COBEM UBERLÂNDIA 2019

ABCM Associação Brasileira de Engenharia e Ciências Mecânicas

25th ABCM International Congress of Mechanical Engineering
October 20-25, 2019, Uberlândia, MG, Brazil

COB-2019-1227
EFFECTS OF NUMERICAL PARAMETERS AND ARTIFICIAL DISSIPATION IN THE SIMULATION OF INVISCID NOZZLE FLOWS

Frederico Bolsoni Oliveira
Instituto Tecnológico de Aeronáutica – DCTA/ITA, 12228-900 – São José dos Campos – SP – Brazil
fredericobolsoni@gmail.com

João Luiz F. Azevedo
Instituto de Aeronáutica e Espaço, DCTA/IAE/ALA, 12228-904 – São José dos Campos – SP – Brazil
joaoluiz.azevedo@gmail.com

Abstract. *A two-dimensional transonic nozzle flow is simulated using four different numerical schemes: Beam and Warming implicit approximate factorization algorithm, Steger and Warming implicit flux vector splitting algorithm, van Leer explicit flux vector splitting algorithm and Liou's AUSM$^+$ method. The effects of numerical parameters over the simulation output and the influence of artificial dissipation in the flow features are evaluated. The present results show good agreement with the available literature and demonstrate that some flow structures can be severely dampened or incorrectly captured by the direct or indirect inclusion of excessive artificial dissipation.*

Keywords: *Computational Fluid Dynamics, Nozzle Flow, Artificial Dissipation, Euler Equations, Flux Vector Splitting*

1. INTRODUCTION

The numerical computation of transonic internal flows is of high importance to a broad range of aerospace applications. In particular, the numerical determination of the flow properties inside a transonic nozzle during its design phase could result in an overall decrease in engineering costs, since the experimental characterization of the flow to a high degree of detail can be time consuming as well as expensive, and sometimes might not even be able to fully capture all of the individual flow features needed for optimizing its internal geometry as well as identifying possible design flaws.

Although usually being less expensive than performing experimental procedures, numerical methods can lead to incorrect conclusions regarding the properties of the flow of interest. If the user is not careful enough to evaluate the impact of the chosen numerical scheme and its control parameters over the output data as well as the independence of the results in relation to the mesh currently in use, non-physical behavior could arise or even physical properties could end up becoming hidden in the final solution that would ultimately lead the user to arrive at misleading conclusions regarding the structure of the flow.

This paper aims to study the behavior of four different numerical schemes regarding the computation of a representative nozzle flow. The geometry chosen for the nozzle has been previously experimentally studied by Mason *et al.* (1980) and later numerically simulated by MacCormack (1985), and Azevedo *et al.* (1995). While the numerical data shown in these papers are in good agreement with the experimental results, it has been found that the meshes used to discretize the spatial domain in these simulations are coarse enough to the point where the schemes used become considerably dissipative, preventing the development of shock waves downstream of the nozzle's throat. Nevertheless, their results are used here for comparison purposes in order to understand the impact of the mesh refinement over the output data.

All methods used here to solve the system of governing equations are implemented in general curvilinear coordinates and limited to bidimensional domains only. The first method is Beam and Warming's implicit approximate factorization scheme (Beam and Warming, 1978), which uses an implicit Euler time march coupled with a second-order centered scheme for the discretization of the spatial derivatives. An approximate factorization of the Implicit Alternating Direction (ADI) type is used in order to keep the implicit matrix operator in a block-tridiagonal form. Since it uses a centered spatial scheme, artificial dissipation must be applied in order to dampen high frequency oscillations that might cause the algorithm to diverge. Pulliam's nonlinear artificial dissipation model (Pulliam, 1986a) is used in the explicit part of the equations, while a simplified version of it is applied to the implicit operator.

The other three methods are based on the Flux Vector Splitting strategy. The first of them is Steger and Warming's own implicit upwind algorithm (Steger and Warming, 1981), where an implicit Euler time march is used coupled with one-sided space derivatives discretization. Here, both first and second order space discretizations were implemented. The second one is an explicit van Leer scheme (Van Leer, 1982), which modifies the governing equations' flux vectors in such

a way that the eigen values of their respective Jacobian matrices are now a continuous function of the flow's local Mach number, solving some intrinsic problems of the Steger and Warming scheme. Both the first and second order of the spatial operators of the van Leer's method are implemented here.

Lastly, we have Liou's Advection Upstream Splitting Method Plus ($AUSM^+$) explicit method (Liou, 1996). In this scheme, originally written in numerical flux notation, the splitting of the flux vectors is obtained by interpreting them as being the sum of the transport of a passive scalar component and a local pressure component. By splitting these two components, an overall splitting of the flux vectors is achieved. Only the first order discretization of the spatial derivatives is implemented.

In order to determine the influence of the mesh over the final results, four different single-block structured meshes have been created, with increasingly higher order of refinement. All the aforementioned schemes are applied to each of these four meshes.

2. TWO-DIMENSIONAL FORMULATION

The two-dimensional form of the Euler Equations can be written in conservation-law form and curvilinear coordinates as follows:

$$\frac{\partial \hat{Q}}{\partial \tau} + \frac{\partial \hat{E}}{\partial \xi} + \frac{\partial \hat{F}}{\partial \hat{\eta}} = 0 \tag{1}$$

where $\hat{Q}$ is the vector of conserved variables:

$$\hat{Q} = \frac{1}{J} \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ e \end{Bmatrix} \tag{2}$$

and $\hat{E}$ and $\hat{F}$ are the flux vectors for the inviscid case:

$$\hat{E} = \frac{1}{J} \begin{Bmatrix} \rho U \\ \rho u U + \xi_x p \\ \rho v U + \xi_y p \\ U(e+p) - \xi_t p \end{Bmatrix}, \quad \hat{F} = \frac{1}{J} \begin{Bmatrix} \rho V \\ \rho u V + \eta_x p \\ \rho v V + \eta_y p \\ V(e+p) - \eta_t p \end{Bmatrix} \tag{3}$$

Here, ρ is the local fluid density, u and v are the Cartesian components of the velocity vector, e is the total energy per unity of volume and p is the static pressure. ξ and η are the main coordinates of the curvilinear domain, τ is the time variable in the new system (made equal to t), and J is the Jacobian of the domain transformation, expressed as:

$$J^{-1} = x_\xi y_\eta - x_\eta y_\xi \tag{4}$$

where the subscript means a partial derivative operator, and the mapping from the Cartesian domain to the curvilinear domain is done in such a way that only discrete unitary variations occur between two adjacent points. The metric terms ξ_t, ξ_x, ξ_y, η_t, η_x ans η_y are calculated from the Cartesian domain as shown below:

$$\xi_x = J y_\eta, \quad \xi_y = -J x_\eta, \quad \eta_x = -J y_\eta, \quad \eta_y = J x_\xi, \quad \xi_t = -x_\tau \xi_x - y_\tau \xi_y, \quad \eta_t = -x_\tau \eta_x - y_\tau \eta_y \tag{5}$$

The variables U and V are the components of the contravariant velocity vector. They can be expressed as functions of the metric terms as follows:

$$U = \xi_t + \xi_x u + \xi_y v, \qquad V = \eta_t + \eta_x u + \eta_y v \tag{6}$$

For a thermally and calorically perfect gas, the pressure p can be expressed as a function of the primitive variables:

$$p = (\gamma - 1)\left[e - \frac{1}{2}\rho(u^2 + v^2)\right] \tag{7}$$

It must be said that all variables mentioned so far are used in their nondimensional form by taking a suitable reference value which will always be stated whenever needed.

3. NUMERICAL SCHEMES

Each of the four numerical schemes used in this work are briefly described below.

3.1 Beam and Warming Method

By using a generic notation for the artificial dissipation terms, this scheme can be written for each point (i, j) in the internal domain as follows:

$$\left(I + \Delta t \delta_\xi \hat{A}^n + D_{\xi_{impl.}}\right)\left(I + \Delta t \delta_\eta \hat{B}^n + D_{\eta_{impl.}}\right)\Delta\hat{Q}^n = -\Delta t\left[\delta_\xi \hat{E}^n + \delta_\eta \hat{F}^n\right] + D_\xi + D_\eta \tag{8}$$

where all the values written within parenthesis in the left member of Eq. (8) are operators that operate over whatever is written to their right. The superscript n indicates the current timestep in which the variables are being evaluated at. The value Δt is simply the local time difference between the current timestep and the next, whereas $\Delta\hat{Q}^n$ is the change in $\hat{Q}$ between these same timesteps. The $\delta\xi$ and $\delta\eta$ are second-order central difference operators in relation to ξ and η, respectively. The identity matrix is represented by I, while the components of the artificial dissipation model are represented by D_ξ, D_η, $D_{\xi_{impl.}}$ and $D_{\eta_{impl.}}$, where the first two are explicitly evaluated, while the other two are introduced in the scheme as implicit operators.

