Investigation of WENO and several fluxes schemes for high-speed flows on overall unstructured meshes

Francisco Augusto Aparecido Gomes
UTFPR

Marcos Aurélio Ortega
ITA
WENO ("Weighted Essentially Non-Oscillatory") is the evolutionary product of a sequence of additive studies, which started with the seminal work of Godunov in the last years of the 50s of the last century. From the pure Riemann reconstruction of Godunov the high-resolution schemes passed by the MUSCL ("Monotonic Upstream-centered Scheme for Conservation Laws"), the ENO ("Essentially Non Oscillatory"), finally reaching the WENO stage. WENO schemes are high order accurate schemes designed for problems with piecewise smooth solutions containing discontinuities. The key idea lies at the approximation level, where a nonlinear adaptive procedure is used to automatically choose the locally smoothest stencil, hence avoiding crossing discontinuities in the interpolation procedure as much as possible. Following closely this evolutionary path, another very important feature is connected to approximating as accurately as possible the Riemann fluxes at the boundary of the tesselation cell. Many ideas for this approximation appeared along the way. The schemes of Lax-Friedrichs, Roe, Steger-Warming, AUSM, AUSMPW, HLL, HLLE, are all in use today. A high-resolution, high-order computer code based on the WENO algorithm, called Fweno, for the simulation of flows at high speeds and with eventually piece-wise smooth solutions, was developed by the present authors. Many of the most efficient flux calculation schemes were implemented, and, in this paper, a thorough comparison between Lax-Friedrichs, Roe, AUSM and AUSMPW is explored on strictly unstructured grids. The validation problems are all benchmarking test cases usually found in the literature: the internal supersonic ramp, the internal supersonic forward-facing step, and the supersonic flow about a circular cylinder. After validation the code was applied to the hypersonic flow about a pressure probe. In the near future cases of hypersonic flow of real gases about blunt bodies will be simulated and analysed.

**Palavras-chave:** WENO Scheme, Finite Volume Method, Convective Fluxes, High-Speed Flows, Unstructured Meshes.
INTRODUCTION

Numerical codes for the solution of hypersonic flows have to have good stability characteristics, especially when faced with discontinuities or regions of large gradients in the flow region. On the other hand, in order to obtain a high-order-of-accuracy scheme, one has to rely on an appropriate reconstruction of the primitive variables in each cell, on a precise solution of the Riemann problem at the faces, and on a careful treatment of the many variables at the boundaries of the computational domain (see Gooch et al., 2009). Among the most well known high-resolution schemes WENO seems to be one of the best choices due to its good reconstruction features (Abgrall, 1994, Sonar, 1997, Hu and Shu, 1999, Wang et al. 2009, Dumbser and Käser, 2007, Käser and Iske, 2005). The evaluation of the convective fluxes at the interfaces of control volumes deserves special attention so as to avoid anomalies in the solution such as the “carbuncle” phenomenon (Kitamura et al., 2010, Kim et al., 2001, Perry and Imlay, 1998).

Approximation to the Riemann problem is another fundamental issue directly responsible for the sparing of computer time. The family of schemes that conserve the total enthalpy is one of the best choices. Belonging to this family one finds the method of Hänel et al., 1987, the AUSM+ (Liou, 1996, Liou, 2000) and the AUSMPW+ (Kim et al., 2001), the flux difference methods of Roe (Toro, 2009), and schemes of the type HLLE+ (Tramel et al., 2009). In general, in the presence of discontinuities, those schemes are reliable and efficient in preserving stability (Kim et al., 2001, Lee et al., 2010, Kitamura et al., 2010). Amongst the variety of schemes AUSMPW calls attention for its distinctive collection of good qualities: efficiency and robustness, constancy of the total enthalpy, reduced numerical dissipation, high resolutions at discontinuities, elimination of overshoots, and no carbuncle phenomenon.

The present paper focuses specifically on unstructured grids. The numerical kernel is the WENO and the following flux schemes will be tested and compared: Lax-Friedrichs, Roe, AUSM+ and AUSMPW. It is a good check for AUSMPW which has been tested mostly on structured grids. A brief description of the governing equations of a calorically perfect gas is given in Section 2. Also in Section 2 we describe the main points of the WENO strategy. In Section 3 a comparative discussion of the many flux schemes are introduced. A discussion about the time stepping and boundary conditions enforcement is presented in Section 4. Results and discussions are presented in Section 5, where attention is called to the behaviour of AUSMPW for the case of unstructured grids. A conclusion is offered in Section 5.
MATHEMATICAL AND NUMERICAL MODELLING

Basic equations

We consider the flow of a Newtonian compressible fluid. The physical situation is mathematically modelled by the two-dimensional Navier-Stokes equations written here as:

\[
\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial (\mathbf{E}_c - \mathbf{E}_v)}{\partial x} + \frac{\partial (\mathbf{F}_c - \mathbf{F}_v)}{\partial y} = \mathbf{S},
\]

(1)

where \( \mathbf{Q} \) is the vector of conserved variables, \( \mathbf{E}_c \) and \( \mathbf{F}_c \) are the \( x \) and \( y \) convective flux vectors, and \( \mathbf{E}_v \) and \( \mathbf{F}_v \) are the \( x \) and \( y \) viscous flux vectors, respectively. \( \mathbf{S} \) stands for a possible geometric or physical source. In Cartesian coordinates these vectors read

\[
\mathbf{Q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \end{pmatrix}, \quad \mathbf{E}_c = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (\rho e_t + p)v \end{pmatrix}, \quad \mathbf{F}_c = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (\rho e_t + p)u \end{pmatrix}
\]

(2)

Because in this paper we shall only deal with non-viscous problems, there is no need to introduce the general algebraic expressions for the viscous fluxes. The symbol \( e_t \) in the equations above stands for the following sum \( [e + (1/2)u_j u_j] \), where \( e \) is the specific internal energy. For a thermally and calorically perfect gas (Liepmann and Roshko, 1956) the equation of state can be written as

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

(3)

with \( \gamma = 1.4 \) for air

Equation (1) is to be solved numerically. Space discretization is attained by the well known WENO scheme. Time advancement is performed by a TVD (“Total Variation Diminishing”) type Runge-Kutta scheme of third order of accuracy. The weighted essentially non-oscillatory scheme was first proposed by Liu et al. (1994) and, for the past two decades, has found many applications in science and engineering. WENO is a high-order-accurate discretization method especially fit for piecewise smooth solutions that might eventually contain discontinuities. Much of WENO schemes success is a consequence of the use of a dynamic set of stencils, where a nonlinear convex combination of lower order polynomials adapts either to a higher order approximation at smooth parts of the solution, or to an upwind spatial discretization that avoids interpolation across discontinuities and provides the necessary dissipation for shock capturing.
WENO - The numerical modelling

In general every high-order (larger than 1) method is prone to oscillations when faced with discontinuities in the flow. As Godunov’s theorem states (Toro, 2009) only the first order scheme is TVD guaranteed. With the introduction of MUSCL-type algorithms (second order of accuracy) the necessity for the control of oscillations came to the fore. The remedy came with the classical slope limiters. But slope limiters frequently identify regions near smooth extrema as requiring limiting, and this typically results in a reduction of the optimal high-order convergence rate.

