

General Riemann Solvers and Dissipation Control for Third-Order Edge-Based Schemes for Metric-Based Adapted Simulations

C. Tarsia Morisco*, H. Nishikawa** and F. Alauzet*

Corresponding author: cosimo.tarsia-morisco@inria.fr

* GammaO, Equipe Inria-ONERA – Centre Inria de Saclay, Université Paris-Saclay, F-91120 Palaiseau, France.

** National Institute of Aerospace, Hampton, VA 23666, USA.

Abstract: One of the most widely used flux functions is the Roe solver. However, many practical simulations require more sophisticated flux functions that offer greater robustness. In this paper, we present a universal flux-correction method that enables a high-order edge-based Roe solver to be extended to any Riemann solver. The technique, initially developed for the third-order edge scheme, is shown here to be applicable to second-order low-dissipation schemes, which exhibit high-order behavior on smooth meshes. Third-order-accurate simulations using an implicit edge-based gradient method for anisotropic metric-based mesh adaptation are presented and analyzed for several Riemann solvers.

Keywords: Computational Fluid Dynamics, Edge-Based Numerical Schemes, High-order schemes, Anisotropic Mesh Adaptation.

1. Introduction

In the last decade, second-order schemes for anisotropic metric-based mesh adaptation have been shown to be robust and very accurate for industrial applications on complex geometries [1, 2, 3, 4, 5]. However, these schemes provide by definition first-order gradients and consequently engineering functionals involving such quantities (*e.g.*, skin friction, heat flux, *et*) are poorly captured. The challenge of the future is developing robust high-order schemes for highly anisotropic adapted meshes. In this scenario, the third-order edge-based method is one of the most efficient third-order accurate discretization algorithms for general tetrahedral meshes [6, 7, 8]. One main advantage is that it only requires a single numerical flux evaluation per edge just like the second-order method. In this way, the residuals can be computed efficiently in a loop over edges followed by a loop over boundary elements, and third-order accuracy is achieved with flux and solution reconstruction using second-order accurate gradient algorithms, not requiring second derivatives [6, 9], nor high-order curved meshes [7, 10]. The most common algorithm for second-order accurate gradients is the quadratic least-square (QLS) [6]. However, this method is generally known to destabilize a flow solver especially for very thin and highly curved meshes [9]. For inviscid flow problems, it has been shown that a quadratic implicit edge-based gradient (QIEBG) is a superior alternative and improves both the accuracy and iterative convergence of the third-order edge-based Euler solver [11]. Moreover, a robustness similar to second-order schemes was recently shown also on highly anisotropic adapted meshes, providing great convergence improvements with respect to second-order simulations for smooth and shock-dominated flows [12, 13]. Moreover, a clever dedicated implementation makes the QIEBG algorithm as convenient as QLSQ in terms of CPU cost [12]. All of that makes the former the current best candidate to handle third-order accurate solutions for industrial applications.

Up to now, the third-order edge-based method is easily applicable only to the Roe flux function or those in a similar form, *i.e.*, the averaged flux with a dissipation term, but not to others. This is because

rewriting general flux functions in a similar form is not necessarily a straightforward process. In this paper, we present a flux-correction form of the third-order edge-based method to enable us to directly call general flux functions. Specifically, we focus on its applications to adaptive tetrahedral meshes towards efficient and automated computational fluid dynamics (CFD) simulations with anisotropic metric-based mesh adaptation. Adapted mesh and solutions obtained for different approximate Riemann solvers will be compared and discussed.

2. Edge-based vertex-centered discretization for Euler Equations

Consider the following conservation law with a source term

$$\frac{\partial \mathbf{u}}{\partial t} + \text{div} \mathcal{F} = \mathbf{s} \quad (1)$$

where $\mathbf{u}$ is a vector of solution variables, $\mathcal{F}$ is a flux vector and $\mathbf{s}$ is a source (or forcing) term vector. In this paper, our target system is the Euler equations defined by the following flux and state vectors

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho \mathbf{v} \\ \rho E \end{bmatrix}, \quad \mathcal{F} = \begin{bmatrix} \rho \mathbf{v} \\ \rho \mathbf{v} \otimes \mathbf{v} + p \mathbf{I} \\ \rho \mathbf{v} H \end{bmatrix}, \quad (2)$$

where $\otimes$ denotes the dyadic product, $\mathbf{I}$ is the identity matrix, ρ is density, $\mathbf{v} = (u, v, w)^t$ is the vector of the velocity (the superscript t denotes transpose) with Cartesian components u , v and w , p is the static pressure, and $H = \gamma p / \{\rho(\gamma - 1)\} + \mathbf{v} \cdot \mathbf{v} / 2$ is the specific total enthalpy with $\gamma = 1.4$ (air), E is the specific total energy, and $E = H - p / \rho$.

Let $\mathcal{H}$ be a mesh of domain Ω , the vertex-centered formulation consists in associating with each vertex P_i of the mesh a control volume denoted C_i . The discretized domain Ω_h can be written as the union of the elements or the union of the finite volume cells: $\Omega_h = \bigcup_{i=1}^{N_K} K_i = \bigcup_{i=1}^{N_V} C_i$, where N_K is the number of elements and N_V the number of vertices. r accuracy ($p=1$) as

$$\frac{\partial \mathbf{u}_i}{\partial t} = \frac{\partial \mathbf{u}_i}{\partial t} C_i, \quad \Phi(\mathbf{u}_i, \mathbf{u}_j, \hat{\mathbf{n}}_{ij}) = \frac{1}{2} [\mathbf{f}(\mathbf{w}_L) + \mathbf{f}(\mathbf{w}_R)] - \frac{1}{2} |\mathbf{A}(\mathbf{w}_i, \mathbf{w}_j, \hat{\mathbf{n}}_{ij})| (\mathbf{w}_R - \mathbf{w}_L), \quad \tilde{\mathbf{s}}_i = \mathbf{s}_i C_i \quad (3)$$