The matrices $\hat{A}$ and $\hat{B}$ are the Jacobian matrices of the flux vectors $\hat{E}$ and $\hat{F}$, respectively, and assume the following form as shown by Pulliam (1986b):

$$\begin{aligned}
&\hat{A} = \frac{\partial \hat{E}}{\partial \hat{Q}}, \qquad \hat{B} = \frac{\partial \hat{F}}{\partial \hat{Q}}, \\
&\left(\hat{A}, \hat{B}\right) = \begin{bmatrix} \kappa_t & \kappa_x & \kappa_y & 0 \\ -u\theta + \kappa_x\phi^2 & \kappa_t + \theta - (\gamma-2)\kappa_x u & \kappa_y u - (\gamma-1)\kappa_x v & (\gamma-1)\kappa_x \\ -v\theta + \kappa_y\phi^2 & \kappa_x v - (\gamma-1)\kappa_y u & \kappa_t + \theta - (\gamma-2)\kappa_y v & (\gamma-1)\kappa_y \\ \theta(\phi^2 - a_1) & \kappa_x a_1 - (\gamma-1)u\theta & \kappa_y a_1 - (\gamma-1)vU & \gamma\theta + \kappa_t \end{bmatrix}
\end{aligned} \tag{9}$$

in which κ is taken to be ξ for $\hat{A}$ and η for $\hat{B}$, while ϕ^2, θ and a_1 are defined as:

$$\phi^2 = \frac{1}{2}(\gamma - 1)(u^2 + v^2) \tag{10}$$

$$\theta = \kappa_x u + \kappa_y v \tag{11}$$

$$a_1 = \gamma\left(\frac{e}{p}\right) \tag{12}$$

where γ is the heat capacity ratio of the fluid. As specified before, the adopted values for the artificial dissipation terms of Eq. (8) comes from the nonlinear 4th differences Pulliam model, whose explicit components are:

$$\begin{aligned}
D_\xi &= \nabla_\xi\left(\frac{\sigma^n_{i+1,j}}{J_{i+1,j}} + \frac{\sigma^n_{i,j}}{J_{i,j}}\right)\left(\epsilon^{(2)}_{i,j}\Delta_\xi \hat{Q}^n_{i,j} - \epsilon^{(4)}_{i,j}\Delta_\xi\nabla_\xi\Delta_\xi\hat{Q}^n_{i,j}\right) \\
D_\eta &= \nabla_\eta\left(\frac{\sigma^n_{i,j+1}}{J_{i,j+1}} + \frac{\sigma^n_{i,j}}{J_{i,j}}\right)\left(\epsilon^{(2)}_{i,j}\Delta_\eta \hat{Q}^n_{i,j} - \epsilon^{(4)}_{i,j}\Delta_\eta\nabla_\eta\Delta_\eta\hat{Q}^n_{i,j}\right)
\end{aligned} \tag{13}$$

In Eq. (13), the operators Δ_ξ and ∇_ξ are first-order forward and backward operators in ξ, respectively. The way that the ϵ's are defined is what makes the model behave in a nonlinear manner. Their expressions are:

$$\begin{aligned}
\epsilon^{(2)}_{i,j} &= K_2\Delta t max\left(\gamma_{i+1,j}, \gamma_{i,j}, \gamma_{i-1,j}\right) \\
\epsilon^{(4)}_{i,j} &= max\left(0, K_4\Delta t - \epsilon^{(2)}_{i,j}\right)
\end{aligned} \tag{14}$$

where typical values of the constants are $K_2 = 1/4$ and $K_4 = 1/100$, although values as low as $K_2 = 1/8$ and $K_4 = 1/200$ were seen to still be capable of keeping the system stable provided that the mesh was fine enough. The

function γ is a pressure gradient "sensor", used for reducing the original forth difference model to second difference one in the neighborhood of pressure discontinuities (i.e. shockwaves), in order to dampen the oscillations intrinsic to the behavior of the central forth difference scheme near these regions. Together with the sum σ of the spectral radii of $\hat{A}$ and $\hat{B}$, we have:

$$\begin{aligned} \gamma_{i,j} &= \frac{|p_{i+1,j} - 2p_{i,j} + p_{i-1,j}|}{|p_{i+1,j} + 2p_{i,j} + p_{i-1,j}|} \\ \sigma_{i,j} &= |U| + a\sqrt{\xi_x^2 + \xi_y^2} + |V| + a\sqrt{\eta_x^2 + \eta_y^2} \end{aligned} \tag{15}$$

in which a is defined as being the local speed of sound. Equivalent expressions for Eq. (13), Eq. (14) and Eq. (15) are used for the η direction. For the implicit operators $D_{\xi_{impl.}}$ and $D_{\eta_{impl.}}$, a simple second difference operator is used, in order to keep the overall structure of the implicit operator as being two-sets of block tridiagonal matrices, which is reasonably cheaper to compute than a block pentadiagonal system of equations (resultant from the application of a forth difference implicit operator). Thus:

$$D_{\xi_{impl.}} = -\epsilon_{impl.} J^{-1} \nabla_\xi \Delta_\xi J, \qquad \epsilon_{impl.} = 3\left(\epsilon^{(2)} + \epsilon^{(4)}\right) \tag{16}$$

Analogue definitions can be made for the η direction.

3.2 Steger and Warming Scheme

When applying the Steger and Warming flux vector splitting method coupled with the explicit Euler time march scheme, the following expression is obtained for each internal point of the domain after applying an approximate LU factorization of the implicit operator (which reduces to an exact factorization in one-dimensional domains due to the way the flux splitting is done):

$$\begin{aligned} \left[I + \Delta t\left(\frac{\nabla_\xi \hat{A}^{+^n}}{\Delta\xi} + \frac{\nabla_\eta \hat{B}^{+^n}}{\Delta\eta}\right)\right]\left[I + \Delta t\left(\frac{\Delta_\xi \hat{A}^{-^n}}{\Delta\xi} + \frac{\Delta_\eta \hat{B}^{-^n}}{\Delta\eta}\right)\right]\Delta\hat{Q}^n = \\ = -\Delta t\left(\nabla_\xi \hat{E}^{+^n} + \Delta_\xi \hat{E}^{-^n} + \nabla_\eta \hat{F}^{+^n} + \Delta_\eta \hat{F}^{-^n}\right) \end{aligned} \tag{17}$$

where $\hat{E}^+$, $\hat{E}^-$, $\hat{F}^+$ and $\hat{F}^-$ are the split flux vectors and $\hat{A}^+$, $\hat{A}^-$, $\hat{B}^+$ and $\hat{B}^-$ are their respective Jacobian matrices, whose eigenvalues are strictly positive or negative, depending of which separated flux vector they are associated with. One important property added to the scheme by using a LU factorization of the implicit operator is that now the solution to the system of equations can be obtained by solving two distinct block (upper and lower) triangular systems. Equation (17) could be rewritten by the sequence:

$$\begin{aligned} \left[I + \Delta t\left(\frac{\nabla_\xi \hat{A}^{+^n}}{\Delta\xi} + \frac{\nabla_\eta \hat{B}^{+^n}}{\Delta\eta}\right)\right]\Delta\hat{Q}^{*^n} &= -\Delta t\left(\nabla_\xi \hat{E}^{+^n} + \Delta_\xi \hat{E}^{-^n} + \nabla_\eta \hat{F}^{+^n} + \Delta_\eta \hat{F}^{-^n}\right) \\ \left[I + \Delta t\left(\frac{\Delta_\xi \hat{A}^{-^n}}{\Delta\xi} + \frac{\Delta_\eta \hat{B}^{-^n}}{\Delta\eta}\right)\right]\Delta\hat{Q}^n &= \Delta\hat{Q}^{*^n} \end{aligned} \tag{18}$$

which is essentially a sequence of multiple triangular block implicit operators. If the implicit operator were built by traversing the transformed internal domain starting from the bottom left node and going into increasingly higher values of the ξ coordinate, followed by increasingly higher values of the η coordinate, each line of constant η in the transformed domain would correspond to a lower triangular system of equations to be solved for $\Delta\hat{Q}^{*^n}$. By doing the same for the second equation of Eq. (18), but now starting from the top right internal node and traversing into increasingly lower values of ξ and η, each line of constant η would now correspond to an upper triangular system of equations for $\Delta\hat{Q}^n$. Both of these types of systems have straightforward solutions (forward substitution for the first one and backward substitution for the second one) and thus are computationally efficient to solve.