The ENO (“Essentially Non-Oscillatory”) family of schemes introduced by Harten et al, (1987) addressed this issue by allowing oscillations up to the order of the truncation error. This action overcame the drawbacks and limitations of limiter-based schemes. ENO schemes avoid interpolation across high-gradient regions through biasing of the reconstruction. This biasing is achieved by reconstructing the solution on several stencils at each location and selecting the reconstruction which is in some sense the smoothest. This allows ENO schemes to retain higher-order accuracy near high gradient regions. However, the selection process can lead to convergence problems and loss of accuracy in regions with smooth solution variations. To counterbalance these problems, the so-called weighted ENO scheme (Liu et al., 1994), is designed to present better convergence rate for steady state problems, better smoothing for the flux vectors, and better accuracy using the same stencils than the ENO scheme. WENO scheme uses a suitably weighted combination of all reconstructions rather than just the one which is judged to be the smoothest. The weighting is designed to favor the smooth reconstruction in the sense that its weight is small, if the oscillation of a reconstructed polynomial is high and its weight is of order one, if a reconstructed polynomial has low oscillation.

In the sequel we shall list the main steps in the cell-centered finite volume reconstruction of a generic variable \( f \) by the WENO scheme. Firstly the domain of computation, \( \Omega \), has to be divided into a number of elements, or cells, \( \Omega_j \). We shall concentrate on triangulated subdivisions, and define a cell-centered scheme for which the punctual reference is considered to be the barycenter of each triangle. A “stencil” is an specific collection of elements.

The main steps of the WENO scheme are:

- Find a number of admissible stencils \( S_1, S_2, ..., S_m \) for a cell \( \Omega_j \) consisting of some specific neighboring cells, such that the cell itself \( \Omega_j \) belongs to each stencil. Note that a stencil can be thought of as a set of specific cells that can be used to obtain a polynomial reconstruction.
- Reconstruct the function \( f \) for each stencil \( S_j \) by a polynomial \( p_j \) based on the mean
values of the function $f$ on each cell in the stencil $S_j$.

- Compute an oscillation indicator $a_j$ for each reconstructed polynomial $p_j$.
- Calculate weights $w_j$ for each $p_j$ using the oscillation indicators such that the sum of all the $w_j$ is one.
- Find the reconstruction polynomial $p$ as the weighted sum of the $p_j$, $p = \sum_{j=1}^{m} w_j p_j$.

Obviously, the selection of stencils plays a vitally critical role to the success of the reconstruction. In principle, a simple geometrical condition guaranteeing that a stencil is admissible for a polynomial of degree greater than one is not yet at hand. Therefore one selects stencils that, most probably, are admissible. Later during the computation of the polynomials $p_j$ the inadmissibility is detected and those stencils are skipped.

The selection of stencils has to cope with some contradictory arguments. On the one hand, the stencils should have a small diameter and should be well centered with respect to $\Omega_j$ to obtain high accuracy and stability in smooth regions (see Fig. 1). The number of stencils should be small to reduce the computational cost. On the other hand, ENO methods are based on the idea that in case of non-smooth regions one-sided stencils are chosen to avoid interpolation across discontinuities. In consequence, the number of stencils should be large enough to avoid any oscillation in the solutions and keep the scheme stable. Figure 2 illustrates how each polynomial is constructed based on the immediate neighborhood of the “target cell”.

The target cell is that cell for which the primitive variables are reconstructed, remembering that integrations are approximated by means of Gauss quadratures along the edges.

**Figure 1.** Molecule of calculation which is used to form each polynomial basis (neighbourhood) for the WENO scheme.

![Figure 1](image1.png)

**Figure 2.** An example of a typical neighborhood for the second order WENO scheme.

![Figure 2](image2.png)

The reconstruction step is more involved, and in a way represents the core of the algorithm. Initially we define the mean of a function, $f(\bar{x}), \bar{f}_{\Omega_j}$, in a generic cell $\Omega_j$. 
\[ \vec{f}_{ij} = \frac{1}{|a_j|} \int_{\Omega_j} f(\vec{x}) \, d\vec{x}. \] (4)

The polynomials are “assembled” by using the following approach. Let us indicate the position of the barycenters by a vector \( \vec{b} \). Then we define a polynomial by the following expression,

\[ p(\vec{x}) = \sum_{|\alpha| \leq n} \vec{q}_\alpha (\vec{x} - \vec{b})^\alpha, \] (5)

where \( n \) is the order of the polynomial and \( \alpha \) is a multi-index such that \( \alpha = (\alpha_i, \alpha_j, \alpha_k), |\alpha| = \alpha_i + \alpha_j + \alpha_k \) and \( \vec{x}^\alpha = x_i^{\alpha_i} x_j^{\alpha_j} x_k^{\alpha_k} \). The \( \vec{q}_\alpha \in \mathbb{R}, |\alpha| \leq n \), is the unknown vector of polynomial coefficients.

To obtain the coefficients \( \vec{q}_\alpha \) one proceeds as follows. First let us state that the number of all \( \alpha \) with \( |\alpha| \leq n \) is indicated by \( N(n) \). For example, in two dimensions, we have \( N(0) = 1, N(1) = 3, N(2) = 6 \). To get to the coefficients we use an interpolation technique, and to this end we need a total of \( N(n) \) conditions. Let us suppose that the reference cell is indicated by \( \Omega_j \) and, to simplify, let us indicate the proper collection of additional \( N(n)-1 \) cells as \( \Omega_{j+1}, \Omega_{j+2}, \ldots, \Omega_{j+N(n)-1} \). The set of cells

\[ S = (\Omega_j, \Omega_{j+1}, \ldots, \Omega_{j+N(n)-1}) \] (6)

is one of the stencils referred above. Fig. 2 is representative of a second order scheme.