with $\mathbf{f} = \mathcal{F} \hat{\mathbf{n}}_{ij}$, $|\mathbf{A}|$ is the dissipation matrix of the Roe flux computed without reconstructed solutions [14], and $\mathbf{w}_L$ and $\mathbf{w}_R$ are the linearly extrapolated values of the primitive variables $\mathbf{w} = (\rho, \mathbf{v}^t, p)$ obtained using the the following MUSCL reconstruction [15, 16, 17]

$$\mathbf{w}_L = \mathbf{w}_i + \frac{1}{2} \nabla \mathbf{w}_i \cdot \Delta \mathbf{r}_{ij}, \quad \mathbf{w}_R = \mathbf{w}_j - \frac{1}{2} \nabla \mathbf{w}_j \cdot \Delta \mathbf{r}_{ij}. \quad (4)$$

where $\Delta \mathbf{r}_{ij} = \mathbf{x}_j - \mathbf{x}_i$ and $\nabla \mathbf{w}_{i/j}$ are first-order accurate discrete operator gradients. Specifically, for standard edge-based schemes [18, 19] the gradient operator is evaluated at each vertex, while for β -schemes [20, 21] the gradient operator is defined for each edge. The V4 scheme [21] belongs to the latter and solution reconstruction is designed as follows

$$\nabla \mathbf{w}_i^{V4} \cdot \Delta \mathbf{r}_{ij} = (1 - \beta) \nabla \mathbf{w}_{ij}^C \cdot \Delta \mathbf{r}_{ij} + \beta \nabla \mathbf{w}_{ij}^U \cdot \Delta \mathbf{r}_{ij}, \quad (5)$$

$$\nabla \mathbf{w}_j^{V4} \cdot \Delta \mathbf{r}_{ij} = (1 - \beta) \nabla \mathbf{w}_{ij}^C \cdot \Delta \mathbf{r}_{ij} + \beta \nabla \mathbf{w}_{ij}^D \cdot \Delta \mathbf{r}_{ij}, \quad (6)$$

where the centered gradient $\nabla \mathbf{w}^C$ is defined along the edge $[i, j]$ by the following relation

$$\nabla \mathbf{w}_{ij}^C \cdot \Delta \mathbf{r}_{ij} = \mathbf{w}_j - \mathbf{w}_i,$$

and the upwind (resp. downwind) gradient $\nabla \mathbf{w}^U$ (resp. $\nabla \mathbf{w}^D$) is computed according to the definition of the upwind (resp. downwind) element of the edge $[i, j]$, denoted by K_{ij} (resp. K_{ji}). This element is the one where the edge facing P_i (resp. P_j) is intersected by the line $P_i P_j$. Note that unlike other edge-based methods, gradients are computed "on the fly" and not stored at nodes except at boundary nodes, where upwind/downwind gradients are not available.

The V4 scheme achieves third-order accuracy only for a linear advection equation on two-dimensional Friedrich-Keller triangular structured meshes when β is set at $1/3$.

3. Third-Order Edge-Based Discretization for steady Euler equations

Consider the third-order edge-based discretization method, which is known to achieve third-order accuracy on arbitrary simplex-element meshes without computing nor storing second derivatives. For steady problems, the spatial discretization at a vertex P_i results in

$$\mathbf{Res}_i^{\Delta x} = \sum_{j \in \mathcal{V}_i} \Phi(\mathbf{u}_i, \mathbf{u}_j, \hat{\mathbf{n}}_{ij}) |\mathbf{n}_{ij}| - \tilde{\mathbf{s}}_i = \mathbf{0}. \quad (7)$$

The left and right states and fluxes are linearly extrapolated as

$$\mathbf{w}_L = \mathbf{w}_i + \frac{1}{2} \nabla \mathbf{w}_i \cdot \Delta \mathbf{r}_{ij}, \quad \mathbf{w}_R = \mathbf{w}_j - \frac{1}{2} \nabla \mathbf{w}_j \cdot \Delta \mathbf{r}_{ij}, \quad (8)$$

$$\mathbf{f}_L = \mathbf{f}_i + \frac{1}{2} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right)_i \nabla \mathbf{w}_i \cdot \Delta \mathbf{r}_{ij}, \quad \mathbf{f}_R = \mathbf{f}_j - \frac{1}{2} \left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right)_j \nabla \mathbf{w}_j \cdot \Delta \mathbf{r}_{ij}, \quad (9)$$

where $(\nabla \mathbf{w}_i, \nabla \mathbf{w}_j)$ are gradients at nodes (P_i, P_j) . These fluxes should be used in the numerical flux in place of $\mathbf{f}(\mathbf{w}_L)$ and $\mathbf{f}(\mathbf{w}_R)$:

$$\Phi(\mathbf{u}_i, \mathbf{u}_j, \hat{\mathbf{n}}_{ij}) = \frac{1}{2} [\mathbf{f}_L + \mathbf{f}_R] - \frac{1}{2} |\mathbf{A}(\mathbf{w}_i, \mathbf{w}_j, \hat{\mathbf{n}}_{ij})| (\mathbf{w}_R - \mathbf{w}_L). \quad (10)$$

To obtain third-order accuracy, gradients must be computed with a method that is exact for quadratic functions [19]. In this paper, we compute them with the quadratic IEBG (QIEBG) method, which requires access to neighbors only at each relaxation step. Point-value evaluations of the source terms are sufficient up to second-order accuracy, but not sufficiently accurate to guarantee global third-order accuracy [22]. For this reason, the following accuracy-preserving quadrature formula [8] is used

$$\tilde{\mathbf{s}}_i = \frac{1}{2} \sum_{j \in \mathcal{V}_i} [c' \mathbf{s}_i + (2 - c') (\mathbf{s}_j - \nabla \mathbf{s}_i \cdot \Delta \mathbf{r}_{ij})] V_{ij}, \quad (11)$$

where $V_{ij} = (\mathbf{x}_j - \mathbf{x}_i) \cdot \mathbf{n}_{ij} / (2D)$ is the partial dual volume related to the edge $P_i P_j$, and $c' = (3D + 4) / (D + 2)$, with $D = 2$ (resp. 3) for triangular (resp. tetrahedral) meshes. The source gradient $\nabla \mathbf{s}_i$ is similarly computed with the QIEBG method.

