and the hypersonic flow around a blunt body. The scheme was based on the modified flux function approximation of [4] and its extension from the two-dimensional space to the axis-symmetrical treatment was developed. Results were compared to the [6] algorithm's solutions. High resolution aspects, capability of shock capture and robustness properties of this TVD scheme were investigated.

In relation to [1-2], second order spatial accuracy can be achieved by introducing more upwind points or cells in the schemes. It has been noted that the projection stage, whereby the solution is projected in each cell face (i-1/2,j; i+1/2,j) on piecewise constant states, is the cause of the first order space accuracy of the [8] schemes ([9]). Hence, it is sufficient to modify the first projection stage without modifying the Riemann solver, in order to generate higher spatial approximations. The state variables at the interfaces are thereby obtained from an extrapolation between neighboring cell averages. This method for the generation of second order upwind schemes based on variable extrapolation is often referred to in the literature as the MUSCL ("Monotone Upstream-centered Schemes for Conservation Laws") approach. The use of nonlinear limiters in such procedure, with the intention of restricting the amplitude of the gradients appearing in the solution, avoiding thus the formation of new extrema, allows that first order upwind schemes be transformed in TVD ("Total Variation Diminishing") high resolution schemes with the appropriate definition of such nonlinear limiters, assuring monotone preserving and total variation diminishing methods.

Traditionally, implicit numerical methods have been praised for their improved stability and condemned for their large arithmetic operation counts ([10]). On the one hand, the slow convergence rate of explicit methods become they so unattractive to the solution of steady state problems due to the large number of iterations required to convergence, in spite of the reduced number of operation counts per time step in comparison with their implicit counterparts. Such problem is resulting from the limited stability region which such methods are subjected (the Courant condition). On the other hand, implicit schemes guarantee a larger stability region, which allows the use of CFL numbers above 1.0, and fast convergence to steady state conditions. Undoubtedly, the most significant efficiency achievement for multidimensional implicit methods was the introduction of the Alternating Direction Implicit (ADI) algorithms by [11], [12], and [13], and fractional step algorithms by [14]. ADI approximate factorization methods consist in approximating the Left Hand Side (LHS) of the numerical scheme by the product of one-dimensional parcels, each one associated with a different spatial coordinate direction, which retract nearly the original implicit operator. These methods have been largely applied in the CFD community and, despite the fact of the error of the approximate factorization, it allows the use of large time steps, which results in significant gains in terms of convergence rate in relation to explicit methods.

In this work, the [1-5;7] schemes are implemented, on a finite volume context and using an upwind and a structured spatial discretization, to solve the Euler equations, in two-dimensions, and are compared with themselves. All schemes are implemented in its second order version in space and are applied to the solution of the supersonic flow along a compression corner. Considering [1-2], a MUSCL approach is employed using five different types of nonlinear limiters, which assure second order and TVD properties, namely: Van Leer limiter, Van Albada limiter, minmod limiter, Super Bee limiter and β-limiter. A spatially variable time step procedure is implemented aiming to accelerate the convergence of the schemes to the steady state condition. The effective gains in terms of

convergence ratio with this procedure are reported in [15-16].

The results have demonstrated that the most accurate solutions are provided by the [1] TVD scheme in its Super Bee variant.

The motivation and justification of this work is to present TVD high resolution schemes, which are reported in the CFD literature as able to provide numerical solutions free of oscillations and test their abilities to provide good shock capturing properties. Furthermore, the CFD literature describes these schemes on a finite difference context and using a generalized coordinate system. Hence, this work represents an original contribution in the sense that the studied TVD schemes are described and implemented on a finite volume context. Moreover, an implicit formulation is also applied, which contributes to the originality of this manuscript too.

2 Euler Equations

The fluid movement is described by the Euler equations, which express the conservation of mass, of linear momentum and of energy to an inviscid, heat non-conductor and compressible mean, in the absence of external forces. In the integral and conservative forms, these equations can be represented by:

$$\partial/\partial t \int_V Q dV + \int_S \left(E_e n_x + F_e n_y\right) dS = 0, \qquad (1)$$

where Q is written to a Cartesian system, V is a cell volume, n_x and n_y are the components of the normal unity vector to the flux face, S is the surface area and E_e and F_e represent the components of the convective flux vector. Q, E_e and F_e are represented by:

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{Bmatrix}, \quad E_e = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e+p)u \end{Bmatrix}, \quad F_e = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (e+p)v \end{Bmatrix}, \qquad (2)$$

being ρ the fluid density; u and v the Cartesian components of the velocity vector in the x and y directions, respectively; e the total energy per unit volume of the fluid mean; and p the static pressure of the fluid mean.

In all problems, the Euler equations were dimensionless in relation to the freestream density, ρ_∞, and in relation to the freestream speed of sound, a_∞. The matrix system of the Euler equations is closed with the state equation of a perfect gas:

$$p = (\gamma - 1)\left[e - 0.5\rho(u^2 + v^2)\right], \qquad (3)$$

being γ the ratio of specific heats. The total enthalpy is determined by $H = (e+p)/\rho$.

3 [1] Algorithm

The [1] algorithm, first order accurate in space, is specified by the determination of the numerical flux vector at *(i+½,j)* interface. At the *(i,j+½)* interface, the implementation is straightforward.

Following a finite volume formalism, which is equivalent to a generalized system, the right and left cell volumes, as well the interface volume, necessary to coordinate change, are defined by:

$$V_R = V_{i+1,j}, \; V_L = V_{i,j} \text{ and } V_{int} = 0.5(V_R + V_L), \qquad (4)$$

in which "*R*" and "*L*" represent right and left states, respectively. The cell volume is defined by:

$$V_{i,j} = 0.5\left|\left(x_{i,j} - x_{i+1,j}\right)y_{i+1,j+1} + \left(x_{i+1,j} - x_{i+1,j+1}\right)y_{i,j} + \left(x_{i+1,j+1} - x_{i,j}\right)y_{i+1,j}\right| + 0.5\left|\left(x_{i,j} - x_{i+1,j+1}\right)y_{i,j+1} + \left(x_{i+1,j+1} - x_{i,j+1}\right)y_{i,j} + \left(x_{i,j+1} - x_{i,j}\right)y_{i+1,j+1}\right|, \qquad (5)$$

where a computational cell and its flux surfaces are defined in Fig. 1.

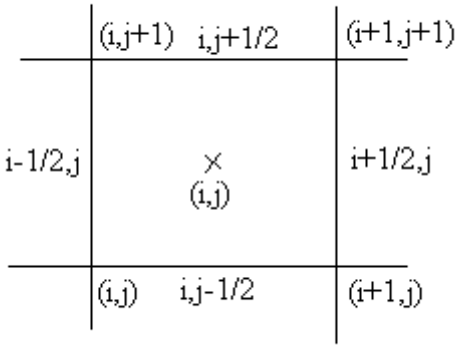

Figure 1: Computational cell.

The area components at interface are defined by: $S_{x\_\text{int}} = s_x' S$ and $S_{y\_\text{int}} = s_y' S$, where s_x' and s_y' are defined as: $s_x' = s_x / S$ and $s_y' = s_y / S$, being $S = \left(s_x^2 + s_y^2\right)^{0.5}$. Expressions to s_x and s_y, which represent the S_x and S_y components always adopted

in the positive orientation, are given in Tab. 1. The metric terms to this generalized coordinate system are defined as:

$$h_x = S_{x\_int}/V_{int}\ ,\ h_y = S_{y\_\mathrm{int}}/V_{\mathrm{int}}\ ,\ h_n = S/V_{int}\ . \tag{6}$$

Table 1: Normalized values of s_x and s_y.

Surface	s_x	s_y
i,j-1/2	$-\left(y_{i+1,j} - y_{i,j}\right)$	$\left(x_{i+1,j} - x_{i,j}\right)$
i+1/2,j	$\left(y_{i+1,j+1} - y_{i+1,j}\right)$	$\left(x_{i+1,j} - x_{i+1,j+1}\right)$
i,j+1/2	$\left(y_{i,j+1} - y_{i+1,j+1}\right)$	$\left(x_{i+1,j+1} - x_{i,j+1}\right)$
i-1/2,j	$\left(y_{i,j+1} - y_{i,j}\right)$	$-\left(x_{i,j+1} - x_{i,j}\right)$

The properties calculated at the flux interface are obtained either by arithmetical average or by [1] average. In this work, the arithmetical average was used:

$$\rho_{\mathrm{int}} = 0.5\left(\rho_R + \rho_L\right),\ u_{\mathrm{int}} = 0.5\left(u_R + u_L\right);$$

$$v_{\mathrm{int}} = 0.5\left(v_R + v_L\right) \text{ and } H_{\mathrm{int}} = 0.5\left(H_R + H_L\right). \tag{7}$$

$$a_{\mathrm{int}} = \sqrt{(\gamma-1)\left[H_{\mathrm{int}} - 0.5\left(u_{\mathrm{int}}^2 + v_{\mathrm{int}}^2\right)\right]}, \tag{8}$$

where a_{int} is the speed of sound at the flux interface. The eigenvalues of the Euler equations, in the ξ direction, are given by:

$$U_{cont} = u_{int}h_x + v_{int}h_y,\ \lambda_1 = U_{cont} - a_{int}h_n;$$

$$\lambda_2 = \lambda_3 = U_{cont} \text{ and } \lambda_4 = U_{cont} + a_{int}h_n\,. \tag{9}$$

The jumps of the conserved variables, necessary to the construction of the [4] dissipation function, are given by:

$$\Delta e = V_{\mathrm{int}}\left(e_R - e_L\right),\ \Delta\rho = V_{\mathrm{int}}\left(\rho_R - \rho_L\right);$$

$$\Delta(\rho u) = V_{\mathrm{int}}\left[(\rho u)_R - (\rho u)_L\right] \text{ and } \Delta(\rho v) = V_{\mathrm{int}}\left[(\rho v)_R - (\rho v)_L\right]. \tag{10}$$

The α vectors at the *(i+½,j)* interface are calculated by the following expressions:

$$\alpha^1 = 0.5(aa - bb),\ \alpha^2 = \Delta\rho - aa,\ \alpha^3 = cc \quad \text{and}$$
$$\alpha^4 = 0.5(aa + bb). \tag{11}$$

with:

$$aa = (\gamma-1)/a_{\mathrm{int}}^2\left[\Delta e + 0.5\left(u_{\mathrm{int}}^2 + v_{\mathrm{int}}^2\right)\Delta\rho - u_{\mathrm{int}}\Delta(\rho u) - v_{\mathrm{int}}\Delta(\rho v)\right]; \tag{12}$$

$$bb = 1/a_{\mathrm{int}}\left[h_x'\Delta(\rho u) - \left(h_x'u_{\mathrm{int}} + h_y'v_{\mathrm{int}}\right)\Delta\rho + h_y'\Delta(\rho v)\right]; \tag{13}$$

$$cc = h_x'\Delta(\rho v) + \left(h_y'u_{\mathrm{int}} - h_x'v_{\mathrm{int}}\right)\Delta\rho - h_y'\Delta(\rho u); \tag{14}$$

$$h_x' = h_x/h_n \quad \text{and} \quad h_y' = h_y/h_n\,. \tag{15}$$

The [1] dissipation function uses the right-eigenvector matrix of the normal to the flux face Jacobian matrix in generalized coordinates:

$$R_{i+1/2,j} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ u_{\mathrm{int}} - h_x'a_{\mathrm{int}} & u_{\mathrm{int}} & -h_y' & u_{\mathrm{int}} + h_x'a_{\mathrm{int}} \\ v_{\mathrm{int}} - h_y'a_{\mathrm{int}} & v_{\mathrm{int}} & h_x' & v_{\mathrm{int}} + h_y'a_{\mathrm{int}} \\ H_{\mathrm{int}} - h_x'u_{\mathrm{int}}a_{\mathrm{int}} - h_y'v_{\mathrm{int}}a_{\mathrm{int}} & 0.5\left(u_{\mathrm{int}}^2 + v_{\mathrm{int}}^2\right) & h_x'v_{\mathrm{int}} - h_y'u_{\mathrm{int}} & H_{\mathrm{int}} + h_x'u_{\mathrm{int}}a_{\mathrm{int}} + h_y'v_{\mathrm{int}}a_{\mathrm{int}} \end{bmatrix}. \tag{16}$$

The entropy condition is implemented of the following way:

$$\psi_l = \begin{cases} |\lambda_l|, & \text{if } |\lambda_l| \geq \varepsilon_{\xi_l} \\ 0.5\left(\lambda_l^2 + \varepsilon_{\xi_l}^2\right)/\varepsilon_{\xi_l}, & \text{if } |\lambda_l| < \varepsilon_{\xi_l} \end{cases}, \text{ non-linear}$$

fields, and $\psi_l = |\lambda_l|$, linear fields, (17)

with "*l*" assuming values of 1 and 4 to non-linear fields and 2 and 3 to linear fields; and ε_{ξ_l} assuming a value of 0.2, as recommended by [1]. The [1] dissipation function is finally constructed by the following matrix-vector product:

$$\left\{D_{Roe}\right\}_{i+1/2,j} = \left[R\right]_{i+1/2,j}\left\{-\psi\alpha\right\}_{i+1/2,j}. \tag{18}$$

The convective numerical flux vector to the *(i+½,j)* interface is described by:

$$F^{(l)}_{i+1/2,j} = \left(E^{(l)}_{\text{int}} h_x + F^{(l)}_{\text{int}} h_y\right) V_{\text{int}} + 0.5 D^{(l)}_{Roe}, \text{ with:}$$
$$E^{(l)}_{\text{int}} = 0.5\left(E^{(l)}_R + E^{(l)}_L\right) \text{ and } F^{(l)}_{\text{int}} = 0.5\left(F^{(l)}_R + F^{(l)}_L\right). \quad (19)$$

The explicit time integration follows the time splitting method, first order accurate, which divides the integration in two steps, each one associated with a specific spatial direction. In the initial step, it is possible to write for the ξ direction:

$$\Delta Q^{*}_{i,j} = -\Delta t_{i,j} / V_{i,j} \left(F^{n}_{i+1/2,j} - F^{n}_{i-1/2,j}\right);$$
$$Q^{*}_{i,j} = Q^{n}_{i,j} + \Delta Q^{*}_{i,j}; \quad (20)$$

and at the end step, η direction:

$$\Delta Q^{n+1}_{i,j} = -\Delta t_{i,j} / V_{i,j} \left(F^{*}_{i,j+1/2} - F^{*}_{i,j-1/2}\right);$$
$$Q^{n+1}_{i,j} = Q^{*}_{i,j} + \Delta Q^{n+1}_{i,j}. \quad (21)$$

4 [2] Algorithm

The approximation to the integral equation (1) to a rectangular finite volume yields an ordinary differential equation system with respect to time:

$$V_{i,j}\, dQ_{i,j}/dt = -R_{i,j}, \quad (22)$$

with $R_{i,j}$ representing the neat flux (residual) of the conservation of mass, of linear momentum and of energy in the $V_{i,j}$ volume. The residual is calculated as:

$$R_{i,j} = R_{i+1/2,j} - R_{i-1/2,j} + R_{i,j+1/2} - R_{i,j-1/2}, \quad (23)$$

where $R_{i+1/2,j} = R^{c}_{i+1/2,j}$, in which "c" is related to the flow convective contribution. The discrete convective flux calculated by the AUSM scheme ("Advection Upstream Splitting Method") can be interpreted as a sum involving the arithmetical average between the right (*R*) and the left (*L*) states of the *(i+½,j)* cell face, related to cells *(i,j)* and *(i+1,j)*, respectively, multiplied by the interface Mach number, and a scalar dissipative term, as shown in [17]. Hence,

$$R_{i+1/2,j} = |S|_{i+1/2,j} \left\{ \frac{1}{2} M_{i+1/2,j} \left[\begin{pmatrix} \rho a \\ \rho a u \\ \rho a v \\ \rho a H \end{pmatrix}_L + \begin{pmatrix} \rho a \\ \rho a u \\ \rho a v \\ \rho a H \end{pmatrix}_R \right] - \frac{1}{2} \phi_{i+1/2,j} \left[\begin{pmatrix} \rho a \\ \rho a u \\ \rho a v \\ \rho a H \end{pmatrix}_R - \begin{pmatrix} \rho a \\ \rho a u \\ \rho a v \\ \rho a H \end{pmatrix}_L \right] \right\}_{i+1/2,j} + \begin{pmatrix} 0 \\ S_x p \\ S_y p \\ 0 \end{pmatrix}, \quad (24a,b)$$

where $S_{i+1/2,j} = \begin{bmatrix} S_x & S_y \end{bmatrix}^T_{i+1/2,j}$ defines the normal area vector to the *(i+½,j)* surface. The "*a*" quantity represents the speed of sound. $M_{i+½,j}$ defines the advective Mach number in the *(i+½,j)* face of the cell *(i,j)*, which is calculated according to [17] as:

$$M_{i+1/2,j} = M^{+}_{L} + M^{-}_{R}, \quad (25)$$

where the $M^{+/-}$ separated Mach numbers are defined by [2] as:

$$M^{+} = \begin{cases} M, & \text{if } M \geq 1; \\ 0.25(M+1)^2, & \text{if } |M| < 1; \text{ and} \\ 0, & \text{if } M \leq -1; \end{cases}$$
$$M^{-} = \begin{cases} 0, & \text{if } M \geq 1; \\ -0.25(M-1)^2, & \text{if } |M| < 1; \\ M, & \text{if } M \leq -1. \end{cases} \quad (26)$$

M_L and M_R represent the Mach numbers associated to the left and right states, respectively. The advection Mach number is defined as:

$$M = \left(S_x u + S_y v\right) / \left(a|S|\right). \quad (27)$$

The pressure at the *(i+½,j)* face of the *(i,j)* cell is calculated from a similar way:

$$p_{i+1/2,j} = p^{+}_{L} + p^{-}_{R}, \quad (28)$$

with $p^{+/-}$ representing the pressure separation defined according to [2]:

$$p^{+} = \begin{cases} p, & \text{if } M \geq 1; \\ 0.25\,p(M+1)^2(2-M), & \text{if } |M| < 1; \text{ and} \\ 0, & \text{if } M \leq -1; \end{cases}$$
$$p^{-} = \begin{cases} 0, & \text{if } M \geq 1; \\ 0.25\,p(M-1)^2(2+M), & \text{if } |M| < 1; \\ p, & \text{if } M \leq -1. \end{cases} \quad (29)$$

The definition of the ϕ dissipation term determines the particular formulation to the convective fluxes. The following choice corresponds to the [2] scheme, according to [18]:

$$\phi_{i+1/2,j} = \phi^{VL}_{i+1/2,j}, \quad (30)$$

where:

$$\phi^{VL}_{i+1/2,j} = \begin{cases} \left|M_{i+1/2,j}\right|, & if\ \left|M_{i+1/2,j}\right| \geq 1; \\ \left|M_{i+1/2,j}\right| + 0.5\left(M_R - 1\right)^2, & if\ 0 \leq M_{i+1/2,j} < 1; \\ \left|M_{i+1/2,j}\right| + 0.5\left(M_L + 1\right)^2, & if\ -1 < M_{i+1/2,j} \leq 0. \end{cases} \quad (31)$$

The explicit time integration follows the method presented in the [1] scheme [Eqs. (20) and (21)]. This version of the [2] algorithm is first order accurate in space.