Notice that, although Eq. (17) was written using first order forward and backward operators, nothing stops us of writing the same method with second-order one-sided operators (δ_ξ^b e δ_ξ^f, for backward and forward in ξ, respectively), while keeping the triangular structure of the matrix operators:

$$\left[I+\Delta t\left(\frac{\delta_\xi^b \hat{A}^{+^n}}{\Delta\xi}+\frac{\delta_\eta^b \hat{B}^{+^n}}{\Delta\eta}\right)\right]\left[I+\Delta t\left(\frac{\delta_\xi^f \hat{A}^{-^n}}{\Delta\xi}+\frac{\delta_\eta^f \hat{B}^{-^n}}{\Delta\eta}\right)\right]\Delta\hat{Q}^n = \\ =-\Delta t\left(\delta_\xi^b \hat{E}^{+^n}+\delta_\xi^f \hat{E}^{-^n}+\delta_\eta^b \hat{F}^{+^n}+\delta_\eta^f \hat{F}^{-^n}\right) \tag{19}$$

Due to the change in the stencil size from 2 points in Eq. (17) to 3 points in Eq. (19), a special treatment should be given to internal points located right next to boundary nodes. Here, we took the easy route and simply decreased the order of the spatial discretization for these points, essentially applying Eq. (17) instead of Eq. (19) in those regions.

The split flux vectors can be easily written by using the generalized flux vector f for two-dimensional problems, which is capable of reconstructing any flux vector provided the eigenvalues of its Jacobian matrix are given. First, the split eigenvalues $\hat{\lambda}^+$ and $\hat{\lambda}^-$ are calculated from their original values λ:

$$\begin{aligned}
&\lambda_1=\lambda_2=\theta=\kappa_x u+\kappa_y v\\
&\lambda_3=\lambda_1+a\sqrt{\kappa_x^2+\kappa_y^2}\\
&\lambda_4=\lambda_1-a\sqrt{\kappa_x^2+\kappa_y^2}\\
&\hat{\lambda}^+=\frac{\lambda+|\lambda|}{2},\qquad \hat{\lambda}^-=\frac{\lambda-|\lambda|}{2}
\end{aligned} \tag{20}$$

Then, by defining $\tilde{\kappa}_x$ and $\tilde{\kappa}_y$ as

$$\tilde{\kappa}_x=\frac{\kappa_x}{\sqrt{\kappa_x^2+\kappa_y^2}},\qquad\qquad \tilde{\kappa}_y=\frac{\kappa_y}{\sqrt{\kappa_x^2+\kappa_y^2}} \tag{21}$$

the generalized flux vector can be written as shown:

$$f=\frac{\rho}{2\gamma J}\left\{\begin{array}{c}
2(\gamma-1)\hat{\lambda}_1+\hat{\lambda}_3+\hat{\lambda}_4\\
2(\gamma-1)\hat{\lambda}_1 u+\hat{\lambda}_3(u+a\tilde{\kappa}_x)+\hat{\lambda}_4(u-a\tilde{\kappa}_x)\\
2(\gamma-1)\hat{\lambda}_1 v+\hat{\lambda}_3(v+a\tilde{\kappa}_y)+\hat{\lambda}_4(v-a\tilde{\kappa}_y)\\
(\gamma-1)\hat{\lambda}_1\left(u^2T+v^2\right)+\frac{\hat{\lambda}_3}{2}\left[(u+a\tilde{\kappa}_x)^2+(v+a\tilde{\kappa}_y)^2\right]+\frac{\hat{\lambda}_4}{2}\left[(u-a\tilde{\kappa}_x)^2+(v-a\tilde{\kappa}_y)^2\right]+\\
+\frac{(3-\gamma)\left(\hat{\lambda}_3+\hat{\lambda}_4\right)a^2}{2(\gamma-1)}
\end{array}\right\} \tag{22}$$

Thus, if the user wanted to calculate $\hat{E}^+$ for example, the κ's would be changed to ξ, and then each of the positive separated eigenvalues $\hat{\lambda}_1^+$, $\hat{\lambda}_2^+$, $\hat{\lambda}_3^+$ and $\hat{\lambda}_4^+$ would be calculated from Eq. (20) and the auxiliary variables $\tilde{\kappa}$ from Eq. (21), which would be used in Eq. (22) for computing $\hat{E}^+$. Lastly, since this is an implicit method that uses a quasi-linear extrapolation in order to estimate the values of the flux vectors in the next timestep, it is required that the split Jacobian matrices be calculated as well. Their separated eigenvalues are already known from Eq.(20), consequently the split Jacobian matrices could be reconstructed by using their eigenvector matrices $[T]$ and separated eigenvalue matrix $\hat{\lambda}$. The form of these eigenvector matrices was shown by Pulliam and Chaussee (1981), and are replicated below:

$$\left[\hat{\lambda}\right]_\kappa=\begin{bmatrix}\hat{\lambda}_1 & 0 & 0 & 0\\ 0 & \hat{\lambda}_2 & 0 & 0\\ 0 & 0 & \hat{\lambda}_3 & 0\\ 0 & 0 & 0 & \hat{\lambda}_4\end{bmatrix} \tag{23}$$

$$\alpha=\frac{\rho}{\alpha\sqrt{2}},\qquad\qquad \beta=\frac{1}{\rho a\sqrt{2}},\qquad\qquad \tilde{\theta}=\tilde{\kappa}_x u+\tilde{\kappa}_y v \tag{24}$$

$$[T]_\kappa=\begin{bmatrix}1 & 0 & \alpha & \alpha\\ u & \tilde{\kappa}_y\rho & \alpha(u+\tilde{\kappa}_x a) & \alpha(u-\tilde{\kappa}_x a)\\ v & -\tilde{\kappa}_x\rho & \alpha(v+\tilde{\kappa}_y a) & \alpha(v-\tilde{\kappa}_y a)\\ \frac{\phi^2}{\gamma-1} & \rho(\tilde{\kappa}_y u-\tilde{\kappa}_x v) & \alpha\left[\frac{\phi^2+a^2}{\gamma-1}+a\tilde{\theta}\right] & \alpha\left[\frac{\phi^2+a^2}{\gamma-1}-a\tilde{\theta}\right]\end{bmatrix} \tag{25}$$

$$[T]_{\kappa}^{-1} = \begin{bmatrix} \left(1-\frac{\phi^2}{a^2}\right) & \frac{(\gamma-1)u}{a^2} & \frac{(\gamma-1)v}{a^2} & -\frac{(\gamma-1)}{a^2} \\ -\frac{(\tilde{\kappa}_y u - \tilde{\kappa}_x v)}{\rho} & \frac{\tilde{\kappa}_y}{\rho} & -\frac{\tilde{\kappa}_x}{\rho} & 0 \\ \beta\left(\phi^2 - a\tilde{\theta}\right) & \beta\left[\tilde{\kappa}_x a - (\gamma-1)u\right] & \beta\left[\tilde{\kappa}_y a - (\gamma-1)v\right] & \beta(\gamma-1) \\ \beta\left(\phi^2 + a\tilde{\theta}\right) & -\beta\left[\tilde{\kappa}_x a + (\gamma-1)u\right] & -\beta\left[\tilde{\kappa}_y a + (\gamma-1)v\right] & \beta(\gamma-1) \end{bmatrix} \tag{26}$$

$$[P] = [T]_{\kappa}\,[\hat{\lambda}]_{\kappa}\,[T]_{\kappa}^{-1} \tag{27}$$

where $[P]$ is being used to describe any of the possible split flux vector Jacobian matrices. So, if $\hat{A}^+$ was to be calculated, then $[\hat{\lambda}^+]_{\xi}$ would be defined by Eq. (23), $[T]_{\xi}$ and $[T]_{\xi}^{-1}$ would be determined by using Eq. (25) and Eq. (26) and $\hat{A}^+$ would then be calculated from Eq. (27).

Since the usage of upwind schemes intrinsically adds artificial dissipation to the solution (a property of one-sided operators), no extra terms are added to Eq. (17) and Eq. (19). Theoretically, the amount of artificial dissipation being added to the solution will be a function of the local grid space of the mesh. The finer it is, the lower is the amount of artificial dissipation being indirectly introduced to the system.