4. Flux Correction Form for General Flux Functions

There exist many flux functions not formulated in the form (10). The HLLC flux is one example. To employ such a flux in the third-order edge-based method, we need to arrange it in the form (10) and then apply the flux and solution reconstructions. However, this may not always be a simple task, although often possible, for flux functions involving complex algorithms and physics. It is desirable to be able to directly employ any numerical flux in the third-order edge-based method. This is possible by modifying the numerical flux at each edge as

$$\Phi(\mathbf{w}_L, \mathbf{w}_R, \hat{\mathbf{n}}_{ij}) + \frac{1}{8} \left[\left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right)_i \nabla \mathbf{w}_i - \left(\frac{\partial \mathbf{f}}{\partial \mathbf{w}} \right)_j \nabla \mathbf{w}_j \right] \cdot \Delta \mathbf{r}_{ij}, \quad (12)$$

where $\Phi(\mathbf{w}_L, \mathbf{w}_R, \hat{\mathbf{n}}_{ij})$ is a general numerical flux function of two states $\mathbf{w}_L$ and $\mathbf{w}_R$. See Ref. [23] for a proof that third-order accuracy is preserved with the above flux-correction form of a numerical flux. The coefficient 1/8 is derived by generating a compatible second-order truncation error term that vanishes when combined with another second-order error arising from the source discretization (see Ref. [8] for details of the error cancellation mechanism). Note that we do not add a dissipation term explicitly as in the form (10) and assume that the numerical flux $\Phi(\mathbf{w}_L, \mathbf{w}_R, \hat{\mathbf{n}}_{ij})$ has dissipation. Note also that the two states, $\mathbf{w}_L$ and $\mathbf{w}_R$, must be computed by a quadratic extrapolation, not linear, in order to accurately approximate the flux at the edge midpoint. This can be achieved by the U-MUSCL scheme with $\kappa = 1/2$ [24]:

$$\mathbf{w}_L = \kappa \frac{\mathbf{w}_i + \mathbf{w}_j}{2} + (1 - \kappa) \left[\mathbf{w}_i + \frac{1}{2} \nabla \mathbf{w}_i \cdot \Delta \mathbf{r}_{ij} \right], \quad \mathbf{w}_R = \kappa \frac{\mathbf{w}_i + \mathbf{w}_j}{2} + (1 - \kappa) \left[\mathbf{w}_j - \frac{1}{2} \nabla \mathbf{w}_j \cdot \Delta \mathbf{r}_{ij} \right], \quad (13)$$

In this work, we employ the above flux-correction form to directly call numerical fluxes implemented in our code, which allows us to perform more practical comparisons with the second-order method.

4.1 Flux-correction for NLV4 scheme: truncation error analysis

The Non-Linear V4 (NLV4) Scheme is a new version of the second-order V4 scheme, that extends the supraconvergence of the former scheme on regular meshes (i.e. third-order accuracy for linear advection), to any nonlinear PDEs. This is achieved with a direct reconstruction of flux and solution and it is originally tailored for the Roe flux. The resulting scheme consists in

$$\Phi^{NLV4} = \frac{1}{2} [\mathbf{f}_L^{NLV4} + \mathbf{f}_R^{NLV4}] - \frac{1}{2} |\mathbf{A}(\mathbf{w}_i, \mathbf{w}_j, \hat{\mathbf{n}}_{ij})| (\mathbf{w}_R - \mathbf{w}_L),$$

where

$$\mathbf{f}_L^{NLV4} = \mathbf{f}_i + \frac{1}{2} \left(A \frac{\partial \mathbf{f}}{\partial \mathbf{w}} \Big|_i + B \frac{\partial \mathbf{f}}{\partial \mathbf{w}} \Big|_j \right) \nabla \mathbf{w}_i^{V4} \cdot \Delta \mathbf{r}_{ij}, \quad (14)$$

$$\mathbf{f}_R^{NLV4} = \mathbf{f}_j - \frac{1}{2} \left(A \frac{\partial \mathbf{f}}{\partial \mathbf{w}} \Big|_j + B \frac{\partial \mathbf{f}}{\partial \mathbf{w}} \Big|_i \right) \nabla \mathbf{w}_j^{V4} \cdot \Delta \mathbf{r}_{ij}, \quad (15)$$

with

$$A = \frac{1}{3} \left(\frac{1}{2\beta} + 1 \right) \quad B = \frac{1}{3} \left(\frac{1}{2\beta} - 1 \right). \quad (16)$$

In order to extend this scheme to general Riemann solvers, consider the same approach as in Ref. [23], and a 1D EDP with general nonlinear flux as

$$\text{div} \mathcal{F}(U) = \frac{dF(U)}{dx} = F_x = F_u U_x, \quad (17)$$

being F_u the flux jacobian.

The Taylor expansion, on an uniform 1D grid, of the 1D NLV4 flux at the mid-edge point P_{i+1} of the

edge $[i, i + 1]$ results in

$$\begin{aligned} \Phi^{NLV4} = & F_i \\ & + \frac{F_x \Delta x}{2} \\ & + \frac{[F_u U_{xx} + F_{uu}(U_x)^2] \Delta x^2}{12} \\ & + \frac{[30F_u U_{xxx} + 12F_{uu} U_x U_{xx} (6\beta + 1) + 18F_{uu}(U_{xx})^2] \Delta x^4}{864} \\ & + \frac{[36F_{uuu} U_{xx}(U_x)^2 + 6F_{uuuu}(U_x)^4] \Delta x^4}{864} + O(\Delta x^5) \Big] = 0. \end{aligned} \quad (18)$$