5 [3] Algorithm

The [3] algorithm, second order accurate in space, follows the Eqs. (4)-(16). The next step consist in determine the entropy function. Two options to the ψ_l entropy function, responsible to guarantee that only relevant physical solutions are to be considered, are implemented aiming an entropy satisfying algorithm:

$$\nu_l = \Delta t \lambda_l = Z_l \quad \text{and} \quad \psi_l = Z_l^2 + 0.25; \quad (32)$$

Or:

$$\psi_l = \begin{cases} \left|Z_l\right|, & if\ \left|Z_l\right| \geq \delta_f \\ 0.5\left(Z_l^2 + \delta_f^2\right)/\delta_f, & if\ \left|Z_l\right| < \delta_f \end{cases}, \quad (33)$$

where "*l*" varies from 1 to 4 (two-dimensional space) and δ_f assuming values between 0.1 and 0.5, being 0.2 the value recommended by [3]. In the present studies, Eq. (32) was used to perform the numerical experiments.

The $\tilde{g}$ function at the *(i+½,j)* interface is defined by:

$$\tilde{g}^l = 0.5\left(\psi_l - Z_l^2\right)\alpha^l. \quad (34)$$

The *g* numerical flux function, which is a limited function to avoid the formation of new extremes in the solution and is responsible to the second order accuracy of the scheme, is given by:

$$g^l_{i,j} = \text{signal}_l \times \text{MAX}\left[0.0; \text{MIN}\left(\left|\tilde{g}^l_{i+1/2,j}\right|, \tilde{g}^l_{i-1/2,j} \times \text{signal}_l\right)\right], \quad (35)$$

where $signal_l$ is equal to 1.0 if $\tilde{g}^l_{i+1/2,j} \geq 0.0$ and -1.0 otherwise.

The θ term, responsible to the artificial compression, which enhances the resolution of the scheme at discontinuities, is defined as follows:

$$\theta^l_{i,j} = \begin{cases} \left|\alpha^l_{i+1/2,j} - \alpha^l_{i-1/2,j}\right| \Big/ \left(\left|\alpha^l_{i+1/2,j}\right| + \left|\alpha^l_{i-1/2,j}\right|\right), & \text{if } \left|\alpha^l_{i+1/2,j}\right| + \left|\alpha^l_{i-1/2,j}\right| \neq 0.0 \\ 0.0, & \text{if } \left|\alpha^l_{i+1/2,j}\right| + \left|\alpha^l_{i-1/2,j}\right| = 0.0 \end{cases}; \quad (36)$$

The β parameter at the *(i+½,j)* interface, which introduces the artificial compression term in the algorithm, is given by the following expression:

$$\beta_l = 1.0 + \omega_l MAX\left(\theta^l_{i,j}, \theta^l_{i+1,j}\right), \quad (37)$$

in which ω_l assumes the following values: $\omega_1 = 0.25$ (non-linear field), $\omega_2 = \omega_3 = 1.0$ (linear field) and $\omega_4 = 0.25$ (non-linear field). The numerical characteristic speed, φ_l, at the *(i+½,j)* interface, which is responsible to transport the numerical information associated to the g numerical flux function, is defined by:

$$\varphi_l = \begin{cases} \left(g^l_{i+1,j} - g^l_{i,j}\right)/\alpha^l, & if\ \alpha^l \neq 0.0 \\ 0.0, & if\ \alpha^l = 0.0 \end{cases}. \quad (38)$$

The entropy function is redefined considering φ_l and β_l: $Z_l = \nu_l + \beta_l \varphi_l$, and ψ_l is recalculated according to Eq. (32) or to Eq. (33). Finally, the [3] dissipation function, to second order of spatial accuracy, is constructed by the following matrix-vector product:

$$\left\{D_{YWH}\right\}_{i+1/2,j} = \left[R\right]_{i+1/2,j} \left\{\left(\beta\left(g_{i,j} + g_{i+1,j}\right) - \psi\alpha\right)/\Delta t_{i,j}\right\}_{i+1/2,j}. \quad (39)$$

The convective numerical flux vector to the *(i+½,j)* interface is described by:

$$F^{(l)}_{i+1/2,j} = \left(E^{(l)}_{\text{int}} h_x + F^{(l)}_{\text{int}} h_y\right) V_{\text{int}} + 0.5 D^{(l)}_{YWH}, \quad (40)$$

with:

$$E_{\text{int}}^{(l)} = 0.5\left(E_R^{(l)} + E_L^{(l)}\right) \text{ and } F_{\text{int}}^{(l)} = 0.5\left(F_R^{(l)} + F_L^{(l)}\right). \quad (41)$$

The explicit time integration follows the method presented in the [1] scheme [Eqs. (20) and (21)]. A first order method was implemented as the explicit time integration is used, because only steady state solutions are aimed and, with it, time accurate solutions are not intended.

6 [4] Algorithm

The [4] algorithm, second order accurate in space, follows the Eqs. (4) to (16). The next step is the definition of the entropy condition, which is defined by Eq. (17).

The $\tilde{g}$ function at the *(i+½,j)* interface is defined according to Eq. (34) and the g numerical flux function is given by Eq. (35). The numerical characteristic speed φ_l at the *(i+½,j)* interface is defined according to Eq. (38).

The entropy function is redefined considering φ_l: $Z_l = \nu_l + \varphi_l$, and ψ_l is recalculated according to Eq. (17). Finally, the [4] dissipation function, to second order spatial accuracy, is constructed by the following matrix-vector product:

$$\left\{D_{Harten}\right\}_{i+1/2,j} = \left[R\right]_{i+1/2,j}\left\{\left(g_{i,j} + g_{i+1,j} - \psi\alpha\right)/\Delta t_{i,j}\right\}_{i+1/2,j}. \quad (42)$$

Equations (40) and (41) are used to conclude the numerical flux vector of the [4] scheme and the explicit time integration is performed by the time splitting method defined by Eqs. (20-21).

7 [5] Algorithm

The [5] algorithm, second order accurate in space, follows Eqs. (4) to (16). The next step consists in determining the θ function. This function is defined in terms of the differences of the gradients of the characteristic variables to take into account discontinuities effects and is responsible to artificial compression:

$$\theta_{i,j}^{l} = \begin{cases} \dfrac{\left|\alpha_{i+1/2,j}^{l} - \alpha_{i-1/2,j}^{l}\right|}{\alpha_{i+1/2,j}^{l} + \alpha_{i-1/2,j}^{l}}, & \text{if } \left(\alpha_{i+1/2,j}^{l} + \alpha_{i-1/2,j}^{l}\right) \neq 0.0 \\ 0.0, & \text{if } \left(\alpha_{i+1/2,j}^{l} + \alpha_{i-1/2,j}^{l}\right) = 0.0 \end{cases}. \quad (43)$$

The κ function at the *(i+½,j)* interface is defined as follows:

$$\kappa_l = 1/8\left(1 + \omega_l MAX\left(\theta_{i,j}^{l}, \theta_{i+1,j}^{l}\right)\right), \quad (44)$$

The *g* numerical flux function is determined by:

$$g_{i,j}^{l} = \text{signal}_l \times \text{MAX}\left[0{,}0; \text{MIN}\left(\left|\alpha_{i+1/2,j}^{l}\right|, \alpha_{i-1/2,j}^{l} \times \text{signal}_l\right)\right], \quad (45)$$

where $signal_l$ assumes value 1.0 if $\alpha_{i+1/2,j}^{l} \geq 0.0$ and -1.0 otherwise. The numerical characteristic speed φ_l at the *(i+½,j)* interface is calculated by the following expression:

$$\varphi_l = \begin{cases} \kappa_l\left(g_{i+1,j}^{l} - g_{i,j}^{l}\right)/\alpha^{l}, & \text{if } \alpha^{l} \neq 0.0 \\ 0.0, & \text{if } \alpha^{l} = 0.0 \end{cases}. \quad (46)$$

The ψ_l entropy function at the *(i+½,j)* interface is defined by:

$$\psi_l = \left(\nu_l + \varphi_l\right)^2 + 0.25, \quad (47)$$

with ν_l defined according to Eq. (17). Finally, the [5] dissipation function, to second order spatial accuracy, is constructed by the following matrix-vector product:

$$\left\{D_{Yee/Kutler}\right\}_{i+1/2,j} = \left[R\right]_{i+1/2,j}\left\{\left(\kappa\left(g_{i,j} + g_{i+1,j}\right) - \psi\alpha\right)/\Delta t_{i,j}\right\}_{i+1/2,j}. \quad (48)$$

8 [7] Algorithm

The [7] algorithm, second order accurate in space, follows Eqs. (4) to (16). The next step consists in determining the g numerical flux function. To non-linear fields (l = 1 and 4), it is possible to write:

$$g_{i,j}^{l} = \begin{cases} \dfrac{\alpha_{i+1/2,j}^{l}\alpha_{i-1/2,j}^{l} + \left|\alpha_{i+1/2,j}^{l}\alpha_{i-1/2,j}^{l}\right|}{\alpha_{i+1/2,j}^{l} + \alpha_{i-1/2,j}^{l}}, & \text{if } \left(\alpha_{i+1/2,j}^{l} + \alpha_{i-1/2,j}^{l}\right) \neq 0.0 \\ 0.0, & \text{if } \left(\alpha_{i+1/2,j}^{l} + \alpha_{i-1/2,j}^{l}\right) = 0.0 \end{cases}. \quad (49)$$

To linear fields (l = 2 and 3), it is possible to write:

$$g_{i,j}^{l} = \text{signal}_l \times \text{MAX}\left[0.0; \text{MIN}\left(\left|\alpha_{i-1/2,j}^{l}\right|, \alpha_{i+1/2,j}^{l} \times \text{signal}_l\right)\right], \quad (50)$$

where $signal_l$ is equals to 1.0 if $\alpha^l_{i-1/2,j} \geq 0.0$ and -1.0 otherwise. After that, Equations (17) is employed and the σ_l term at the *(i+½,j)* interface is defined:

$$\sigma_l = 0.5\left(\psi_l - Z_l^2\right). \quad (51)$$

The φ_l numerical characteristic speed at the *(i+½,j)* interface is defined by:

$$\varphi_l = \begin{cases} \sigma_l \left(g^l_{i+1,j} - g^l_{i,j}\right) \big/ \alpha^l, & if\ \alpha^l \neq 0.0 \\ 0.0, & if\ \alpha^l = 0.0 \end{cases}. \quad (52)$$

The entropy function is redefined considering the φ_l term: $Z_l = \nu_l + \varphi_l$ and ψ_l is recalculated according to Eq. (18). Finally, the [7] dissipation function, to second order accuracy in space, is constructed by the following matrix-vector product:

$$\left\{D_{Hughson/Beran}\right\}_{i+1/2,j} = \left[R\right]_{i+1/2,j} \left\{\left(\sigma\left(g_{i,j} + g_{i+1,j}\right) - \psi\alpha\right) \big/ \Delta t_{i,j}\right\}_{i+1/2,j}. \quad (53)$$

Equations (40) and (41) are used to conclude the numerical flux vector of [7] scheme and the explicit time integration is performed by the time splitting method defined by Eqs. (20) and (21).

9 MUSCL Procedure

Second order spatial accuracy can be achieved by introducing more upwind points or cells in the schemes. It has been noted that the projection stage, whereby the solution is projected in each cell face (i-1/2,j; i+1/2,j) on piecewise constant states, is the cause of the first order space accuracy of the [8] schemes ([9]). Hence, it is sufficient to modify the first projection stage without modifying the Riemann solver, in order to generate higher spatial approximations. The state variables at the interfaces are thereby obtained from an extrapolation between neighboring cell averages. This method for the generation of second order upwind schemes based on variable extrapolation is often referred to in the literature as the MUSCL ("Monotone Upstream-centered Schemes for Conservation Laws") approach. The use of nonlinear limiters in such procedure, with the intention of restricting the amplitude of the gradients appearing in the solution, avoiding thus the formation of new extrema, allows that first order upwind schemes be transformed in TVD high resolution schemes with the appropriate definition of such nonlinear limiters, assuring monotone preserving and total variation diminishing methods. Details of the present implementation of the MUSCL procedure, as well the incorporation of TVD properties to the schemes, are found in [9]. The expressions to calculate de fluxes following a MUSCL procedure and the nonlinear flux limiter definitions employed in this work, which incorporates TVD properties, are defined as follows.

The conserved variables at the interface *(i+½,j)* can be considered as resulting from a combination of backward and forward extrapolations. To a linear one-sided extrapolation at the interface between the averaged values at the two upstream cells (i,j) and (i-1,j), one has:

$$Q^L_{i+1/2,j} = Q_{i,j} + \frac{\varepsilon}{2}\left(Q_{i,j} - Q_{i-1,j}\right), \text{ cell (i,j);} \quad (55)$$

$$Q^R_{i+1/2,j} = Q_{i+1,j} - \frac{\varepsilon}{2}\left(Q_{i+2,j} - Q_{i+1,j}\right), \text{ cell (i+1,j),} \quad (56)$$

leading to a second order fully one-sided scheme. If the first order scheme is defined by the numerical flux

$$F_{i+1/2,j} = F\left(Q_{i,j}, Q_{i+1,j}\right) \quad (57)$$

the second order space accurate numerical flux is obtained from

$$F^{(2)}_{i+1/2,j} = F\left(Q^L_{i+1/2,j}, Q^R_{i+1/2,j}\right). \quad (58)$$

Higher order flux vector splitting or flux difference splitting methods, such as those studied in this work, are obtained from:

$$F^{(2)}_{i+1/2,j} = F^+\left(Q^L_{i+1/2,j}\right) + F^-\left(Q^R_{i+1/2,j}\right). \quad (59)$$

All second order upwind schemes necessarily involve at least five mesh points or cells. To reach high order solutions without oscillations around discontinuities, nonlinear limiters are employed, replacing the term ε in Eqs. (55) and (56) by these limiters at the left and at the right states of the flux interface. To define such limiters, it is necessary to calculate the ratio of consecutive variations of the conserved variables. These ratios are defined as follows:

$$r^+_{i-1/2,j} = \left(Q_{i+1,j} - Q_{i,j}\right) \big/ \left(Q_{i,j} - Q_{i-1,j}\right) \text{ and}$$
$$r^+_{i+1/2,j} = \left(Q_{i+2,j} - Q_{i+1j}\right) \big/ \left(Q_{i+1,j} - Q_{i,j}\right), \quad (60)$$

where the nonlinear limiters at the left and at the right states of the flux interface are defined by $\Psi^L = \Psi\left(r^+_{i-1/2,j}\right)$ and $\Psi^R = \Psi\left(1/r^+_{i+1/2,j}\right)$. In this work, five options of nonlinear limiters were considered to the numerical experiments. These limiters are defined as follows:

$$\Psi_l^{VL}(r_l) = \frac{r_l + |r_l|}{1 + r_l}, \text{ [19] limiter;} \quad (61)$$

$$\Psi_l^{VA}(r_l) = \frac{r_l + r_l^2}{1 + r_l^2}, \text{ Van Albada limiter;} \quad (62)$$

$$\Psi_l^{MIN}(r_l) = signal_l MAX\left(0, MIN\left(|r_l|, signal_l\right)\right), \text{ minmod limiter;} \quad (63)$$

$$\Psi_l^{SB}(r_l) = MAX\left(0, MIN\left(2r_l, 1\right), MIN\left(r_l, 2\right)\right), \text{ "Super Bee" limiter, due to [20];} \quad (64)$$

$$\Psi_l^{\beta-L}(r_l) = MAX\left(0, MIN\left(\beta r_l, 1\right), MIN\left(r_l, \beta\right)\right), \text{ β-limiter,} \quad (65)$$

with "l" varying from 1 to 4 (two-dimensional space), $signal_l$ being equal to 1.0 if $r_l \geq 0.0$ and -1.0 otherwise, r_l is the ratio of consecutive variations of the lth conserved variable and β is a parameter assuming values between 1.0 and 2.0, being 1.5 the value assumed in this work. With the implementation of the numerical flux vectors of [1-2] following this MUSCL procedure, second order spatial accuracy and TVD properties are incorporated in the algorithms.