3.3 Van Leer Scheme

The explicit implementation of van Leer's flux vector splitting method would be represented in the same manner as Steger and Warming's explicit versions of Eq. (17) and Eq. (19). In other words, the implicit operator on those equations is made equal to the identity matrix:

$$\hat{Q}^{n+1} = \hat{Q}^n - \Delta t\left(\nabla_{\xi}\hat{E}_{VL}^{+^n} + \Delta_{\xi}\hat{E}_{VL}^{-^n} + \nabla_{\eta}\hat{F}_{VL}^{+^n} + \Delta_{\eta}\hat{F}_{VL}^{-^n}\right) \tag{28}$$

$$\hat{Q}^{n+1} = \hat{Q}^n - \Delta t\left(\delta_{\xi}^{b}\hat{E}_{VL}^{+^n} + \delta_{\xi}^{f}\hat{E}_{VL}^{-^n} + \delta_{\eta}^{b}\hat{F}_{VL}^{+^n} + \delta_{\eta}^{f}\hat{F}_{VL}^{-^n}\right) \tag{29}$$

where Eqs. Eq. (28) and Eq. (29) are the explicit Euler versions of van Leer's scheme using first and second order difference operators, respectively. The VL subscript that follows each split flux vector specifies that they are being calculated using van Leer's definition of flux vector splitting. Essentially, his method keeps the eigenvalues of the Jacobian matrices of each split flux vector continuously differentiable throughout all values of local Mach number (which was not true in the Steger and Warming scheme), increasing the methods robustness and improving the quality of the solution in regions close to discontinuities. In order to appropriately specify them, first it is necessary to define q as being the velocity vector magnitude and M_{κ} as the Mach Number calculated only using the velocity component in the κ direction, as shown below:

$$q^2 = u^2 + v^2 \tag{30}$$

$$M_{\kappa} = \frac{\tilde{\kappa}_x u + \tilde{\kappa}_y v}{a} \tag{31}$$

Then, the split flux $g^{\pm}$ related to the continuity equation is computed by the following equation:

$$g^{\pm} = \pm\frac{\rho a\left(M_{\kappa} \pm 1\right)^2\sqrt{\kappa_x^2 + \kappa_y^2}}{4J} \tag{32}$$

For a subsonic region of the flow in the κ direction, the generalized van Leer split flux vector $f_{VL}^{\pm}$ is calculated according to Eq. (33), whereas for the supersonic region, the van Leer split reduces to the Steger and Warming one, from Eq. (22) using the eigenvalues from Eq. (20):

$$\begin{aligned} f_{VL}^{\pm} &= g^{\pm}\begin{Bmatrix} 1 \\ u + \frac{\tilde{\kappa}_x a(-M_{\kappa}\pm 2)}{\gamma} \\ v + \frac{\tilde{\kappa}_y a(-M_{\kappa}\pm 2)}{\gamma} \\ \frac{q^2-(M_{\kappa}a)^2}{2} + a^2\frac{[(\gamma-1)M_{\kappa}\pm 2]^2}{2(\gamma^2-1)} \end{Bmatrix}, \qquad for\,|M_{\kappa}| < 1. \\ f_{VL}^{\pm} &= f, \qquad for\,|M_{\kappa}| \geq 1. \end{aligned} \tag{33}$$

Thus, the next time step in the simulation can be simply determined by calculating $f_{VL}^{\pm}$ and substituting them on either Eq. (28) or Eq. (29), depending on the desired order for the space difference operators.

3.4 Liou's AUSM$^+$ Scheme

Liou's sequel to the Advection Upstream Splitting Method, entitled AUSM$^+$, follows a different approach to perform the split of the flux vectors. Here, the flux vector is seen as a composition of the flux of passive scalars being convected by the velocity field and pressure terms that compose a pressure flux. These two fluxes (taken as numerical fluxes) are split in different manners, coupled with a special definition for the speed of sound evaluated at a numerical frontier between two adjacent nodes in the domain, in order to keep the unification between the Mach Number and the corresponding local velocity. More information regarding the development of the method can be found in Liou (1996), but the overall algorithm adapted to the Euler Equations in general curvilinear coordinates applied to the context of finite differences will be shown here.

First, it is necessary to define the speed of sound a evaluated at the numerical frontier $i+1/2$ (which is located between two adjacent points of the transformed domain, either in the ξ or η direction):

$$a_{i+\frac{1}{2}} = min\left(\tilde{a}_i, \tilde{a}_{i+1}\right), \qquad \tilde{a} = \frac{{a^*}^2}{max\left(a^*, |M_\kappa a|\right)} \tag{34}$$

where a^* is the local critical speed of sound. Then, two functions $\mathcal{M}$ and $\mathcal{P}$ are defined in order to perform the split of the Mach Number M_κ and the pressure terms. Just as it was achieved by the van Leer method, these functions ensure that the properties of the system are continuous and differentiable. The $\mathcal{M}$ function is defined as follows:

$$\begin{aligned}
&\mathcal{M}^{\pm}(\tilde{M}) = \begin{cases} \frac{1}{2}(\tilde{M} \pm |\tilde{M}|), & if\ |\tilde{M}| \geq 1, \\ \mathcal{M}_\beta^{\pm}(\tilde{M}), & otherwise. \end{cases} \\
&\mathcal{M}_\beta^{\pm}(\tilde{M}) = \pm\frac{1}{4}(\tilde{M} \pm 1)^2 \pm \beta(\tilde{M}^2 - 1)^2, \quad -\frac{1}{16} \leq \beta \leq \frac{1}{2}
\end{aligned} \tag{35}$$

where β is a control parameter whose recommended value is $\beta = 1/8$. For the $\mathcal{P}$ function, we have:

$$\begin{aligned}
&\mathcal{P}^{\pm}(\tilde{M}) = \begin{cases} \frac{1}{2}\left(1 + sign(\tilde{M})\right), & if\ |\tilde{M}| \geq 1, \\ \mathcal{P}_\alpha^{\pm}(\tilde{M}), & otherwise. \end{cases} \\
&\mathcal{P}_\alpha^{\pm}(\tilde{M}) = \frac{1}{4}(\tilde{M} \pm 1)^2(2 \mp \tilde{M}) \pm \alpha\tilde{M}(\tilde{M}^2 - 1)^2, \ -\frac{3}{4} \leq \alpha \leq \frac{3}{16}
\end{aligned} \tag{36}$$

Once again, α is a control parameter whose recommended value is $\alpha = 3/16$. The values of $\tilde{M}$ present on both Eq. (35) and Eq. (36) are local Mach Numbers in the κ direction evaluated using $a_{i+1/2}$:

$$\tilde{M}_i = \frac{(M_\kappa a)_i}{a_{i+\frac{1}{2}}}, \qquad \tilde{M}_{i+1} = \frac{(M_\kappa a)_{i+1}}{a_{i+\frac{1}{2}}} \tag{37}$$

The Mach Number $m_{i+1/2}$ evaluated at the numerical frontier $i + 1/2$ can then be defined as:

$$m_{i+\frac{1}{2}} = \mathcal{M}^+(\tilde{M}_i) + \mathcal{M}^-(\tilde{M}_{i+1}) \tag{38}$$

The split of this value can be achieved in a similar manner to the split of eigenvalues used in the Steger and Warming method:

$$m_{i+\frac{1}{2}}^{\pm} = \frac{m_{i+\frac{1}{2}} \pm \left|m_{i+\frac{1}{2}}\right|}{2} \tag{39}$$

The pressure component p evaluated at the numerical frontier $i + 1/2$ follows the same idea:

$$p_{i+\frac{1}{2}} = \mathcal{P}^+(\tilde{M}_i)p_i + \mathcal{P}^-(\tilde{M}_{i+1})p_{i+1} \tag{40}$$

The final form of the generic numerical flux vector $f_{i+1/2}$ is:

$$f_{i+\frac{1}{2}} = a_{i+\frac{1}{2}} \left[m^{+}_{i+\frac{1}{2}} \left(\frac{\sqrt{\kappa_x^2 + \kappa_y^2}}{J} \right)_i \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ e+p \end{Bmatrix}_i + m^{-}_{i+\frac{1}{2}} \left(\frac{\sqrt{\kappa_x^2 + \kappa_y^2}}{J} \right)_{i+1} \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ e+p \end{Bmatrix}_{i+1} \right] + \\ + \begin{Bmatrix} 0 \\ p_{i+\frac{1}{2}} \left[\left(\kappa_x \mathcal{P}^{+}(\tilde{M}) \right)_i + \left(\kappa_x \mathcal{P}^{-}(\tilde{M}) \right)_{i+1} \right] \\ p_{i+\frac{1}{2}} \left[\left(\kappa_y \mathcal{P}^{+}(\tilde{M}) \right)_i + \left(\kappa_y \mathcal{P}^{-}(\tilde{M}) \right)_{i+1} \right] \\ 0 \end{Bmatrix} \tag{41}$$

A time march can then be performed as follows:

$$\hat{Q}^{n+1} = \hat{Q}^n - \Delta t \left(\hat{E}^n_{i+\frac{1}{2},j} - \hat{E}^n_{i-\frac{1}{2},j} + \hat{F}^n_{i,j+\frac{1}{2}} - \hat{F}^n_{i,j-\frac{1}{2}} \right) \tag{42}$$

4. SPATIAL DISCRETIZATION AND BOUNDARY CONDITIONS

The overall geometry of the nozzle used here has been kept unchanged in relation to the one used by Azevedo *et al.* (1995). The four meshes generated from said geometry, though, follow a different approach. Since one of our main goals is to identify the impact of the mesh refinement over the simulation's output data, it has been decided that all four meshes would be built by using simple algebraic schemes, with approximate constant spacing between adjacent nodes in the ξ and η directions. The reasoning behind this is that the differences between each mesh are constrained to only different values of node points per unity area, which is approximately constant throughout the entire discrete domain of a given mesh.