Consider the V4 reconstruction in Eqs.(5)-(6). It is easy to verify that Eq.(4) guarantees the exactness for a quadratic solution at the edge-mid point for $\beta = 1/4$. In this case, we can approximate

$$\begin{aligned} \Phi^{V4}(U_L, U_R) \approx F(U_{i+1/2}) = & F_i \\ & + \frac{F_x \Delta x}{2} \\ & + \frac{[F_u U_{xx} + F_{uu}(U_x)^2] \Delta x^2}{8} \\ & + \frac{[F_u U_{xxx} + 3F_{uu} U_x U_{xx} + F_{uuu}(U_x)^3] \Delta x^4}{48} \\ & + \frac{[F_u U_{xxxx} + 4F_{uu} U_x U_{xxx} + 3F_{uu}(U_{xx})^2 + 6F_{uuu}(U_x)^2 U_{xx} + F_{uuuu}(U_x)^4] \Delta x^4}{384} \\ & + O(\Delta x^5) \Big] = 0. \end{aligned} \quad (19)$$

It is possible to demonstrate that there is a coefficient $C = 1/16$, so that

$$\Phi^{NLV4} = F(U_L, U_R) + C \Delta F_{ij}^{NLV4} + O(\Delta x^4), \quad (20)$$

with

$$\Delta F_{ij}^{NLV4} = (A F_u + B F_u|_{i+1}) \nabla \mathbf{w}_i^{V4} \cdot \Delta \mathbf{r}_{ij} - (A F_u|_{i+1} + B F_u) \nabla \mathbf{w}_j^{V4} \cdot \Delta \mathbf{r}_{ij}. \quad (21)$$

4.2 Flux-correction for NLV4 scheme: discretization error analysis

To verify the accuracy of the flux-correction for the NLV4, we use the method of manufactured solutions [25, 26], in which an *ad hoc* function can be made an exact solution of the two-dimensional Euler equations by including a suitable forcing term $\mathbf{s}(\mathbf{x})$ as

$$\text{div}(\mathcal{F}^{inv}) = \mathbf{s}(\mathbf{x}). \quad (22)$$

As an exact solution, we consider the following functions:

$$\rho = 1.3 + 0.20 \exp(-1.05x + 1.10y + 1.15z), \quad (23)$$

$$u = 0.5 + 0.10 \exp(1.05x - 1.10y + 1.15z), \quad (24)$$

$$v = -0.1 + 0.02 \exp(1.50x + 3.10y - 1.50z), \quad (25)$$

$$p = 1.2 + 0.20 \exp(1.10x + 1.10y + 1.15z), \quad (26)$$

and compute the forcing term $\mathbf{s}(\mathbf{x})$ numerically at each vertex by evaluating the Euler equations with their exact gradients (analytically derived and evaluated at each vertex), as

$$\mathbf{s}(\mathbf{x}) = \begin{bmatrix} \nabla \cdot (\rho \mathbf{u}) \\ \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) + \nabla p \\ \nabla \cdot (\rho E + p) \mathbf{u} \end{bmatrix}. \quad (27)$$

Point-value evaluations of the source terms are sufficient for second-order schemes, but generally not so accurate to guarantee global third-order accuracy [27]. However, the NLV4 scheme produces $\mathcal{O}(h^3)$ -truncation error on regular grids. In this case, a point-value evaluation is exact and precise. Therefore, the spatial discretization at a vertex j results in

$$\frac{\mathbf{Res}_j}{V_j} = \frac{1}{V_j} \sum_{P_j \in \mathcal{V}(P_i)} \Phi^{NLV4}(\mathbf{w}_L, \mathbf{w}_R, \hat{\mathbf{n}}_{ij}) |\mathbf{n}_{ij}| - \mathbf{s}(\mathbf{x}_j). \quad (28)$$

A family of consistently-refined uniform triangular grids in a unit square domain is considered and the corresponding numerical solution $\mathbf{u}_j$ is computed (see Figure 1, on the left).

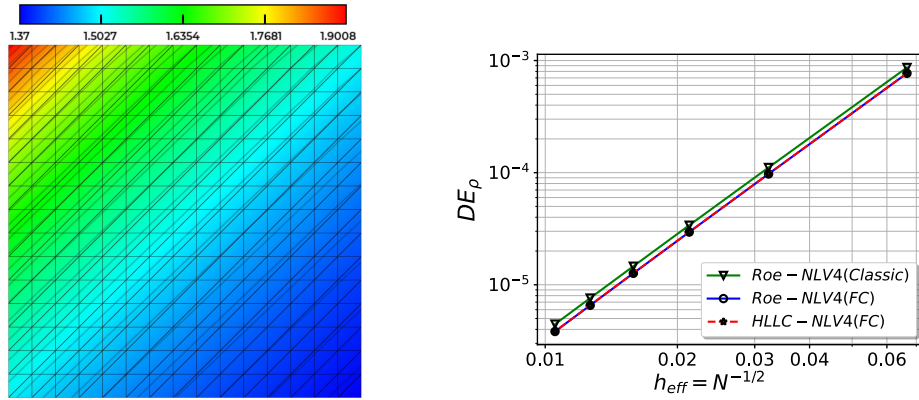


Figure 1: Accuracy verification for the Euler equations on 2D uniform triangular meshes. Left, mesh and numerical density field. Right, L^2 norm of the discretization error on density field: comparison between original version of NLV4 (Roe-flux tailored) and the NLV4 scheme with flux-correction. Accuracy trends at interior vertices for strong Dirichlet boundary conditions.

These meshes are generated by subdividing a Cartesian grid of $n \times n \times n$ nodes, where $n = [16, 32, 48, 64, 80, 96]$. Strong boundary conditions imposing the exact solution are used at boundaries. Each calculation is considered converged when residuals stagnate between 10^{-14} and 10^{-16} . The discretization error is measured in the following norm [28],

$$DE_p = \left(\frac{1}{N} \sum_j |u_j - u_j^{ex}|^p \right)^{\frac{1}{p}}, \quad (29)$$

where u_j and u_j^{ex} are the numerical and the exact solution at the vertex j respectively, N is the number of vertices and p is the order of the norm considered. Here, we employed $p = 2$, for which the above indicator corresponds to the root mean square. For consistently refined meshes, one can consider the following effective mesh size

$$h_{eff} = \frac{1}{N} \sum_{j=1}^N V_j^{1/3}. \quad (30)$$

The L^2 norm of the discretization error for the density field is plotted against the effective mesh size in Figure 1. In the picture on the right, we can see that the baseline version of the NLV4 for the roe-flux

produces third-order accuracy, as expected. In addition, the flux-correction permits to get third-order accuracy even when a different flux function (HLLC, red line) is used. Moreover, if one compares the original version of the NLV4 (green line) with the new version using Flux-Correction (blue line), the former appears to give a solution lightly more accurate. This is probably due two aspects: (i) given that the flux-correction is built to guarantee the quadratically exactness of the mid-edge point flux, it is more precise; (ii) the β required by the Flux-Correction is $1/4$, while it is a free parameter in the original version, and fixed at $1/3$ in this simulation.