Writing down the interpolation relations one obtains

\[
\begin{align*}
\vec{p}(\vec{x})\Omega_j & = \vec{f}_{\Omega_j}, \\
\vec{p}(\vec{x})\Omega_{j+1} & = \vec{f}_{\Omega_{j+1}}, \\
\vec{p}(\vec{x})\Omega_{j+2} & = \vec{f}_{\Omega_{j+2}}, \\
\ldots & \\
\vec{p}(\vec{x})\Omega_{j+N(n)-1} & = \vec{f}_{\Omega_{j+N(n)-1}},
\end{align*}
\] (7)

and it should be observed that the number of equations is equal to \( N(n) \). By expanding the many \( \vec{p}(\vec{x})\Omega \), a linear system of algebraic equations for the unknown \( N(n) \) coefficients \( \vec{q}_\alpha, |\alpha| \leq n \), is established. This expansion reads
where the overbar indicates mean values as given by Eq. 4. In matrix form the system is given by

\[ M \tilde{q} = \bar{f}, \]

where the matrix M is obtained from

\[ M_{k, \alpha} = \frac{1}{\Omega_k} \int_{\Omega_k} (\bar{x} - \bar{\Omega}_j)^{\alpha} d\bar{x}, \quad j \leq k \leq (N(n) + j - 1), |\alpha| \leq n. \]  

As we have established before, in the above equations, \( \bar{q} \) is the vector of N(n) unknown coefficients \( q_{\alpha}, |\alpha| \leq n \), and \( \bar{f} \) is the vector of the given mean values \( \bar{f} = (\bar{f}_j, \bar{f}_{j+1}, \ldots, \bar{f}_{j+N-1}) \).

If the system has a unique solution, what means that the matrix M is invertible, then the stencil S is said to be admissible.

After being reconstructed, the polynomials have to be investigated in order to assess their “quality”, in other words, one has to determine how they behave in terms of eventual oscillations. To this end we have relied on the scheme of Hu and Shu (1999). An oscillator indicator for the polynomial \( p_j \) is defined as

\[ o_j = \left[ \sum_{1 \leq |\alpha| \leq n} \int_{\Omega_j} h^{(\alpha)-1} \left( \frac{\partial h_{p_i}}{\partial x_{\alpha}} \right)^2 d\Omega \right]^{1/2} \]

where \( \alpha\ ), \( \alpha \in (0,1,2,\ldots) \) and \( h = \text{mes} (\Omega_j) \) is the characteristic length of cell \( \Omega_j \).

The idea now is to reconstruct a new polynomial \( p = \sum_{i=1}^{m} w_i p_i \) by relying upon the calculated polynomials \( p_i \) and a collection of weights \( w_i \). The weights, one for each polynomial, are obtained by means of the following definition (Friedrich, 1998)

\[ w_i = \frac{(\epsilon + o_i p_i)^{-\beta}}{\sum_{i=1}^{m} (\epsilon + o_k p_k)^{-\beta}} \]

where \( \beta \) is a positive number.

Note that this type of limiting is fundamentally different from the one used in TVD schemes (like the MUSCL for example). Reconstruction schemes based on the WENO limiting
weight gradients, obtained from neighboring stencils, have the elegance of continuously eliminating these polynomials which cause oscillations.

# THE MANYFLUX SCHEMES

The idea of reconstructing the variables in the discretization cell by means of the solution of a Riemann problem (Godunov, 1959) has opened an avenue for the development of the class of the high-resolution methods. The fact that the scheme incorporates the Riemann problem in its DNA makes it especially appropriate for the solution of problems with eventual discontinuities in the domain of calculation. Notwithstanding this, the precise solution of a Riemann problem in each and every face of the domain tessellation renders the method almost impractical in terms of computer costs, especially when one thinks about a complex three-dimensional geometry. The proper assessment of this new difficulty was made possible by the appearance of the so called “approximate Riemann solvers”. There is an assortment of these approximate techniques, each one with its proper characteristics. We shall explore in this study four of the most well known, and most successful schemes in the literature.

## The Roe and Lax-Friedrichs schemes

To briefly introduce the approximateflux schemes let us imagine that the domain is tesselated into a set \( \{ J \} \) of triangular or rectangular cells, or elements. The mean-in-the-cell variables’ values are allocated at the barycenter of each cell (cell-centred scheme). For an element \( j \in \{ J \} \) the edges will be indicated by a countereg. The numerical flux, either precise or approximate, of the Riemann problem at an edge \( eg \), can be obtained by solving the following one-dimensional equation

\[
\frac{\partial Q}{\partial t} + \frac{\partial S_{n_{eg}}}{\partial n_{eg}} = 0. 
\]  

where \( S_{n_{eg}} = [E,F] \cdot n_{eg} \), and \( n_{eg} \) is an unit vector normal to edge \( eg \) directed from the element outwards. The initial condition for equation (13) corresponds to \( Q = Q_L \) and \( Q = Q_R \) separated by the \( eg \) interface.

Of all the approximate Riemann solvers the Roe scheme is one of the most well known. The method is based on the Roe-averaged state \( \hat{Q} \) that satisfies the following (Roe, 1981):

\[
\Delta S_n = \hat{A}_n \Delta Q,
\]  

where \( \Delta S_n = S_n(Q_R) - S_n(Q_L) \), \( \Delta Q = Q_R - Q_L \), and \( \hat{A}_n = \partial S_n / \partial Q \) evaluated at the Roe-averaged quantities,
where $h$ denotes the total enthalpy.

The flux at the interface is defined by

$$\mathbb{S}_{\text{Roe}} = \frac{1}{2} [S_n(Q_L) + S_n(Q_R)] - \frac{1}{2} \hat{R}_n |\hat{\Lambda}_n| \hat{\Lambda}_n \Delta Q,$$

where $\hat{R}_n$ is the matrix of right eigen-vectors of $\hat{A}_n$, $\hat{\Lambda}_n = \hat{R}_n^{-1}$, and $|\hat{\Lambda}_n|$ is the eigen-values diagonal matrix. The form of those matrices can be easily found in the literature — see, for example, Nishikawa and Kitamura (2008).

