doi: 10.5028/jatm.v9i3.707

Further Development and Application of High-Order Spectral Volume Methods for Compressible Flows

Carlos Breviglieri[1], João Luiz F Azevedo[2]

ABSTRACT: The present paper investigates the high-order spectral finite volume method with emphasis on applicability aspects for compressible flows. The intent is to improve the understanding and implementation of numerical techniques related to high-order unstructured grid schemes. In that regard, a hierarchical moment limiter and high-order mesh capability are developed for a 2-dimensional Euler spectral finite volume solver. The limiter formulation and geometry interpreter for high-order mesh generation are new contributions for the spectral finite volume method. Literature test cases are evaluated to assess the interaction of curved mesh, limiter and spatial reconstruction features of the spectral finite volume scheme. An order-of-accuracy study is presented along with steady and unsteady problems with strong shock waves and other discontinuities typical of compressible flows. Moreover, second, third and fourth-order spatial resolution analyses are explored and the spectral finite volume results are compared with those from different numerical methods.

KEYWORDS: Spectral volume method, Compressible flows, High-order reconstruction, Unstructured grids.

INTRODUCTION

High-order numerical schemes represent the natural extension of current Computational Fluid Dynamics (CFD) methods, which were developed over the past 30 years for aerospace simulations. The current generation methods are mostly 2nd-order accurate in space and have achieved a level of maturity and robustness desirable for everyday use in aeronautical engineering scenarios. Likewise, several complementary methods were developed for time integration, convergence acceleration, shock capturing and geometry flexibility. However, there are many problems that cannot be fully simulated using low-order methods, such as vortex dominated flows. This observation has motivated the CFD community to consider high-order methods for unstructured meshes. Application of discretization orders larger than 2nd-order has been an area of ongoing research for the last decades (Abgrall 1994; Barth and Frederickson 1990; Ollivier-Gooch 1997; Wang *et al.* 2013). The present paper is aligned with such effort.

The spectral finite volume (SFV) scheme, as proposed by Wang and co-workers (Liu *et al.* 2006; Sun *et al.* 2006; Wang 2002; Wang and Liu 2002, 2004; Wang *et al.* 2004), shares the functionality of a finite volume solver, copes with unstructured meshes and is an efficient alternative to other classes of 2-dimensional (2-D) high-order methods, for instance, the essentially non-oscillatory (ENO) and weighted ENO (WENO) families of schemes (Wolf and Azevedo 2006, 2007). The CFD solution resolution, or quality, is directly related to

1.Departamento de Ciência e Tecnologia Aeroespacial – Instituto Tecnológico de Aeronáutica – Divisão de Ciências da Computação – São José dos Campos/SP – Brazil. **2.**Departamento de Ciência e Tecnologia Aeroespacial – Instituto de Aeronáutica e Espaço – Divisão de Aerodinâmica – São José dos Campos/SP – Brazil.

Author for correspondence: João Luiz F Azevedo | Departamento de Ciência e Tecnologia Aeroespacial | Instituto de Aeronáutica e Espaço – Divisão de Aerodinâmica | Praça Marechal Eduardo Gomes, 50 – Vila das Acácias | CEP: 12.228-904 – São José dos Campos/SP – Brazil | E-mail: joaoluiz.azevedo@gmail.com

Received: May 06, 2016 | **Accepted:** Aug. 02, 2016

the spatial discretization and numerical methods involved. The geometric discretization typically consists of a pre-processing step in which a mesh is generated around the object of interest. Such mesh must be loaded in memory during computation to provide data on model geometry and domain boundaries.

The present study uses a 2-D, finite volume, unstructured CFD solver as a starting point for the development of a high-order method (Breviglieri *et al.* 2010a). The primary objective is to use the SFV numerical method to solve compressible flows at various speed regimes. The SFV method in 2-D allows one to assess the difficulties and advantages of high-order reconstruction in representative test cases, with proper discontinuity and boundary treatment techniques, in a manageable framework. Such techniques are, in principle, extendable to other high-order methods and also for 3-dimensional (3-D) problems.