5. Numerical result

The mesh-adaptive solution platform used in this work includes the solver `WOLF` [29], the anisotropic local remesher `FEFLO.A` [30] and the interpolator software `INTERPOL` [31, 32]. Each step of the adaptation process is detailed in Ref [33]. For each calculation, we selected a maximum number of 40 symmetric Gauss-Seidel (SGS) iterations and 2 orders of magnitude by which the relative linear system residual must be decreased at each iteration. When using the third-order EB scheme, as explained in Section 3, we need to solve an additional linear system for the gradients (i.e., QIEGB algorithm). In this case, we perform at most 10 gradient relaxations per implicit defect-correction iteration, with 5 orders of magnitude QIEGB linear system residual reduction.

In order to achieve high efficiency, the evolution of the local CFL is automatically managed by the flow solver, and increased with a geometric law from 0.1 to 1×10^5 [29].

5.1 Inviscid Steady Subsonic Flow over a 3D Smooth Bump

The first case considered is the steady subsonic flow over a smooth bump through a duct. The Mach number consider is $M = 0.3$. The smooth bump is defined by

$$z_b = 0.05 \sin^4 \left[\frac{\pi (10x - 3)}{9} \right], \quad x \in [0.3, 1.2],$$

and placed on the bottom boundary of a rectangular duct with size $(x, y, z) = [-0.25, 1.75] \times [0, 1] \times [0, 1]$. All boundary conditions are imposed weakly through the right state using the linearly exact formula (see Appendix B of Ref [34]): free stream boundary condition at the inflow; back pressure condition with unit static pressure ratio at the outflow; slip condition on all the other boundaries. As pointed out in Ref [35], any boundary state involving normals to boundaries (e.g., slip condition) requires sufficiently accurate nodal normals to preserve the designed accuracy. In this case, this is achieved by the knowledge of the exact tangent vector, as

$$\mathbf{n}_i^{BC} = (dz_b/dx, 0, 1),$$

and $\hat{\mathbf{n}}_i^{BC} = \mathbf{n}_i^{BC} / |\mathbf{n}_i^{BC}|$. A mesh-convergence study is run on eight anisotropic meshes with complexities:

$$C = \{ 5000, \quad 10\,000, \quad 20\,000, \quad 40\,000, \quad 80\,000, \quad 160\,000, \quad 320\,000 \}.$$

The final adapted mesh and pressure field solution is shown in Figure 3.

Convergence is considered achieved when non-linear residuals fall below $R_{tg} = 1 \times 10^{-11}$. In order to evaluate the accuracy, the following entropy error is used

$$\epsilon_e = \frac{p}{\rho^\gamma} - \frac{p_\infty}{\rho_\infty^\gamma} = \frac{\hat{p}}{\hat{\rho}^\gamma} - \frac{1}{\gamma M_\infty^2}, \quad \text{with} \quad \hat{p} = \frac{p}{\rho_\infty v_\infty^2}, \quad \text{and} \quad \hat{\rho} = \frac{\rho}{\rho_\infty},$$

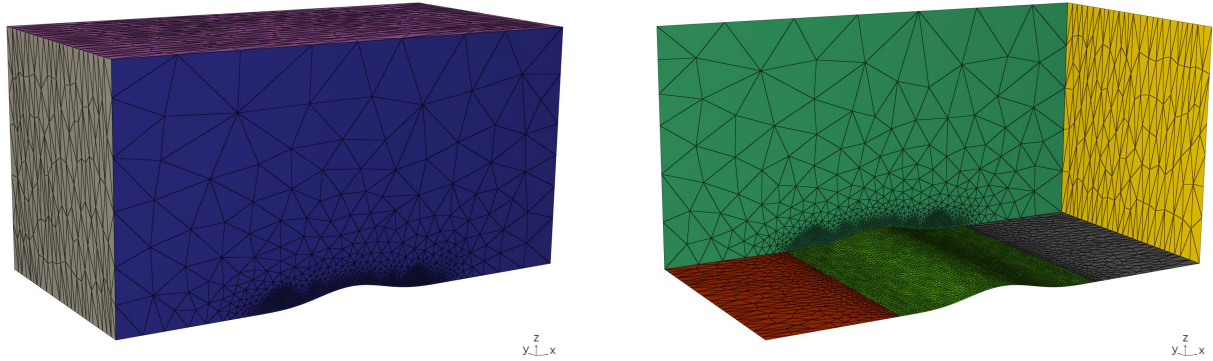


Figure 2: 3D square duct with smooth bump: Initial coarse mesh only composed of 10 026 vertices, 42 244 tetrahedra and 11 262 triangles on the boundaries. Exterior (left) and interior (right) view.