It is very well documented in the literature that the Roe flux admits non-physical expansion shocks; this can be avoided by introducing a small correction to the value of the eigen-values. On the other hand, the undesirable carbuncle phenomenon typically appears in calculation with this method.

Another type of Riemann solver uses the Lax-Friedrichs flux, also known as Rusanov flux; this is a one-wave approximation to the exact Riemann solver, and can be obtained from the Roe flux by replacing the eigen-value matrix $|\hat{\Lambda}_n|$ by a single scalar $\hat{s}_n$ defined as the spectral radius of $\hat{A}_n$,

$$\hat{s}_n = \rho(\hat{A}_n) = \max(|\hat{u}_n + \hat{c}|, |\hat{u}_n - \hat{c}|) = |\hat{u}_n| + \hat{c}. $$

In order to obtain the new flux, one sets

$$|\hat{\Lambda}_n| = \hat{s}_n I,$$

where $I$ is the identity matrix, and from Eq. (20) one obtains,

$$\mathbb{S}_{\text{LP}} = \frac{1}{2} [S_n(Q_L) + S_n(Q_R)] - \frac{1}{2} \hat{R}_n(\hat{s}_n I)\hat{\Lambda}_n \Delta Q = \frac{1}{2} [S_n(Q_L) + S_n(Q_R)] - \frac{1}{2} \hat{s}_n \Delta Q.$$
Because the coefficient of the dissipation-like term ($\delta_n$) is positive the method is stable. It is also robust and carbuncle-free.

The AUSM and AUSMPW scheme

In the last few decades much effort has been spent in the development of numerical schemes for hypersonic flows, which are, at the same time, robust and accurate. Hypersonic flow problems generally include severe viscous dissipation in a boundary layer and strong shock waves leading to large gradient of flow properties (it is important to bear in mind that the ultimate objective of the present authors is to use the code in the treatment of hypersonic flows). Initially, investigators tried to extend some of the supersonic algorithms, mostly of the upwind-biased type, to the hypersonic range. The flux vector splitting (FVS) approach of Steger and Warming, a very simple and well suited method for implicit time advancement, was one that was adapted. Unhappily, in spite of its very desirable efficiency and robustness, the FVS have shown poor accuracy due mainly to a large amount of numerical diffusion. Specifically, numerical diffusion is responsible for accuracy problems in resolving shear layer regions, which occur intensively in hypersonic flow. Flux-difference-splitting methods (FDS) such as the Roe scheme (Roe, 1981) was also adapted. In this case, where the method is well suited for explicit upwinding, results have shown good accuracy but at the expense of an increase in operation count and complexity in the linearization process. Much of the following works, therefore, were directed towards a blend of both FVS and FDS with the idea in mind of arriving at a robust, accurate and at the same time efficient formulation.

Liou and Steffen (1993) pioneered in this direction with the AUSM scheme — “Advection Upstream Splitting Method”. The core of the new approach was the proper definition of an “advection Mach number”, which is used for extrapolation of convective quantities. Besides being simple and robust AUSM is accurate enough to resolve a shear layer, what made it a good candidate for the blending of FVS and FDS as referred above. Notwithstanding those desirable qualities, it was verified that the characteristic of advection in AUSM may still induce oscillations in the solution. An improved version called AUSM+ was implemented by Liou (1996), and a novel definition of a proper sound speed at the interface between cells was introduced. After some research effort, it was found that these oscillations could be cured through the introduction of weighting functions based on pressure, what gave rise to AUSMPW — “AUSM by Pressure-based Weight Functions” (Kim and Rho, 1997, 1998). AUSMPW have shown to possess many desirable features such as no carbuncle phenomena, elimination of overshoots, accurate numerical dissipation, and preservation of the total enthalpy. In spite of the many improvements in those methods for fluxes calculation some unphysical expansion shocks were still observed with AUSM+ and AUSMPW. Kim et al. (2001) have therefore
suggested still another modification, which they called AUSMPW\(^+\), and arrived at a method that has shown to be superior in relation to all the predecessor ones in that it does not present the undesirable expansion shocks. The new method is derived by improving AUSMPW by the introduction of a new definition of the shock speed at the interfaces (following an approach similar to AUSM\(_{\alpha}\)). The many steps in the development of the schemes are algebraically rather involved. Therefore, initially, and for completeness, we simply furnish the final expressions of the fluxes in order to acquaint the reader to the schemes. In the following we will discuss some very important details related to the internal structure of those schemes. These details, some of them at the level of the implementation itself, are instrumental to the success of the code. The reader should be aware that these kind of information are rarely found in the literature. We have, for the two-dimensional AUSM\(_{\alpha}\) and AUSMPW, respectively,

\[
\tilde{S}_{\text{AUSM}^+} = \frac{1}{2} \left( [M_L]_\beta^+ + [M_R]_\beta^- \right) \Phi_3 + \left( [P_L]_\alpha^+ + [P_R]_\alpha^- \right) \vec{P}_R
\]  

where \( \beta = 1/8, \alpha = 3/16 \) (see Liou, 1986),

\[
\tilde{S}_{\text{AUSMPW}} = \frac{1}{2} \left( [M_L]_\alpha^+ + [M_R]_\alpha^- \right) \Phi_{L,R} + \left( [P_L]_\alpha^+ + [P_R]_\alpha^- \right) \vec{P}_R
\]

where, for AUSMPW, \( \alpha = 3/16 \). The “+” and “-” in the equations above are understood to be associated with the right and left-running waves that arrive at a certain edge.

**Some specifics of the AUSM and AUSMPW schemes**

Most of the time, the reader, in his pursuit for detailed data especially at the implementation level, gets frustrated, because literature information in this respect is very rare. Inspired by this necessity we shall discuss some of those specific points in relation to the development of the AUSM\(_{\alpha}\) and AUSMPW subroutines. To this end we start with Fig. 3.