Two complementary techniques for the high-order method are explored here, namely, the high-order boundary representation and limited reconstruction. In order to keep the high-order method competitive with the lower-order ones, a high-order representation of the geometric boundaries is necessary to reduce the total number of mesh cells. Such high-order boundary representation improves the numerical resolution and convergence aspects of the method, as indicated in Wang and Liu (2006). The use of limiters is necessary when the flow solution contains discontinuities, in order to remove spurious oscillations that may eventually lead to divergence of the numerical solution. Previous study on limiter implementations for high-order methods (Breviglieri *et al.* 2010b, 2008) was based on problem-dependent parameters to find out which cells need limiting, which can limit too many or too few elements of the solution. In the first case, the order of the method is seriously reduced and, in the second one, divergence can occur. To circumvent this drawback, the study here discussed uses a parameter-free generalized moment limiter (Yang and Wang 2009) based reconstruction to deal with discontinuities. The new limiter does not require input constants from the user, rendering the code more robust.

The numerical solver is implemented for the solution of the 2-D Euler equations in a cell-centered finite volume context for triangular meshes. The reported findings and tools are relevant for the long-term goal of numerical analysis over complex 3-D configurations. The study is also motivated by the authors' institute mission, which is to design and build satellite launchers and probes. As a consequence, the research addresses high Mach number flows in the context of high-order schemes.

GOVERNING EQUATIONS

The 2-D Euler equations are solved in their integral form as:

$$\frac{\partial}{\partial t}\int_V Q dV + \int_V (\nabla \cdot \vec{P}) dV = 0 \ , \tag{1}$$

where $\vec{P} = E\hat{\imath} + F\hat{\jmath}$. The application of the divergence theorem to Eq. 1 yields:

$$\frac{\partial}{\partial t}\int_V Q dV + \int_S (\vec{P} \cdot \vec{n}) dS = 0 \ . \tag{2}$$

The vector of conserved variables, Q, and the convective flux vectors, E and F, are given by:

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ e_t \end{Bmatrix}, E = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (e_t + p)u \end{Bmatrix}, F = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (e_t + p)v \end{Bmatrix} . \tag{3}$$

The standard CFD nomenclature is being used here. Hence, ρ is the density, u and v are the Cartesian velocity components in the x and y directions, respectively, p is the pressure, and e_t is the total energy per unit volume. The system is closed by the equation of state for a perfect gas:

$$p = (\gamma - 1)\left[e_t - \frac{1}{2}\rho(u^2 + v^2)\right] , \tag{4}$$

where the ratio of specific heats, γ, is set as 1.4 for all computations in this study. In the finite volume context, for stationary meshes, Eq. 2 can be rewritten for the i-th mesh cell as:

$$\frac{\partial Q_i}{\partial t} = -\frac{1}{V_i}\int_{S_i} (\vec{P} \cdot \vec{n}) dS \ , \tag{5}$$

where Q_i is the cell averaged value of Q at time t; V_i is the volume, or area in 2-D, of the i-th mesh element; S_i is the surface that surrounds the V_i volume.

NUMERICAL FORMULATION

SPATIAL DISCRETIZATION

In order to solve Eq. 5, a k-th-order approximation of the integral is computed. The computational domain, Ω, is discretized into N non-overlapping triangles such that:

$$\Omega = \bigcup_{i=1}^{N} SV_i \ . \tag{6}$$

These triangles are referred to as the spectral volumes (SVs).

The solution process involves the definition of proper initial and boundary conditions to the computational domain.

For a given order of spatial accuracy, k, using the SFV method, each SV_i cell must be further divided in sub-cells or control volumes (CVs).

$$N_k = \frac{(k+d-1)!}{(k-1)!\,d!} \tag{7}$$

where d is the physical dimension of the problem of interest. If one denotes by $CV_{i,j}$ the jth control volume of SV_i, the cell-averaged conserved variables, q, for $CV_{i,j}$ at time t are computed as:

$$q_{i,j} = \frac{1}{V_{i,j}} \int_{CV_{i,j}} q(x,y)dV, \tag{8}$$

where $V_{i,j}$ is the volume of $CV_{i,j}$.

Once the CV cell-averaged conserved variables are available for all CVs within SV_i, a polynomial $p_i(x,y) \in P^{k-1}$ of degree $k-1$ can be reconstructed to approximate each component of q as:

$$p_i(x,y) = q(x,y) + O(h^{k-1}),\ (x,y) \in SV_i, \tag{9}$$

where h represents the maximum edge length of all CVs within SV_i. The polynomial reconstruction process is discussed in detail in the following section. For now, it is sufficient to say that this high-order reconstruction is used to update the cell-averaged state variables at the next time step for all the CVs within the computational domain. Note that this polynomial approximation is valid within SV_i and the use of numerical fluxes are necessary across SV boundaries.