Table 1. Number of nodes of each mesh. Finer meshes were created by doubling the amount of nodes of the previous mesh in each direction.

Mesh	Number of ξ Nodes	Number of η Nodes	Total Number of Nodes
1	25	12	300
2	51	24	1224
3	103	48	4944
4	207	96	19872

The first mesh, referred here as simply "Mesh 1", was created by using 25 points in the ξ direction and 12 points in the η direction. The constant ξ lines were kept aligned with the y axis for simplicity, even though in general keeping these lines perpendicular to the wall reduces the error introduced by extrapolating pressure and temperature during the enforcement of the boundary conditions at this location. Azevedo *et al.* (1995) has shown that since the angles of the convergent and divergent sections of this nozzle are sufficiently small, this simplification has no significant impact in the solution of the Euler Equations. The process of generating each of the three other consecutive meshes is straightforward and done by approximately doubling the number of points in each curvilinear direction. Table 1 shows the number of nodes used in each direction for all four meshes, while Fig 1 and Fig 2 shows Mesh 1 and Mesh 4. Notice that an extra line of constant η has been added beyond the nozzle's center line. This has been done in order to facilitate the implementation of the symmetry boundary condition in this region.

The nozzle has a total length of $0.1158\ m$ in which the throat is located exactly at its center. The convergent section has an angle of $22.33°$ in relation to the x axis measured counterclockwise, while the divergent section has an angle of $1.21°$ measured clockwise in relation to the same axis. The centerline is taken to be a simple symmetry plane and the bottom boundary is an eulerian wall with $0.0274\ m$ of curvature radius near the throat. The left boundary is the nozzle's inlet, while the right one is its output. The fluid is taken to be atmospheric air with a constant heat capacity ratio γ of 1.4 and an individual gas constant R of approximately $287\ J/(kg\ K)$.

During the entire simulation the flow in the convergent section of the domain is kept at subsonic speeds and thus it is known, from the characteristic relations, that three different boundary conditions must be specified at the left boundary, while the fourth one is obtained by performing an extrapolation from the inner domain. Supposing that there is a reservoir upstream of the nozzle's intake where the fluid is kept at a known stagnation state, it seems reasonable to define the left boundary by means of the fluid's stagnation properties, mainly a total pressure P_t of $1.0136 \times 10^5\ Pa$ and a total

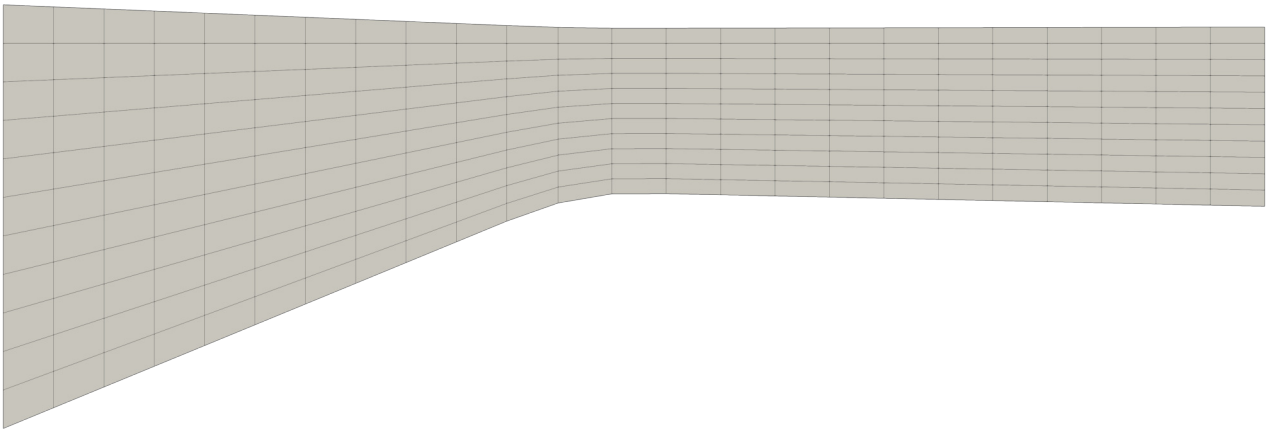

Figure 1. Computational Mesh 1.

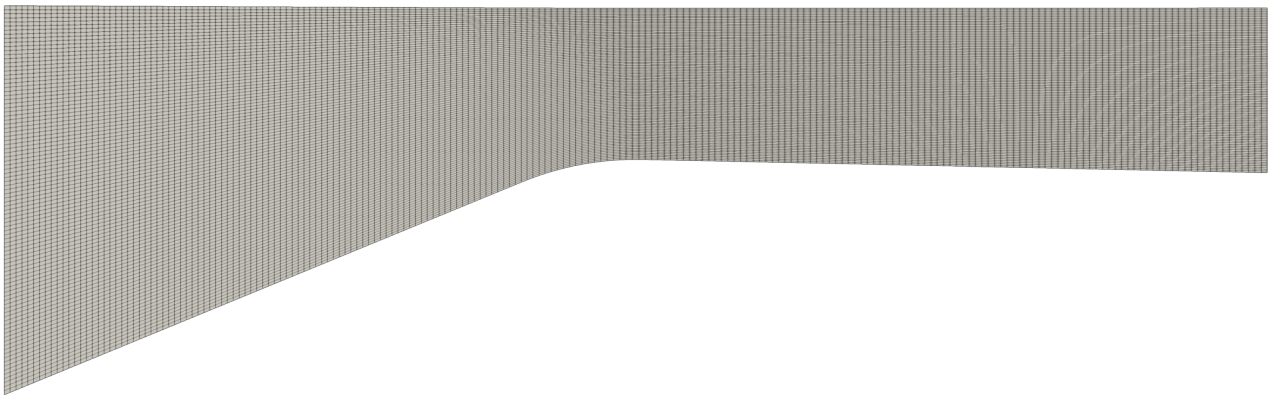

Figure 2. Computational Mesh 4.

temperature T_t of $295.1\ K$. The third condition is simply taken as a zero flow entrance angle. Although at a first glance this last condition might not be necessarily considered to be realistic, since it is expected that the direction of the velocity profile smoothly transitions from being essentially aligned with the x axis in the centerline to being tangent to the wall at the wall itself, it has been seen that no significant differences in results are obtained by considering a curvature in the domain's intake region. Lastly, the horizontal velocity component u is obtained by zero-order extrapolation.

At the output plane it is expected the flow to be supersonic, making its state vector completely defined by the solution in the inner domain at any given time. Therefore, all 4 boundary conditions in this region are obtained by zero-order extrapolation. As will be seen later, the simulation starts in a state where the entire domain is subsonic, requiring that one boundary condition to be specified in this region. While not supersonic, the static pressure p of the output is kept constant and equal to $P_t/3$, which was seen to ensure that the flow will become supersonic without introducing instabilities to the solution during the time march.

In order to treat the symmetry condition, an extra line of constant η has been introduced past the centerline by mirroring the original mesh. Scalar properties of the nodes that are part of this line are made equal to their counterparts in the inner domain. On the other hand, vectorial variables (velocity components, for example) are taken to have the same magnitude of their inner domain counterparts, but opposite direction, in order to keep the perpendicular flux across the centerline equal to zero for any property.