**The AUSM\(_{\alpha}\) scheme**

For convenience we repeat here the expression for the numerical flux AUSM\(_{\alpha}\),

\[
\tilde{S}_{\text{AUSM}^+} = c_{1/2} \left( [M_L]_\beta^+ + [M_R]_\beta^- \right) \Phi_{L,R} + \left( [P_L]_\alpha^+ + [P_R]_\alpha^- \right) \vec{P}_R
\]

For two-dimensional problems the vectors in the last equation read,
The splittings of both Mach number and pressure for the calculation of the AUSM+ flux \( \mathbf{\mathcal{F}}_{\text{AUSM}^+} \), respectively,

\[
\Phi_{L,R} = \begin{pmatrix}
\rho_{L,R} \\
\rho_{L,R} u_{L,R} \\
\rho_{L,R} v_{L,R} \\
\rho_{L,R} h_{L,R}
\end{pmatrix},
\quad \mathbf{\mathcal{P}}_L = \begin{pmatrix}
0 \\
n_x p_L \\
0 \\
0
\end{pmatrix},
\quad \mathbf{\mathcal{P}}_R = \begin{pmatrix}
0 \\
0 \\
n_y p_R \\
0
\end{pmatrix}
\quad (27)
\]

where the “+” and “-” signs are indicative of waves coming from the left and the right relative to the edge in question — see Fig. 3. The Mach number on each side of the edge is given by the following

\[
M_{L,R} = \frac{U_{L,R}}{c_{\frac{1}{2}}}, 
\quad (30)
\]

where \( U_{L,R} \) stands for the component of the velocity vector which happens to be normal to the interface between cells.

\[
U_{L,R} = n_x u_{L,R} + n_y v_{L,R}, 
\quad (31)
\]

and

\[
c_{\frac{1}{2}} = \min(\tilde{c}_L, \tilde{c}_R)
\quad (32)
\]

such that
The critical sound speed, $C_{LR}^*$, can be obtained as follows

$$\hat{c}_{LR} = \frac{C_{LR}^*}{\text{max} \left( |U_{LR}|, c_{LR}^* \right)}$$

(33)

The critical sound speed, $C_{LR}^*$, can be obtained as follows

$$h_{t_{LR}} = c_{LR} + \frac{p_{LR}}{\rho_{LR}} + \frac{1}{2} \left( v_{LR}^2 + v_{LR}^2 \right)$$

(34)

where $h_{t_{LR}}$ is the total enthalpy at the cell interface,

$$h_{t_{LR}} = c_{LR} + \frac{p_{LR}}{\rho_{LR}} + \frac{1}{2} \left( v_{LR}^2 + v_{LR}^2 \right)$$

(35)

where $e$ stands for the specific internal energy.

The AUSMPW scheme

Rewriting the AUSMPW flux,

$$\tilde{S}_{\text{AMPW}} = c_{1/2} \left( \tilde{M}_L^+ + \tilde{M}_R^- \right) \tilde{\Phi}_{LR} + \left( [P_L^+]_a \tilde{P}_L + [P_R^-]_a \tilde{P}_R \right)$$

(36)

where, more specifically,

2. For $\frac{m_1}{2} < 0$:

$$\tilde{M}_L^+ = \tilde{M}_L^+ |_{\beta} + \tilde{M}_L^+ |_{\beta} - \omega \tilde{M}_R^- |_{\beta} (1 + f_R) + (f_L \tilde{M}_L^+ |_{\beta} + f_L \tilde{M}_L^+ |_{\beta})$$

(37)

$$\tilde{M}_R^+ = \omega \tilde{M}_R^- |_{\beta} (1 + f_R)$$

(38)

and

2. For $\frac{m_1}{2} < 0$:

$$\tilde{M}_L^+ = \omega \tilde{M}_L^- |_{\beta} (1 + f_L)$$

(39)

$$\tilde{M}_R^- = \tilde{M}_L^+ |_{\beta} + \tilde{M}_R^- |_{\beta} - \omega \tilde{M}_L^- |_{\beta} (1 + f_L) + (f_L \tilde{M}_L^+ |_{\beta} + f_L \tilde{M}_L^+ |_{\beta})$$

(40)

where

$$m_\frac{1}{2} = \tilde{M}_L^+ + \tilde{M}_R^-$$

(41)

$$\omega(p_L,p_R) = 1 - \min \left( \frac{p_L}{p_R}, \frac{p_R}{p_L} \right)^3$$

(42)

The shock instability, which is the main drive behind the carbuncle phenomenon appearance and development, can be controlled introducing a limiting mechanism. This limiter, in a way, introduces a balance between two confronting actions, namely, the numerical dissipation.
and the split of pressure introduced across the cell interface. The limiter can be written as a function of the tangential component of the velocity at the cell interface as follows,

\[
    f_{L,R} = \left( \left( \frac{p_{L,R}}{p_s} \right) - 1 \right) p^l(p_{L,R}, p_{R,L}) |M_{L,R}| \begin{cases} \min \left( 1, \left( \frac{V_{L,R}}{c_s} \right)^{0.25} \right), & |M_{L,R}| < 1, \\ 0, & |M_{L,R}| > 1, \end{cases}
\]

where

\[
    p_s = p^+_{L|\alpha=\frac{\pi}{4}} p_L + p^-_{R|\alpha=\frac{\pi}{4}} p_R
\]

and

\[
    p^l(x, y) = \begin{cases} 4 \min \left( \frac{x}{y}, \frac{y}{x} \right) - 3, & 0 \leq \min \left( \frac{x}{y}, \frac{y}{x} \right) < 1, \\ 0, & 0 \leq \min \left( \frac{x}{y}, \frac{y}{x} \right) \leq 0.25, \end{cases}
\]

The splittings of the Mach number and pressure are the same as those for the AUSM\textsuperscript{+} scheme — Eq. (28) and (29). The Mach number on each side of the cell interface is also defined as in Eq. (30). As for the tangential velocity at the interface one has

\[
    V_{L,R} = -n_y u_{L,R} + n_x v_{L,R}.
\]

**The “Edge-Based” scheme**

The reader should be aware that at the implementation level one relies on a strategy known in the literature as the “edge-based scheme”, which was introduced in its basics by D. J. Mavriplis in a seminal work of 1988. Why “edge-based”? Most probably, and the user should bear that aspect in his mind, the terminology comes because most of the action of the calculation is happening at the edges of the elements. And this is so, because, in order to reach the balance in the element, as imposed by the system of equations that is being solved, the prediction of integrals at the faces (edges in two-dimensional geometries) represents an extensive part of the whole effort.

First it is important to call attention to a detail relative to the initial data structure (basically the connectivity tables). The unstructured grid, as delivered by the ANSYS\textsuperscript{®} package, is organized in such way that, in order to account for all the nodes, the triangles are numbered and the nodes, for each triangle, are also numbered, and this numbering of nodes is done by moving along the edges of the target triangle according to a counterclockwise direction. One could call this data bank as the “master connectivity table”.