Integrating Eq. 5 in $CV_{i,j}$, one can obtain the integral form for the CV mean state variable:

$$\frac{dq_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{nf} \int_{A_r} (\vec{f} \cdot \vec{n}) dS = 0, \tag{10}$$

where $\vec{f} = E\hat{i} + F\hat{j}$ at the CV level; A_r represents the length of the r-th edge of $CV_{i,j}$; nf is the number of edges of $CV_{i,j}$. The boundary integral in Eq. (10) can be further discretized into the convective operator form:

$$C(q_{i,j}) \equiv \int_{S_{i,j}} (\vec{f} \cdot \vec{n}) dS = \sum_{r=1}^{nf} \int_{A_r} (\vec{f} \cdot \vec{n}) dS. \tag{11}$$

The integration on the right side of Eq. 11 can be performed numerically with a k-th order accurate Gaussian quadrature formula as:

$$C(q_{i,j}) = \sum_{r=1}^{nf} \sum_{q=1}^{nq} w_{rq} \vec{f}(q(x_{rq}, y_{rq})) \cdot \vec{n}_r A_r + \\ + O(A_r h^k)\ , \tag{12}$$

where (x_{rq}, y_{rq}) and w_{rq} are, respectively, the Gaussian quadrature point coordinates and the weights on the r-th edge of $CV_{i,j}$; nq = integer$[(k+1)/2]$ is the number of quadrature points required on the r-th edge for k-th order accuracy.

Due to the discontinuity of the reconstructed values of the conserved variables over SV boundaries, one must use a numerical flux function to approximate the flux values along the spectral volume boundaries. This means that $\vec{f}(q(x_{rq}, y_{rq}))$, which appears in Eq. 12, must come from an appropriate numerical flux if the r-th edge of $CV_{i,j}$ is also an external edge of SV_i. Moreover, at any moment during the simulation, one can compute the SV-averaged conserved variable vector, Q_i, for the i-th spectral volume, as:

$$Q_i = \frac{1}{V_i} \sum_{j=1}^{N_k} q_{i,j} V_{i,j}\ . \tag{13}$$

The calculation of the SV-averaged values is important at the end of the computation in order to analyze the high-order numerical solution at the original grid level. The average is also used to recover the conserved variable vectors for the SVs, which are required for the limited reconstruction process as discussed in the section "Limited Reconstruction".

The flux integration across CV boundaries that lie on the SV edges involves 2 discontinuous states, to the left and to the right of the edge. A numerical flux is used to solve for this discontinuous state. Note that the normal flux component needs to be continuous in order to maintain conservation. In the present study, the Roe flux difference splitting method (Roe 1981) with entropy fix is used to compute the numerical flux. As the method is based on one of the forms of a Riemann solver, it introduces the upwind effects and, hence, the artificial dissipation terms into the SFV method.

The semi-discrete SFV scheme for updating the values of conserved properties for the CVs can be written as:

$$\frac{dq_{i,j}}{dt} = -\frac{1}{V_{i,j}} C(q_{i,j}) \\ = -\frac{1}{V_{i,j}} \sum_{r=1}^{nf} \sum_{q=1}^{nq} w_{rq} f_{Roe}(q_L(x_{rq}, y_{rq}), q_R(x_{rq}, y_{rq}), \vec{n}_r) A_r\ , \tag{14}$$

where the summations in the right hand side of Eq. 14 are

the equivalent convective operator, $C(q_{i,j})$, for the j-th control volume of SV_i. The numerical flux can be expressed as:

$$\vec{f}\cdot\vec{n} = f_{roe}(q_L, q_R, \vec{n}) = \frac{1}{2}\left[\vec{f}(q_L)+\vec{f}(q_R)\right]\cdot\vec{n} - \frac{1}{2}|B|\,(q_R - q_L)\ . \tag{15}$$

Here, B is the Roe matrix (Roe 1981) in the edge-normal direction, which has 4 real eigenvalues, namely, $\lambda_1 = v_n - a$, $\lambda_2 = \lambda_3 = v_n$, $\lambda_4 = v_n + a$ where v_n is the velocity component normal to the edge and a is the speed of sound. Let T be the matrix composed of the right eigenvectors of B. Then, this matrix can be diagonalized as:

$$T^{-1}BT = \Lambda\ , \tag{16}$$

where Λ is the diagonal matrix composed of the eigenvalues of B, which can be written as:

$$\Lambda = diag\left[v_n - a, v_n, v_n, v_n + a\right]\ . \tag{17}$$

Hence, the $|B|$ matrix is formed as:

$$|B| = T\,|\Lambda|\,T^{-1}, \tag{18}$$

where Λ and T are calculated as a function of the Roe averaged properties (Roe 1981). Furthermore, $|\Lambda|$ is formed with the magnitude of the eigenvalues.

