V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

NEW LIMITER AND GRADIENT RECONSTRUCTION METHOD FOR HLLC-FINITE VOLUME SCHEME TO SOLVE NAVIER-STOKES EQUATIONS

L. Remaki, O. Hassan and K. Morgan

School of Engineering,Swansea University,
Singleton Park SA2 8PP, Swansea, Wales, UK.
e-mails: l.remaki@swansea.ac.uk,O.hassan@swansea.ac.uk,K.morgan@swansea.ac.uk

Key words: Fluid Dynamics, Finite Volume, HLLC, limiters, Gradient reconstruction

Abstract. *This paper presents new developments within the numerical resolution of Navier-Stokes equations using HLLC-Finite volume method. The main contributions include new limiter design based on a stability analysis and an accurate gradient reconstruction. Viscous external flow tests are used to demonstrate the efficiency of the method.*

1 INTRODUCTION

Unstructured mesh methods are now widely employed for the simulation of general industrial aerodynamic flow problems. This is mainly due to the ease and rapidity with which complex geometrical domains may be meshed, using automatic mesh generation procedures. An additional attractive feature of the approach is that it allows naturally for the incorporation of adaptive mesh techniques, though this has not been fully exploited to date. A variety of solution algorithms have been proposed for aerodynamic flow simulations on unstructured meshes, employing both finite volume methods and continuous or discontinuous finite element methods [1, 2, 3, 4, 5, 6, 7]. The major difficulty that is often encountered with these methods is the maintenance of an acceptable level of accuracy while, simultaneously, ensuring robustness and stability over a range of flow speeds.

In this paper, we concentrate upon the use of the finite volume approach. Several variants of this method have been developed and applied to the solution of the Navier–Stokes equations, with the essential difference between the methods generally being the manner in which the contributions of the inviscid fluxes are computed. For centered schemes, these fluxes are taken simply as the average of adjacent fluxes and, in this case, an artificial viscosity operator is required to ensure stability. The alternative approach is to evaluate the fluxes using an approximate solution to a Riemann problem, with the implicit upwinding that is involved resulting in stability without the requirement for the explicit addition of artificial viscosity. Here, an upwind approach is incorporated within our standard cell vertex finite volume based system for the simulation of compressible high speed flows [8, 9]. The method chosen involves the use of the HLLC Riemann solver to evaluate the contributions of the inviscid. First a stability and conservation analysis is developed, based on, a new robust and suitable limiter is designed. As part of this algorithm, a new accurate method for constructing the solution gradient is achieved. A number of 3D Navier-Stokes and Euler simulations are included to demonstrate the convergence and stability of the method, over a range of flow speeds from subsonic to supersonic, without the requirement of any tuning of user defined parameters.

2 Governing Equations

The equations governing three dimensional unsteady viscous compressible flow are expressed, relative to a Cartesian (x_1, x_2, x_3) coordinate system, over a fixed volume $\mathscr{V}$ with a closed surface $\mathscr{S}$, in the integral form

$$\frac{d}{dt}\int_{\mathscr{V}} \vec{Q}\, d\mathscr{V} = \int_{\mathscr{S}} \vec{F}^{\alpha}(\vec{Q})\, n^{\alpha}\, d\mathscr{S} - \int_{\mathscr{S}} \vec{G}^{\alpha}(\vec{Q})\, n^{\alpha}\, d\mathscr{S} \qquad \alpha = 1, 2, 3 \tag{1}$$

where the summation convention is employed and $\vec{n} = (n^1, n^2, n^3)$ denotes the unit outward normal vector to $\mathscr{S}$. In this equation,

$$\vec{Q} = \begin{pmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_2 \\ E \end{pmatrix} \qquad \vec{F}^{\alpha} = \begin{pmatrix} \rho u_\alpha \\ \rho u_1 u_\alpha + p\delta_{\alpha_1} \\ \rho u_2 u_\alpha + p\delta_{\alpha_2} \\ \rho u_2 u_\alpha + p\delta_{\alpha_3} \\ (E+p)u_\alpha \end{pmatrix} \qquad \vec{G}^{\alpha} = \begin{pmatrix} 0 \\ \tau_{1\alpha} \\ \tau_{2\alpha} \\ \tau_{3\alpha} \\ u_\beta \tau_{\beta\alpha} - q_\alpha \end{pmatrix}$$

where ρ, p and E denote the density, pressure and total energy of the fluid respectively, u_α is the velocity of the fluid in direction x_α, t is the time and $\delta_{\alpha,\beta}$ is the Kronecker delta. The averaged deviatoric stress tensor is defined by

$$\tau_{\beta\alpha} = -\frac{2}{3}\mu \frac{\partial u_k}{\partial x_k}\delta_{\beta\alpha} + \mu\left(\frac{\partial u_\beta}{\partial x_\alpha} + \frac{\partial u_\alpha}{\partial x_\beta}\right) \tag{2}$$

and the averaged heat flux is

$$q_\alpha = -k\frac{\partial T}{\partial x_\beta} \tag{3}$$

Here, μ denotes the sum of the laminar and turbulent viscosities, k is the sum of the laminar and turbulent thermal conductivities and T is the averaged absolute temperature.

The equation set is completed by the addition of the perfect gas equation of state in the form

$$p = (\gamma - 1)(E - 0.5\rho u_\alpha u_\alpha) \tag{4}$$

where γ is the ratio of the specific heats. Steady state solutions of this equation set are sought in a fixed spatial computational domain Ω.

3 Finite Volume Method

The domain Ω is discretised into a mesh of tetrahedral cells, using a Delaunay mesh generation process with automatic point creation [8, 9]. To enable the implementation of a cell vertex finite volume solution approach, a median dual mesh is constructed by connecting edge midpoints, element centroids and face centroids such that only one node is present in each control volume [8, 9]. Each edge of the grid is associated with a segment of the dual mesh interface between the nodes connected to the edge. The dual mesh interface inside the computational domain surrounding node I is denoted Γ_I. The lines which define the control volume interface surrounding node I are denoted byΓ_I^k. The segment of the dual associated with an edge is a surface. This surface is defined using triangular facets, where each facet is connected to the midpoint of the edge, a neighboring element centroid and the centroid of an element face connected to the edge. This is illustrated in Figure 1. The midpoint of the edge between node I and J is termed x_m^{IJ}, the centroid of the face with vertices I, J and K is named and the element centroid is designated by x_c. The bold lines on the dual mesh in the figure illustrate the boundaries between the edges of which the dual mesh segment is associated. With this dual mesh

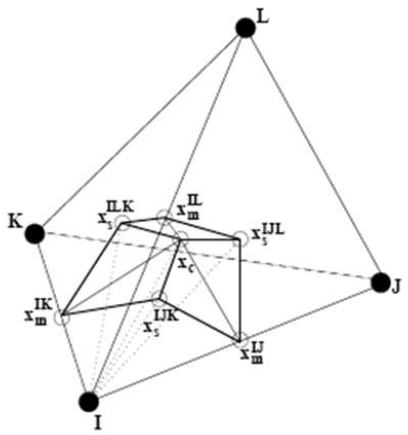

Figure 1: Illustration of that part of the dual mesh surrounding node I that is contained within a tetrahedral cell.