At the wall we have used a strong form of Mavriplis (1990) boundary condition, where the contravariant velocity component U is obtained by zero-order extrapolation from the adjacent node and V is set to zero in order to keep the flow tangent to the wall, thus enabling the determination of u and v. By imposing a zero normal pressure and temperature gradients at the wall, the two other properties can be calculated. It must be said that due to the existence of a wall curvature near the throat, forcing a zero pressure gradient at this boundary is merely a simplification of the fluid dynamics in this region, since the correct approach would be to take the wall curvature into consideration. No significant errors, though, were found to be introduced in the results by making this simplification.

The domain is initialized by considering that the flow at every inner node starts at zero velocity, with pressure and temperature equal to the values set at the inlet. In order to also verify the impact that clustering nodes at the throat has over the solution, four other meshes have been generated based on the ones shown before, but now applying an exponential growth in the η direction near the wall and in the ξ direction towards the throat. Figure 3 shows the equivalent of Mesh 4

made under these new assumptions.

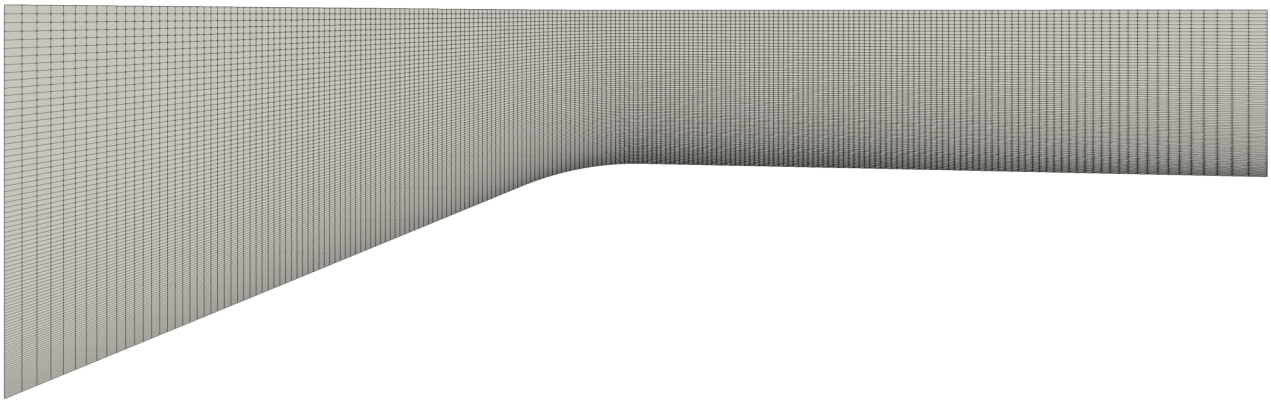

Figure 3. Computational Mesh 4 created using exponential growth in the near wall region and towards the throat.

5. RESULTS AND DISCUSSION

Values of pressure, made dimensionless in relation to the inlet total pressure, along the span of the nozzle, made dimensionless in relation to its half-length l, for the centerline and wall regions calculated using the Beam and Warming scheme coupled with the nonlinear 4th differences Pulliam model with $K_2 = 1/8$ and $K_4 = 1/200$ are shown in Fig 4 and Fig 5 for the four original meshes. As seen, the results for the convergent section ($x/l \leq 0$) are approximately equal for all four meshes, for both the centerline and the wall regions. There is, though, one notable difference: the presence of oscillations near the left boundary associated with the coarse meshes 1 and 2. These oscillations are related to the zero-order extrapolation used here to compute some of the flow properties at the boundaries. This simplification introduces an extra restriction to the solution by indirectly forcing the gradient of the property being extrapolated to be approximately zero at the boundaries. For coarse meshes, this behaviour can lead to severe constraints in the model that do not reflect the actual flow dynamics, specially at the boundaries where the fluid flux across them is significant. This culminates in the emergence of the oscillations seen not only for the left boundary, but also the right boundary. Notice that these oscillations are completely absent from the finer meshes 3 and 4, since the distance between the nodes located at these boundaries to the ones adjacent to them is sufficiently small to the point where the introduction of this indirect constrain results in a property value that is not off from the actual value that would be obtained if, for example, characteristic relations were used instead. In order to avoid this problem, the unknown properties at the boundaries should be either calculated by using equations that closely resembles the dynamics of these regions (the characteristic relations, for example) or even, at least for the case studied here, use first order extrapolation instead, since the restriction that causes these oscillations would be lifted on both cases. Caution should be taken if the latter solution is used, since it was seen that it can introduce instabilities in the time-march if strong discontinuities are present near the boundary. Figure 6 shows a comparison regarding the pressure distribution along the wall calculated using zero-order and first-order extrapolations at the boundaries applied to Mesh 1. It clearly shows that the usage of first-order extrapolation eliminates the oscillations while, at the same time, also slightly increases the computed pressure drop across the throat, improving the results if compared with the ones obtained for Mesh 4.

Turning the attention back to the results displayed in Fig. 4 and Fig. 5, it is possible to notice that the values obtained for the pressure along the divergent part of the nozzle are substantially different from mesh to mesh. As the grid refinement increases, there is a clear tendency of oblique shockwaves to develop at the throat, appearing first at the wall (as seen from the local minimum located at $x/l \approx 0$ in Fig. 5), followed by alternating reflections at the centerline and wall, respectively. Both Mesh 3 and Mesh 4 obtained similar results, albeit the latter was capable of better defining the property gradient across the shock, as expected from the usage of more grid points. The problem arises when looking at the solution obtained for the coarse meshes 1 and 2. Both of them correctly determine that the flow in the divergent region is supersonic and also that the average Mach number across the output is around 1.3, but they completely underestimate the pressure drop across the throat while also inhibit the development of the shock. It is interesting to notice, though, that the local pressure peaks obtained at both the centerline and at the wall when using Mesh 3 and Mesh 4 seems to coincide with the pressure values obtained by Mesh 1 at these regions, which is probably the reason why this coarse mesh is capable of defining the macro flow structures even if the shockwave is not present. Mesh 2, on the other hand, is somewhat capable of identifying the start of the shock structure up to the first shock reflection, but quickly dissipates it to the point where the shockwave seizes to exist. Since the artificial dissipation model used when calculating the solution of all four cases were equal while also using the same control parameters, the total or partial absence of the shock in Mesh 1 and Mesh 2 leads us to believe that the artificial dissipation naturally added by the spatial discretization to the model being used in this central differences

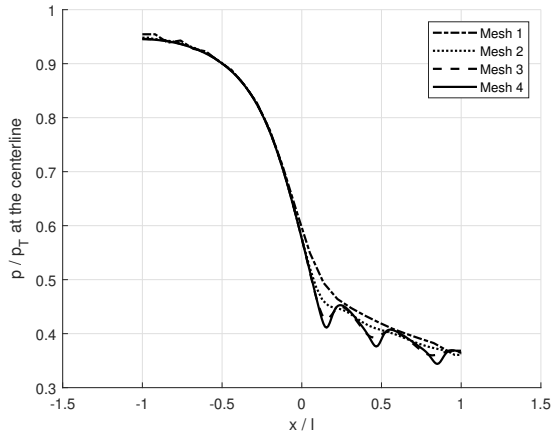

Figure 4. Pressure distribution at the centerline for the four different meshes without exponential growth.

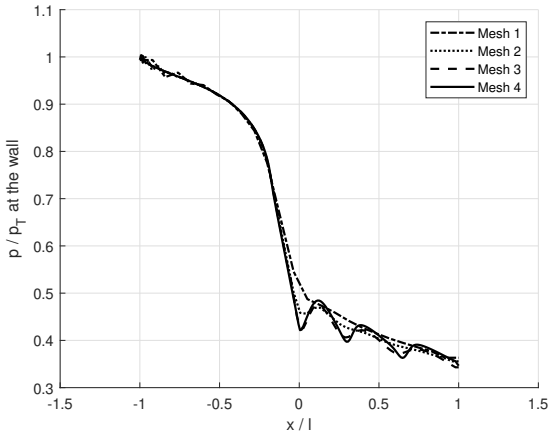

Figure 5. Pressure distribution at the wall for the four different meshes without exponential growth.