Now, let us focus our attention on one particular edge — for example, edgeegin Fig. 3. In this instance, and after reading completely the master connectivity table, each edge
is going to be read two times. When passing by triangle L the edge eg will be annotated as formed by nodes nd1 and nd2, and in this order, nd1–nd2. When passing by triangle R the edge eg will be annotated by nodes nd2 and nd1 and in this order nd2–nd1. What happens is that now the interest relies completely on the edge. Therefore, another connectivity table has to be generated and the leading entry now is the edge. How does this table is organized?

The edge is evidently numbered by nd1 and nd2, the order read when the edge eg was located by the first time, but only in this order. The other possibility nd2 to nd1 that came with triangle R is abandoned. On the other hand, the triangle L, defined as located to the left of edge eg(edge nd1–nd2) is annotated, in the new table, as connected to this specific edge. The triangle R that was located in the first overall search as having the edge nd2–nd1 is also annotated and connected to the edge eg, but defined as situated to the right of edge.

In short, the new table, having as its main entry the edge eg, known only by the two nodes nd1 and nd2, in the order nd1–nd2, also informs what are the left, L, and right, R, triangles, relative to edge eg. As an important result, there is never any doubt about the direction of the vector \( \hat{n} \) which is always directed from L to R.

The Fortran sequence that predicts the fluxes components reads

```fortran
  do i=1,nfaces
    !
    ! Assignment of each face.
    !
    if1=face(i,1) !Cell to the left
    if2=face(i,2) !Cell to the right
    !
    ! Calculation of the flux on each face
    !
    F = compute.flux.on.face(i)
    Flux(:,if1) = Flux(:,if1) + F * S(i)
    Flux(:,if2) = Flux(:,if2) - F * S(i)
  end do
```

The array “face(i,1:2)” corresponds to the edge connectivity table which should contain, for all the individual edges of the grid, the following data:

“For each edge of the mesh store: L, R, nd1, nd2.”

The value of F as returned by the function “compute.flux.on.face(i)” represents the numerical prediction of the total flux at the edge, that is, \( (F = (\vec{F}_c - \vec{F}_v) \cdot \hat{n}S) \), where \( \vec{F}_c = n_x F_c + n_y F_c \).
is the convective (non-viscous) parcel, while \( \tilde{F}_v = n_x F_x + n_y F_y \) is the viscous parcel. \( S \) is the face area and \( \tilde{n} = n_x \hat{i} + n_y \hat{j} \) is the normal vector to the face directed outwards.

The non-viscous flux, \( \tilde{F}_v \), can be obtained by means of any of the flux strategies that were discussed above, that is, Lax-Friedrichs, Roe, AUSM*, or AUSMPW. For the calculation of the viscous parcel, \( \tilde{F}_v \), one needs the variables’ gradients, which, in our case, were obtained by a variant of the Green-Gauss method, and not as usual by relying upon the ENO polynomials. These gradients represent the primitive variables variation at each Gauss quadrature point on each face.

## TIME STEPPING AND BOUNDARY CONDITIONS

The TVD Runge-Kutta time stepping

In order to advance the reconstructed cell averages in time a third-order TVD Runge-Kutta scheme is applied. Dropping, for simplicity, any subscript indicative of spatial position, one may write,

\[
\begin{align*}
Q^{n+\frac{1}{2}} &= Q^n + \Delta t R(Q^n), \\
\tilde{Q}^{n+\frac{1}{2}} &= \frac{3}{4} Q^n + \frac{1}{4} \tilde{Q}^{n+\frac{1}{2}} + \frac{1}{4} \Delta t R\left(\tilde{Q}^{n+\frac{1}{2}}\right) \\
\tilde{Q}^{n+1} &= \frac{1}{3} Q^n + \frac{2}{3} \tilde{Q}^{n+\frac{1}{2}} + \frac{2}{3} \Delta t R\left(\tilde{Q}^{n+\frac{1}{2}}\right)
\end{align*}
\]  

(47)

where \( R = d\tilde{Q}/dt \) represents the residual of Eq. (1) after spatial discretization, and is equal to the sum of the fluxes crossing all the edges that delimit cell \( \Omega_j \). A suggested time step is the one that obeys the following condition

\[
\Delta t = C \min_j (\Delta t_j), \quad \Delta t_j = \frac{h_j}{V_j}
\]  

(48)

where \( C \leq 1/2 \) is the CFL number, \( h_j \) is the characteristic length of the cell and \( V_j \) is a characteristic velocity (Titarev et al. 2010, Gottlieb and Shu, 1998).

**Boundary conditions**

Proper specification of boundary conditions is absolutely fundamental for the success of a physical problem numerical simulation. These conditions must be correctly specified in accordance with speed regimes at inlet and outlet sections, viscous or nonviscous interactions on solid walls, one-dimensional or multidimensional geometries, symmetric surfaces, reflecting and non-reflecting boundaries, and farfield interactions. The basic strategy employed here
is the creation of ghost volumes along the boundaries of domain. The values attributed at
the centroid of the ghost volumes are such as to fit the proper conditions at the boundaries.

At inflow and outflow sections the number of boundary conditions is determined ac-
cording to the eigenvalue spectrum of the Jacobian matrices in terms of the primitive vari-
ables. It should be noted that the commanding direction is the normal to the face. At solid walls
slip or nonslip conditions are enforced according to the mathematical model that is being
used, Euler or Navier-Stokes, respectfully. At solid walls the pressure is extrapolated from
inside according to boundary-layer directives, especially the fact that, for large wall curvature,
the potential pressure is imposed upon the wall. The temperature at the wall is established
according to an a priori specification, what corresponds to a Dirichlet approach, or to an ex-
trapolation in case the heat flux is specified (Neumann approach). Farfield boundaries are
tackled by means of the Riemann invariants.