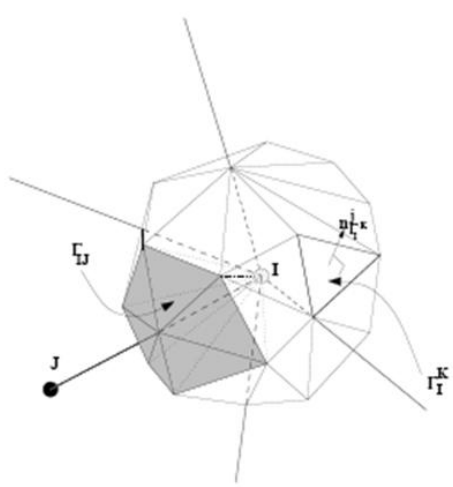

Figure 2: Illustration of the dual mesh surrounding an internal node I.

definition, the control volume can be thought of as constructed by a set of tetrahedra with base on the dual mesh. A complete dual mesh cell around an internal node I is shown in Figure 2.

Equation (1) is applied to each cell Ω_I of the dual mesh in turn. To perform the numerical integration of the inviscid fluxes over the surface $\partial\Omega_I$ of this cell, a set of coefficients is calculated for each edge using the dual mesh segment associated with the edge. The values of these coefficients for an internal edge are evaluated as

$$n^{\alpha}_{IJ} = \sum_{K \in \Gamma_{IJ}} A_{\Gamma^K_I} n^{\alpha}_{\Gamma^K_I} \tag{5}$$

where $A_{\Gamma^K_I}$ is the area of facet Γ^K_I and $n^{\alpha}_{\Gamma^K_I}$ is the component, in direction x_α, of the outward unit normal vector of the facet from the viewpoint of node I. The integral of the inviscid flux over the surface $\partial\Omega_I$ is then approximated as

$$\int_{\partial\Omega_I} \vec{F}^{\alpha} n^{\alpha} dS \approx \sum_{J \in \Lambda_I} \tilde{\vec{F}}_{IJ} \tag{6}$$

$$\tilde{\vec{F}} = \begin{pmatrix} \rho q_{IJ} \\ \rho u_1 q_{IJ} + p n^1_{IJ} \\ \rho u_2 q_{IJ} + p n^2_{IJ} \\ \rho u_3 q_{IJ} + p n^3_{IJ} \\ (E+p) q_{IJ} \end{pmatrix}$$

is a consistent numerical flux function. The solution is advanced in time to steady state using an explicit multi-stage Runge Kutta procedure and the convergence is accelerated by the use of local time stepping and by the addition of an agglomerated multigrid process.

4 HLLC Flux Function

The numerical flux function is computed using the HLLC Riemann solver [10, 11], which is a modication of the original HLL scheme [11]. The central idea is to assume a wave configuration for the solution that consists of three waves separating four constant states, as illustrated in Figure 3. The solution of the Riemann problem in this case consists of a contact wave and two acoustic waves, which may be either shocks or expansion fans. The solver is based on Godunov's method, where the approximate solutions are constructed by averaging intermediate states in the exact solution, respecting certain principles, such as exactly resolving isolated shocks and contact discontinuities. The HLLC solver employed is based on an exact resolution of a Riemann problem, while averaging the wave speeds S^L , S^M and S^R in an appropriate manner. In this paper the acoustic waves approximation proposed in [11] are modified to improve the transition from subsonic to supersonic speeds. Suppose $\vec{Q}_I$that is the numerical solution at

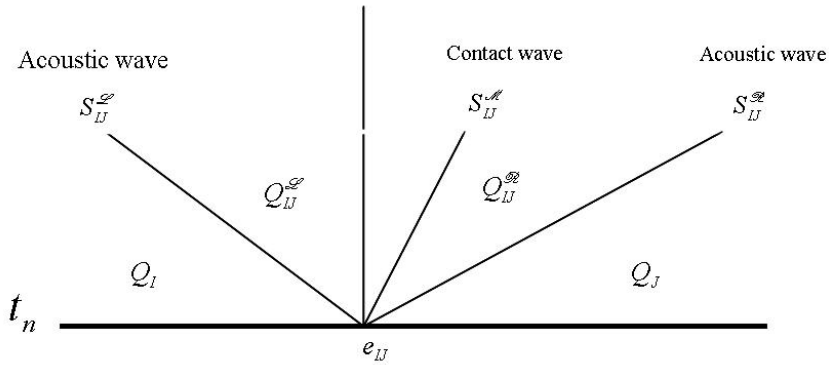

Figure 3: Schematic illustration of the HLLC Riemann solver.

node I. The entries in the vector $\vec{Q}_{IJ}^{\mathscr{L}}$ are then evaluated as

$$\begin{aligned}
\rho_{IJ}^{\mathscr{L}} &= \rho_I \frac{(S_{IJ}^{\mathscr{L}} - q_{IJ}^{\mathscr{L}})}{S_{IJ}^{\mathscr{L}} - S_{IJ}^{\mathscr{M}}} \\
(\rho u_\alpha)_{IJ}^{\mathscr{L}} &= \frac{(S_{IJ}^{\mathscr{L}} - q_{IJ}^{\mathscr{L}})\rho_I u_{\alpha I} + (p_{IJ}^* - p_I) n_{IJ}^\alpha}{S_{IJ}^{\mathscr{L}} - S_{IJ}^{\mathscr{M}}} \\
(E)_{IJ}^{\mathscr{L}} &= \frac{(S_{IJ}^{\mathscr{L}} - q_{IJ}^{\mathscr{L}})E_I - p_I q_{IJ}^{\mathscr{L}} + p_{IJ}^* S_{IJ}^{\mathscr{M}}}{S_{IJ}^{\mathscr{L}} - S_{IJ}^{\mathscr{M}}}
\end{aligned} \tag{7}$$

and the entries in the vector $\vec{Q}_{IJ}^{\mathscr{R}}$ are defined similarly. In these equations,

$$p_{IJ}^* = \rho_I (q_{IJ}^{\mathscr{L}} - S_{IJ}^{\mathscr{L}})(q_{IJ}^{\mathscr{L}} - S_{IJ}^{\mathscr{M}}) + p_I, q_{IJ}^{\mathscr{L}} = u_{\alpha I} n_{IJ}^\alpha, q_{IJ}^{\mathscr{R}} = u_{\alpha J} n_{IJ}^\alpha \tag{8}$$

and the wave speeds are computed as

$$S_{IJ}^{\mathscr{M}} = \frac{\rho_J q_{IJ}^{\mathscr{R}}(S_{IJ}^{\mathscr{R}} - q_{IJ}^{\mathscr{R}}) - \rho_I q_{IJ}^{\mathscr{L}}(S_{IJ}^{\mathscr{L}} - q_{IJ}^{\mathscr{L}}) + p_I - p_J}{\rho_J (S_{IJ}^{\mathscr{R}} - q_{IJ}^{\mathscr{R}}) - \rho_I (S_{IJ}^{\mathscr{L}} - q_{IJ}^{\mathscr{L}})} \tag{9}$$

The HLLC flux function is then evaluated as

$$\tilde{\vec{F}}_{IJ}^{HLLC} = \begin{cases} \vec{F}(\vec{Q}_I) & \text{if } S_{IJ}^{\mathscr{L}} > 0 \\ \vec{F}_{IJ}^{\mathscr{L}} & \text{if } S_{IJ}^{\mathscr{L}} \leq 0 < S_{IJ}^{\mathscr{M}} \\ \vec{F}_{IJ}^{\mathscr{R}} & \text{if } S_{IJ}^{\mathscr{M}} \leq 0 \leq S_{IJ}^{\mathscr{R}} \\ \vec{F}(\vec{Q}_J) & \text{if } S_{IJ}^{\mathscr{R}} < 0 \end{cases} \tag{10}$$