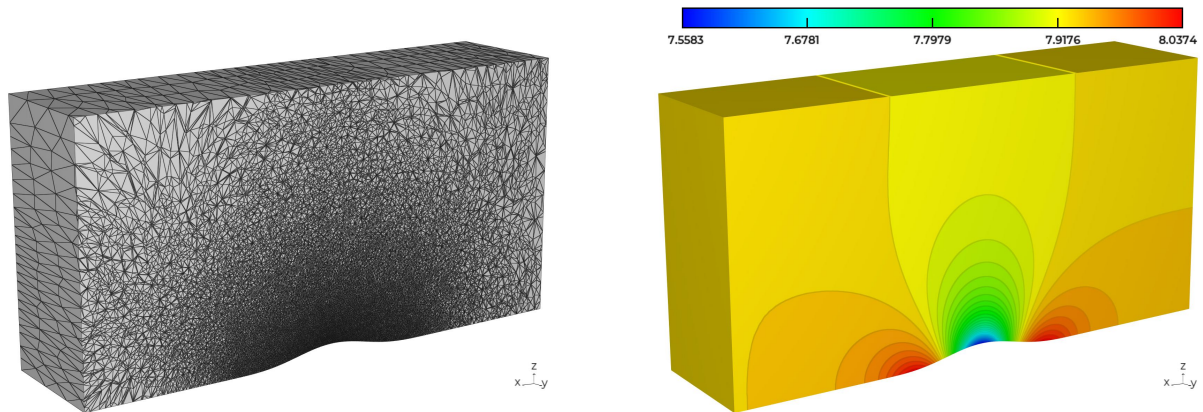


Figure 3: 3D subsonic flow over a smooth bump. *Multi-scale* anisotropic metric-based adaptation: adapted mesh composed of 350 000 vertices, 110 000 boundary triangles and 19 000 tetrahedra (left) and pressure field $\hat{p}$ (right).

and measured in the following norm

$$DE_p = \left(\frac{1}{N} \sum_j |\epsilon_{e,j}|^p \right)^{\frac{1}{p}}, \quad (31)$$

where N is the number of vertices and p is the order of the norm considered. Here, we employed $p = 2$, for which the above indicator corresponds to the root mean square. The L_2 norm of entropy error is plotted against the effective mesh size h_{eff} in Figure 4. We can see that both the original third-order EB scheme developed for the Roe flux (circles), and the new version employing Flux-correction using the HLLC flux (stars) exhibit third-order slope and very similar level of accuracy during the entire adaptive process. The same trend is observed both at interior and boundary vertices. As expected, the V4 scheme (diamonds) shows second-order mesh-convergence. In terms of entropy error, a third-order EB scheme produces a solution that is $4\times$ and $100\times$ more accurate on the first adaptive mesh and final adapted mesh, respectively.

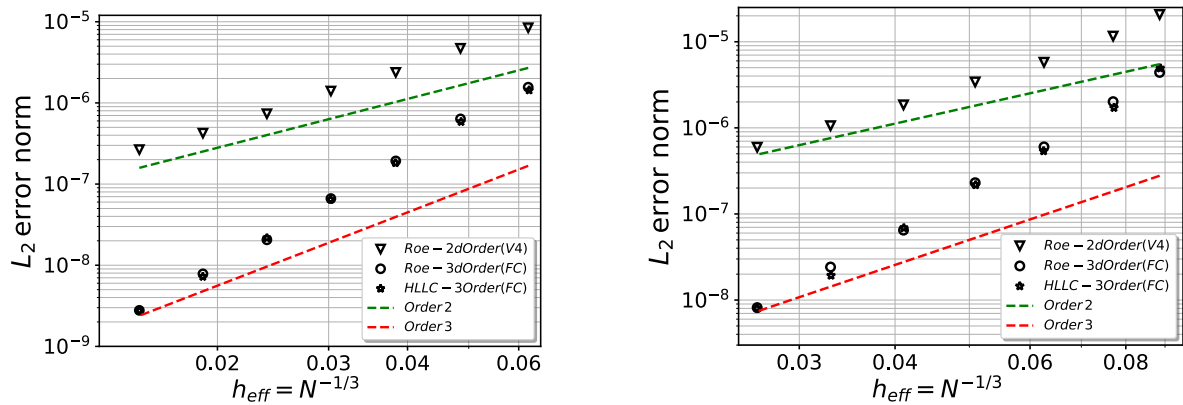


Figure 4: 3D subsonic flow over a smooth bump. Solutions obtained during the mesh-adaptive process. V4 and third-order EB schemes. L_2 norms of entropy error for interior points (left) and boundary vertices (right).

5.2 Inviscid subsonic flow past a 2D NACA0012 airfoil

The second test case here considered is the two-dimensional steady subsonic flow past the 2D NACA0012 airfoil at Angle of Attack $AoA = 2^\circ$ and free-stream Mach number $M_\infty = 0.63$. Free stream boundary condition is imposed at the outer boundary of the domain; slip condition on the airfoil surface. As previously done for the bump, boundary conditions are weakly enforced, and the normal vector in the slip condition right state is the exact one, based on the shape coefficient of the NACA0012 airfoil.

A mesh-convergence study is run using a *multi-scale* error estimate based on the L^2 norm of the Mach number field interpolation error on the following eight complexities:

$$C = \{ 5000, \quad 10\,000, \quad 20\,000, \quad 40\,000 \quad 80\,000, \quad 160\,000, \quad 320\,000, \quad 640\,000 \}.$$

The adapted mesh and Mach number field solution obtained at $C = 160\,K$ are depicted in Fig. 5.

The schemes considered in this calculation are: the baseline V4 scheme, the original third-order EB scheme tailored for the Roe flux function, and the new general third-order EB scheme. For the former and the latter, the HLLC flux function is used.

The evolution of the aerodynamic coefficients during the adaptive process is illustrated in Figure 6.

The green line indicates the results obtained with the baseline second-order V4 scheme, using the HLLC flux function. The red line indicates the coefficients obtained using the third-order EB with the HLLC flux function as well. For the sake of comparison, an additional mesh-adaptive process is run with the