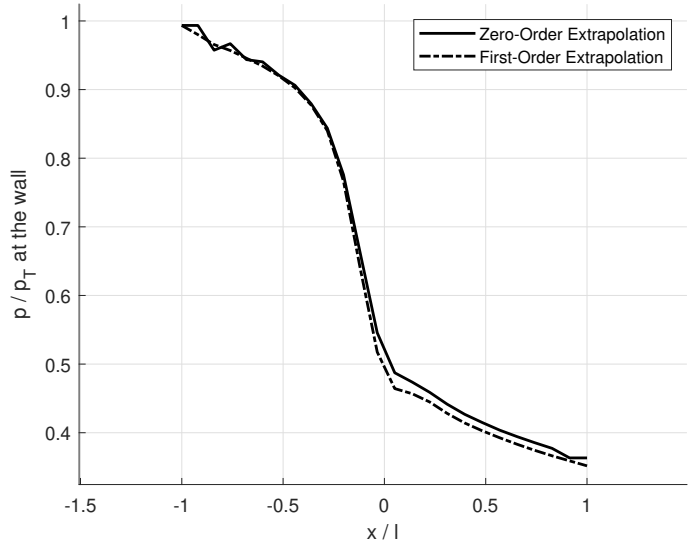

Figure 6. Dimensionless pressure distribution along the wall calculated using the Beam and Warming applied to Mesh 1 using both zero-order and first-order extrapolations for the unknown variables at the boundaries.

scheme is the main culprit of this behaviour. It is well known that the amount of artificial dissipation introduced locally to a discrete solution is proportional to the grid spacing used throughout the numerical domain, becoming lower the finer the mesh is at a given location. This is apparent especially for upwind schemes, where the artificial dissipation can be explicitly written as a function of the grid spacing if the modified equation is determined.

In order to verify if this behaviour is indeed promoted by the added numerical dissipation, due to the usage of a coarse grid spacing, and is not restricted to the Beam and Warming method, the results for the dimensionless pressure distributed along the wall obtained using the the Beam and Warming, Steger and Warming and van Leer second order schemes are plotted in Fig. 7 and in Fig. 8 for Mesh 1 and Mesh 2, respectively. When referring to Mesh 1, both upwind schemes obtained almost identical results for the convergent section of the domain, as well as the region near the nozzle's output in the divergent section. This behaviour was expected, since the main differences between the Steger and Warming scheme and the van Leer one lies on the way that the split flux vectors are evaluated when the flow is in a near sonic regime. In this case, this condition is achieved in the area that surrounds the throat, thus explaining why both methods behave differently in this region. The Beam and Warming method, though, underestimates the pressure values along the convergent section by around 10% in relation to the upwind schemes, but the overall shape of the pressure profile appears to be concise with each other. It is also possible to notice that, exactly at the throat, the van Leer scheme obtained a result that closely approaches the values calculated using the Beam and Warming scheme, while the Steger and Warming one displays what appears to be a discontinuity in the pressure gradient along the wall. This is a property of the latter method, where the spatial derivatives of the split flux vectors are not necessarily continuous across a sonic line, leading

to discrepancies in the flow features as this one. This is not observed in the van Leer scheme simply because the method itself was built with the purpose of solving this problem. When using Mesh 1, none of the methods were able to capture the development of shockwaves, completely dampening them without altering the average pressure profile along the wall in a significant manner. It is also interesting to notice that these two upwind schemes appear to be less susceptible to the zero-order extrapolation induced oscillations at the boundaries, even for this coarse mesh. This is probably because they naturally add sufficient numerical dissipation to the model when coupled with a coarse mesh, while the non-linear artificial dissipation model used here with the Beam and Warming scheme has to be treated differently at the boundaries (due to the size of the stencil), thus not introducing enough dissipation to smooth out oscillations in the property's profile.

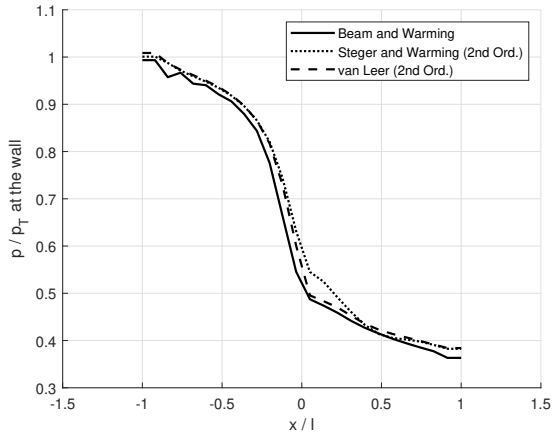

Figure 7. Pressure distribution along the wall for Mesh 1 calculated using three different second order schemes.

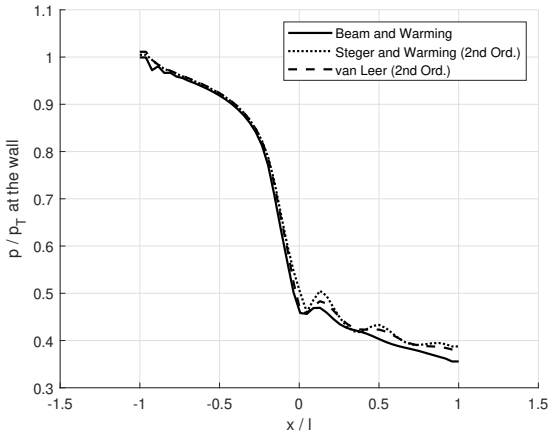

Figure 8. Pressure distribution along the wall for Mesh 2 calculated using three different second order schemes.

When turning the attention to the results obtained for Mesh 2, it is seen that the finer grid spacing leads the differences between the schemes to be considerably reduced at the convergent section of the domain, to the point where they are now essentially identical. From the throat and onwards, though, there is a sensible difference not only in the pressure values, but also in the overall shape of the curve. The three schemes were able to successfully identify the formation of the shockwave, followed by a sequence of wall reflections. Observing the first local maximum located after the throat, it is seen that the Beam and Warming scheme coupled with the artificial dissipation model used here dampens the strength of the shock considerably in relation to the other methods. Furthermore, the following shock reflections are only perceived by the Beam and Warming scheme as almost negligible inflection points in the curve, while the upwind schemes are able to reasonably maintain the shock definition. Thus, the presence of a strong shockwave dampening along the nozzle's length when using the Beam and Warming scheme points to the presence of excessive numerical dissipation, since Mesh 2 was proven to have a fine enough grid to correctly portray the shock reflections, as seen from the results obtained by the upwind schemes. Unfortunately, not much can be done besides refining the mesh in order to obtain a better definition of the shockwave when using the second order Beam and Warming scheme as used here, since decreasing the control constants of the artificial dissipation model leads to problems in the convergence of the solution.

If first order schemes were used instead, the results shown on Fig. 9 would be obtained for the dimensionless pressure distribution along the wall when applied to Mesh 4. These results include not only the first order versions of the Steger and Warming and van Leer schemes, but also the first order Liou's AUSM$^+$. As expected, first order upwind schemes naturally introduce more artificial dissipation than their second order counterparts, severely dampening the shock reflections but still managing to acknowledge the existence of a shockwave in the first place. The values obtained for the pressure drop at the throat are similar between the van Leer and AUSM$^+$ schemes and are in line with the values that were obtained with the second order schemes applied to the same mesh. The first order Steger and Warming scheme not only underestimates this pressure drop, but also significantly widens the shock's contact region at the throat. Following shock reflections are almost non-existent in the Steger and Warming scheme, safe for some inflection points that can be identified in the pressure plot. These shock reflections are more pronounced in both van Leer and AUSM$^+$ methods, where Liou's scheme closely resembles the results obtained for the Beam and Warming method when using Mesh 2, while both first order van Leer and first order Steger and Warming Schemes are not capable of achieving the same shock definition as their second order counterparts using Mesh 2. This behaviour shows that even though, in theory, the order of magnitude of the spatial discretization error introduced by the first order schemes using Mesh 4 are approximately equal to the ones introduced by the second order schemes using Mesh 2, the amount of artificial dissipation introduced by the former methods is much higher than the latter, amplifying the diffusive effects of the solution and smoothing out the oscillations promoted by the presence of the shockwave. This effect is less apparent in the results shown for the Liou's scheme, probably due to the separate treatment that exists between the pressure flux and the convection of passive scalar variables in this method.

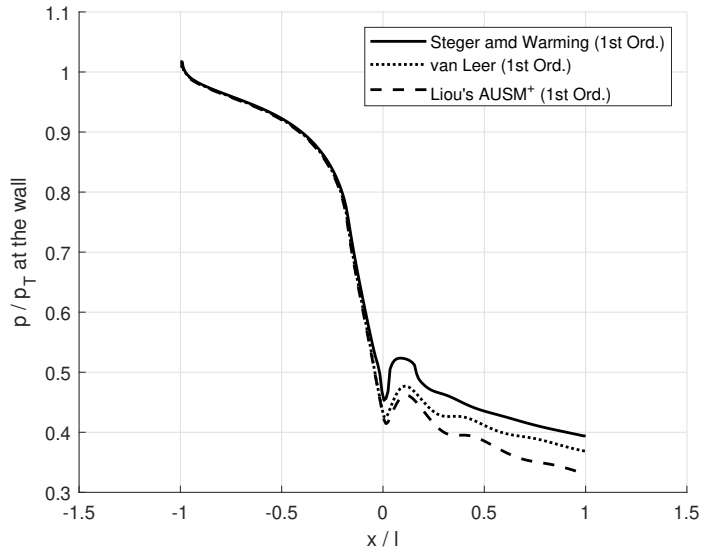

Figure 9. Pressure distribution along the wall for Mesh 4 calculated using three different first order upwind schemes.