Where $\vec{F}_{IJ}^{\mathscr{L}}$ and $\vec{F}_{IJ}^{\mathscr{R}}$ are obtained using the Rankine–Hugoniot conditions

$$\vec{F}_{IJ}^{\mathscr{L}} = \vec{F}(\vec{Q}_I) + S_{IJ}^{\mathscr{L}}(\vec{Q}_{IJ}^{\mathscr{L}} - \vec{Q}_I) \qquad \vec{F}_{IJ}^{\mathscr{R}} = \vec{F}(\vec{Q}_J) + S_{IJ}^{\mathscr{R}}(\vec{Q}_{IJ}^{\mathscr{R}} - \vec{Q}_J) \tag{11}$$

5 Second Order HLLC

The HLLC solver described above turns the finite volume scheme first order, and it is well known to be excessively diffusive. To overcome this shortcoming a second order HLLC is generally used. This is obtained by a second order approximation in the Riemann problem of the left and right primary variables, namely; density, velocity field and energy, That is

$$\begin{aligned} \tilde{\vec{Q}}^{\mathscr{L}}_{IJ} &= \vec{Q}^{\mathscr{L}}_{IJ} + \delta\vec{Q}_{IJ} \\ \tilde{\vec{Q}}^{\mathscr{R}}_{IJ} &= \vec{Q}^{\mathscr{R}}_{IJ} + \delta\vec{Q}_{JI} \end{aligned} \tag{12}$$

where $\delta\vec{Q}_{IJ}$ and $\delta\vec{Q}_{JI}$ are some gradient approximation.

However this makes the scheme instable around discontinuities and steep gradients. To recover stability a limiter is used to turn the method first order around discontinuities and clip excessive values of gradient. In the following we derive a robust, parameter free and efficient limiter for a wide range of flow speeds. The performance of any limiter depends strongly on the accuracy of the reconstructed gradient. Therefore we propose first a new accurate method for gradient reconstruction.

6 Iteratively Corrected Least–Squares (ICLS) Method

First we consider the classical Least–Squares (LS) method. It is shown for instance in [20] that this method produces accurate results on isotropic meshes, while a significant loss of accuracy is observed in highly stretched meshes. It is proved in the same paper that accuracy is recovered for vertex-based discretizations using weighted LS version by the inverse distance, the method still exact for linear functions only however. Now, let us recall the LS formulation: if I is a given vertex, we try to find a 3×3 $\nabla\vec{Q}_I$ vector such that

$$\vec{h}_{IJ} \cdot \nabla\vec{Q}_I = \vec{Q}_J - \vec{Q}_I \qquad \text{for} \qquad J \in N_I \tag{13}$$

where N_I is the set of node I neighbors' indexes and $\vec{h}_{IJ}$ is the edge vector connecting node I to J. Writing equation (13) in matrix form gives

$$\vec{A}_I \nabla\vec{Q}_I = \triangle\vec{Q}_I \qquad \text{for} \qquad I \in N_I \tag{14}$$

Where $(\vec{A}_I)_{JK} = (\vec{h}_{IJ})_K$ and $(\triangle\vec{Q}_I)_J = \vec{Q}_J - \vec{Q}_I$.

In general $\vec{A}_I$ is not a square matrix, and then equation (14) has no solution. Then alternatively, we look for the best plan that fits the set $\{\vec{Q}_K, K \in N_I \cup \{I\}\}$ using a least squares method. This consists of finding a vector $\vec{V}_I$ minimizing the norm

$$\|\vec{A}_I\vec{V} - \triangle\vec{Q}_I\|_2 = \sqrt{\langle \vec{A}_I\vec{V} - \triangle\vec{Q}_I, \vec{A}_I\vec{V} - \triangle\vec{Q}_I\rangle} \qquad \text{that is}$$

$$\vec{V}_I = \arg(\min_{V \in IR^3} \|\vec{A}_I\vec{V} - \triangle\vec{Q}_I\|_2) \tag{15}$$

and then set $\nabla\vec{Q}_I \approx \vec{V}_I$.

Since in equation (14) the number of unknown is less than the number of equations, the problem formulated in equation (15) is classical in optimization theory which solution is given by

$$\vec{V}_I = \left(\vec{A}_I^T \vec{A}_I\right)^{-1} \vec{A}_I^T \triangle\vec{Q}_I \tag{16}$$

The matrix $\left(\vec{A}_I^T \vec{A}_I\right)^{-1} \vec{A}_I^T$ is known as the pseudo–inverse of the matrix $\vec{A}_I$.

Now re–write equation (14) as

$$\frac{\vec{h}_{IJ}}{|\vec{h}_{IJ}|} \cdot \nabla\vec{Q}_I = \frac{\vec{Q}_J - \vec{Q}_I}{|\vec{h}_{IJ}|} \qquad \text{for} \qquad J \in N_I \tag{17}$$

In equation (14) we try to fit the dot product of the gradient with vectors representing the edges to the variable differences, while 17 tries to fit the derivatives in directions defined by the connected edges to the corresponding first order derivatives approximation, which is more consistent. Equations 14 and 17 are obviously the same, however the LS corresponding solutions are different. When the 2-norm used in 15 is replaced by a weighted 2-norm we get what known as the weighed LS method. The weighted 2-norm is defined as

$$\|\vec{A}_I\vec{V} - \triangle\vec{Q}_I\|_{2,w}^2 = \sum_{J\in N_I} w_{IJ}^2 \left(\vec{h}_{IJ}\vec{V} - (\vec{Q}_J - \vec{Q}_I)\right)^2$$

Now let us express the unweighted norm for equation (17)

$$\|\vec{A}_I\vec{V} - \triangle\vec{Q}_I\|_2^2 = \sum_{J\in N_I} \left(\nabla\vec{Q}_I \frac{\vec{h}_{IJ}}{|\vec{h}_{IJ}|} - \frac{(\vec{Q}_J - \vec{Q}_I)}{|\vec{h}_{IJ}|}\right)^2 = \sum_{J\in N_I} \frac{1}{|\vec{h}_{IJ}|} \left(\nabla\vec{Q}_I\vec{h}_{IJ} - (\vec{Q}_J - \vec{Q}_I)\right)^2$$

Consequently, the unweighted LS solution of 17 is obtained from the weighted LS solution of equation 14 with weights $w_{IJ} = 1/|\vec{h}_{IJ}|$ the inverse distance, we then end with the method proposed in [20].

In the following we still refer by $\vec{A}_I$ and $\triangle\vec{Q}_I$ to the matrix and the right hand side of matrix form of equation 17, in this case

$$(\vec{A}_I)_{IJ} = \left(\frac{\vec{h}_{IJ}}{|\vec{h}_{IJ}|}\right) \qquad\qquad (\vec{Q}_I)_J = \frac{\vec{Q}_J - \vec{Q}_I}{|\vec{h}_{IJ}|} \tag{18}$$

Now we propose to improve this method by an interactive process. We refer to the new method by ICLS standing for iteratively corrected LS method. Using the second order Taylor's formula we have

$$\frac{\vec{Q}_J - \vec{Q}_I}{|\vec{h}_{IJ}|} = \frac{\vec{h}_{IJ}}{|\vec{h}_{IJ}|}\nabla\vec{Q}_I + \frac{1}{2|\vec{h}_{IJ}|}\left\langle H(\eta)\vec{h}_{IJ}, \vec{h}_{IJ}\right\rangle \tag{19}$$