10 Implicit Formulations

All implicit schemes studied in this work used an ADI formulation to solve the algebraic nonlinear system of equations. Initially, the nonlinear system of equations is linearized considering the implicit operator evaluated at the time "n" and, posteriorly, the five-diagonal system of linear algebraic equations is factored in two three-diagonal systems of linear algebraic equations, each one associated with a particular spatial direction. Thomas algorithm is employed to solve these two three-diagonal systems. The implicit schemes studied in this work were only applicable to the solution of the Euler equations, which implies that only the convective contributions were considered in the RHS ("Right Hand Side") operator.

10.1 [1] TVD implicit scheme

The ADI form of the [1] TVD scheme is defined by the following two step algorithm:

$$\left\{I + \Delta t_{i,j}\Delta^-_\xi K^+_{i+1/2,j} + \Delta t_{i,j}\Delta^+_\xi K^-_{i+1/2,j}\right\}\Delta Q^*_{i,j} = \left[RHS_{(Roe)}\right]^n_{i,j}, \text{ to the } \xi \text{ direction;} \quad (66)$$

$$\left\{I + \Delta t_{i,j}\Delta^-_\eta J^+_{i,j+1/2} + \Delta t_{i,j}\Delta^+_\eta J^-_{i,j+1/2}\right\}\Delta Q^{n+1}_{i,j} = \Delta Q^*_{i,j}, \text{ to the } \eta \text{ direction;} \quad (67)$$

$$Q^{n+1}_{i,j} = Q^n_{i,j} + \Delta Q^{n+1}_{i,j}, \quad (68)$$

where:

$$K^\pm_{i\pm1/2,j} = \left[R\right]^n_{i\pm1/2,j}\Omega^\pm_{i\pm1/2,j}\left[R^{-1}\right]^n_{i\pm1/2,j};$$
$$J^\pm_{i,j\pm1/2} = \left[R\right]^n_{i,j\pm1/2}\Phi^\pm_{i,j\pm1/2}\left[R^{-1}\right]^n_{i,j\pm1/2}; \quad (69)$$
$$\Omega^\pm_{i\pm1/2,j} = diag\left[\left(\lambda^l_\xi\right)^\pm\right]^n_{i\pm1/2,j};$$
$$\Phi^\pm_{i,j\pm1/2} = diag\left[\left(\lambda^l_\eta\right)^\pm\right]^n_{i,j\pm1/2}; \quad (70)$$
$$\left(\lambda^l_\xi\right)^\pm = 0.5\left(\lambda^l_\xi \pm \left|\lambda^l_\xi\right|\right),\ \left(\lambda^l_\eta\right)^\pm = 0.5\left(\lambda^l_\eta \pm \left|\lambda^l_\eta\right|\right),$$
$$\Delta^-_\xi = (\cdot)_{i,j} - (\cdot)_{i-1,j},\ \Delta^+_\xi = (\cdot)_{i+1,j} - (\cdot)_{i,j}; \quad (71)$$
$$\Delta^-_\eta = (\cdot)_{i,j} - (\cdot)_{i,j-1},\ \Delta^+_\eta = (\cdot)_{i,j+1} - (\cdot)_{i,j}. \quad (72)$$

In Equation (70), $diag[\cdot]$ is a diagonal matrix; in Eqs. (70) and (71), "l" assumes values from 1 to 4 and λ's are the eigenvalues of the Euler equations, described by Eq. (9). The matrix R^{-1} is defined as:

$$R^{-1} = \begin{bmatrix} \frac{1}{2}\left[\frac{\gamma-1}{a^2_{\text{int}}}\frac{\left(u^2_{\text{int}}+v^2_{\text{int}}\right)}{2} + \frac{1}{a_{\text{int}}}\left(u_{\text{int}}h'_x + v_{\text{int}}h'_y\right)\right] & \frac{1}{2}\left[-\frac{\gamma-1}{a^2_{\text{int}}}u_{\text{int}} - \frac{h'_x}{a_{\text{int}}}\right] & \frac{1}{2}\left[-\frac{\gamma-1}{a^2_{\text{int}}}v_{\text{int}} - \frac{h'_y}{a_{\text{int}}}\right] & \frac{\gamma-1}{2a^2_{\text{int}}} \\ 1 - \frac{\gamma-1}{a^2_{\text{int}}}\frac{\left(u^2_{\text{int}}+v^2_{\text{int}}\right)}{2} & \frac{\gamma-1}{a^2_{\text{int}}}u_{\text{int}} & \frac{\gamma-1}{a^2_{\text{int}}}v_{\text{int}} & -\frac{\gamma-1}{a^2_{\text{int}}} \\ -\left(h'_x v_{\text{int}} - h'_y u_{\text{int}}\right) & -h'_y & h'_x & 0 \\ \frac{1}{2}\left[\frac{\gamma-1}{a^2_{\text{int}}}\frac{\left(u^2_{\text{int}}+v^2_{\text{int}}\right)}{2} - \frac{1}{a_{\text{int}}}\left(u_{\text{int}}h'_x + v_{\text{int}}h'_y\right)\right] & \frac{1}{2}\left[-\frac{\gamma-1}{a^2_{\text{int}}}u_{\text{int}} + \frac{h'_x}{a_{\text{int}}}\right] & \frac{1}{2}\left[-\frac{\gamma-1}{a^2_{\text{int}}}v_{\text{int}} + \frac{h'_y}{a_{\text{int}}}\right] & \frac{\gamma-1}{2a^2_{\text{int}}} \end{bmatrix}, \quad (73)$$

The interface properties are defined either by arithmetical average or by [1] average. In this work, the arithmetical average was used. The $RHS_{(Roe)}$ operator required in Eq. (66) is defined as:

$$\left[RHS_{(Roe)}\right]_{i,j}^{n} = -\Delta t_{i,j}/V_{i,j}\left(F_{i+1/2,j}^{(Roe)} - F_{i-1/2,j}^{(Roe)} + F_{i,j+1/2}^{(Roe)} - F_{i,j-1/2}^{(Roe)}\right)^{n}, \quad (74)$$

with $F_{i+1/2,j}^{(Roe)}$ calculated according to Eq. (19). This implementation is first order accurate in time due to the definition of Ω and of Φ, as reported in [21], but is second order accurate in space due to the RHS solution at the steady state, when a MUSCL procedure is employed.

10.2 [2] TVD implicit scheme

The ADI form of the [2] TVD scheme is defined by the following two step algorithm:

$$\left\{I + \Delta t_{i,j}\Delta_{\xi}^{-}A_{i+1/2,j}^{+} + \Delta t_{i,j}\Delta_{\xi}^{+}A_{i+1/2,j}^{-}\right\}\Delta Q_{i,j}^{*} = \left[RHS_{(VL)}\right]_{i,j}^{n},$$

to the ξ direction; (75)

$$\left\{I + \Delta t_{i,j}\Delta_{\eta}^{-}B_{i,j+1/2}^{+} + \Delta t_{i,j}\Delta_{\eta}^{+}B_{i,j+1/2}^{-}\right\}\Delta Q_{i,j}^{n+1} = \Delta Q_{i,j}^{*},$$

to the η direction; (76)

$$Q_{i,j}^{n+1} = Q_{i,j}^{n} + \Delta Q_{i,j}^{n+1}, \quad (77)$$

where the matrices $A^{\pm}$ and $B^{\pm}$ are defined as:

$$\begin{aligned} A_{i\pm1/2,j}^{\pm} &= \left[T\right]_{i\pm1/2,j}^{n}\Omega_{i\pm1/2,j}^{\pm}\left[T^{-1}\right]_{i\pm1/2,j}^{n}; \\ B_{i,j\pm1/2}^{\pm} &= \left[T\right]_{i,j\pm1/2}^{n}\Phi_{i,j\pm1/2}^{\pm}\left[T^{-1}\right]_{i,j\pm1/2}^{n}; \end{aligned} \quad (78)$$

$$\begin{aligned} \Omega_{i\pm1/2,j}^{\pm} &= diag\left[\left(\lambda_{\xi}^{l}\right)^{\pm}\right]_{i\pm1/2,j}^{n}; \\ \Phi_{i,j\pm1/2}^{\pm} &= diag\left[\left(\lambda_{\eta}^{l}\right)^{\pm}\right]_{i,j\pm1/2}^{n}; \end{aligned} \quad (79)$$

with the similarity transformation matrices defined by:

$$T = \begin{bmatrix} 1 & 0 & \alpha \\ u_{\text{int}} & h_y^{'}\rho_{\text{int}} & \alpha\left(u_{\text{int}} + h_x^{'}a_{\text{int}}\right) \\ v_{\text{int}} & -h_x^{'}\rho_{\text{int}} & \alpha\left(v_{\text{int}} + h_y^{'}a_{\text{int}}\right) \\ \dfrac{\phi^2}{\gamma-1} & \rho_{\text{int}}\left(h_y^{'}u_{\text{int}} - h_x^{'}v_{\text{int}}\right) & \alpha\left(\dfrac{\phi^2 + a_{\text{int}}^2}{\gamma-1} + a_{\text{int}}\tilde{\theta}\right) \end{bmatrix}$$

$$\begin{bmatrix} \alpha \\ \alpha\left(u_{\text{int}} - h_x^{'}a_{\text{int}}\right) \\ \alpha\left(v_{\text{int}} - h_y^{'}a_{\text{int}}\right) \\ \alpha\left(\dfrac{\phi^2 + a_{\text{int}}^2}{\gamma-1} - a_{\text{int}}\tilde{\theta}\right) \end{bmatrix}; \quad (80)$$

$$\alpha = \rho_{\text{int}}/\left(\sqrt{2}a_{\text{int}}\right),\ \beta = 1/\left(\sqrt{2}\rho_{\text{int}}a_{\text{int}}\right); \quad (81)$$

$$\phi^2 = (\gamma-1)\frac{u_{\text{int}}^2 + v_{\text{int}}^2}{2},\ \tilde{\theta} = h_x^{'}u_{\text{int}} + h_y^{'}v_{\text{int}}; \quad (82)$$

$$T^{-1} = \begin{bmatrix} 1 - \dfrac{\phi^2}{a_{\text{int}}^2} & (\gamma-1)\dfrac{u_{\text{int}}}{a_{\text{int}}^2} & (\gamma-1)\dfrac{v_{\text{int}}}{a_{\text{int}}^2} & -\dfrac{\gamma-1}{a_{\text{int}}^2} \\ -\dfrac{h_y^{'}u_{\text{int}} - h_x^{'}v_{\text{int}}}{\rho_{\text{int}}} & \dfrac{h_y^{'}}{\rho_{\text{int}}} & -\dfrac{h_x^{'}}{\rho_{\text{int}}} & 0 \\ \beta\left(\phi^2 - a_{\text{int}}\tilde{\theta}\right) & \beta\left[h_x^{'}a_{\text{int}} - (\gamma-1)u_{\text{int}}\right] & \beta\left[h_y^{'}a_{\text{int}} - (\gamma-1)v_{\text{int}}\right] & \beta(\gamma-1) \\ \beta\left(\phi^2 + a_{\text{int}}\tilde{\theta}\right) & -\beta\left[h_x^{'}a_{\text{int}} + (\gamma-1)u_{\text{int}}\right] & -\beta\left[h_y^{'}a_{\text{int}} + (\gamma-1)v_{\text{int}}\right] & \beta(\gamma-1) \end{bmatrix}. \quad (83)$$

The properties defined at interface are calculated by arithmetical average. The $RHS_{(VL)}$ operator required in Eq. (75) is defined as:

$$\left[RHS_{(VL)}\right]_{i,j}^{n} = -\Delta t_{i,j}/V_{i,j}\left(R_{i+1/2,j}^{(VL)} - R_{i-1/2,j}^{(VL)} + R_{i,j+1/2}^{(VL)} - R_{i,j-1/2}^{(VL)}\right)^{n}, \quad (84)$$

with the numerical flux vector $R_{i+1/2,j}^{(VL)}$ calculated according to Eq. (24).

10.3 [3-5; 7] TVD implicit schemes

In schemes [3-5; 7] studied in this work, a backward Euler method is applied followed by an ADI approximate factorization to solve a resulting three-diagonal system in each direction. The ADI form to these four schemes is defined by the following two-steps algorithm:

$$\left\{I - \Delta t_{i,j}J_{i+1/2,j}^{-}\Delta_{\xi}^{+} + \Delta t_{i,j}J_{i-1/2,j}^{+}\Delta_{\xi}^{-}\right\}^{n}\Delta Q_{i,j}^{*} = RHS_{i,j}^{n},$$

in the ξ direction; (85)

$$\left\{I - \Delta t_{i,j}K_{i,j+1/2}^{-}\Delta_{\eta}^{+} + \Delta t_{i,j}K_{i,j-1/2}^{+}\Delta_{\eta}^{-}\right\}^{n}\Delta Q_{i,j}^{n+1} = \Delta Q_{i,j}^{*},$$

in the η direction; (86)

$$Q_{i,j}^{n+1} = Q_{i,j}^{n} + \Delta Q_{i,j}^{n+1}, \quad (87)$$

where:

$$J^{\pm}_{i\pm1/2,j} = \left[R\right]^{n}_{i\pm1/2,j}\,\Omega^{\pm}_{i\pm1/2,j}\left[R^{-1}\right]^{n}_{i\pm1/2,j};$$
$$K^{\pm}_{i,j\pm1/2} = \left[R\right]^{n}_{i,j\pm1/2}\,\Phi^{\pm}_{i,j\pm1/2}\left[R^{-1}\right]^{n}_{i,j\pm1/2}; \quad (88)$$

$$\Omega^{\pm}_{i\pm1/2,j} = diag\left\{1/2\left[Q\left(\lambda^{l}_{\xi}+\gamma^{l}_{\xi}\right)\pm\left(\lambda^{l}_{\xi}+\gamma^{l}_{\xi}\right)\right]^{n}_{i\pm1/2,j}\right\}; \quad (89)$$

$$\Phi^{\pm}_{i,j\pm1/2} = diag\left\{1/2\left[Q\left(\lambda^{l}_{\eta}+\gamma^{l}_{\eta}\right)\pm\left(\lambda^{l}_{\eta}+\gamma^{l}_{\eta}\right)\right]^{n}_{i,j\pm1/2}\right\}; \quad (90)$$

$$\left(\gamma^{l}_{\xi}\right)_{i+1/2,j} = \begin{cases}\left(g^{l}_{i+1,j}-g^{l}_{i,j}\right)/\alpha^{l}, & if\ \alpha^{l}\neq 0.0\\ 0.0, & if\ \alpha^{l}=0.0\end{cases}; \quad (91)$$

$$\Delta^{-}_{\xi} = \left(\cdot\right)_{i,j} - \left(\cdot\right)_{i-1,j},\ \Delta^{+}_{\xi} = \left(\cdot\right)_{i+1,j} - \left(\cdot\right)_{i,j};$$

$$\Delta^{-}_{\eta} = \left(\cdot\right)_{i,j} - \left(\cdot\right)_{i,j-1},\ \Delta^{+}_{\eta} = \left(\cdot\right)_{i,j+1} - \left(\cdot\right)_{i,j}. \quad (92)$$

In Equation (88), the R matrix is defined by Eq. (16); in Eqs. (89-91), "*l*" varies from 1 to 4 (two-dimensional case); h'_x and h'_y are defined by Eq. (15); and $g^{l}_{i,j}$ is defined by:

$$g^{l}_{i,j} = signal_l \times MAX\left[0,0;\ MIN\left(\sigma^{l}_{i+1/2,j}\left|\alpha^{l}_{i+1/2,j}\right|,,\ signal_l\times\sigma^{l}_{i-1/2,j}\alpha^{l}_{i-1/2,j}\right)\right] \quad (93)$$

where $signal_l$ is equal to 1.0 if $\alpha^{l}_{i+1/2,j} \geq 0.0$ and -1.0 otherwise; $\sigma^{l}\left(\lambda_l\right) = 0.5Q_l\left(\lambda_l\right)$; and *Q*, the entropy function, is determined by:

$$Q_l\left(W_l\right) = \begin{cases}\left|W_l\right|, & if\ \left|W_l\right|\geq\delta_f\\ 0.5\left(W_l^2+\delta_f^2\right)/\delta_f, & if\ \left|W_l\right|<\delta_f\end{cases}, \quad (94)$$

with δ_f assuming values between 0.1 and 0.5, being 0.2 the value recommended by [21].

The $RHS^{n}_{i,j}$ operator is determined by the [3-5; 7] schemes as:

$$\left[RHS\right]^{n}_{i,j} = -\Delta t/V_{i,j}\left(F^{Scheme}_{i+1/2,j} - F^{Scheme}_{i-1/2,j} + F^{Scheme}_{i,j+1/2} - F^{Scheme}_{i,j-1/2}\right)^{n}, \quad (95)$$

where the superscript "Scheme" of the numerical flux vectors is related to the scheme under analysis, being: Scheme = YWH to the [3] algorithm; Scheme = H to the [4] algorithm; Scheme = YK to the [5] algorithm; and Scheme = HB to the [7] algorithm.

This implementation is second order accurate in space and first order accurate in time, appropriated to steady state problems, conform definition of Ω and Φ (details in [21]).

