
Three-dimensional Adaptive Central Schemes on Unstructured Staggered Grids

Aziz Madrane¹

Airbus/Institut for Aerospace-Technology aziz.madrane@airbus.com

Summary. We present a new formulation of three-dimensional central finite volume methods on unstructured staggered grids for solving systems of hyperbolic equations. Based on the Lax-Friedrichs and Nessyahu-Tadmor one-dimensional central finite difference schemes, the numerical methods we propose involve a staggered grids in order to avoid solving Riemann problems at cell interfaces. The cells are barycentric, while those of the staggered grid are diamond-shaped. In order to reduce artificial viscosity, we start with an adaptively refined primal grid in 3D, where the theoretical *a posteriori* result of the first-order scheme is used to derive appropriate refinement indicators. We apply those methods and solve Euler equations. Our numerical results are in good agreement with corresponding results appearing in the literature.

1 Introduction

Staggered central finite volume schemes have been introduced by Nessyahu and Tadmor in 1990 [NT90]. The main advantage of these schemes is that no information about solutions to local Riemann problems is needed. Using staggered grids one can replace the upwind fluxes by central differences. The price one has to pay is the occurrence of excessive numerical viscosity since the resulting scheme can be interpreted as a Lax-Friedrichs scheme. Therefore, a MUSCL-type higher-order scheme in one spatial dimension was proposed in [NT90]. Later in [AVM97] and [AMS01] the central schemes have been generalized to multi-dimensional schemes on unstructured grids. The present paper focusses on the derivation of high-order central schemes on unstructured staggered grids. Furthermore we extend an adaptation strategy of [KO02] for first- and second-order methods where the theoretical *a posteriori* result of first-order schemes is used to derive appropriate refinement indicators.

2 Mathematical modelling

2.1 Governing equations

Let $\Omega \subset R^3$ be the domain of interest of the flow with boundary Γ . We write $\Gamma = \Gamma_B \cup \Gamma_\infty$, where Γ_B denotes the part of the body boundary which is relevant for the computational domain and Γ_∞ is the (upwind) farfield boundary. Three-dimensional compressible inviscid flows are described by Euler's equations, written in their conservation form

$$\frac{\partial U}{\partial t} + \nabla \cdot \vec{\mathcal{F}}(U) = 0, \quad (1)$$

where

$$U = (\rho, \rho u, \rho v, \rho w, E)^T, \quad \vec{\mathcal{F}}(U) = (F(U), G(U), H(U))^T.$$

Here $\vec{\mathcal{F}}(U)$ denotes the convective flux, ρ is the density, $\vec{V} = (u, v, w)^T$ is the velocity vector, $E = \rho e = \rho e + \frac{1}{2} \rho(u^2 + v^2 + w^2)$ is the total energy per unit volume, and p is the pressure of the fluid.

3 Space and time discretization

3.1 Definitions

We assume that Ω is a bounded polyhedral domain of R^3 and we start from an arbitrary FEM tetrahedral grid \mathcal{T}_h , where h is the maximal length of the edges in \mathcal{T}_h .

A dual finite volume partition is derived from the construction of median planes, that is, for every vertex i of \mathcal{T}_h , a cell C_i is defined around i as follows.

Every tetrahedron having i as a vertex is subdivided into 24 sub-tetrahedra by planes containing an edge and the midpoint of the opposite edge; then the cell C_i is the union of sub-tetrahedra having i as a vertex (see Fig. 1). In

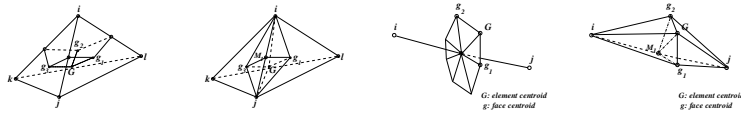


Fig. 1. Barycentric cell C_i and diamond cell L_{ij} and Sketch of $\partial C_{ij} = \partial C_i \cap \partial C_j$ and $\partial L_{ij} = \partial L_{ij} \cap \tau$

particular, the boundary ∂C_i of C_i is the union of $\partial C_{ij} = \partial C_i \cap \partial C_j$ that can be defined as the union of triangles (see Fig. 1) such that As for two-dimensional

extensions [AVM97], the present three-dimensional extension also uses a dual grid, with dual cells L_{ij} associated with the edges of \mathcal{T}_h . The dual (“diamond”) cell L_{ij} is composed of the sub-tetrahedra (defined above) sharing edge $[i, j]$ (see Fig. 1). The following notation will be needed.