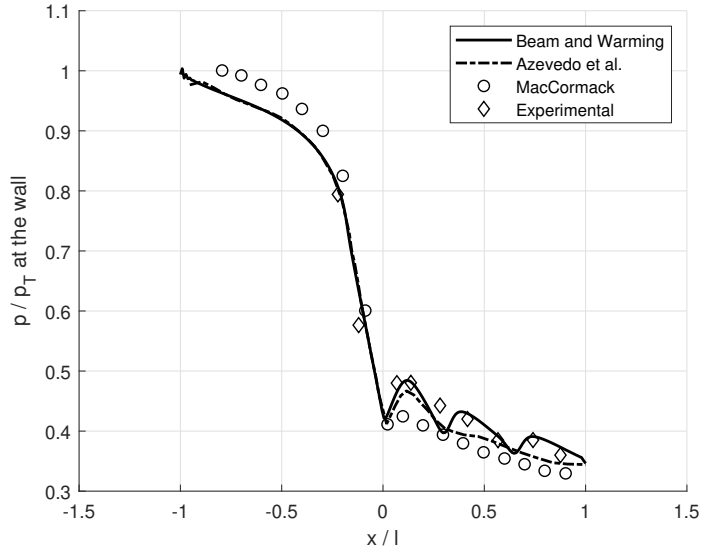

Figure 10. Pressure distribution along the wall as numerically determined by Azevedo *et al.* (1995) and MacCormack (1985), as well as the experimental data obtained by Mason *et al.* (1980) in comparison to the values obtained here using the Beam and Warming scheme applied to Mesh 4.

The numerical results from MacCormack (1985) and Azevedo *et al.* (1995) are now plotted in Fig. 10 against the ones obtained here using Mesh 4 with the Beam and Warming method. Also included for validation purposes are the experimental results from Mason *et al.* (1980). Despite using an inviscid formulation, the numerical results presented were able to achieve satisfactory agreement with the available experimental data, pointing towards the idea that this flow is mainly governed by inertial forces and viscous forces do not contribute significantly towards the dynamics in play. Both MacCormack's and Azevedo's results approximately coincides with the values obtained here for the local minimum pressure in the divergent section, but the former essentially does not capture the presence of a shockwave, while the latter correctly estimates the formation of a shock, but quickly dissipates it, akin to the behaviour displayed by the second order Beam and Warming method results obtained using Mesh 2. Notice that the experimental data shows higher values of pressure than the ones determined by Azevedo et al., comparable in magnitude with the data shown for the finer mesh used here. Thus, it can be said that even though the mesh used by Azevedo et al. had a sufficient small grid spacing for correctly capturing all the shock reflections (as shown here when using Mesh 2 with the second order upwind schemes), the artificial dissipation model coupled with the intrinsic numerical dissipation introduced by the spatial discretization culminated in the destruction of the shock reflection structures. Thus, a denser mesh should have been used in order to

enable the propagation of the shockwave throughout the divergent section of the nozzle. Just for the sake of completeness, since the node disposition of the meshes used here and the ones present in the referred works are not the same, a similar analysis to the one shown in this paper was also performed using the meshes created with the exponential growth discussed in the previous section. No significant differences were observed for this new case and the conclusions herein displayed continued to be valid.

6. CONCLUDING REMARKS

A simulation of a two-dimensional transonic flow along a convergent-divergent nozzle has been performed using four different numerical schemes for the solution of the Euler Equations in the context of finite differences. Four different meshes of increasingly higher degree of refinement were used in order to access the influence of the spatial discretization over the amount of artificial dissipation naturally introduced to the solution. It has been seen that not using a sufficiently fine mesh can lead to the introduction of an excessive amount of numerical dissipation, which resulted in the dampening of shockwave structures that would otherwise be present in the flow along the divergent section of the domain. This behaviour was seen to be more severe when using the second order Beam and Warming centered scheme, coupled with Pulliam's 4th differences nonlinear artificial dissipation model, than when using the second order Steger and Warming and van Leer upwind schemes. The latter methods were able to correctly capture the series of shock reflections that appear downstream the nozzle's throat even when using a mesh that would otherwise dampen them if the centered scheme was used instead.

The simulation configurations that were seen to naturally introduce less artificial dissipation to the solution are the ones that achieved the best agreement in relation to experimental data. First order schemes correctly captured the formation of the shock, but would otherwise introduce errors in other flow features even when using a fine mesh. Exception to this observation is the Liou's AUSM$^+$ method, which was able to correctly estimate the pressure values along the wall of the divergent section, albeit strongly dampening the shock reflections.

It was also seen that the numerical results obtained by MacCormack (1985) and Azevedo *et al.* (1995), although able to correctly estimate the average static flow pressure along the wall, introduced excessive artificial dissipation to the solution, preventing the shockwaves to develop at the divergent section and leading to a small but sensible underestimation of the pressure values in this region of the domain. It is expected that by simply improving the overall node density of their discrete domains by a factor of 2, for example, would enable the shockwave to propagate without being damped before reaching the nozzle's output. A natural follow-up to the current study would be to verify the impact of the inclusion of viscous terms over the simulation results.

7. ACKNOWLEDGEMENTS

The authors wish to express their gratitude to *Coordenação de Aperfeiçoamento de Pessoal de Nível Superior*, CAPES, which has supported the present research through a graduate scholarship for the first author. The authors also gratefully acknowledge the support for the present research provided by *Conselho Nacional de Desenvolvimento Científico e Tecnológico*, CNPq, under the Research Grant No. 309985/2013-7. The work is further supported by *Fundação de Amparo à Pesquisa do Estado de São Paulo*, FAPESP, under the Research Grant No. 2013/07375-0.

8. REFERENCES

Azevedo, J.L.F., Fico Junior, N.G.C.R. and Ortega, M.A., 1995. "Two-dimensional and axisymmetric nozzle flow computations using the euler equations". *Journal of the Braz. Soc. Mechanical Sciences*, Vol. XVII, No. 2, pp. 147–170.

Beam, R.M. and Warming, R.F., 1978. "An implicit factored scheme for the compressible navier-stokes equations". *AIAA Journal*, Vol. 16, No. 4, pp. 393–402.

Liou, M.S., 1996. "A sequel to ausm: Ausm$^+$". *Journal of Computational Physics*, Vol. 129, pp. 364–382.

MacCormack, R.W., 1985. "Current status of numerical solutions of the navier-stokes equations". *AIAA 23rd Aerospace Sciences Meeting*, , No. 85-0032.

Mason, M.L., Putnam, L.E. and Re, R.J., 1980. "The effect of throat contouring on two-dimensional converging-diverging nozzles at static conditions". *Nasa Technical Paper*, , No. 1704.

Mavriplis, D.J., 1990. "Accurate multigrid solution of the euler equations on unstructed and adaptive meshes". *AIAA Journal*, Vol. 28, No. 2, pp. 213–221.

Pulliam, T.H., 1986a. "Artificial dissipation models for the euler equations". *AIAA Journal*, Vol. 24, No. 12, pp. 1931–1940.

Pulliam, T.H., 1986b. "Solution methods in computational fluid dynamics". Notes for the von Kármán Institute for Fluid Dynamics Lecture Series, von Kárman Institute, Rhode-St-Genese, Belgium (available in pdf from <https://scholar.google.com/citations?user=Po3__8gAAAAJ&hl=en>, last time accessed March 2019).

Pulliam, T.H. and Chaussee, D.S., 1981. "A diagonal form of na implicit approximate-factorization algorithm". *Journal*

of Computational Physics, Vol. 39, pp. 347–363.

Steger, J.L. and Warming, R.F., 1981. "Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods". *Journal of Computational Physics*, Vol. 40, pp. 263–293.

Van Leer, B., 1982. "Flux-vector splitting for the euler equations". *Lecture Notes in Physics, Springer-Verlag*, Vol. 170, pp. 507–512.

9. RESPONSIBILITY NOTICE

The authors are solely responsible for the printed material included in this paper.