In the case of a nonviscous flow, since there is no friction force, the velocity vector is
tangent to the wall surface. This is equivalent to the condition that there is no flow normal to
the surface, i.e., the wall is impermeable and then, one has \( \nabla \cdot \mathbf{n}_{wx} = 0 \). This translates into the
following relations between components at the ghost and at the adjacent-to-the-wall element,

\[
\begin{align*}
v_g &= (1 - 2n^2_{wx})u_j - 2n_{wx}n_{wy}v_j, \\
v_g &= -2n_{wx}n_{wy}u_j + (1 - 2n^2_{wx})u_j,
\end{align*}
\]  

(49)

where subscript “g” is indicative of the ghost cell, subscript “j” indicates the interior po-
sition, and \( n_{wx} \) and \( n_{wy} \) are the components of the unit vector normal the wall. If the physical
modelling corresponds to a viscous fluid, the wall velocity condition is even simpler,

\[
\begin{align*}
u_g &= -u_j \\
v_g &= -v_j,
\end{align*}
\]  

(50)

and the fluid will get in rest relative to the wall.

For external flows, and for obvious reasons, the domain has to be limited in size. In this
case two important points have to be observed. Firstly, the truncation of the domain should
have no notable effects on the flow solution as compared to the infinite domain. Secondly,
any outgoing disturbances must not be reflected back into the flow field. For external flows
and in general terms, one normally makes use of the Riemann invariants (Toro, 2009). With
this approach, the two sources of errors as identified above is normally avoided. Applying
the notion of characteristics to a one-dimensional scenario defined by the normal direction
to an external surface there results three Riemann invariants,
\[
\begin{align*}
RI_1 &= \hat{V} \cdot \hat{n} - \frac{2c}{(\gamma - 1)} \\
RI_2 &= \hat{V} \cdot \hat{n} + \frac{2c}{(\gamma - 1)} \\
RI_3 &= \frac{p}{\rho^2}
\end{align*}
\] (51)

and the basic technique is such that: (i) supersonic inlet: the three invariants are given from outside; (ii) supersonic outlet: the three invariants from inside; (iii) subsonic inlet: \(RI_1\) and \(RI_3\) from outside and \(RI_2\) from inside; (iv) subsonic outlet: \(RI_2\) and \(RI_3\) from inside and \(RI_1\) from outside.

## RESULTS AND DISCUSSION

### The internal supersonic ramp: Scramjet flame holder problem

This a classical benchmark problem for high-speed codes, much cited in the literature — see, for example, Ripley et al. (2004), Chung (2010). An internal supersonic flow approaches a two-dimensional ramp deflected against the oncoming flow according to an angle of 10°. After the compression corner and still along the lower wall a 10° expansion deflection follows. The result is an oblique shock wave that gets anchored at the first corner and that provides for the proper initial direction change. Because the flow is contained inside a parallel-walled duct a series of shock and expansion reflections follows downstream until the exit section is reached.

The entrance density and pressure are respectively equal to 1.2215\(Kg\; l/m^3\) and 101,314.08\(N\;/\;m^2\) and two values for the entrance Mach number were tested, 2 and 3. It is important to keep in mind that the exit Mach number is supersonic along all the section height. The grid is unstructured with 16,118 elements. Fig. 4 shows a detail of the grid at the ramp region. For comparison purposes the total length of the duct was made equal to five inlet heights just like in the work of Ripley et al. The comparison between our figures 5 and 6 and figure 11 of Ripley et al. shows an outstanding coincidence of values all over the field, what demonstrates the accuracy of the present version of code Fweno. The only slight difference can be detected at the position where the reflected shock crosses the exit section. Most probably this is due to the grid resolution. The reader can observe that our grid is much coarser; figure 11 of Ripley et al. corresponds to an adapted grid (shown in their figure 11 (c)) with 50,837 cells.

Table 1 shows values of the Mach number in regions A, B, C, and D — see Fig. 7. The table presents analytical as well as numerical values from Ripley et al. (2004). The reader
can observe that our results compare quite well with the references even in this case where
the grid is rather coarse.

**Figure 4.** Detail of the mesh for the supersonic ramp test case. The total number of elements is 16,118.

![Figure 4. Detail of the mesh for the supersonic ramp test case. The total number of elements is 16,118.](image)

**Table 1.** Values of the Mach number for selected regions. “An. Sol.” indicates analytical solutions, and the columns “Roe”, “AUSM+” and “AUSMPW” refer to the present study.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2.00</td>
<td>2.000</td>
<td>1.995</td>
<td>1.999</td>
<td>1,998</td>
</tr>
<tr>
<td>B</td>
<td>1.65</td>
<td>1.639</td>
<td>1.640</td>
<td>1.637</td>
<td>1.639</td>
</tr>
<tr>
<td>C</td>
<td>1.30</td>
<td>1.287</td>
<td>1.290</td>
<td>1.291</td>
<td>1.292</td>
</tr>
<tr>
<td>D</td>
<td>2.00</td>
<td>1.985</td>
<td>1.995</td>
<td>1.970</td>
<td>1,998</td>
</tr>
</tbody>
</table>

The ramp problem was also solved taking into account an entrance Mach number equal
to 3. Fig. 8 shows the solution corresponding to Mach number contours. The resulting physics
corroborates the numerical results considering that the shock system is washed downstream
as expected because the initial deflecting angle is the same 10° and the initial Mach number
has increased from 2 to 3.
Figure 5. Pressure field as calculated by the AUSM+ scheme. Entrance Mach number equal to 2.

Figure 6. Pressure field as calculated by the AUSMPW scheme. Entrance Mach number equal to 2.

The internal forward-facing step: A Mach 3 wind tunnel with a step

This is also a very well known benchmark case, with a much more stringent condition than the last problem. The reason is evidently the abruptness of the forward-facing step. A uniform supersonic stream with M=3 enters a two-dimensional duct. The lower wall of the duct is turned abruptly by a forward-facing step — see Fig. 9. This figure shows details of the mesh at the corner region; the grid has a total of 9224 elements. The meshes for this investigation — for all cases — were generated using the ANSYS® package. The reader can observe in Fig. 9 that the mesh is very homogeneous and uniform, especially at the corner neighborhood, a very desirable geometric scenario, considering that the corner is in fact a geometrical discontinuity in the flow field.
Figure 7. Iso-Mach-lines as calculated by the AUSM+ scheme. Entrance Mach number equal to 2.

Figure 8. Mach number field for an entrance Mach number equal to 3. The flux scheme is the AUSMPW.