Schemes [3,5; 7] studied in this work present steady state solutions which depend of the time step; hence, in the implicit use of these algorithms a high CFL number does not can be considered, because the solution could be destroyed. Schemes with the "RHS" defined as function of the time step have this problem - time step dependent solutions.

11 Spatially Variable Time Step

The basic idea of this procedure consists in keeping constant the CFL number in all calculation domain, allowing, hence, the use of appropriated time steps to each specific mesh region during the convergence process. Hence, according to the definition of the CFL number, it is possible to write:

$$\Delta t_{i,j} = CFL\left(\Delta s\right)_{i,j}/c_{i,j}, \quad (96)$$

where CFL is the "Courant-Friedrichs-Lewy" number to provide numerical stability to the scheme; $c_{i,j} = \left[\left(u^2+v^2\right)^{0.5}+a\right]_{i,j}$ is the maximum characteristic speed of information propagation in the calculation domain; and $\left(\Delta s\right)_{i,j}$ is a characteristic length of information transport. On a finite volume context, $\left(\Delta s\right)_{i,j}$ is chosen as the minor value found between the minor centroid distance, involving the *(i,j)* cell and a neighbor, and the minor cell side length.

12 Initial and Boundary Conditions

12.1 Initial condition

To the physical problems studied in this work, freestream flow values are adopted for all properties as initial condition, in the whole calculation domain ([22-23]). Therefore, the vector of conserved variables is defined as:

$$Q_{i,j} = \left\{1 \quad M_\infty\cos\alpha \quad M_\infty\sin\alpha \quad \frac{1}{\gamma(\gamma-1)}+0.5M_\infty^2\right\}^{T}, \quad (97)$$

being M_∞ the freestream flow Mach number and α the flow attack angle.

12.2 Boundary conditions

The boundary conditions are basically of three types: solid wall, entrance and exit. These conditions are implemented in special cells named ghost cells.

(a) Wall condition: This condition imposes the flow tangency at the solid wall. This condition is satisfied considering the wall tangent velocity component of the ghost volume as equals to the respective velocity component of its real neighbor cell. At the same way, the wall normal velocity component of the ghost cell is equaled in value, but with opposite signal, to the respective velocity component of the real neighbor cell.

The pressure gradient normal to the wall is assumed be equal to zero, following an inviscid formulation. The same hypothesis is applied to the temperature gradient normal to the wall, considering adiabatic wall. The ghost volume density and pressure are extrapolated from the respective values of the real neighbor volume (zero order extrapolation), with these two conditions. The total energy is obtained by the state equation of a perfect gas.

(b) Entrance condition:

(b.1) Subsonic flow: Three properties are specified and one is extrapolated, based on analysis of information propagation along characteristic directions in the calculation domain ([23]). In other words, three characteristic directions of information propagation point inward the computational domain and should be specified. Only the characteristic direction associated to the "$(q_n\text{-}a)$" velocity cannot be specified and should be determined by interior information of the calculation domain. The pressure was the extrapolated variable from the real neighbor volume, to the studied problems. Density and velocity components had their values determined by the freestream flow properties. The total energy per unity fluid volume is determined by the state equation of a perfect gas.

(b.2) Supersonic flow: All variables are fixed with their freestream flow values.

(c) Exit condition:

(c.1) Subsonic flow: Three characteristic directions of information propagation point outward the computational domain and should be extrapolated from interior information ([23]). The characteristic direction associated to the "$(q_n\text{-}a)$" velocity should be specified because it penetrates the calculation domain. In this case, the ghost volume's pressure is specified by its freestream value. Density and velocity components are extrapolated and the total energy is obtained by the state equation of a perfect gas.

(c.2) Supersonic flow: All variables are extrapolated from the interior domain due to the fact that all four characteristic directions of information propagation of the Euler equations point outward the calculation domain and, with it, nothing can be fixed.

13 Results

Tests were performed in a personal computer (notebook) with Pentium dual core processor of 2.20GHz of clock and 2.0Gbytes of RAM memory. Converged results occurred to 4 orders of reduction in the value of the maximum residual. The maximum residual is defined as the maximum value obtained from the discretized conservation equations. The value used to γ was 1.4. To all problems, the attack angle was adopted equal to 0.0°.

In the present results, the following nomenclature is used to represent the studied schemes:

R81 – Represent [1] solutions;
VL82 – Represent [2] solutions
YWH82 – Represent [3] solutions;
H83 - Represent [4] solutions;
YK85 - Represent [5] solutions;
HB91 - Represent [7] solutions.

The reference to the limiters is also abbreviated: Van Leer limiter (VL), Van Albada limiter (VA), minmod limiter (Min), Super Bee limiter (SB) and β-limiter (BL).

To the compression corner physical problem, an algebraic mesh with 60x40 points was used, which is composed of 2,301 rectangular cells and of 2,400 nodes, on a finite volume context. The compression corner configuration is described in Fig. 2.

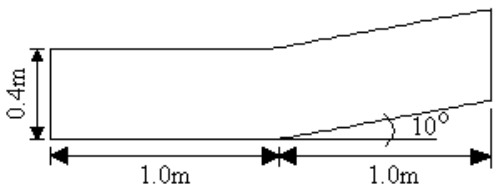

Figure 2 : Compression corner configuration.

The compression corner mesh employed in this work is presented in Fig. 3. The initial condition to the compression corner problem adopts a freetream Mach number of 3.0, which represents a moderate supersonic flow.

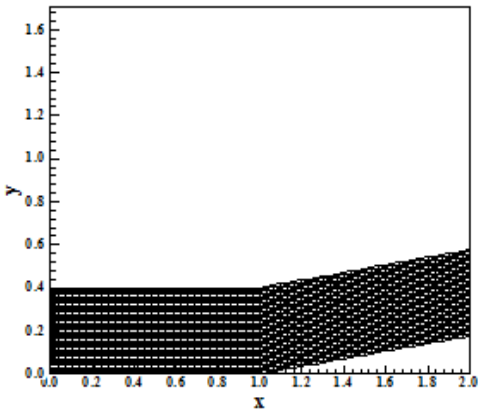

Figure 3 : Compression corner mesh.

13.1 Corner results – Explicit case

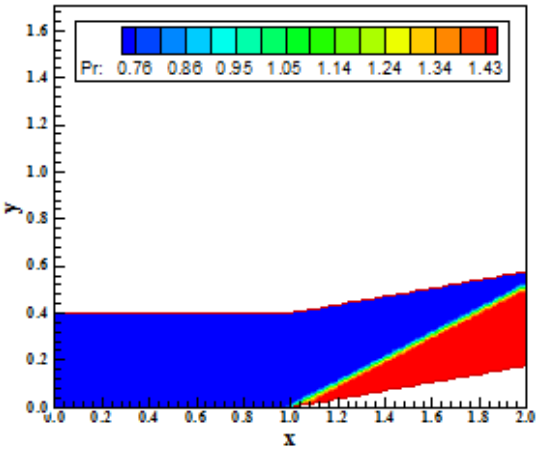

Figure 4 : Pressure contours (R81-VL).

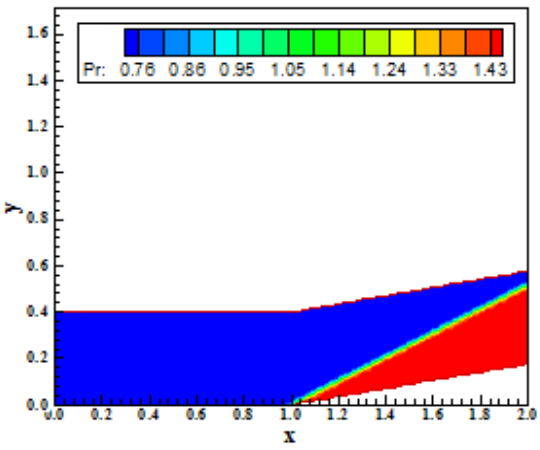

Figure 5 : Pressure contours (R81-VA).

Figures 4 to 17 show the pressure field obtained by the R81, in its five variants; the VL82, in its five variants; the YWH82; the H83; the YK85; and the HB91 schemes, respectively. The pressure field generated by the R81 scheme in its SB variant is the most severe in relation to the other schemes.

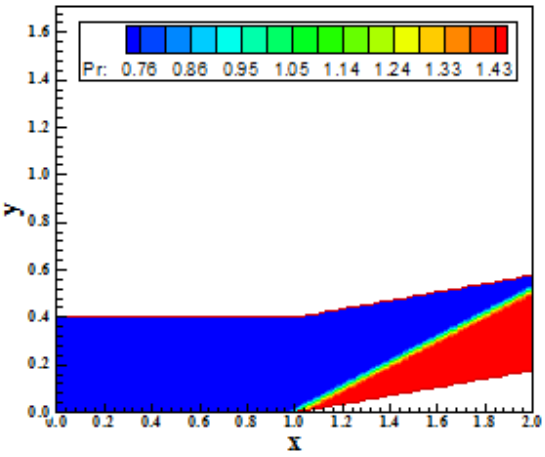

Figure 6 : Pressure contours (R81-Min).

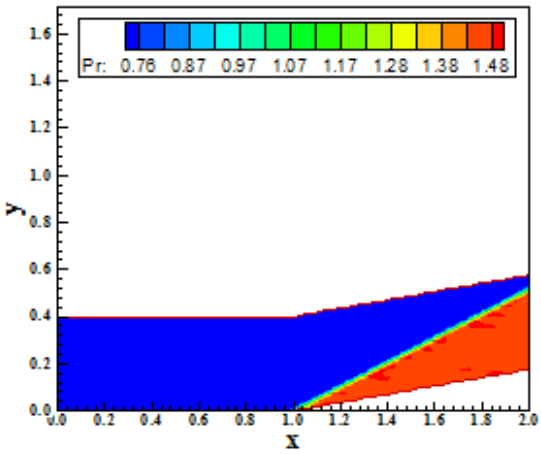

Figure 7 : Pressure contours (R81-SB).

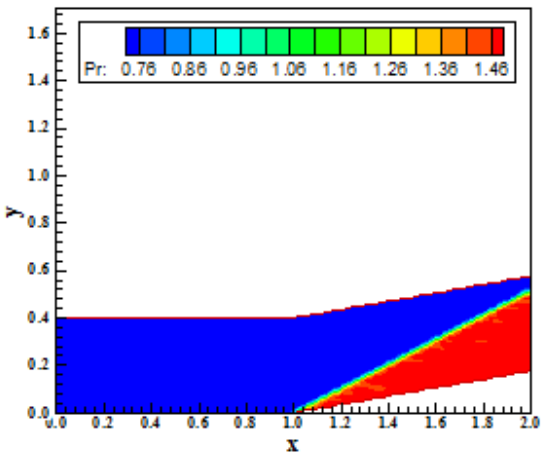

Figure 8 : Pressure contours (R81-BL).

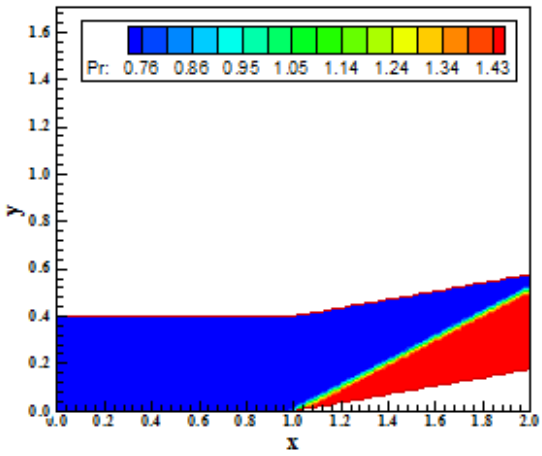

Figure 9 : Pressure contours (VL82-VL).

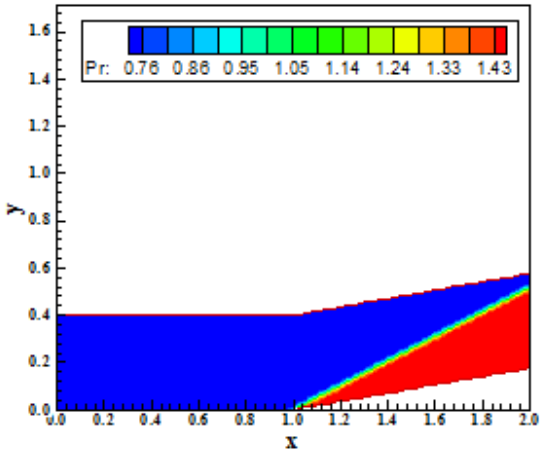

Figure 10 : Pressure contours (VL82-VA).

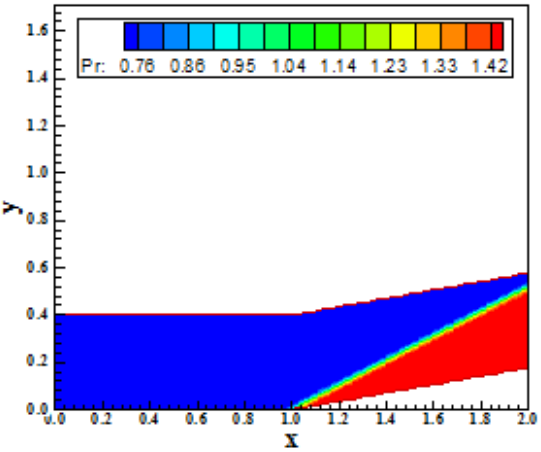

Figure 11 : Pressure contours (VL82-Min).

Figures 18 to 31 exhibit the Mach number field generated by the R81, in its five variants; the VL82, in its five variants; YWH82; the H83; the YK85; and the HB91 schemes, respectively. The Mach number contours generated by the VL82 scheme in its SB variant are the most intense field in relation to the other schemes.

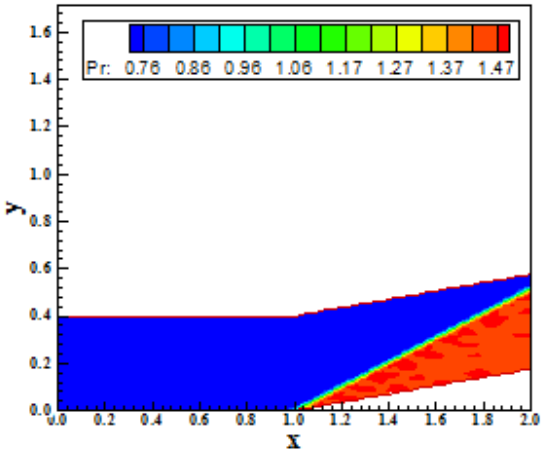

Figure 12 : Pressure contours (VL82-SB).

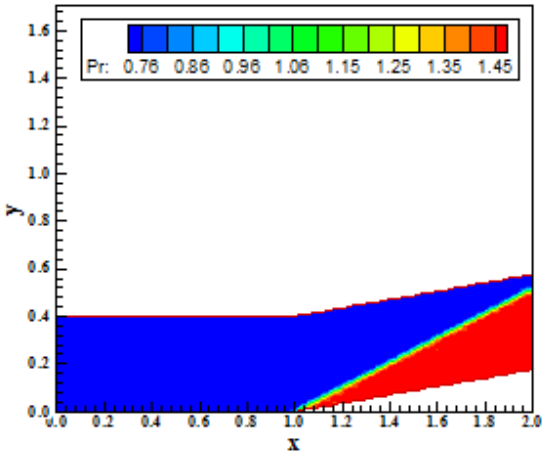

Figure 13 : Pressure contours (VL82-BL).

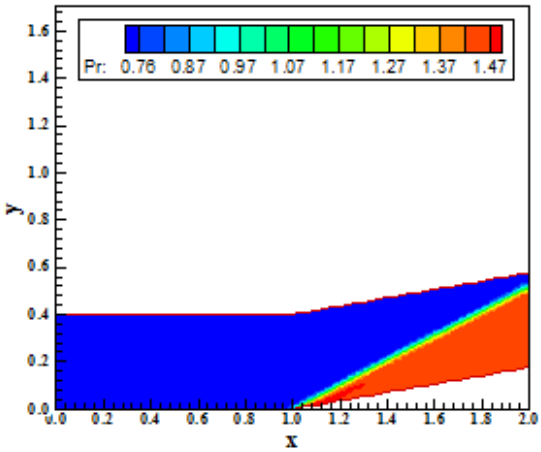

Figure 14 : Pressure contours (YWH82).

Figure 32 shows the wall pressure distributions obtained by all variants of the R81 TVD scheme. They are compared with the oblique shock wave

theory results. As can be observed, some solutions present overshoot at the compression corner, mainly the R81 TVD scheme using the SB limiter.

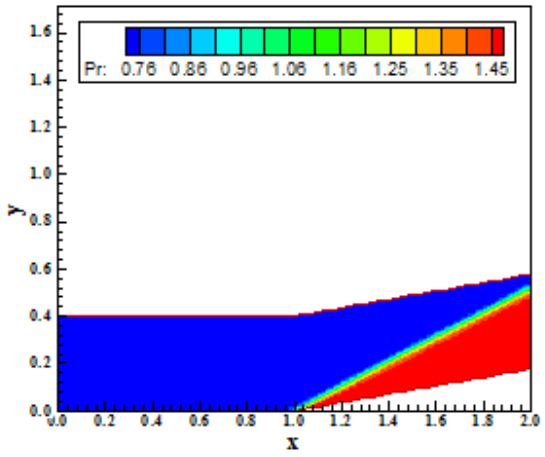

Figure 15 : Pressure contours (H83).