In the remainder term, H is the Hessian and η is some point on the edge $\vec{h}_{IJ}$. By approximating η by the midpoint of $\vec{h}_{IJ}$, the remainder term could is approximated as

$$\left\langle H(\eta)\vec{h}_{IJ}, \vec{h}_{IJ} \right\rangle = \left(\vec{h}_{IJ}\nabla\vec{Q}_J - \vec{h}_{IJ}\nabla\vec{Q}_I\right) \qquad \text{and then}$$

$$\frac{\vec{Q}_J - \vec{Q}_I}{|\vec{h}_{IJ}|} = \frac{\vec{h}_{IJ}}{|\vec{h}_{IJ}|}\nabla\vec{Q}_I + \frac{1}{2|\vec{h}_{IJ}|}\left(\vec{h}_{IJ}\nabla\vec{Q}_J - \vec{h}_{IJ}\nabla\vec{Q}_I\right) \tag{20}$$

In equation 20 $\nabla\vec{Q}_I$ and $\nabla\vec{Q}_J$ are unknown, therefore we propose the following iterative scheme:

1. First compute the LS estimate of the gradient, $\nabla\vec{Q}_I^0$ using 17
2. Compute iteratively LS estimate of $\nabla\vec{Q}_I^n$ using equation

$$\frac{\vec{h}_{IJ}}{|\vec{h}_{IJ}|}\nabla\vec{Q}_I^n = \frac{\vec{Q}_J^n - \vec{Q}_I^n}{|\vec{h}_{IJ}|} - \underbrace{\frac{1}{2|\vec{h}_{IJ}|}\left(\vec{h}_{IJ}\nabla\vec{Q}_J^{n-1} - \vec{h}_{IJ}\nabla\vec{Q}_I^{n-1}\right)}_{\text{Correction term}}$$

The algorithm using the pseudo-inverse solution reads:

1. First compute the LS estimate of the gradient, $\nabla\vec{Q}_I^0$ as:

$$\nabla\vec{Q}_I^0 = \left(\vec{A}_I^T\vec{A}_I\right)^{-1}\vec{A}_I^T\triangle\vec{Q}_I$$

2. Compute iteratively LS estimate of $\nabla\vec{Q}_I^n$

$$\nabla\vec{Q}_I^n = \left(\vec{A}_I^T\vec{A}_I\right)^{-1}\vec{A}_I^T\left(\triangle\vec{Q}_I + \underbrace{\frac{1}{2}\triangle\nabla\vec{Q}_I^{n-1}}_{\text{Correction term}}\right)$$

Note that only the right-hand side in $\nabla\vec{Q}_I^n$ expression is modified, moreover the pseudo-inverse depends only on the mesh, that means that it could be computed ones and stored. This makes the algorithm not expensive. Note also that the algorithm uses larger stencil as it iterates, which improves its accuracy at each step. The efficiency of the method is demonstrated in the numerical results section.

7 Limiter Design

To design our limiter, we investigate in the following some relevant proprieties the scheme should satisfies and then establish the necessary conditions to build such a limiter.

7.1 Conservative Condition

The cell vertex finite volume method is inspired from the conservative principle used to derive the Navier-Stokes equations. The correction 12 to achieve second order HLLC changes the variables value at both sides of cells interface. To preserve conservative principle the integral

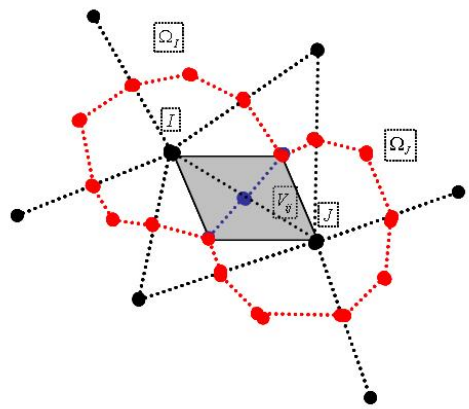

Figure 4: A 2D representation of two adjacent dual cells.

of each variable on the volume defined by the interface and the vertices of the adjacent cells has to be constant. That is:

$$\int_{V_{IJ}} \bar{Q}_{IJ} = \int_{V_{IJ}} Q_{IJ} \tag{21}$$

Where $Q, \bar{Q}$ refer to any variable before and after modification and V_{IJ} is the shaded volume shown in figure 4.

21 implies

$$\bar{Q}^L_{IJ} + \bar{Q}^R_{IJ} = Q^L_{IJ} + Q^R_{IJ}$$

That is

$$\delta Q_{IJ} = -\delta Q_{JI} \tag{22}$$

Relation (25) implies that if derivatives at nodes I and J have opposite sign they have to be zero, and if they are of the same sign they have to have same value. This condition is satisfied using for instance the well known minmod function 23 at values δQ_{IJ} and δQ_{JI}, which could be seen therefore as a conservative condition.

$$Minmod(a,b) = \begin{cases} 0 & \text{if } ab < 0 \\ else & \\ a & \text{if } |a| \leq |b| \\ b & \text{if } |a| \geq |b| \end{cases} \tag{23}$$

7.2 Monotonicity Property

Second order schemes are oscillatory not only in the vicinity of discontinuities but also in regions showing steep gradients. To ensure non-oscillatory scheme, we impose one of the several criteria developed in the literature for this purpose to be satisfied. Among them; the monotonicity preservation, the total variation diminishing (TVD), and the local extremum diminishing (LED), see [21, 22] for details. In practice the last one is easier to enforce by flux limiting ensuring that we always have:

$$\underset{J}{Min}(Q_J - Q_J) \leq \nabla Q_I \vec{h}_{IK} \leq \underset{J}{Max}(Q_J - Q_J) \tag{24}$$

In the following the (LED) creation is selected. To satisfy 24 let us first compute the Barth and Jespersen limiter [21] and express it as in [**?**] but for vertex-centered scheme rather than element-centered scheme. For each edge $\vec{h}_{IK}$, Φ_{IK} is define as

$$\Phi_{IK} = \begin{cases} Min\left(1, \frac{Max_J\, Q_J - Q_I}{\nabla Q_I \vec{h}_{IK}}\right) & \text{if } \nabla Q_I \vec{h}_{IK} > 0 \\ Min\left(1, \frac{Min_J\, Q_J - Q_I}{\nabla Q_I \vec{h}_{IK}}\right) & \text{if } \nabla Q_I \vec{h}_{IK} < 0 \\ 1 & \text{ielse} \end{cases}$$

And then take the minimum on the connected edges to node I

$$\Phi_I = \underset{K \in N_I}{Min}\, \Phi_{IK}$$

And then the gradient $\nabla Q_I \vec{h}_{IK}$ is replaced by $\Phi_I \nabla Q_I \vec{h}_{IK}$.

In practice, this limiter shows an excessive smoothing of the gradient, especially for transonic flows. This is due to the fact that in the case condition 24 is violated, the gradient in all directions are normalized by the dominated one. This causes the relatively small gradients to be reduced to almost zero. The limiter suffers also from non-differentiability, that could affect convergence for high order finite volume schemes see [22]. To obtain gradual gradient normalization and differentiable limiter, we propose the following formula:

$$\Phi_{IK} = \begin{cases} \left(\dfrac{\underset{J}{Max}\, Q_J - Q_I}{\nabla Q_I \vec{h}_{IK}}\right) sg_n\left(\dfrac{\nabla Q_I \vec{h}_{IK}}{\underset{J}{Max}\, Q_J - Q_I}\right) & \text{if } \nabla Q_I \vec{h}_{IK} > 0 \\ \left(\dfrac{\underset{J}{Min}\, Q_J - Q_I}{\nabla Q_I \vec{h}_{IK}}\right) sg_n\left(\dfrac{\nabla Q_I \vec{h}_{IK}}{\underset{J}{Min}\, Q_J - Q_I}\right) & \text{if } \nabla Q_I \vec{h}_{IK} < 0 \\ 1 & \text{else} \end{cases} \tag{25}$$

With $sg_n(t) = \frac{t}{(1+t^n)^{\frac{1}{n}}}$ being a family of functions we derived from a sigmoid function. First note that this limiter is used without taking the minimum over index K which may reduce excessively the gradient. In addition, the sigmoid function has the following nice properties:

$$sg_n(t) \leq t \quad \text{and} \quad sg_n(t) \leq 1, \quad \text{for } t \geq 0 \tag{26}$$

This property guaranty condition (24) to be fulfilled and non amplification of local gradients. $sg_n(t)$ is a good approximation of the $Min(.,.)$ function since we have

$$\underset{n\to\infty}{Limit}(sg_n(t)) = Min(1,t), \quad \text{for } t \geq 0 \tag{27}$$

Note that in practice the value of n is taken to be equal to 4 for inviscid flows and 2 for viscous flows.