This problem has been well explored by Woodward and Colella (1984) with the basic aim of discussing advantages and disadvantages of specific algorithms for the prediction of compressible flows with discontinuities. The problem corresponds to a supersonic stream running in a duct with a step at the lower wall. The entrance height of the passage is indicated by h. The length is given by 3h and the step is located at a position such that the distance to the entrance section is equal to 0.6h. The height of the step is 0.2h. The geometry is considered to be two-dimensional, what means that the width of the tunnel is considered to be infinite. A stream of fluid supposed to be a thermal and calorically perfect gas with $\gamma = 1.4$ is constantly fed through the entrance section with density and pressure equal, respectfully, to 1.2215 Kg/m$^3$ and 101,314.08 N/m$^2$. The entrance Mach number is kept constant and equal to 3. Along the walls of the duct reflecting boundary conditions are applied, and at the exit section the flow conditions are established taking into account that the flow is supersonic.
Initially we have confronted the results obtained with the schemes of Lax-Friedrichs and Roe, both after 2 non-dimensional time units. Figures 10 and 11 show maps of Mach number isolines. When compared to the work of Woodward and Collela (1984), Fig. 3, page 131, one can observe the exactness of our simulation. And the reader has to take into account that our grid is much coarser — $\eta = 1/40$ against $\eta = 1/80$, where $\eta$ is the grid characteristic length. Apparently the reflected shock in the lower wall has much more resolution for the Roe scheme when compared to the Lax-Friedrichs approach. This is the effect of the numerical dissipation. It is well known that the flux of Lax-Friedrichs is much more dissipative when other conditions are kept basically the same.

**Figure 9.** Grid for the forward-facing step problem. Details at the corner region.

![Figure 9](image)

**Figure 10.** Map of Mach number isolines. Lax-Friedrichs scheme. $t = 2$.

![Figure 10](image)

Fig. 12 shows the result of our calculation after a time stretch of 4 non-dimensional time units.
The flux scheme used here is the one due to Roe — Eq. (20). The accuracy of the simulation when compared to the results of Woodward and Colella is outstanding. For example, the points where the shocks reflect on the lower and upper wall and the position at the exit section where the shock leaves the duct are basically coincident. The upper shock stem reaches the wall basically at a right angle as it should, and the slipstream — the contact surface for this case —, after the shock bifurcation point, is clearly resolved. Another point to be called attention to is the smooth solution around the upper corner of the step. This point is a geometrical singularity of the flow and, consequently, one has to be careful when dealing with the simulation in its vicinity.

Supersonic flow with $M_\infty = 6$ about a circular cylinder

The grid for this problem is shown on Fig. 13. The total number of elements is 2816. The supersonic flow about a blunt body like the circular cylinder is an appropriate scenario for the testing of the carbuncle phenomenon, which customarily appears when one uses the flux scheme of Roe (Liou, 1996). The solutions of this problem using the schemes of Roe, AUSM+, and AUSMPW are shown by figures 14, 15, 16.
Fig. 14 depicts the isolines for the Mach number field from the solution of the problem by the Roe scheme. It is clear from the data in the figure that the lines at the shock oscillate markedly, especially in front of the stagnation point. These oscillations grow with time and develop into the carbuncle phenomenon. On the other hand, for the AUSM+ and AUSMPW schemes the phenomenon does not appear. The AUSMPW solution improves upon the AUSM+ in that the shock is more crisply defined, in spite that, in overall terms, the two solutions agree quite well.

Figure 13. Grid for the circular cylinder problem. Total number of elements equal to 2816.

Figure 14. Mach lines for the flow about the cylinder with $M_\infty = 6$. Flux scheme of Roe.

Hypersonic flow with $M_\infty = 10$ about a Pitot probe

The idea for this case study is related to the normal necessity of installing probes inside hypersonic tunnels. The approaching Mach number is equal to 10 and the flow about a blunt body at this level of
speed will be much more demanding for the code in terms of accuracy, robustness and efficiency. The code Fweno behaved quite well and the results are really quite promising. The many numerical tools that were implemented, namely, the edge flux calculations, the mounting of the many polynomials inside the Weno, the Runge-Kutta TVD time stepping approach, all have worked quite well even in this case where the grid is everywhere unstructured.

The grid is shown in Fig. 17. The reader can observe that for this problem we have dealt with a local refined grid. The action of local clustering was done in a posteriori fashion. That is, the problem was solved with a coarse grid, and afterwards, in a post-processing action, the local clustering was enforced taking as reference the shock contour. Refining was also applied between the shock and the body considering the importance of the physics in this region of the flow.

Figures 18, 19, 20 show the results in terms of density fields. These cases were run until $t=5.5$ units of non-dimensional time. The results given by the AUSM+ and AUSMPW are basic coincident in this instance. There is a slight difference in terms of the shock resolution: AUSMPW gives a slightly more crispy shock contour when compared with AUSM+.
Figure 17. Grid for the flow about the probe with $M_\infty = 10$. 

Figure 18. Density isolines for the flow about the probe with $M_\infty = 10$. Roe’s flux scheme. 

Figure 19. Density isolines for the flow about the probe with $M_\infty = 10$. AUSM+ flux scheme.
As already mentioned, the reader can observe that the map of density isolines for these two flux approaches are basically coincident. This is not the case with the result coming from the Roe flux scheme. The overall map of density isolines does not coincide properly with the other two cases, and, especially at the front region of the body, the differences are great. What happens is that the carbuncle phenomenon is starting to appear. In order to stress this point we present a zoom of this region in Fig. 21. It is clear that the isolines are starting to bulge in the region of the symmetry axis. With the passing of time this will grow along and form the characteristic carbuncle protuberance at the shock contour in front of the body.

**CONCLUSIONS**

The application of the WENO strategy, together with four fluxes calculation strategies, LF, Roe, AUSM+, and AUSMPW, with the ultimate aim of numerical integration of the Euler equations, to an essentially unstructured grid was demonstrated. It is important to keep in mind that the fluxes schemes AUSM+ and AUSMPW have been used, most of the time, in conjunction with structured meshes. The code Fweno, recently developed by the present
authors, has shown, even in those very demanding conditions, the usual outstanding qualities of robustness, efficiency and accuracy. The so called edge-based strategy was discussed in detail, and this, much probably, will be of much help to those that are endeavouring to work in this area. Several benchmarking cases were tackled together with the calculation of a hypersonic Pitot probe. In all cases the solutions that came out are physically sounding and accurate. In a next step in the very near future this code will be extended with the basic objective of simulating hypersonic flows for Mach numbers in the range 20-25. That is, we will deal with re-entry aerodynamics and all the chemistry involved in the simulation of the flow around the returning vehicle.

**REFERÊNCIAS**