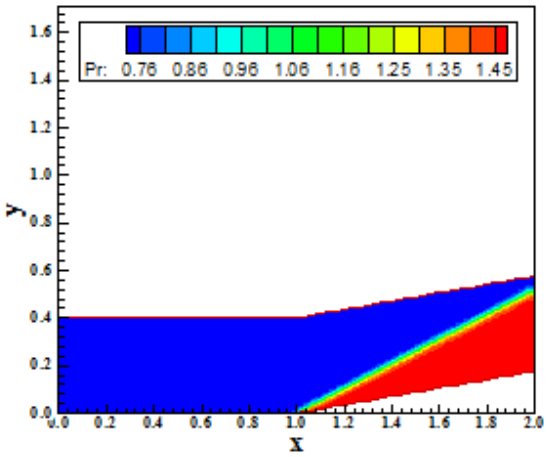

Figure 16 : Pressure contours (YK85).

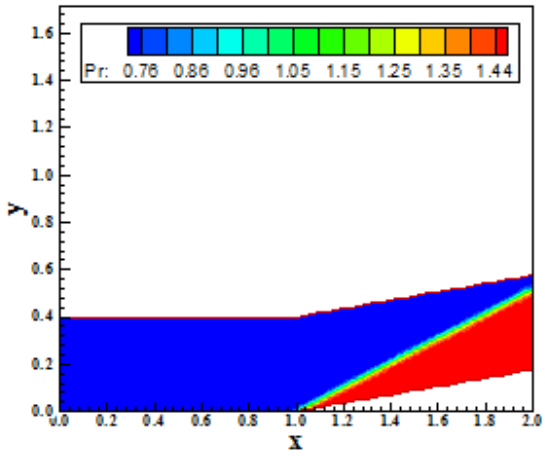

Figure 17 : Pressure contours (HB91).

Figure 33 exhibits the wall pressure distribution obtained by the R81 TVD scheme using VL, VA and Min limiters. As noted, no overshoot or undershoot are observed in the solutions, presenting these ones a smooth behaviour. It is also possible to observe that the shock discontinuity is captured within four cells, which is also a typical number of cells encountered in high resolution schemes to capture accurately shock waves. So the accuracy of the R81 TVD scheme with these three limiters is in accordance with typical results of current high resolution schemes.

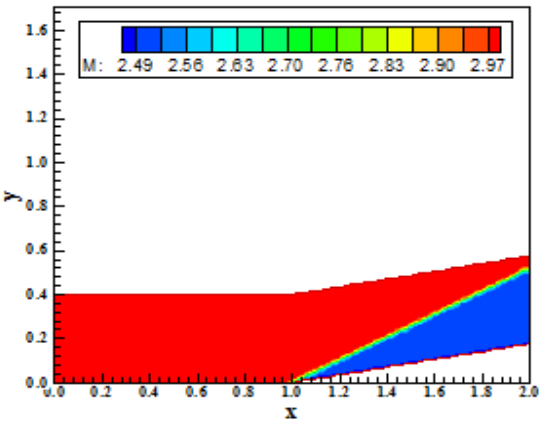

Figure 18 : Mach number contours (R81-VL).

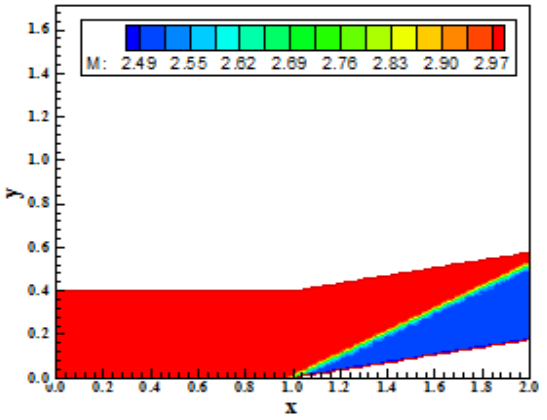

Figure 19 : Mach number contours (R81-VA).

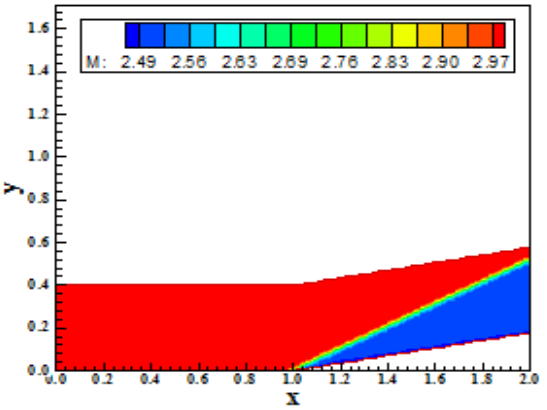

Figure 20 : Mach number contours (R81-Min).

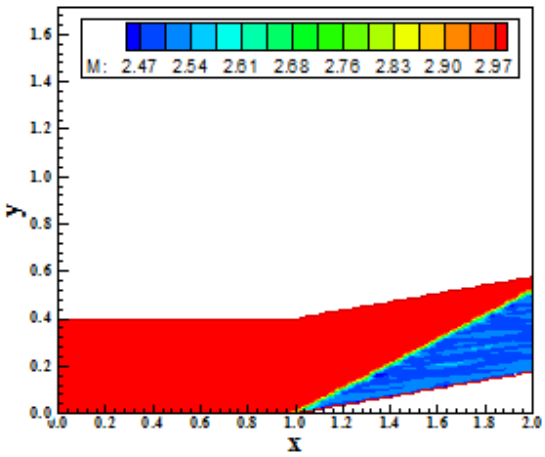

Figure 21 : Mach number contours (R81-SB).

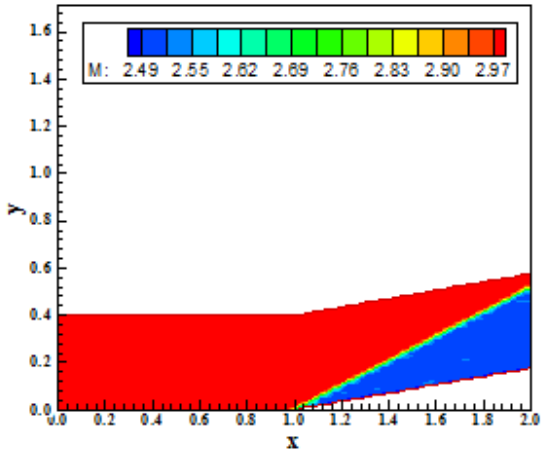

Figure 22 : Mach number contours (R81-BL).

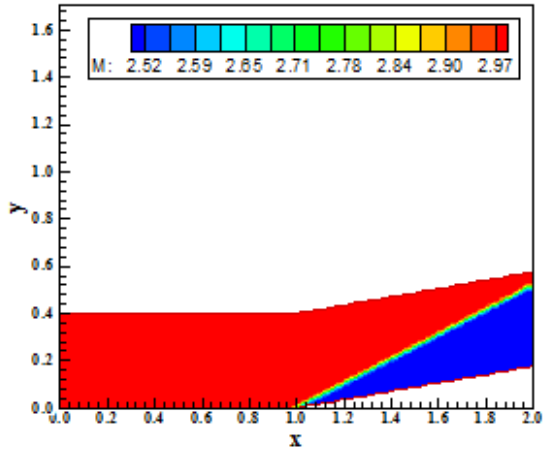

Figure 23 : Mach number contours (VL82-VL).

Figure 34 shows the wall pressure distributions obtained by the R81 TVD scheme using the SB and the BL limiters. The SB limiter yields a pronounced overshoot, but the shock is also captured in four cells, as is the case with the BL limiter. By the results, the best solutions were obtained with VL, VA and Min limiters because detect sharp and smooth pressure distributions at the corner wall.

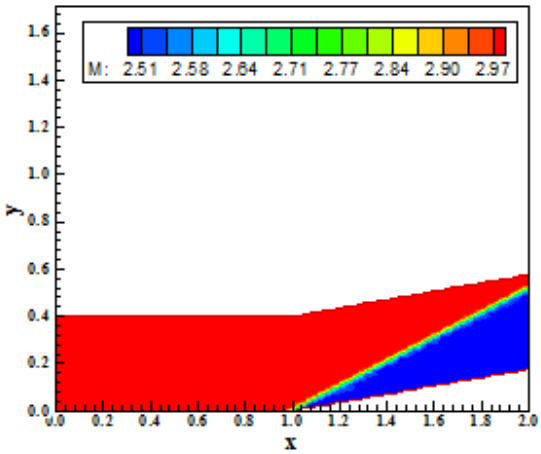

Figure 24 : Mach number contours (VL82-VA).

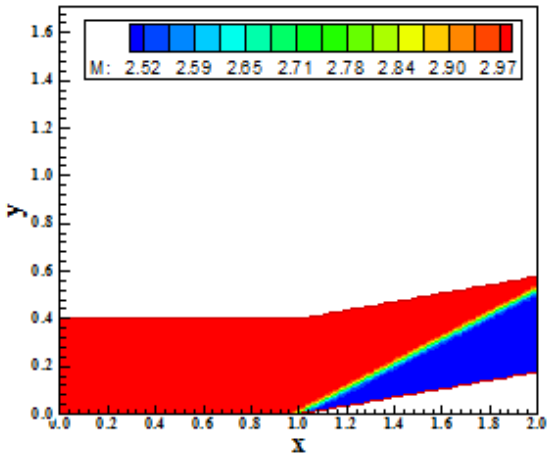

Figure 25 : Mach number contours (VL82-Min).

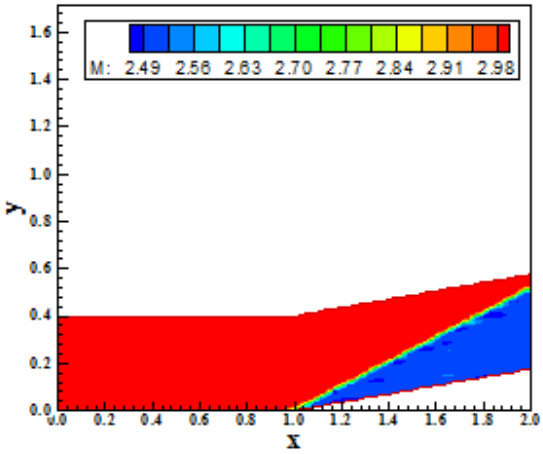

Figure 26 : Mach number contours (VL82-SB).

One way to quantitatively verify if the solutions generated by the R81 TVD scheme are satisfactory

consists in determining the shock angle of the oblique shock wave, β, measured in relation to the initial direction of the flow field. [24] (pages 352 and 353) presents a diagram with values of the shock angle, β, to oblique shock waves. The value of this angle is determined as function of the freestream Mach number and of the deflection angle of the flow after the shock wave, ϕ. To the compression corner problem, ϕ = 10º (ramp inclination angle) and the freestream Mach number is 3.0, resulting from this diagram a value to β equals to 27.5º. Using a transfer in Figures 4 to 8, it is possible to obtain the values of β to the R81 TVD scheme in its variants, as well the respective errors, shown in Tab. 2. As can be observed, the R81 TVD scheme using the SB limiter has yielded the best result in terms of R81 variants.

Table 2 : Shock angle and percentage errors (R81/Explicit case).

Algorithm	β (°)	Error (%)
R81 – VL	27.0	1.82
R81 – VA	27.0	1.82
R81 – Min	27.0	1.82
R81 – SB	27.4	0.36
R81 – BL	26.9	2.18

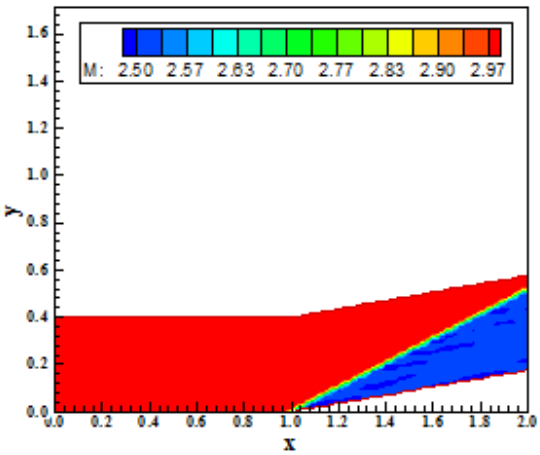

Figure 27 : Mach number contours (VL82-BL).

Figure 35 shows the wall pressure distributions obtained by all variants of the VL82 TVD scheme. They are compared with the oblique shock wave theory results. As can be observed, some solutions present oscillations at the compression corner, mainly the VL82 TVD scheme using the SB limiter, but they are in less frequency than in the solutions of the variants of the R81 TVD scheme. Figure 36 exhibits the wall pressure distributions obtained by the VL82 TVD scheme using VL, VA and Min limiters. As noted, no overshoot or undershoot are observed in the solutions, presenting these ones a smooth behaviour.

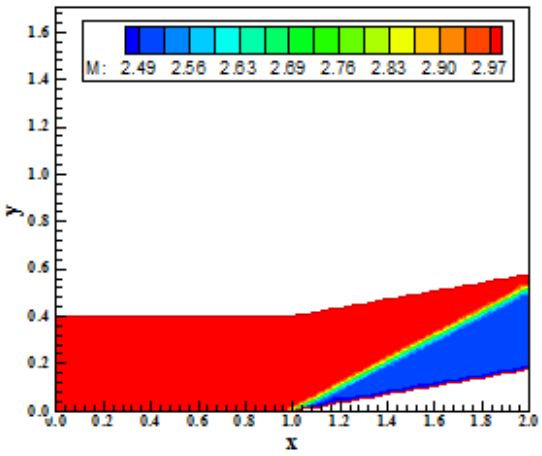

Figure 28 : Mach number contours (YWH82).

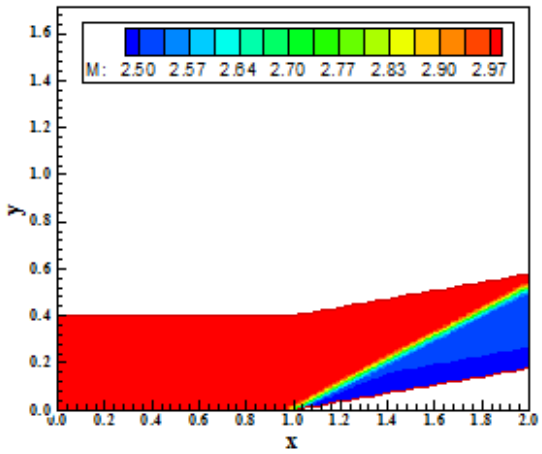

Figure 29 : Mach number contours (H83).

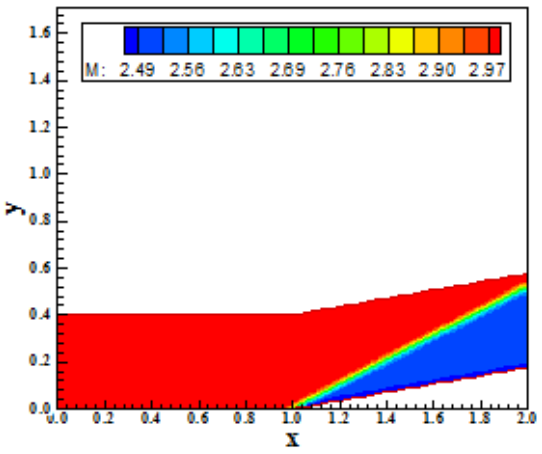

Figure 30 : Mach number contours (YK85).

It is also possible to observe that the shock discontinuity is captured within four cells, which is

a typical number of cells encountered in high resolution schemes to capture accurately shock waves. So the accuracy of the VL82 TVD scheme with these three limiters is in accordance with typical results of current high resolution schemes.

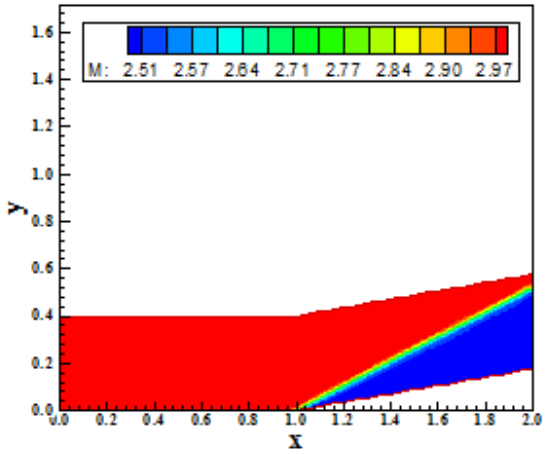

Figure 31 : Mach number contours (HB91).

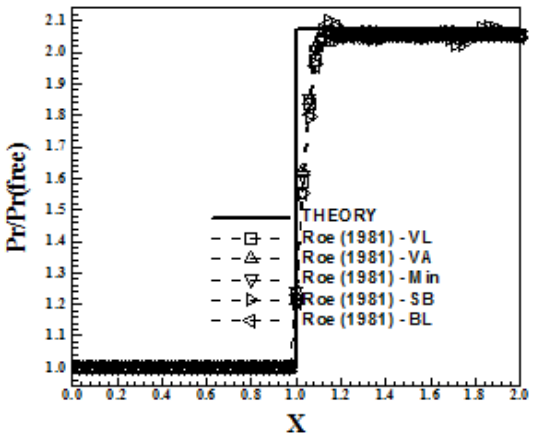

Figure 32 : Wall pressure distributions (R81).