8 Summary of the Method

In summary, we have developed and discussed through sections 6 and 7 the essential ingredients for a suitable limiter ensuring an efficient and robust second order HLLC-Finite volume scheme. This is therefore achieved by modifying the left and right variables values in the HLLC Riemann solver as follows:

$$\begin{aligned} \bar{Q}^L_{IJ} &= Q^L_{IJ} + \delta Q_{IJ} \\ \bar{Q}^R_{IJ} &= Q^R_{IJ} + \delta Q_{JI} \end{aligned}$$

Where $\delta Q_{IJ} = \frac{1}{2} Minmod\left(\Phi_{IJ}\nabla Q_I \vec{h}_{IJ}, -\Phi_{JI}\nabla Q_J \vec{h}_{JI}\right)$ ∇Q_I being the reconstructed gradient using ICLS and $\Phi_{IJ}$is given by (25)

9 Numerical Results

To demonstrate the efficiency of the ICLS, the method is applied to reconstructing a quadratic function and a fourth order polynomial gradient on a cubic domain. Results of the gradient norm obtained by the LS corresponding to the ICLS initial estimation and iterated solutions are compared to the exact solution. Figure 5 shows the tetrahedral mesh used for tests.

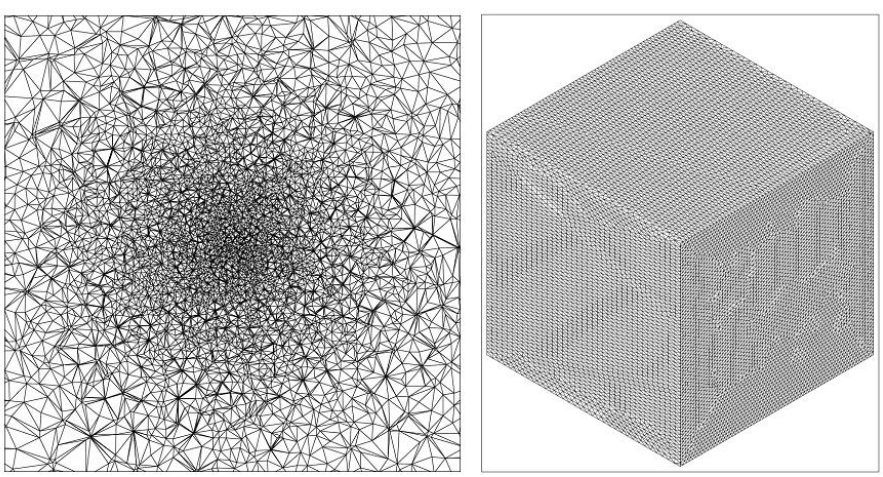

Figure 5: Tetrahedral triangulation of the cubic domain.

Figure 6 shows the exact solution of the quadratic function on a cut through the cube center and on faces. Initial and after 10 iteration results are shown in figures 7 and 8. The gradient

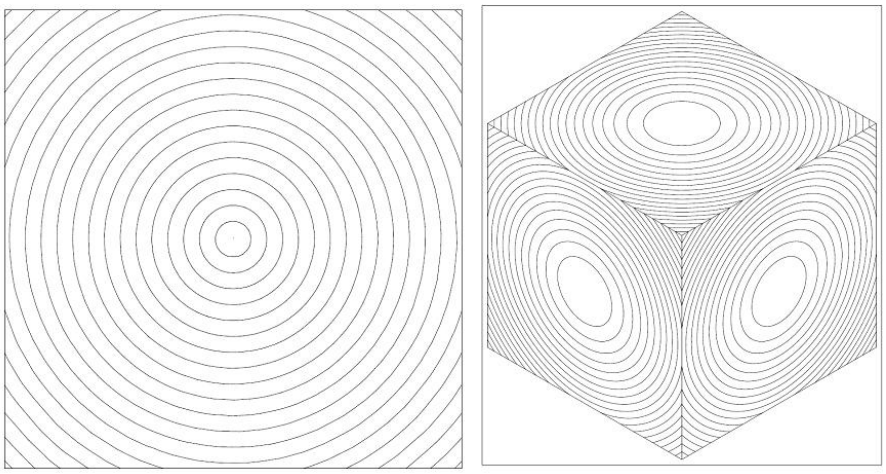

Figure 6: Quadratic function gradient norm: Exact Solution.

is perfectly reconstructed, especially on boundaries where the gradient is much more difficult to estimate. A convergence curve of the l^2 error logarithm is drawn in figure 9 proving the

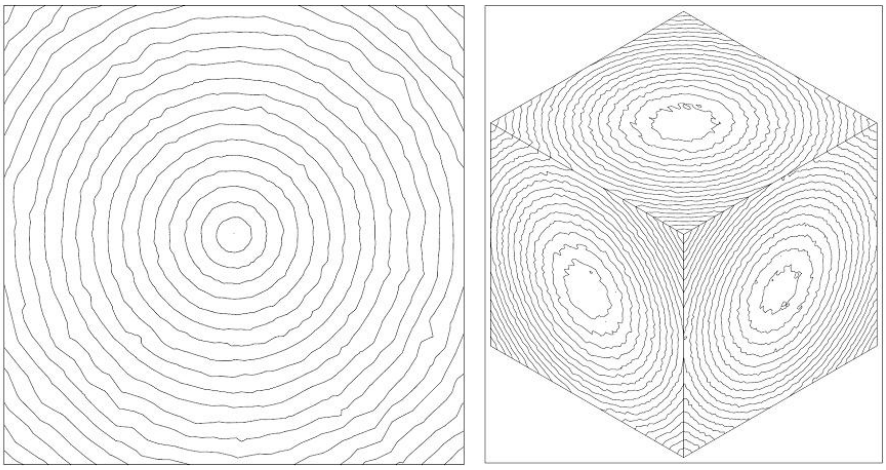

Figure 7: Quadratic function gradient norm: ICLS Initial Estimate (LS method).

convergence of the method to the exact solution. This demonstrates that ICLC is exact for quadratic functions while the classical LS is exact only for linear functions. Figures 10 to 13 show the same results for a fourth order polynomial function, we can appreciate the sensitive improvement with iterations. Note finally that the most improvement occur during the few first iterations, which means that we don't need to iterate for a long time.