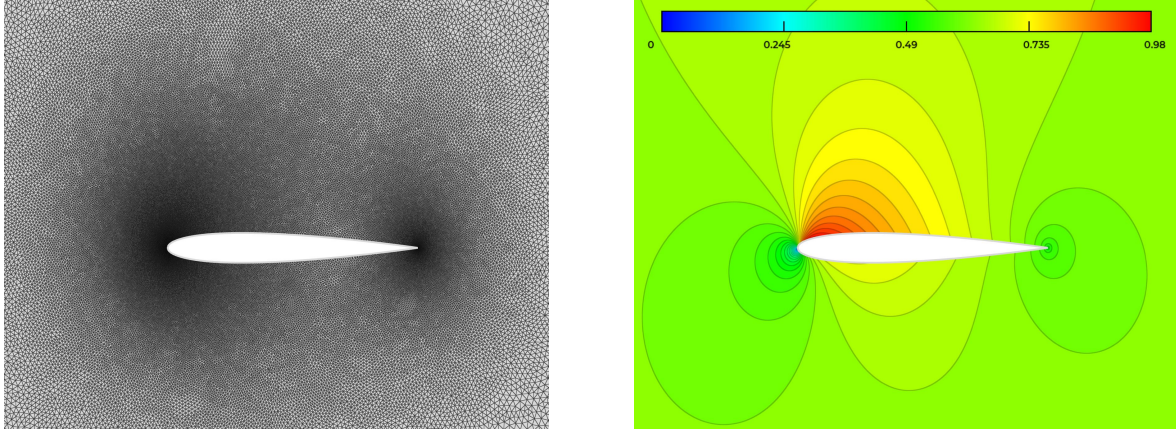


Figure 5: 2D subsonic flow past the NACA0012 airfoil, $M_\infty = 0.63$, $\text{AoA} = 2^\circ$. *Multi-scale* metric-based anisotropic mesh adaptation based on the L^2 norm of the Mach number interpolation error. Anisotropic adapted mesh (left) and Mach number field (right) obtained at $C = 160$ K.

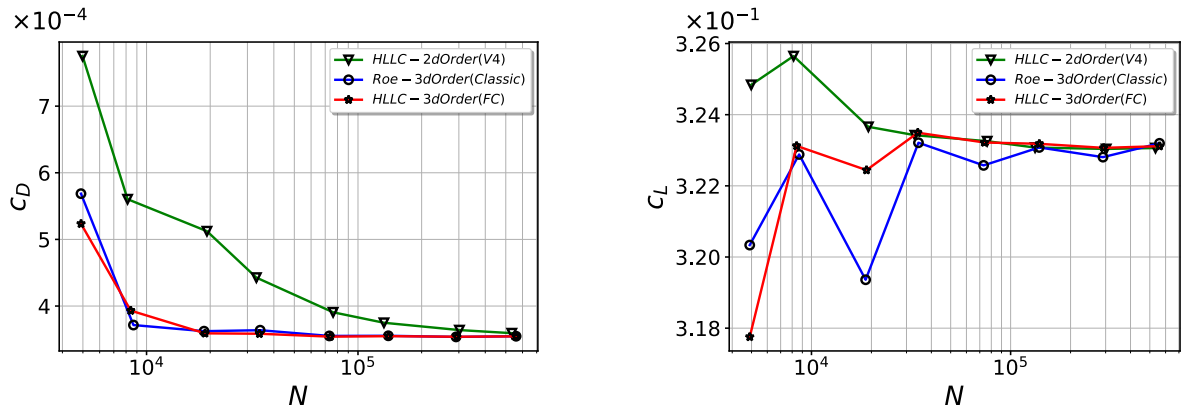


Figure 6: 2D subsonic flow past the NACA0012 airfoil, $M_\infty = 0.63$, $\text{AoA} = 2^\circ$. *Multi-scale* metric-based anisotropic mesh adaptation based on the L^2 norm of the Mach number interpolation error. Total drag c_D and total lift c_L coefficients against the number of vertices N . Comparison between: V4 scheme and third-order EB schemes.

original third-order EB scheme tailored for the Roe flux function (blue line). The first remark that we can draw is that, as expected, the third-order scheme (blue/red lines) converges faster than V4 (green line) for both drag and lift. In particular, with respect to drag, the third-order scheme produce a mesh-converged solution on the fifth adapted mesh ($C = 80\,000$), while the second-order V4 scheme shows an incipient mesh-convergence only on the last adaptive-mesh ($C = 640\,000$). As concerning the lift, convergence is very similar between V4 and third-order EB schemes. However, the new general version of the third-order EB scheme (red line) seems more stable in terms of mesh-convergence w.r.t. the original version developed for the Roe flux function (blue line).

6. Conclusions

In this paper we present a small step forward in the development of a third-order accurate method for robust metric-based anisotropic mesh adaptation. The original version of the third-order EB scheme was tailored for the Roe-flux function. This is maybe unpracticable when one wants to use more sophisticated Riemann solvers. Here, a flux-correction is demonstrated to produce third-order accuracy also in the case of general Riemann solvers with minimum modifications in the solver.

Flux-correction is shown to be applicable also to other edge-based schemes. Specifically, the new NLV4 scheme, which is third-order accurate only on smooth unstructured meshes for the Roe-solver, is extended to generic flux functions. This is demonstrated via a detailed truncation error and discretization error analyses.

The new version of third-order EB scheme including flux-correction is demonstrated to be more accurate and robust on metric-based anisotropic mesh adaptation.

References

- [1] J. Vanharen, A. Loseille, F. Alauzet, and M.A. Park. Near-field anisotropic mesh adaptivity for the Third AIAA Sonic Boom Workshop. *J. Aircr.*, 59(3):683–696, 2022.
- [2] F. Alauzet, T. Gauchery, M. Maunoury, C. Tarsia Morisco, and S. Mouton. 5th AIAA CFD High-Lift Prediction Workshop results using metric-based anisotropic mesh adaptation. In *AIAA SciTech 2025 Forum*, AIAA Paper 2025-0497, Orlando, FL, 2025.
- [3] F. Alauzet and P.R. Spalart. An improved smooth rotation correction for the Spalart-Allmaras turbulence model for better off-body vortex prediction and vortex–vortex interaction effects. *Int. J. Comput. Fluid Dyn.*, 38(2-3):265–297, 2024.
- [4] E. Guilbert, A. Remigi, M. Philit, and F. Alauzet. The ability of anisotropic mesh adaptation to capture complex physics on a transonic tandem compressor. In *Proc. J. Glob. Power Propuls*, GPPS-TC-2024-0076, Sep 2024.
- [5] F. Alauzet, E. Parente, and A. Remigi. High-fidelity film-cooled Rotor High Pressure Turbine simulation using metric-based anisotropic mesh adaptation. In *Proc. J. Glob. Power Propuls*, GPPS-TC-2024-0037, Sep 2024.
- [6] A. Katz and V. Sankaran. An efficient correction method to obtain a formally third-order accurate flow solver for node-centered unstructured grids. *J. Sci. Comput.*, 51:375–393, 2012.
- [7] H. Nishikawa. Accuracy-preserving boundary flux quadrature for finite-volume discretization on unstructured grids. *J. Comput. Phys.*, 281:518–555, 2015.
- [8] H. Nishikawa and Y. Liu. Accuracy-preserving source term quadrature for third-order edge-based discretization. *J. Comput. Phys.*, 344:595–622, 2017.
- [9] B. Diskin and J. L. Thomas. Effects of mesh regularity on accuracy of finite-volume schemes. In *Proc. of 50th AIAA Aerospace Sciences Meeting*, AIAA Paper 2012-0609, Nashville, Tennessee, 2012.
- [10] Y. Liu and H. Nishikawa. Third-order inviscid and second-order hyperbolic Navier-Stokes solvers for three-dimensional inviscid and viscous flows. In *46th AIAA Fluid Dynamics Conference*, AIAA