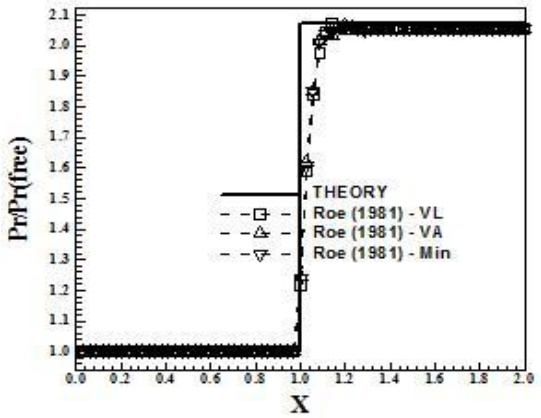

Figure 33 : Wall pressure distributions (R81-1).

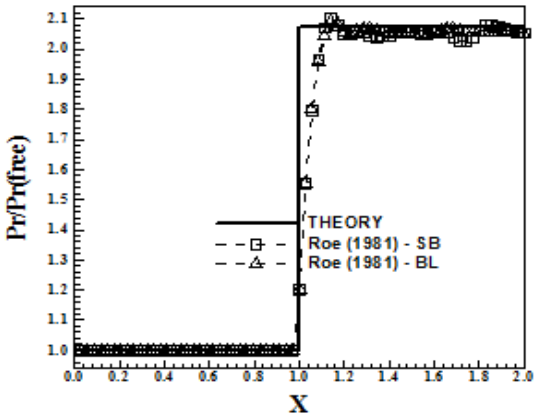

Figure 34 : Wall pressure distributions (R81-2).

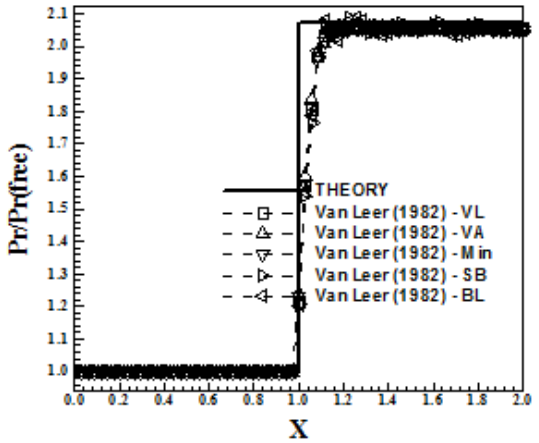

Figure 35 : Wall pressure distributions (VL82).

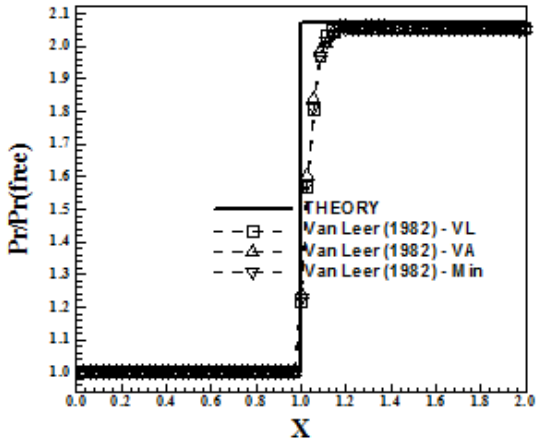

Figure 36 : Wall pressure distributions (VL82-1).

Figure 37 shows the wall pressure distributions obtained by the VL82 TVD scheme using the SB and the BL limiters. The SB limiter yields oscillations along the shock plateau, but the shock is also captured in four cells, as is the case with the BL limiter. By the results, the best solutions were obtained with VL, VA and Min limiters because

detect sharp and smooth pressure distributions at the corner wall.

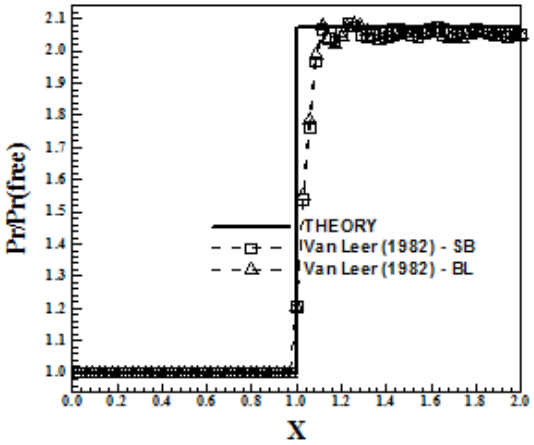

Figure 37 : Wall pressure distributions (VL82-2).

Analysing the oblique shock wave angle, using a transfer in Figures 9 to 13, it is possible to obtain the values of β to each variant of the VL82 TVD scheme, as well the respective errors, shown in Tab. 3. The VL82 TVD scheme using the VL, the VA and the Min limiters have yielded the best results.

Table 3 : Shock angle and percentage errors (VL82/Explicit case).

Algorithm	β (°)	Error (%)
VL82 – VL	27.2	1.09
VL82 – VA	27.2	1.09
VL82 – Min	27.2	1.09
VL82 – SB	27.0	1.82
VL82 – BL	27.0	1.82

Figure 38 shows the pressure distributions along the compression corner wall obtained by the YWH82, the H83, the YK85 and the HB91 schemes. They are compared with the exact solution from oblique shock wave theory. It is possible to note that the solutions generated by the H83, the YK85 and the HB91 schemes are smoother than that generated by the YWH82 scheme, but all solutions present a small pressure peak at the shock region. In the solutions generated by the H83 and the YK85 schemes, the shock presents a small peak in relation to the theory, but the shock is sharp defined. The HB91 scheme presents the smallest value to the pressure peak at the ramp beginning, the shock position, characterizing this scheme as the best of the four Harten's based algorithms under study. All Harten's based schemes under-predict the value of the pressure at the ramp (at the plateau region) in relation to the theoretical solution.

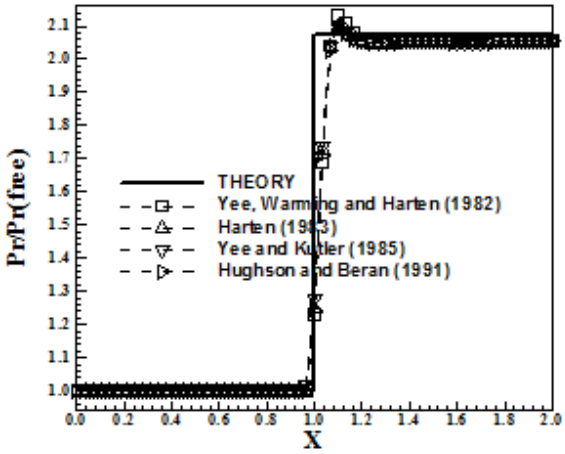

Figure 38 : Pressure distributions at wall.

The width of the constant pressure region after the shock (the plateau) at the ramp is better represented by the YK85 scheme. The shock profile is captured by the schemes using three points, which represents good solutions to high resolution algorithms.

Analysing the oblique shock wave angle, using a transfer in Figures 14 to 17, it is possible to obtain the values of β to each Harten's based TVD scheme, as well the respective errors, shown in Tab. 4. The results highlight the HB91 scheme as the most accurate of the studied Harten's based TVD algorithms.

Table 4 : Shock angle and percentage errors to each scheme (Harten's based schemes/Explicit case).

Algorithm	β (°)	Error (%)
YWH82	28.0	1.82
H83	27.8	1.09
YK85	28.0	1.82
HB91	27.6	0.36

Comparing the overall results, the best scheme was the R81 TVD scheme in its SB variant, presenting a reasonable wall pressure distribution and a very accurate value to the shock angle of the oblique shock wave.

13.2 Corner results – Implicit case

To the implicit case, it was chosen again the compression corner problem due to the accurate shock angle value which can be obtained, as also the wall pressure distribution. Moreover, it allows the visualization of the increasing in the shock wave thickness originated from each scheme due to the use of large time steps for algorithms presenting steady-state-time-dependent solutions.

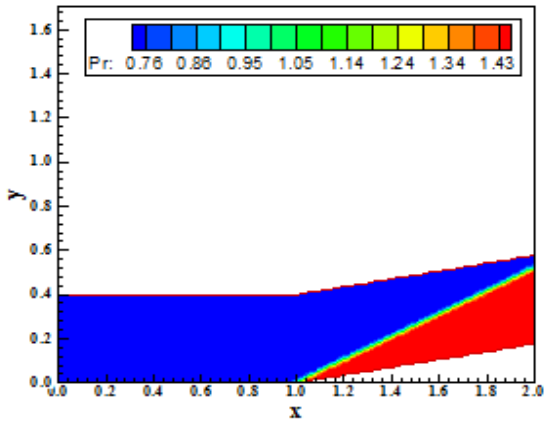

Figure 39 : Pressure contours (R81-VL).

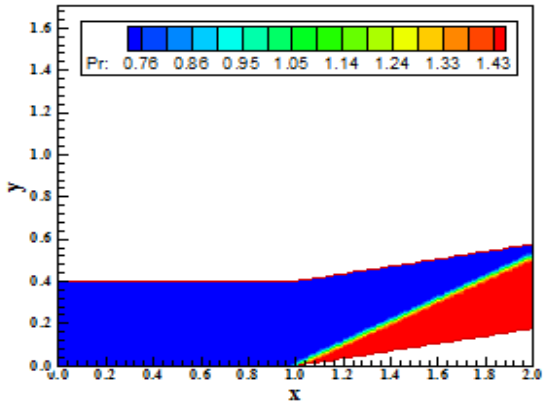

Figure 40 : Pressure contours (R81-VA).

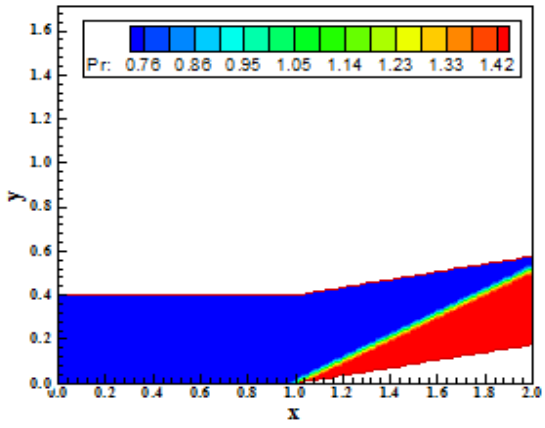

Figure 41 : Pressure contours (R81-Min).

Figures 39 to 51 exhibit the pressure contours obtained by the R81, in its five variants; the VL82, in its five variants; the H83; the YK85; and the HB91 schemes. The YWH82 scheme did not present converged results. As can be observed, the most severe pressure field is due to R81 in its SB variant. Also noted is the increasing in the shock wave thickness to the Harten's based scheme solutions. It occurs because the dissipation function of these schemes is time step dependent. So, in the steady state condition, the solution depends of the time step employed.

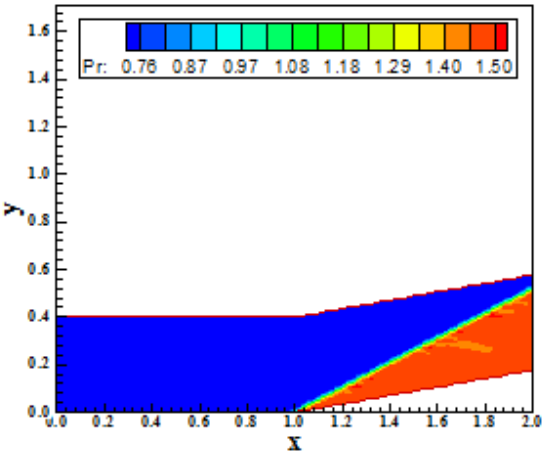

Figure 42 : Pressure contours (R81-SB).

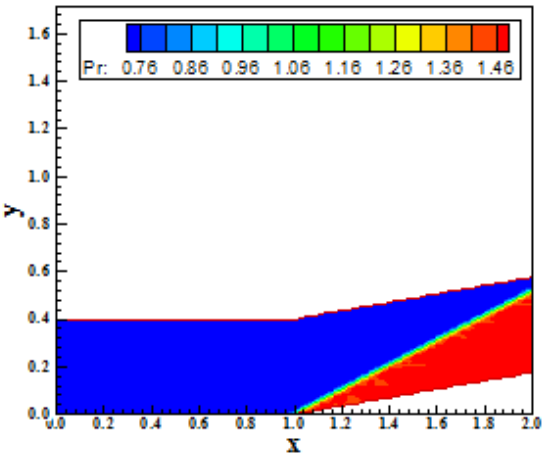

Figure 43 : Pressure contours (R81-BL).

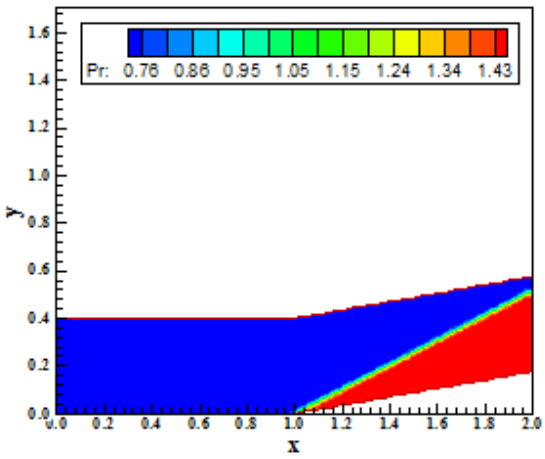

Figure 44 : Pressure contours (VL82-VL).

Figures 52 to 64 show the Mach number contours generated by the R81, in its five variants;

the VL82, in its five variants; the H83; the YK85; and the HB91 schemes. The increasing in the shock wave thickness observed in Figs. 49 to 51 is also clear as compared with their explicit counterparts.

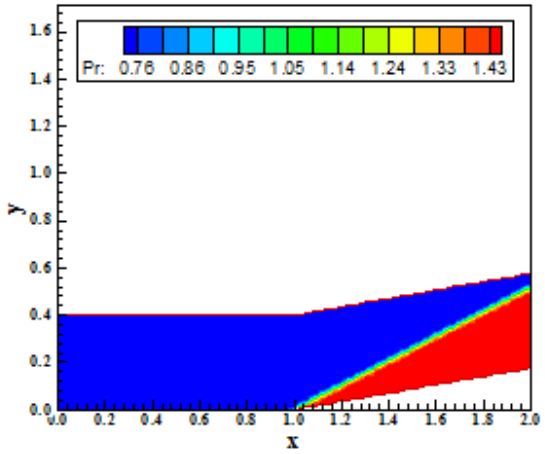

Figure 45 : Pressure contours (VL82-VA).

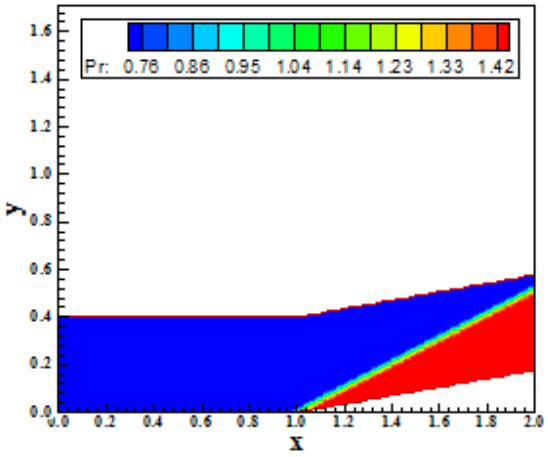

Figure 46 : Pressure contours (VL82-Min).

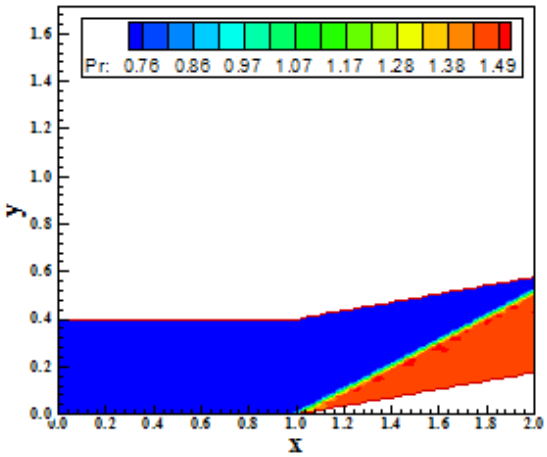

Figure 47 : Pressure contours (VL82-SB).

Figure 65 shows the wall pressure distributions obtained by all variants of the R81 TVD scheme. They are compared with the oblique shock wave theory results. As can be seen, some solutions present overshoot at the compression corner, mainly the R81 TVD scheme using the SB limiter, as occurred in the explicit case.

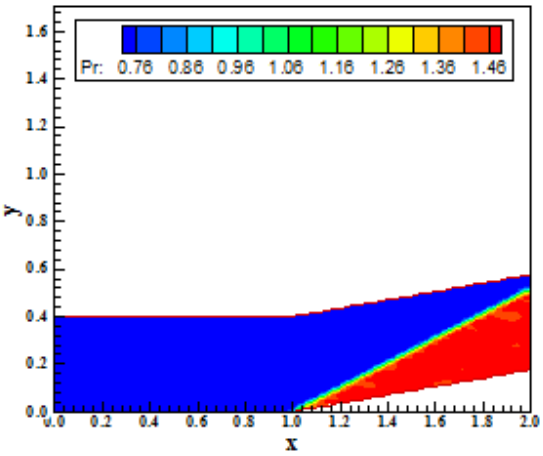

Figure 48 : Pressure contours (VL82-BL).