Notation 1 *Let i, j, k, l be the four nodes defining a tetrahedron τ , $\tau \in \mathcal{T}_h$. Then*

- T_{ij} denotes the set of all tetrahedra which share edge $[i, j]$ as a common edge,
- $K(i)$ is the set of nodes (vertices) which are neighbors of node i
- $C_i = \bigcup_{j \in K(i)} (C_i \cap L_{ij})$ and
- $\partial C_i = \bigcup_{j \in K(i)} \{\partial C_i \cap \partial C_j\} \cup \{\partial C_i \cap \Gamma_B\} \cup \{\partial C_i \cap \Gamma_\infty\},$
- $L_{ij} = \bigcup_{\tau \in T_{ij}} (L_{ij} \cap C_i \cap \tau) \cup (L_{ij} \cap C_i \cap \tau)$ and
- $\partial L_{ij} = \bigcup_{\tau \in T_{ij}} (\partial L_{ij} \cap \tau) \cup (\partial L_{ij} \cap \Gamma_\infty) \cup (\partial L_{ij} \cap \Gamma_B),$
- $\mathbf{n}_{ij} = (n_{ij_x}, n_{ij_y}, n_{ij_z})$ is the unit outward normal vector to ∂L_{ij} ,
- $\boldsymbol{\nu}_i = (\nu_{i_x}, \nu_{i_y}, \nu_{i_z})$ is the unit outward normal vector to ∂C_i .

Let m_{ij} denote the midpoint of edge $[i, j]$, also written as M_1 in Fig. 1, and let $U_i^n \cong U(a_i, t^n)$ and $U_{ij}^{n+1} \cong U(m_{ij}, t^{n+1})$ denote the nodal (resp. cell average) values in the first and second grid at time $t = t^n$ and $t = t^{n+1}$, respectively, (n even). The union of all the barycentric cells constitutes a partition of the computational domain Ω_h and the same holds for diamond cells. Now we can define the two steps of our high-order accurate (staggered, Lax–Friedrichs type) finite volume method.

3.2 High-order accurate approximations

To obtain second-order accuracy, we introduce cellwise piecewise linear interpolation.

First step: We integrate (1) on $L_{ij} \times [t^n, t^{n+1}]$, assuming we have obtained, from the cell average values U_i^n , piecewise linear reconstructions given by

$$U_h(\mathbf{x}, t^n)|_{C_i} = \mathcal{L}_i(\mathbf{x}, t^n) = U_i^n + \nabla U_i^n \cdot (\mathbf{x} - \mathbf{x}_i), \quad \forall \mathbf{x} \in C_i, \mathbf{x} \in R^3. \quad (2)$$

For the integration with respect to time, in order to ensure “nearly” second-order accuracy, we adopt a “quasi-midpoint formula” time discretization, where the convective flux is computed at the intermediate time $t^{n+1/2}$, thus requiring the computation of **predicted** values $U_h(\mathbf{x}, t^{n+1/2})$ given ∂L_{ij} .

Predictor’s first step: On each face of the cell L_{ij} , using Euler’s equations, we define a predicted vector

$$U_{i,g_1,G}^{n+1/2} = U_{i,g_1,G}^n - \frac{\Delta t}{2} \vec{\mathcal{F}}'(U_{i,g_1,G}^n) \cdot \nabla U_i^n \quad (3)$$

where, by (2), the value of U_h^n is taken equal to

$$U_h(\mathbf{x}, t^n) \cong U_i^n + \nabla U_i^n \cdot (\mathbf{x}_{i,g_1,G} - \mathbf{x}_i) \equiv U_{i,g_1,G}^n \quad (4)$$

along the face ig_1G of the diamond cell L_{ij} .

Corrector's first step: By (3) the corrector can be written as follows:

$$\begin{aligned} & \text{Vol}(L_{ij})U_{ij}^{n+1} - \sum_{\tau \in T_{ij}} \left[\int_{L_{ij} \cap C_i \cap \tau} \mathcal{L}(\mathbf{x}, t^n) d\mathbf{x} + \int_{L_{ij} \cap C_j \cap \tau} \mathcal{L}(\mathbf{x}, t^n) d\mathbf{x} \right] \\ & + \Delta t \sum_{\tau \in T_{ij}} \left[\int_{\partial L_{ij}^1 \cap \tau} + \int_{\partial L_{ij}^2 \cap \tau} + \int_{\partial L_{ij}^3 \cap \tau} + \int_{\partial L_{ij}^4 \cap \tau} \right] \vec{\mathcal{F}}(U_h^{n+1/2}) \cdot \mathbf{n}_{ij} dA \\ & + \Delta t \int_{\partial L_{ij} \cap \Gamma_B} \vec{\mathcal{F}}(U_h^n) \cdot \mathbf{n} dA + \Delta t \int_{\partial L_{ij} \cap \Gamma_\infty} \vec{\mathcal{F}}(U_h^n) \cdot \mathbf{n} dA = 0. \end{aligned} \quad (5)$$

Where the volume and the boundary integrals are approximated by the mid-point rule.