- Paper 2016-3969, Washington, D.C., 2016.
- [11] H. Nishikawa. Updates to implicit edge-based gradient methods. In *AIAA SciTech 2024 Forum*, AIAA Paper 2024-1748, Orlando, FL, 2024.
 - [12] C. Tarsia Morisco and F. Alauzet. Applications of implicit edge-based gradients to third-order edge-based scheme for adaptive tetrahedral grids. In *AIAA SciTech 2025 Forum*, AIAA Paper 2025-0302, Orlando, FL, 2025.
 - [13] C. Tarsia Morisco, F. Alauzet, and H. Nishikawa. Recent advancements in third-order edge-based scheme for anisotropic adaptive meshes. In *AIAA SciTech 2026 Forum*, AIAA Paper 2026-2513, Orlando, FL, 2026.
 - [14] Hiroaki Nishikawa. Robust numerical fluxes for unrealizable states. *J. Comput. Phys.*, 408:109244, 2020.
 - [15] Bram Van Leer. Towards the ultimate conservative difference scheme III. Upstream-centered finite-difference schemes for ideal compressible flow. *J. Comput. Phys.*, 23(3):263–275, 1977.
 - [16] Bram Van Leer. Towards the ultimate conservative difference scheme IV. A new approach to numerical convection. *J. Comput. Phys.*, 23(3):276–299, 1977.
 - [17] Bram Van Leer. Towards the ultimate conservative difference scheme V. A second-order sequel to godunov’s method. *J. Comput. Phys.*, 32(1):101–136, 1979.
 - [18] C. O. E. Burg. Higher order variable extrapolation for unstructured finite volume RANS flow solvers. In *17th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2005-4999, Orlando, FL, 2005.
 - [19] A. Katz and V. Sankaran. Mesh quality effects on the accuracy of CFD solutions on unstructured meshes. *J. Comput. Phys.*, 230:7670–7686, 2011.
 - [20] B. Stoufflet, J. Periaux, L. Fezoui, and A. Dervieux. Numerical simulation of 3-D hypersonic Euler flows around space vehicles using adapted finite element. In *AIAA 25th Aerospace Sciences Meeting*, AIAA Paper 1987-0560, Reno, NV, 1987.
 - [21] C. Debiez and A. Dervieux. Mixed-element-volume MUSCL methods with weak viscosity for steady and unsteady flow calculations. *Comput. Fluids*, 29(1):89–118, 2000.
 - [22] H. Nishikawa. A face-area-weighted centroid formula for finite-volume method that improves skewness and convergence on triangular grids. *J. Comput. Phys.*, 401:109001, 2020.
 - [23] H. Nishikawa. A flux-correction form of the third-order edge-based scheme for a general numerical flux function. arXiv:2602.01938v2 [math.NA], 18 March, 2026.
 - [24] E. Padway and H. Nishikawa. Resolving confusion over third-order accuracy of U-MUSCL. *AIAA J.*, 60(3):1415–1439, March 2022.
 - [25] W. L. Oberkampf and C. J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, 2010.
 - [26] P. J. Roache. *Fundamentals of Computational Fluid Dynamics*. Hermosa, 1998.
 - [27] H. Nishikawa and Y. Liu. Accuracy-preserving source term quadrature for third-order edge-based discretization. 2017. in review.
 - [28] B. Diskin and J. L. Thomas. Accuracy analysis for mixed-element finite-volume discretization schemes. *NIA Report No. 2007-08*, 2007.
 - [29] Frédéric Alauzet and Loic Frazza. Feature-based and goal-oriented anisotropic mesh adaptation for RANS applications in aeronautics and aerospace. *Journal of Computational Physics*, 439:110340, 2021.
 - [30] Adrien Loseille and Rainald Löhner. On 3d anisotropic local remeshing for surface, volume and boundary layers. In *Proceedings of the 18th international meshing roundtable*, pages 611–630, Salt Lake City, UT, 2009. Springer.
 - [31] Frédéric Alauzet and Michel Mehrenberger. P1-conservative solution interpolation on unstructured triangular meshes. *International Journal for Numerical Methods in Engineering*, 84:1552–1588, 2010.
 - [32] Frédéric Alauzet. A parallel matrix-free conservative solution interpolation on unstructured tetrahedral meshes. *Computer Methods in Applied Mechanics and Engineering*, 299:116–142, 2016.

- [33] Frederic Alauzet, Francesco Clerici, Adrien Loseille, Cosimo Tarsia Morisco, and Julien Vanharen. Some progress on CFD high lift prediction using metric-based anisotropic mesh adaptation. *AIAA Paper 2022-0388*, January 2022.
- [34] H. Nishikawa. Beyond interface gradient: A general principle for constructing diffusion schemes. In *40th AIAA Fluid Dynamics Conference and Exhibit*, AIAA Paper 2010-5093, Chicago, IL, 2010.
- [35] Y. Liu and H. Nishikawa. Third-order inviscid and second-order hyperbolic Navier-Stokes solvers for three-dimensional unsteady inviscid and viscous flows. In *55th AIAA Aerospace Sciences Meeting*, AIAA Paper 2017-0738, Grapevine, Texas, 2017.