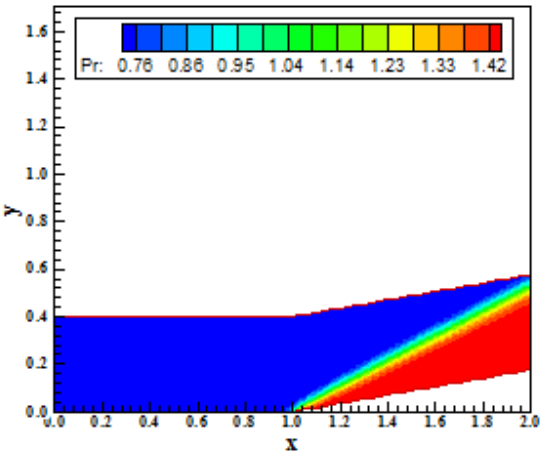

Figure 49 : Pressure contours (H83).

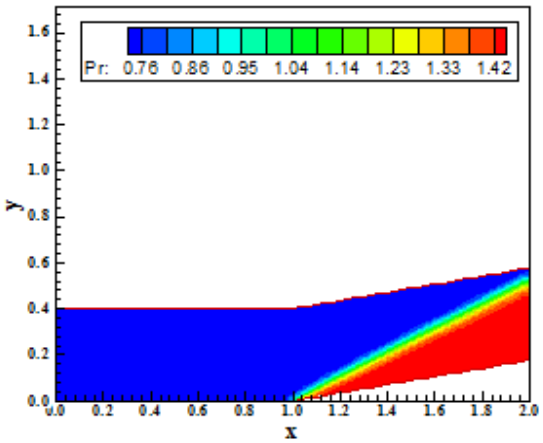

Figure 50 : Pressure contours (YK85).

Figure 66 exhibits the wall pressure distributions obtained by the R81 TVD scheme using VL, VA and Min limiters. As noted, no overshoot is observed in the solutions, presenting these ones a smooth behaviour. It is also possible to observe that the shock discontinuity is captured in three cells, better than the explicit solutions, which is a good number of cells to capture accurately a shock discontinuity by a high resolution scheme. Thus, the accuracy of the R81 TVD scheme with these three limiters to the implicit case is better than the explicit one and is in accordance with typical results of current high resolution schemes.

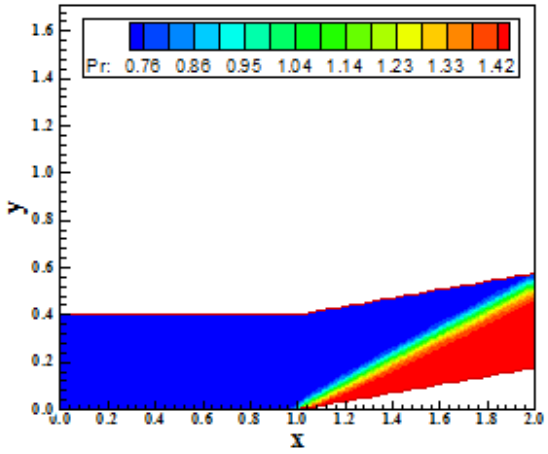

Figure 51 : Pressure contours (HB91).

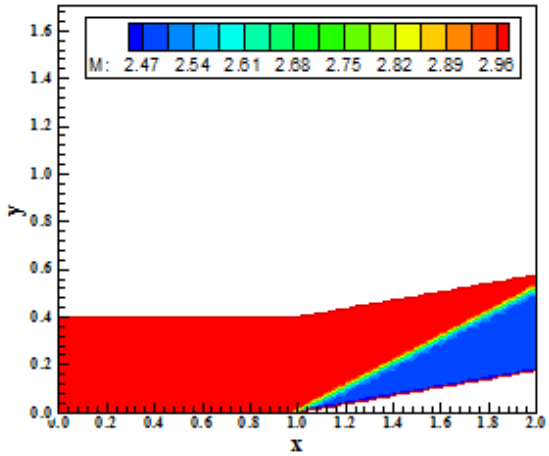

Figure 52 : Mach number contours (R81-VL).

Figure 67 shows the wall pressure distributions obtained by the R81 TVD scheme using SB and BL limiters. The SB and BL limiters yield a pronounced overshoot, but the shock is also captured in three cells. As in the explicit case, these two limiters present problems of oscillations due to the shock, something that should be avoided by the use of adequate region of TVD properties. By the results, the best solutions were obtained with VL, VA and Min limiters because detect sharp and smooth pressure distributions at the corner wall.

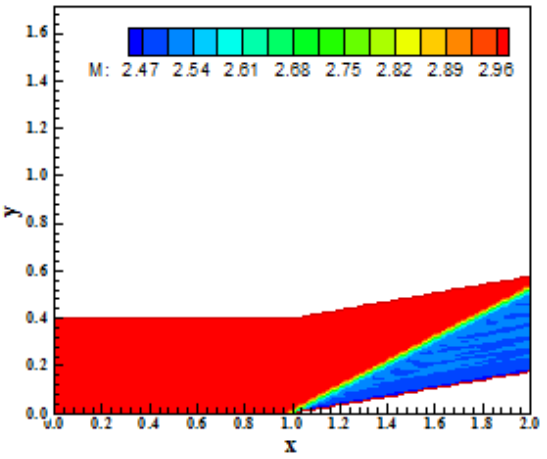

Figure 53 : Mach number contours (R81-VA).

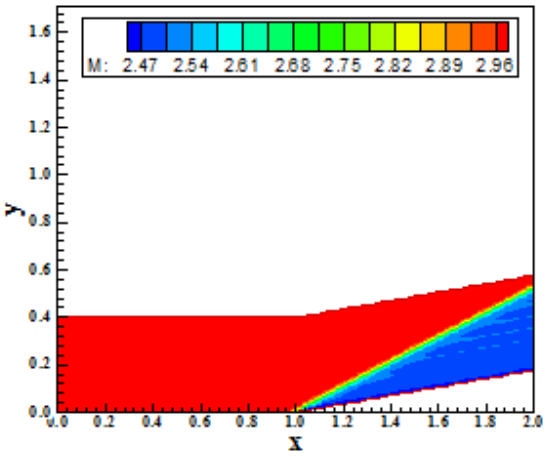

Figure 54 : Mach number contours (R81-Min).

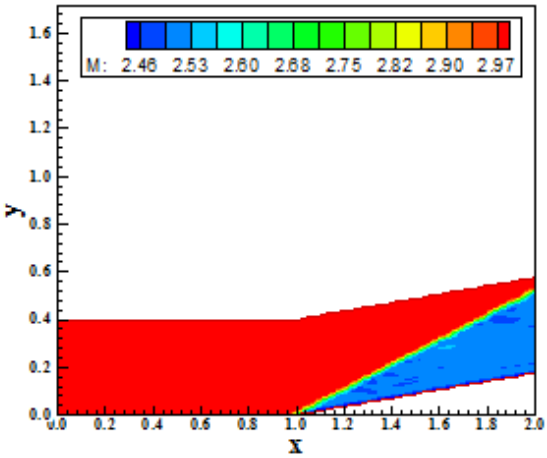

Figure 55 : Mach number contours (R81-SB).

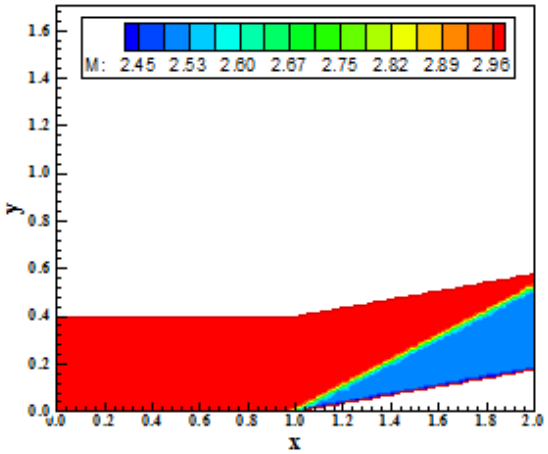

Figure 56 : Mach number contours (R81-BL).

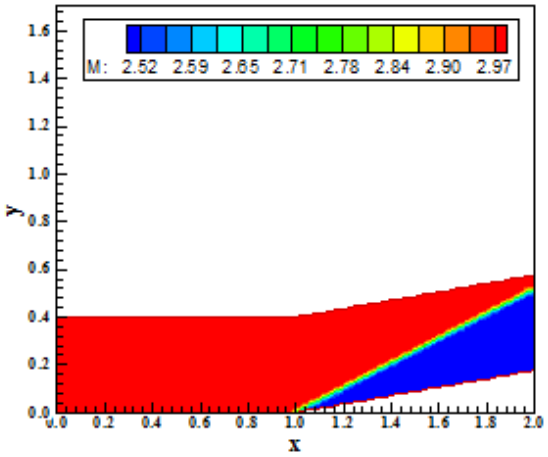

Figure 57 : Mach number contours (VL82-VL).

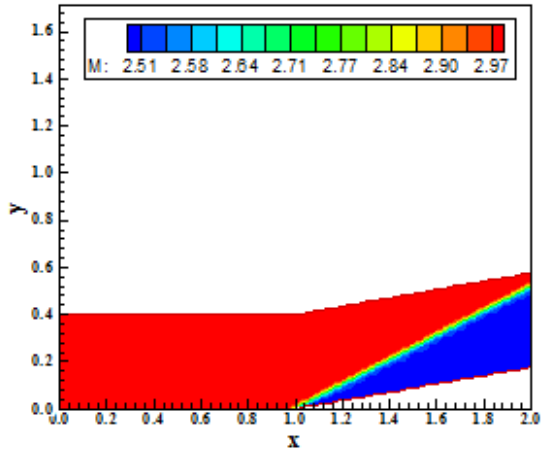

Figure 58 : Mach number contours (VL82-VA).

Analysing the oblique shock wave angle, using a transfer in Figures 39 to 43, it is possible to obtain the values of β to each variant of the R81 TVD scheme, in the implicit case, as also the respective errors, shown in Tab. 5. The R81 TVD scheme using the SB limiter has yielded the best result. The values obtained to β by the implicit solutions were better than by the explicit ones, as well the percentage errors.

Table 5 : Shock angle and percentage errors (R81/Implicit case).

Algorithm	β (°)	Error (%)
R81 – VL	27.2	1.09
R81 – VA	27.6	0.36
R81 – Min	27.7	0.73
R81 – SB	27.5	0.00
R81 – BL	27.2	1.09

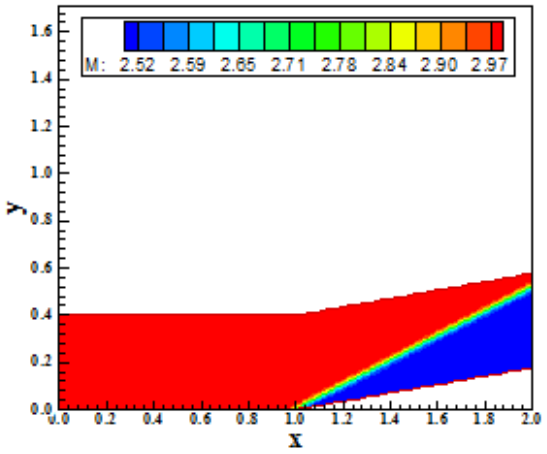

Figure 59 : Mach number contours (VL82-Min).

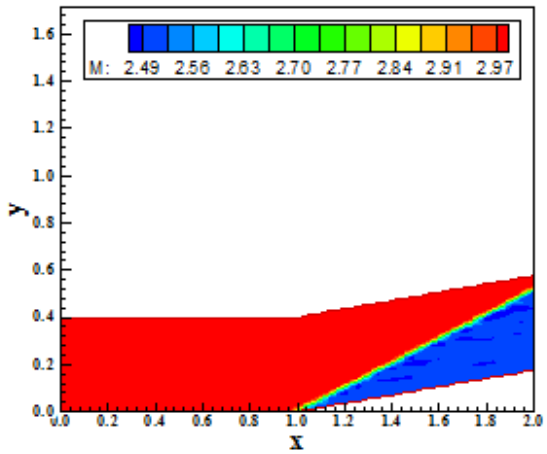

Figure 60 : Mach number contours (VL82-SB).

Figure 68 shows the wall pressure distributions obtained by all variants of the VL82 TVD scheme. They are compared with the oblique shock wave theory results. As noted, some solutions present oscillations at the compression corner, mainly the VL82 TVD scheme using the SB limiter, but they

are in less amount than in the solutions of the R81's variants.

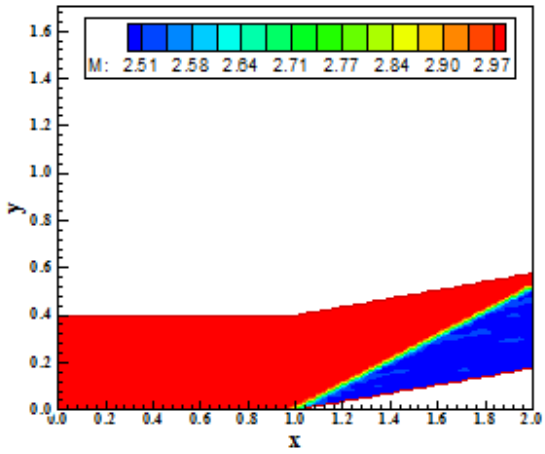

Figure 61 : Mach number contours (VL82-BL).

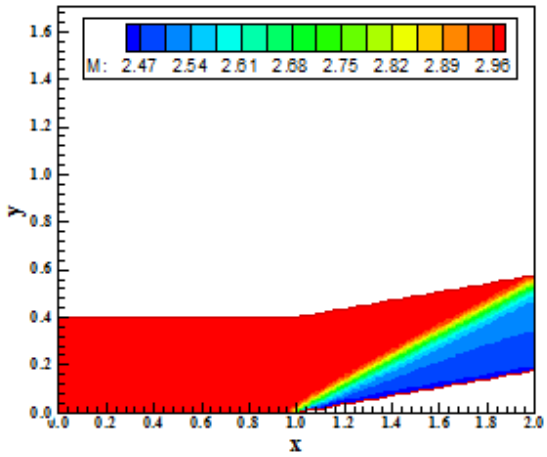

Figure 62 : Mach number contours (H83).

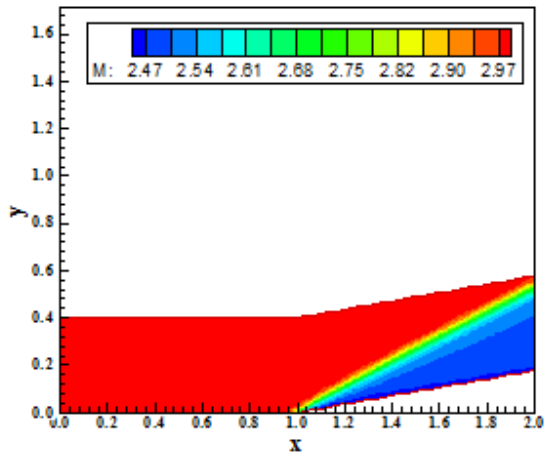

Figure 63 : Mach number contours (YK85).

Figure 69 exhibits the wall pressure distributions obtained by the VL82 TVD scheme using VL, VA and Min limiters. As observed, no overshoots or undershoots are noted in these solutions, presenting these ones a smooth behaviour. It is also possible to observe that the shock discontinuity is captured in five cells, a high number of cells to high resolution schemes capture accurately shock waves. Hence, the accuracy of the VL82 TVD scheme with these three limiters, in the implicit case, is not in agreement with typical results of current high resolution schemes.

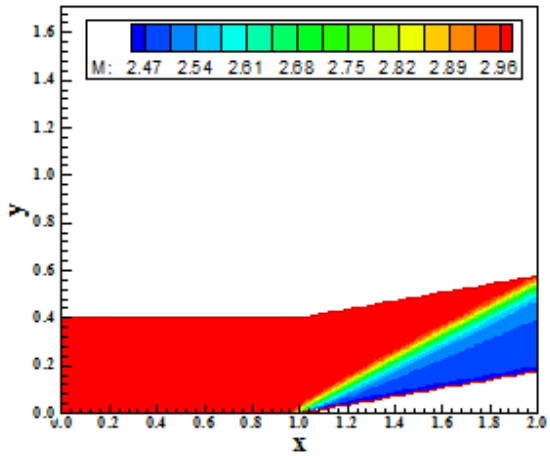

Figure 64 : Mach number contours (HB91).

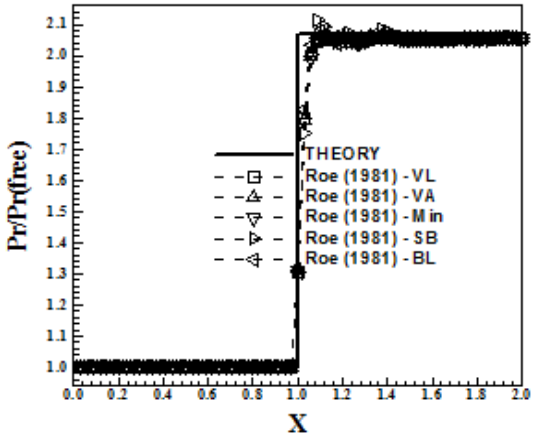

Figure 65 : Wall pressure distributions (R81).