Second step: To obtain the second step of the time discretization, we integrate (1) on the cell $C_i \times [t^{n+1}, t^{n+2}]$, assuming that, from the diamond cell average values U_{ij}^{n+1} computed in the first time step, we have obtained piecewise linear reconstructions given by

$$U_h(\mathbf{x}, t^{n+1})|_{L_{ij}} = \mathcal{L}_{ij}(\mathbf{x}, t^{n+1}) = U_{ij}^{n+1} + \nabla U_{ij}^{n+1} \cdot (\mathbf{x} - \mathbf{x}_{ij}). \quad (6)$$

Predictor's second step: Proceeding as in the first step, we obtain the predictor's second step:

$$U_{M_1g_1G}^{n+3/2} = U_{M_1g_1G}^{n+1} - \frac{\Delta t}{2} \vec{\mathcal{F}}'(U_{M_1g_1G}^{n+1}) \cdot \nabla U_{ij}^{n+1}, \quad (7)$$

where

$$U_h(\mathbf{x}, t^{n+1}) \cong U_{ij}^{n+1} + \nabla U_{ij}^{n+1} \cdot (\mathbf{x}_{M_1,g_1,G} - \mathbf{x}_{M_1}) \equiv U_{M_1,g_1,G}^{n+1} \quad (8)$$

defines an approximation to the value of U on the boundary element $[M_1, g_1, G]$ of cell C_i .

Corrector's second step: The second step is

$$\begin{aligned} & \text{Vol}(C_i)U_i^{n+2} - \sum_{j \in K(i)} \int_{C_i \cap L_{ij}} \mathcal{L}_{ij}(\mathbf{x}, t^{n+1}) d\mathbf{x} \\ & + \Delta t \sum_{j \in K(i)} \int_{\partial C_i \cap \partial C_j} \vec{\mathcal{F}}(U(\mathbf{x}, t^{n+3/2})) \cdot \boldsymbol{\nu}_i dA \\ & + \Delta t \int_{\partial C_i \cap \Gamma_B} \vec{\mathcal{F}}(U_h^{n+1}) \cdot \boldsymbol{\nu} dA + \Delta t \int_{\partial C_i \cap \Gamma_\infty} \vec{\mathcal{F}}(U_h^{n+1}) \cdot \boldsymbol{\nu} dA = 0, \end{aligned} \quad (9)$$

where the volume and the boundary integrals are computed as above.

Approximation of the slopes and limitation

Numerical experiments have led us to choose Green–Gauss’ method [BJ89] for the gradients used in the reconstruction for the cells C_i and a least squares weighted procedure [HB98] for the cells L_{ij} . For the limitation we use Venkatakrishnan’s limiter see [Ven95].

4 Mesh adaptation algorithm

4.1 General description

The theory behind the mesh adaptation technique for central schemes on unstructured staggered grids has been developed in [KO02]. We introduce the following three main steps in this technique.

- First, a strategy to determine where a modification is needed in the field of the grid, e.g. by means of an (*a posteriori*) error estimate.
- Secondly, a rule that selects the elements or edges in \mathcal{T}_h (marking strategy).
- Thirdly, a rule that refines the elements in \mathcal{T}_h (refinement strategy).

A posteriori error estimate:

For stationary problems, following the theory of [KO02], for each edge $e_{ij} \in \mathcal{T}_h$ we have the error estimate $\eta_{e_{ij}}$:

$$\|u - u_h\|_{L^1(e_{ij})} \leq \eta(u_{e_{ij}}) = aQ + b\sqrt{Q}, \quad (10)$$

where $a = 2 + 2\omega$, $\omega = 0.5$, $b = 4 + 2^d$, $d = 3$,

$$\begin{aligned} Q = & \frac{1}{2} \sum_{e_{ij}=1}^{ne} h_{e_{ij}} \text{Vol}(L_{e_{ij}}) \sum_{\tau \in T_{ij}} \frac{\text{Vol}(L_{e_{ij}} \cap C_i \cap \tau)}{\text{Vol}(L_{e_{ij}})} \frac{\text{Vol}(L_{e_{ij}} \cap C_j \cap \tau)}{\text{Vol}(L_{e_{ij}})} |u_i - u_j| \\ & + \sum_{e_{ij}=1}^{ne} \text{Vol}(L_{e_{ij}}) \left| u_{e_{ij}} - \sum_{j \in K(i)} \frac{\text{Vol}(L_{e_{ij}} \cap C_i)}{\text{Vol}(L_{e_{ij}})} u_{e_{ij}} \right| \\ & + 6 \sum_{e_{ij}=1}^{ne} h_{e_{ij}} \sum_{\tau \in T_{ij}} \text{Area}(\partial C_i \cap \partial C_j \cap \tau) |u_i - u_j| \end{aligned} \quad (11)$$

$h_{e_{ij}} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$, and ne is the number of edges of the original finite element triangulation \mathcal{T}_h .

Marking strategy:

In this subsection, we introduce the maximum strategy to determine the set $\widetilde{\mathcal{T}}_h$ in the general adaptive algorithm.

Algorithm 1 (Maximum strategy) *This algorithm determines the set $\widetilde{\mathcal{T}}_h$.*

(a) *Given: a partition \mathcal{T}_h , error estimates $\eta_{e_{ij}}$ for the edges $e_{ij} \in \mathcal{T}_h$, and a threshold $\theta \in (0, 1)$.*

Sought: a subset $\widetilde{\mathcal{T}}_h$ of marked edges that should be refined.

(b) *Compute $\eta_{\mathcal{T}_h, \max} = \max_{e_{ij} \in \mathcal{T}_h} \eta_{e_{ij}}$.*

(c) *If $\eta_{e_{ij}} \geq \theta \eta_{\mathcal{T}_h, \max}$, then mark the edge e_{ij} for refinement and put it into the set $\widetilde{\mathcal{T}}_h = \{e_{ij} \in \mathcal{T}_h | \eta_{e_{ij}} \geq \theta \eta_{\mathcal{T}_h, \max}, \theta \in (0, 1)\}$.*

Refinement strategy:

The set of marked edges is examined, tetrahedron by tetrahedron, and additional edges are marked in an attempt to maintain the grid quality and to get a conforming mesh (see Fig. 2). The final set of marked edges results in tetrahedra with one edge, three edges on one face, or all six edges. A tetrahedron with all six marked edges is shown in Fig. 2. The mesh is then refined by inserting new nodes on the midpoints of the marked edges and reconnecting these nodes into new tetrahedra and boundary faces. For the last configura-

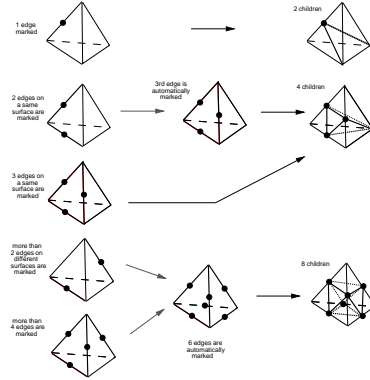


Fig. 2. Refinement strategies for a tetrahedron

tion, cutting off tetrahedra on all four corners leaves an octahedron which can be split into four tetrahedra by adding an inner edge connecting two diagonally opposite corners of the octahedron. To minimize distortions of the created tetrahedra, the shortest of the three possible inner diagonals should be chosen.

Tetrahedral mesh improvement using face and edge swapping

The edge and face swapping techniques effectively improve shape measures. The swapping algorithm minimizes a shape function, such as the aspect ratio, $AR = \frac{1}{3} \frac{\rho_{\text{out}}}{\rho_{\text{in}}}$, for tetrahedra [LJ94] as follows. where ρ_{out} is the tetrahedral circumsphere radius, ρ_{in} is the tetrahedral in-sphere radius. The value of the aspect ratio varies from 1, for an ideal element, to ∞ , for badly shaped elements. Reconnections of tetrahedra with undesirable shape measures are investigated and new local configurations for tetrahedra are selected with more desirable shape measures. Edges on boundary faces can also be swapped. Details of the way in which the face swapping can be implemented in practice can be found in [Joe95].

Boundary modification

The inserted boundary nodes may not be located on the surface geometry of the model to be simulated because they were inserted at the midpoints of existing edges. To address this issue, we have implemented a boundary curvature correction based on Hermite interpolation [Loh96].