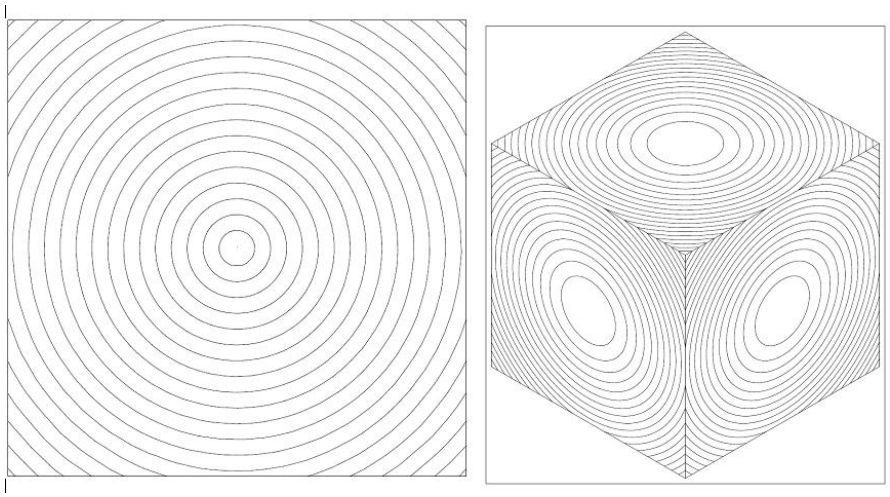

Figure 8: Quadratic function gradient norm: ICLS after 10 Iterations.

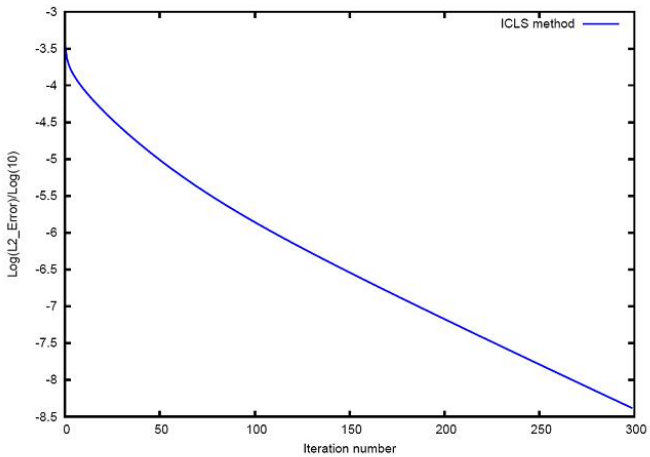

Figure 9: ICLS Quadratic function gradient estimation: Convergence error curve.

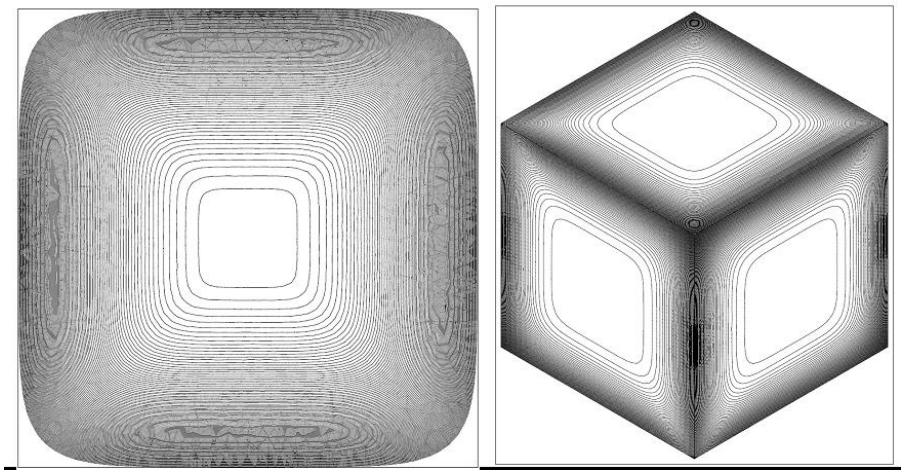

Figure 10: fourth order polynomial function gradient norm: Exact Solution.

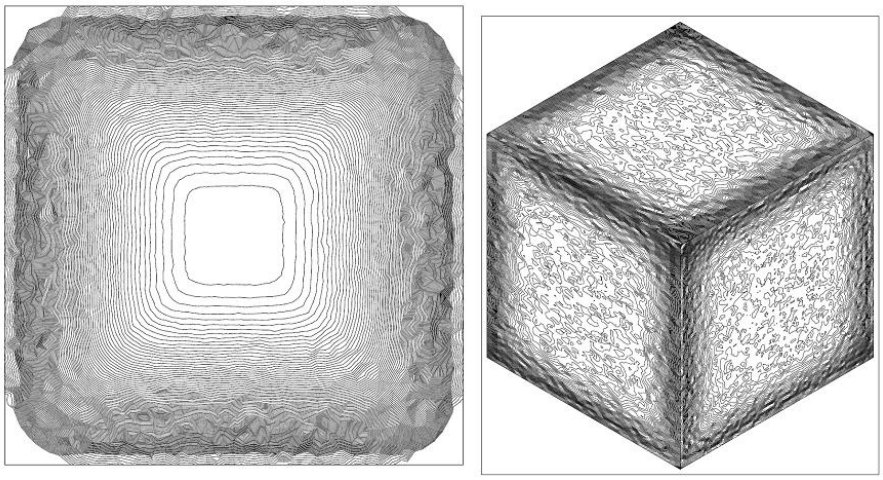

Figure 11: ICLS Fourth order Polynomial function gradient norm: ICLS Initial Estimate.

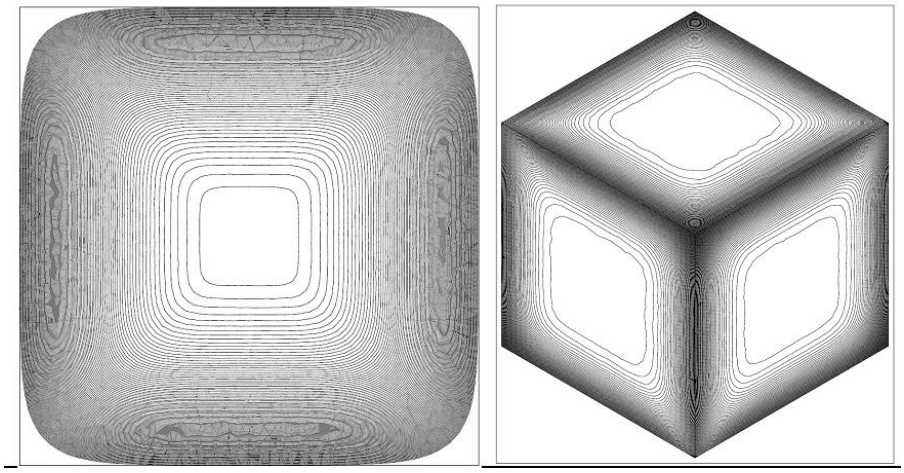

Figure 12: Fourth order Polynomial function gradient norm: ICLS after 100 Iterations.

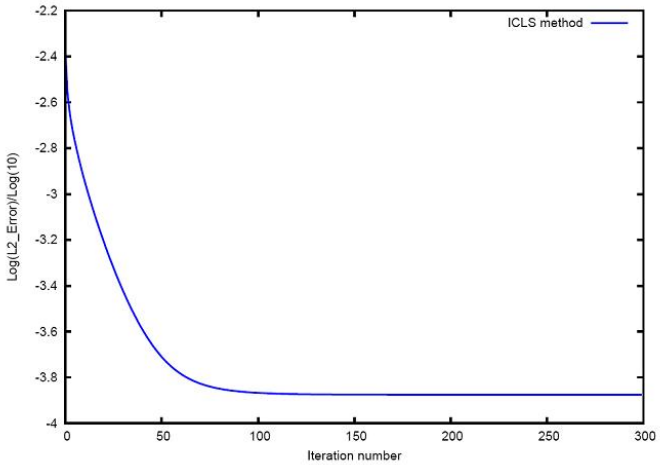

Figure 13: ICLS Fourth order Polynomial function gradient estimation: Convergence error curve.