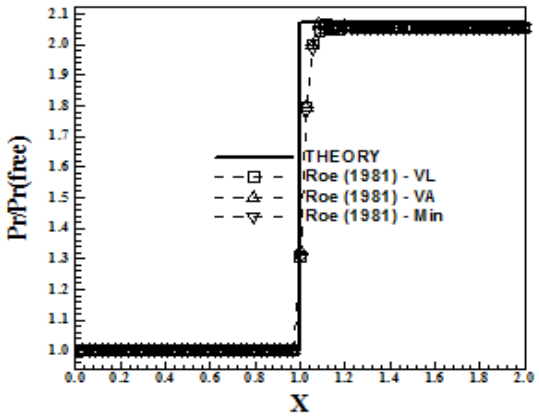

Figure 66 : Wall pressure distributions (R81-1).

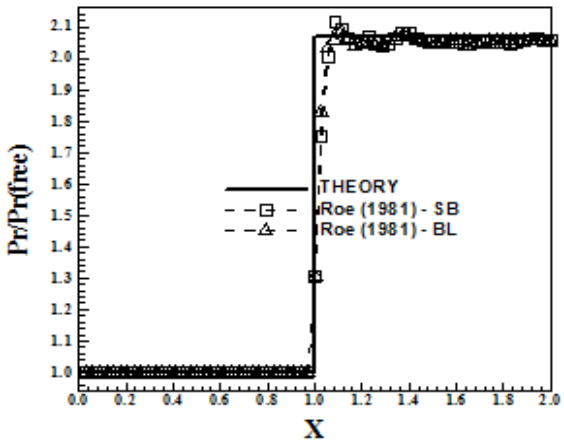

Figure 67 : Wall pressure distributions (R81-2).

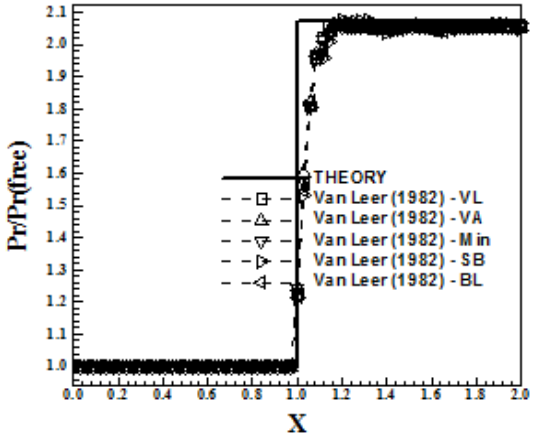

Figure 68 : Wall pressure distributions (VL82).

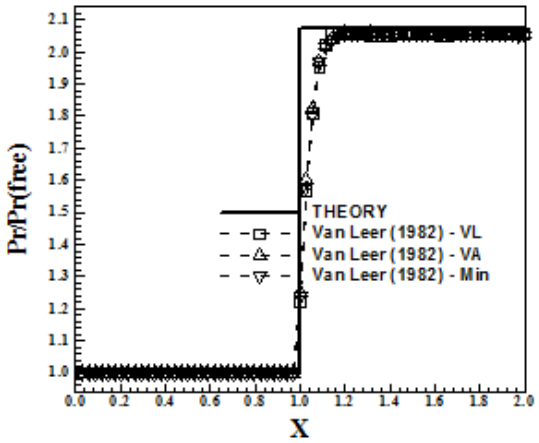

Figure 69 : Wall pressure distributions (VL82-1).

Figure 70 shows the wall pressure distributions obtained by the VL82 TVD scheme using SB and BL limiters. The SB limiter yields oscillations along the shock plateau, but the shock discontinuity is also captured within five cells, as is the case with the BL limiter. By the results, the best solutions were obtained with VL, VA and Min limiters because detect sharp and smooth pressure distributions at the corner wall, although require a high number of cells to capture the shock discontinuity.

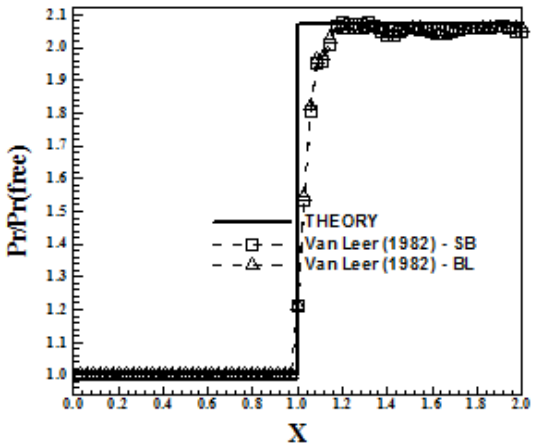

Figure 70 : Wall pressure distributions (VL82-2).

Analysing the oblique shock wave angle, using a transfer in Figures 44 to 48, it is possible to obtain the values of β to each variant of the VL82 TVD scheme, to the implicit case, as well the respective errors, shown in Tab. 6. The VL82 TVD scheme using the SB limiter has yielded the best result in relation to its variants.

Table 6 : Shock angle and percentage errors (VL82/Implicit case).

Algorithm	β (°)	Error (%)
VL82 – VL	27.8	1.09
VL82 – VA	27.0	1.82
VL82 – Min	27.2	1.09
VL82 – SB	27.6	0.36
VL82 – BL	26.9	2.18

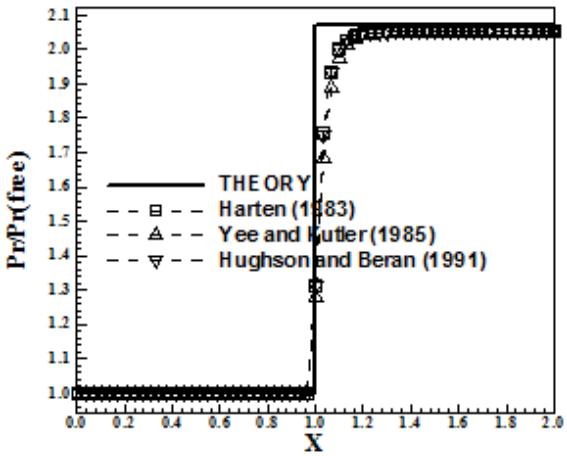

Figure 71 : Wall pressure distributions.

Figure 71 exhibits the wall pressure distributions obtained by the H83, by the YK85 and by the HB91 schemes. They are compared with the oblique shock wave theory. As can be noted, none pressure peak is observed in the solutions. Moreover, the pressure distributions are free of oscillations and extremes. However, the shock discontinuity is captured using five cells, which is excessive to a high resolution scheme.

The shock angle of the oblique shock wave generated by the H83, by the YK85 and by the HB91 is again evaluated. Using a transfer in Figures 49 to 51, it is possible to determine the shock angle as also the percentage error obtained in this measurement. As can be seen in Table 7, the best estimation to this parameter is again predicted by the HB91 scheme, as treating of the Harten's based TVD algorithms. The major percentage errors found in the solutions, in comparison with their explicit counterparts, are due to the smearing that the excessive dissipation provides in the implicit case.

Table 7 : Shock angle and percentage errors to each scheme (Harten's based schemes/Implicit case).

Algorithm	β (º)	Error (%)
H83	28.3	2.91
YK85	28.4	3.27
HB91	27.3	0.72

13.3 Explicit versus implicit comparisons

Figure 72 exhibits the best wall pressure distributions obtained by each scheme in its explicit version. As the R81 and VL82 have three distributions with approximately the same behaviour, it was chosen one of them to represent the scheme solution. The choice was the solution obtained with the Min nonlinear limiter because it is the most conservative among them. So, Figure 72 presents the best curves of each algorithm. All presented solutions in the explicit results capture the shock discontinuity in four cells, as previously emphasized. The best wall pressure distribution in this comparison was obtained by both the R81 and the VL82 TVD schemes using Min limiter. The other solutions present small peaks at the pressure distributions and were disregarding. Figure 73 exhibits the best wall pressure distributions obtained by each scheme in its implicit version. The solutions of the R81 TVD scheme capture the shock discontinuity in three cells, which is an improvement in relation to the explicit solutions. Moreover, the implicit distributions determine solutions more sharp defined than the explicit ones.

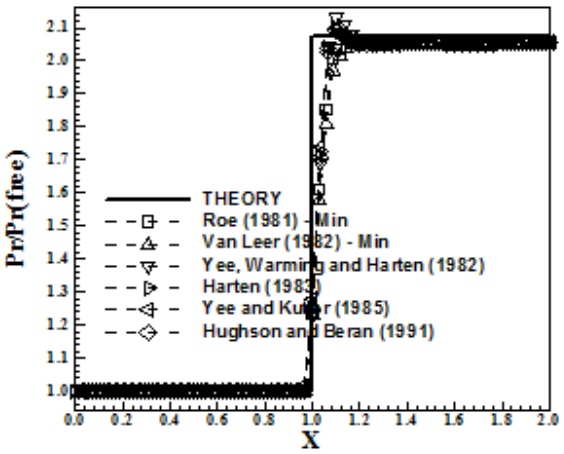

Figure 72 : Best wall pressure distributions (Explicit case).

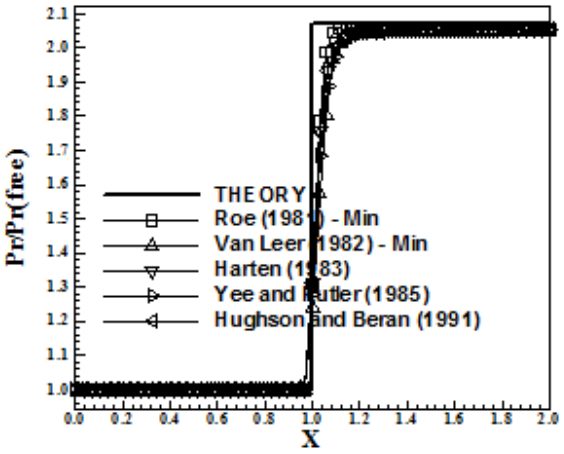

Figure 73 : Best wall pressure distributions (Implicit case.

In other words, the discontinuity profiles generated by the implicit solutions are better vertically defined at the discontinuity (closer to the theoretical solution discontinuity) than the explicit profiles, assuring a better definition at the transition. The solutions of the VL82 TVD scheme capture the shock discontinuity in five cells, which is a weak behaviour to a high resolution scheme. The best wall pressure distribution in this comparison was obtained by the R81 TVD scheme using Min limiter.

Table 8 presents the best values to the shock angle obtained by all schemes in their explicit case. As can be observed, the best result is obtained with the R81 TVD scheme in its SB variant and with HB91 TVD scheme. Table 9 presents the best values to the shock angle of the oblique shock wave obtained by all schemes in their implicit case. Again the R81 TVD

scheme in its SB variant presents the best value to this parameter.

Table 8 : Shock angle and percentage errors to each scheme (Explicit case).

Algorithm	β (º)	Error (%)
R81 – SB	27.4	0.36
VL82-Min	27.2	1.09
YWH82	28.0	1.82
H83	27.8	1.09
YK85	28.0	1.82
HB91	27.6	0.36

Table 9 : Shock angle and percentage errors to each scheme (Implicit case).

Algorithm	β (º)	Error (%)
R81 – SB	27.5	0.00
VL82 – SB	27.6	0.36
H83	28.3	2.91
YK85	28.4	3.27
HB91	27.3	0.72

14 Conclusions

In this work, the [1-5; 7] schemes are implemented, on a finite volume context and using an upwind and a structured spatial discretization, to solve the Euler equations, in two-dimensions, and are compared with themselves. All schemes are implemented in their second order accurate versions in space and are applied to the solution of the supersonic flow along a compression corner configuration. The theories involving the extension of the first order versions of the numerical schemes of [1] and [2] to second order spatial accuracy, incorporating hence TVD properties through a MUSCL approach, and the implicit numerical implementation of all second order schemes under study are detailed. First order time integrations like ADI approximate factorization are programmed. A spatially variable time step procedure is implemented aiming to accelerate the convergence of the schemes to the steady state condition. The effective gains in terms of convergence ratio with this procedure are reported in [15-16].

The results have demonstrated that the most accurate solutions are provided by the [1] scheme in its SB variant. This algorithm has provided the best solutions in the compression corner problem, both in the explicit and implicit cases, due to the best estimative of the shock angle.

The present author strongly recommends the use of the [1] scheme in its SB variant to the final phase of the aerospace vehicle projects, where more refined results are needed at a low computational cost. To the initial phase, where start results are expected without a great refinement, the [7] scheme is suggested to.

15 Acknowledgments

The author acknowledges the CNPq by the financial support conceded under the form of a DTI (Industrial Technological Development) scholarship no. 384681/2011-5. He also acknowledges the infra-structure of the ITA that allowed the realization of this work.

References:

[1] P. L. Roe, Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes, *Journal of Computational Physics*, Vol. 43, 1981, pp. 357-372.

[2] B. Van Leer, Flux-Vector Splitting for the Euler Equations, Proceedings of the 8th International Conference on Numerical Methods in Fluid Dynamics, E. Krause, Editor, *Lecture Notes in Physics*, Vol. 170, 1982, pp. 507-512, Springer-Verlag, Berlin.

[3] H. C. Yee, R. F. Warming, and A. Harten, A High-Resolution Numerical Technique for Inviscid Gas-Dynamic Problems with Weak Solutions, *Proceedings of the 8th International Conference on Numerical Methods in Fluid Dynamics*, E. Krause, Editor, Lecture Notes in Physics, Springer-Verlag, Berlim, Germany, Vol. 170, 1982, pp. 546-552.

[4] A. Harten, High Resolution Schemes for Hyperbolic Conservation Laws, *Journal of Computational Physics*, Vol. 49, 1983, pp. 357-393.

[5] H. C. Yee, and P. Kutler, Application of Second-Order-Accurate Total Variation Diminishing (TVD) Schemes to the Euler Equations in General Geometries, *NASA-TM-85845*, 1985.

[6] R. W. MacCormack, The Effect of Viscosity in Hypervelocity Impact Cratering, *AIAA Paper 69-354*.

[7] M. C. Hughson, P. S. and Beran, Analysis of Hyperbolic Blunt-Body Flows Using a Total Variation Diminishing (TVD) Scheme and the MacCormack Scheme, *AIAA 91-3206-CP*, 1991.

[8] S. K. Godunov, A Difference Scheme for Numerical Computation of Discontinuous Solution of Hydrodynamic Equations, *Math. Sbornik*, Vol. 47, 1959, pp. 271-306.

[9] C. Hirsch, *Numerical Computation of Internal and External Flows – Computational Methods for Inviscid and Viscous Flows*, John Wiley & Sons Ltd, 691p., 1990.

[10] R. M. Beam, and R. F. Warming, An Implicit Factored Scheme for the Compressible Navier-Stokes Equations, *AIAA Journal*, Vol. 16, No. 4, 1978, pp. 393-402.

[11] J. Douglas, On the Numerical Integration of $u_{xx}+u_{yy}=u_t$ by Implicit Methods, *Journal of the Society of Industrial and Applied Mathematics*, Vol. 3, 1955, pp. 42-65.

[12] D. W. Peaceman, and H. H. Rachford, The Numerical Solution of Parabolic and Elliptic Differential Equations, *Journal of the Society of Industrial and Applied Mathematics*, Vol. 3, 1955, pp. 28-41.

[13] J. Douglas, and J. E. Gunn, A General Formulation of Alternating Direction Methods, *Numerische Mathematiks*, Vol. 6, 1964, pp. 428-453.

[14] N. N. Yanenko, *The Method of Fractional Steps*, Springer Verlag, N.Y., EUA, 1971.

[15] E. S. G. Maciel, Analysis of Convergence Acceleration Techniques Used in Unstructured Algorithms in the Solution of Aeronautical Problems – Part I, *Proceedings of the XVIII International Congress of Mechanical Engineering (XVIII COBEM)*, Ouro Preto, MG, Brazil, 2005. [CD-ROM]

[16] E. S. G. Maciel, Analysis of Convergence Acceleration Techniques Used in Unstructured Algorithms in the Solution of Aerospace Problems – Part II, *Proceedings of the XII Brazilian Congress of Thermal Engineering and Sciences (XII ENCIT)*, Belo Horizonte, MG, Brazil, 2008. [CD-ROM]

[17] M. Liou, and Steffen Jr., C. J., A New Flux Splitting Scheme, *Journal of Computational Physics*, Vol. 107, 1993, pp. 23-39.

[18] R. Radespiel, and N. Kroll, Accurate Flux Vector Splitting for Shocks and Shear Layers, *Journal of Computational Physics*, Vol. 121, 1995, pp. 66-78.

[19] B. Van Leer, Towards the Ultimate Conservative Difference Scheme. II. Monotonicity and Conservation Combined in a Second-Order Scheme, *Journal of Computational Physics*, Vol. 14, 1974, pp. 361-370.

[20] P. L. Roe, In *Proceedings of the AMS-SIAM Summer Seminar on Large-Scale Computation in Fluid Mechanics*, Edited by B. E. Engquist et al., Lectures in Applied Mathematics, Vol. 22, 1983, p. 163.

[21] H. C. Yee, R. F. Warming, R. F., and A. Harten, Implicit Total Variation Diminishing (TVD) Schemes for Steady-State Calculations, *Journal of Computational Physics*, 57, 1985, pp. 327-360.

[22] A. Jameson, and D. Mavriplis, Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh, *AIAA Journal*, 24, 1986, pp. 611-618.

[23] E. S. G. Maciel, Simulação Numérica de Escoamentos Supersônicos e Hipersônicos Utilizando Técnicas de Dinâmica dos Fluidos Computacional, *Doctoral Thesis*, ITA, CTA, São José dos Campos, SP, Brazil, 258p, 2002.

[24] J. D. Anderson Jr., *Fundamentals of Aerodynamics*, McGraw-Hill, Inc., EUA, 563p, 1984.