5 Numerical experiments

5.1 Transonic NACA0012

A NACA0012 wing configuration has been employed to demonstrate the transonic shock capturing capability of the present adaptive grid solution method. The flow condition is at a free stream Mach number of 0.85 and incidence angle of 3.5 degrees. A reasonably coarse grid with a chordwise nearly uniform point distribution was generated to serve as the initial grid for adaptation. The grid, contains 76 125 points and 395 203 tetrahedral cells. An inviscid flow computation on this grid reveals the presence of a weak shock wave on the upper surface of the wing. The Mach and C_p contours are also illustrated in Fig. 3. As expected, the shock wave is diffused due to the grid coarseness and excessive numerical viscosity. Using the local remeshing procedures described earlier, four levels of adaptive refinement were performed in this case. The final grid contains 305 458 points and 1 675 668 tetrahedra. A threshold value $\theta = 0.5$ is used. Figure 3 shows the adapted grid and the corresponding Mach and C_p contours. As evidenced, the grid is efficiently refined at the shock location, which shows a sharp shock definition. Figure 3 illustrates the chordwise distributions of the surface pressure coefficient C_p for the initial coarse and adapted grids. As expected, there are significant differences between the adapted and the initial grid results. From the C_p distribution, it appears that the shock location of adapted grids is well captured compared to that of coarse grids. Furthermore, this example emphasizes the advantage of grid adaptation in providing more accurate flow solutions economically.

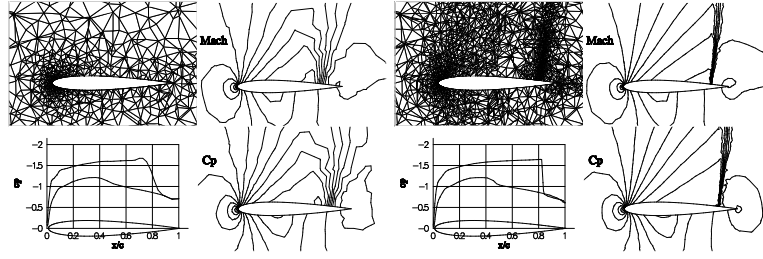


Fig. 3. Mach and Cp contours at $M_\infty = 0.85$ and $\alpha = 3.5^\circ$ for the initial grid and adapted grid

References

- [AVM97] P. ARMINJON, M.C. VIALLOAN AND A. MADRANE, *A finite volume extension of the Lax–Friedrichs and Nessyahu–Tadmor schemes for conservation laws on unstructured grids*, revised version with numerical applications, Int. J. of Comp. Fluid Dynamics 9, No. 1 (1997), pp. 1–22.
- [AMS01] P. ARMINJON, A. MADRANE AND A. ST-CYR, *Numerical simulation of 3D flows with a non-oscillatory central scheme on staggered unstructured tetrahedral grids*, in H. Freistuehler, G. Warnecke, Birkhauser, Eds., Proceedings of the Eighth International Conference on Hyperbolic Problems, Int. Series of Num Math. 140 (2001), pp. 59–68.
- [BJ89] T.J. BARTH AND D.C. JESPERSEN, (1989), *The design and application of upwind schemes on unstructured meshes*, AIAA Paper No. 89-0366, 27th Aerospace Sciences Meeting, January 9–12, 1989, Reno, Nevada.
- [HB98] A. HASELBACHER AND J. BLAZEK, *On the accurate and efficient discretisation of the Navier–Stokes equations on mixed grids*, AIAA J., 99–33552, (1998).
- [Joe95] B. JOE, *Construction of three-dimensional improved quality triangulations using local transformations*, SIAM J. Sci. Comput., 16, 1292–1307, 1995.
- [KO02] M. KÜTHER AND M. OHLBERGER, *Adaptive second-order central schemes on unstructured staggered grids*, in T.Y. Hou and E. Tadmor, Eds., Proceedings of the Ninth International Conference on Hyperbolic Problems held at Caltech, Pasadena CA, March 25–29, 2002, pp. 675–684, Springer Berlin/Heidelberg/New York, 2003.
- [LJ94] A. LIU AND B. JOE, *Relationship between tetrahedron shape measures*, BIT, 34 (1994), pp. 268–287.
- [Loh96] R. LÖHNER, *Regriidding Surface Triangulations*, J.C.P. 126, 1–10, 1996.
- [NT90] H. NESSYAHU AND E. TADMOR, *Non-oscillatory central differencing for hyperbolic conservation laws*, J. Comp. Phys., 87, No. 2 (1990), pp. 408–463.
- [Ven95] V. VENKATAKRISHNAN, *Convergence to steady state solutions of the Euler equations on unstructured grids with limiters*, J. Comp. Phys. 118 (1995), pp. 120–130.