To demonstrate the effectiveness of the proposed flow solver, a viscous transonic flow over an ONERA M6 wing and an inviscid supersonic flow over a fighter F15 have been simulated. For the ONERA M6 a hybrid mesh is generated using FLITE package (described in for instance [8] and [9]). The mesh contains 4880340 tetrahedral elements with a 10 prismatic layers. The imposed flight conditions are M=0.8395, AoA =3.06 and Re=11.e7. Figures 14 shows the mesh and the λ shock of the steady state obtained after 9500 time iterations as shown on figure 15 where the Log of L^2 norm of the residual is drawn.

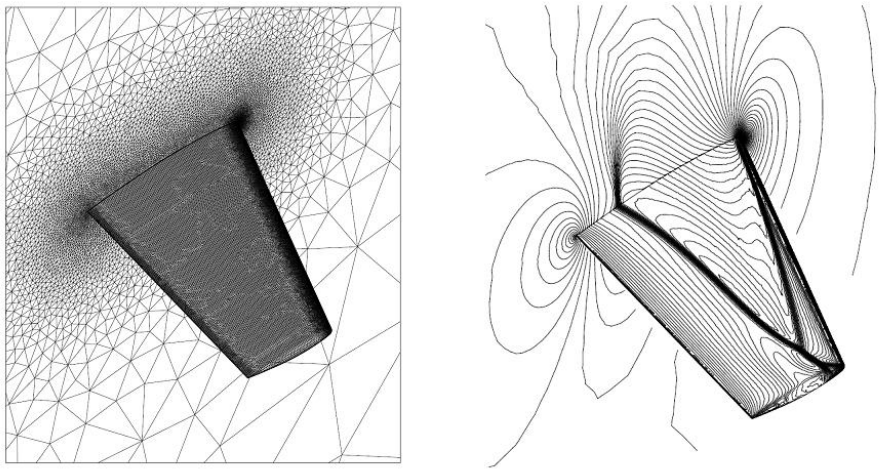

Figure 14: Transonic flow over ONRA M6 wing. Used mesh and Cp profile.

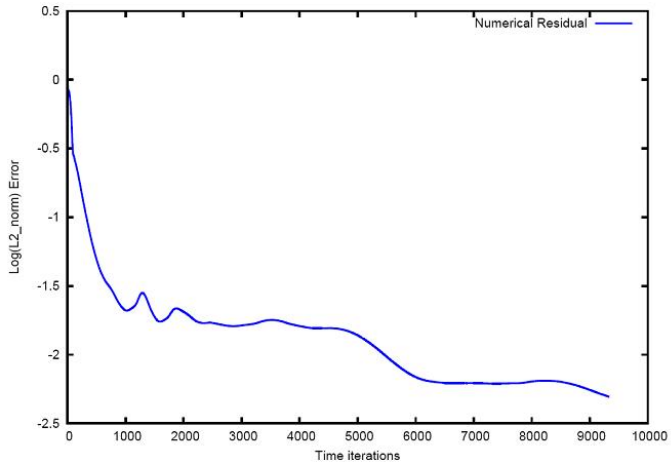

Figure 15: Convergence error curve.

The C_p chordwise profiles are compared to experimental results for different stations in Figure16. The results show a very good agreement with experiment, especially the result at 80% of wing

semi span, which is the most difficult to obtain since the cut is too close to the λ shock corner.

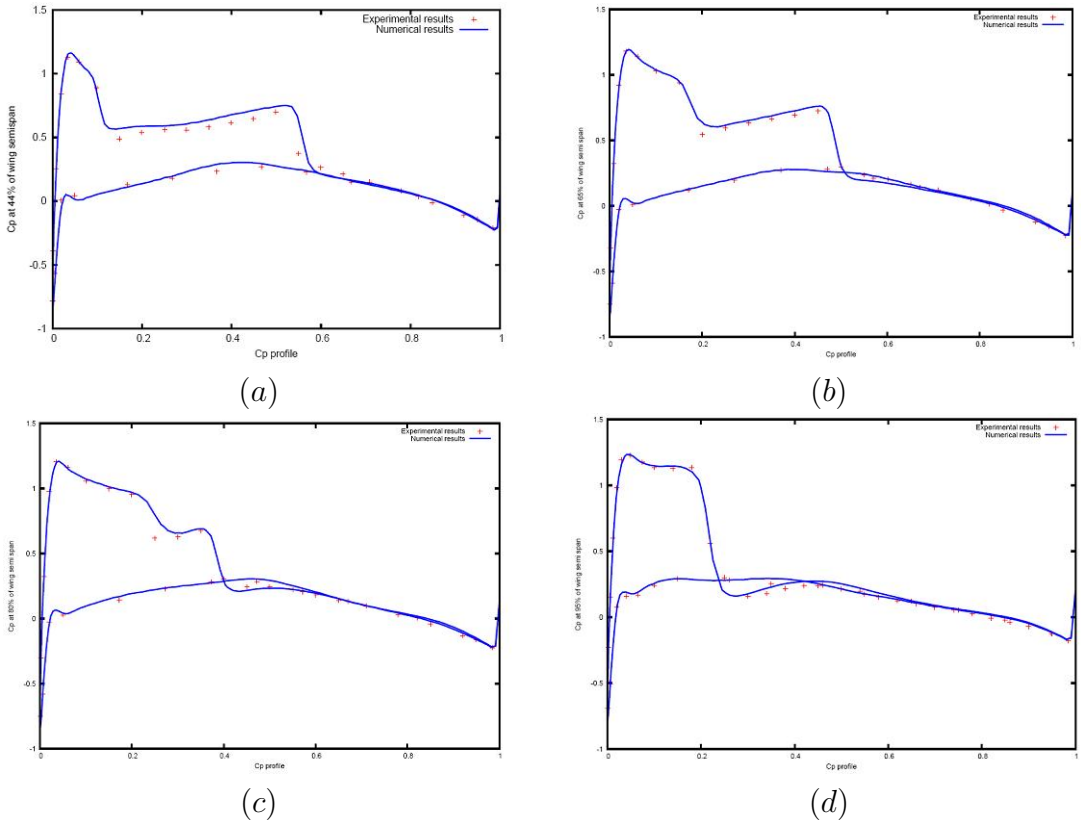

Figure 16: Cp chordwise profile compared to experimental results

To test the robustness of the proposed scheme, it is applied as it is,without tuning any parameter, to inviscid supersonic hight speed flows over F15 fighter. The F15 geometry is discretised using FLITE system to 7599995 tetrahedral elements. The test is run at flight condition $M = 2.$ and $AoA = 3.$ Figure 17 shows the mesh and the density profile, we can appreciate the well capturing of the physics, the shocks especially. Figure 18 shows a good convergence of the residual.

10 Conclusions

We presented in this paper new developments within the numerical resolution of Navier-Stokes equations using HLLC-Finite volume method. The main contributions include HLLC acoustic waves improvement, a new limiter design based on a stability analysis and an accurate gradient reconstruction. Viscous and inviscid external flows simulations over an ONERA M6N wing and F15 fighter are performed. The Mach number was varied from 0.8395 to 2. The results demonstrate the accuracy and robustness in terms of convergence and stability of the

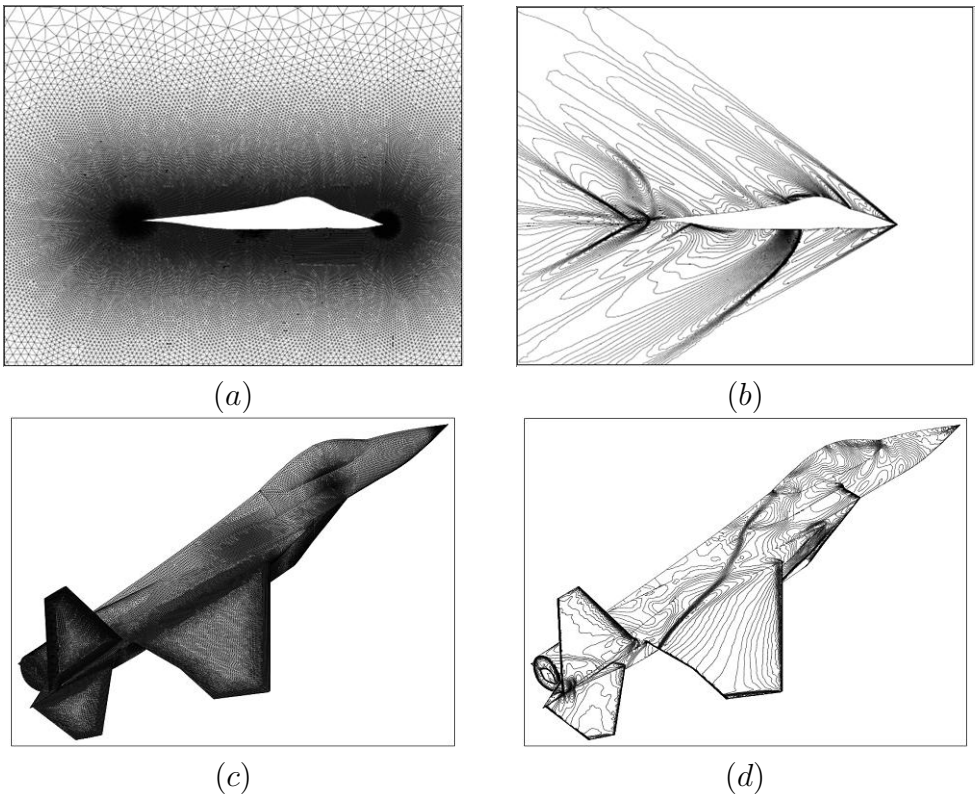

Figure 17: Supersonic flow over the F15 fighter: Mach = 2.

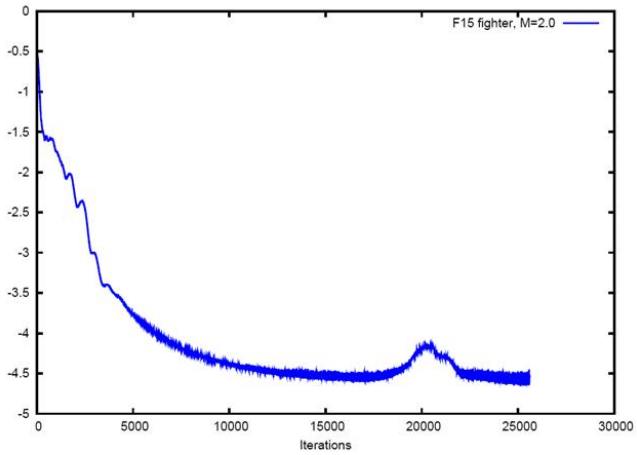

Figure 18: Residual error curve.

method, over a wide range of flow speeds, without the requirement of any tuning of user defined parameters.

REFERENCES

[1] V. Girault, P.–A. Raviart, Finite Element Methods for the Navier Stokes Equations, Springer, Berlin, 1986.

[2] A. Corsini, A FE Method for the Computational Fluid Dynamics of Turbomachinery, Lecture Notes, October 1999.

[3] R. Eymard, T. Galluöet, R. Herbin, Finite volume methods. Handbook of Numerical Analysis, North Holland, Amsterdam, 7:713-1020, 2000.

[4] C. Chainais–Hillairet, *Finite volume schemes for a nonlinear hyperbolic equation: convergence towards the entropy solution and error estimates*, M2AN Math. Model. Numer. Anal., 33(1999), 129–156.

[5] P–H. Cournède, C. Debiez, A. Dervieux, A positive MUSCL scheme for triangulations. INRIA Report 3465, 1998.

[6] J. A. Desideri, A. Dervieux, Compressible flow solvers using unstructured grids. Von Karman Institute Lecture Notes 1988–05, Von Karman Institute for Fluid Dynamics, Belgium, 1988.

[7] R. E. Ewing, T. Lin Y. and Lin, *On the accuracy of the finite volume element method based on piecwise linear polynomials*, SIAM J. Numer. Anal., 39(2002), 1865–1888.

[8] K. A. Sørensen, O. Hassan, K. Morgan, N. P. Weatherill, *A multigrid accelerated time–accurate inviscid compressible fluid flow solution algorithm employing mesh movement and local remeshing*, Int.J.Num.Meth.Fluids, 45(2003), 517–536.

[9] N. P. Weatherill, O. Hassan, K. Morgan, J. W. Jones, B. G. Larwood, K. A. Sørensen, *Aerospace simulations on parallel computers using unstructured grids*, Int.J.Num.Meth.Fluids, 40(2002), 171–187.

[10] E. F Toro, M. Spruce, W. Speares, *Restoration of the contact surface in the HLL–Riemann Solver*, Shock Waves, 4(1994), 25–34.

[11] E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, A Practical Introduction (2nd edn), Springer, Berlin, 1999.

[12] S. Taasan, D. M. Nark, An absorbing buffer zone technique for acoustic wave propagation, AIAA Paper 95–0146, 1995.

[13] J. Peraire, J. Peri, K. Morgan, *Adaptive remeshing for three–dimensional compressible flow computation*, J.Comp.Phys., 103(1992), 269–285.

[14] W. G. Habashi, J. Dompierre, Y. Bourgault, D. Ait–Ali–Yahia, M. Fortin, M–G. Vallet, *Anisotropic mesh adaptation: towards user–independent, mesh–independent and solver–independent CFD solutions. Part I: general principles*, Int.J.Num.Meth.Fluids, 32(2000), 725–744.

[15] E. F. D'Azevedo, R. B. Simpson, *On optimal triangular meshes for minimizing the gradient error*, Nm.Math., 59(1991), 321–348.

[16] M. Fortin, *Anisotropic mesh adaptation through hierarchical error estimators*, SIAM J.Numer.Anal., 26(1998), 788–811.

[17] V. Dolejsi, J. Felcman, Anisotropic mesh adaptation for transonic and supersonic flow simulation, in Proceedings of ALGORITMY Conference on Scientific Computing, 78–85 2002.

[18] L. Remaki, W. G. Habashi, *3–D mesh adaptation on multiple weak discontinuities and boundary layers*, SIAM J.Sci.Comput., 28(2006), 1379–1397.

[19] L. Remaki, C. Lepage, W. G. Habashi, Efficient anisotropic mesh adaptation on weak and multiple shocks, AIAA Paper 2004–0084, 2004.

[20] D. J. Mavriplis, Revisiting the least–squares procedure for gradient reconstruction on unstructured meshes, AIAA Paper 2003–3986, 2003.

[21] T. J. Barth, D. C. Jespersen, The design and application of upwind schemes on unstructured meshes, In 27th Aerospace Sciences Meeting, Jan. 1989. AIAA paper 89-0366.

[22] M. Berger, M. J. Aftosmis, Analysis of Slope Limiters on Irregular Grids, NAS Technical Report NAS-05-007.