It is important to emphasize that some edges of the CVs, resulting from the partition of the SVs, lie inside the original SV in the region where the polynomial is continuous. For such edges, there is no need to compute numerical fluxes, as previously described. Instead, one uses analytical formulas for the flux computation and, hence, no artificial dissipation is required for such edges and the flux computation is extremely fast.

TEMPORAL DISCRETIZATION

In order to advance Eq. 14 in time, an explicit, multi-stage, Runge-Kutta time-stepping method is considered, unless otherwise stated. In particular, the 3-stage total variation diminishing (TVD) Runge-Kutta scheme is used, *i.e.,*

$$\begin{aligned} q_{i,j}^{(1)} &= q_{i,j}^{n} - \frac{\Delta t}{V_{i,j}}\,C(q_{i,j}^{n})\ , \\ q_{i,j}^{(2)} &= \alpha_1 q_{i,j}^{n} + \alpha_2\left[q_{i,j}^{(1)} - \frac{\Delta t}{V_{i,j}}\,C(q_{i,j}^{(1)})\right]\ , \\ q_{i,j}^{n+1} &= \alpha_3 q_{i,j}^{n} + \alpha_4\left[q_{i,j}^{(2)} - \frac{\Delta t}{V_{i,j}}\,C(q_{i,j}^{(2)})\right]\ . \end{aligned} \tag{19}$$

as discussed in Wolf and Azevedo (2006), where the n and $n+1$ superscripts denote, respectively, the values of the properties at the beginning and at the end of the n-th time step. The α coefficients are $\alpha_1 = 3/4$, $\alpha_2 = 1/4$, $\alpha_3 = 1/3$, and $\alpha_4 = 2/3$. The C operator represents the discretized convective operator as indicated in Eq. 14.

GENERAL FORMULATION FOR HIGH-ORDER RECONSTRUCTION

The evaluation of the conserved variables at the quadrature points is necessary in order to perform the flux integration over the mesh cell faces or mesh cell edges, in the 2-D case. These evaluations can be achieved by reconstructing conserved variables in terms of some base functions using the degrees-of-freedom (DOFs) within a SV. The present paper has carried out such reconstructions using polynomial base functions. Hence, let P^{k-1} denote the space of $(k-1)$-th degree polynomials in 2 dimensions. Then, the minimum dimension of the approximation space that allows P^{k-1} to be complete is defined by Eq. 7. In order to reconstruct q in P^{k-1}, it is necessary to partition the SV into N_k non-overlapping CVs, such that:

$$SV_i = \bigcup_{j=1}^{N_k} CV_{i,j}\ . \tag{20}$$

The reconstruction problem, for a given continuous function in SV_i and a suitable partition, can be stated as finding $p_{k-1} \in P^{k-1}$ such that:

$$\int_{CV_{i,j}} p_{k-1}(x,y)dS = \int_{CV_{i,j}} q(x,y)dS. \tag{21}$$

With a complete polynomial basis, $e_l\,(x, y) \in P^{k-1}$, it is possible to satisfy Eq. 21. Hence, p_{k-1} can be expressed as:

$$p_{k-1} = \sum_{\ell=1}^{N_k} b_\ell e_\ell(x,y), \tag{22}$$

where e is the basis function vector, $[e_1, \ldots, e_{Nk}]$; b is the reconstruction coefficient vector, $[b_1, \ldots, b_{Nk}]^T$. It should be further observed that p_{k-1} here denotes the $(k-1)$-th order polynomial for the standard SV cell, *i.e.,* mapped into the standard domain. Hence, the same polynomial can be used for similarly partitioned SVs. The substitution of Eq. 22 into Eq. 21 yields:

$$\frac{1}{V_{i,j}} \sum_{\ell=1}^{N_k} b_\ell \int_{CV_{i,j}} e_\ell(x,y)dS = q_{i,j} \ . \tag{23}$$

If $\overline{q}$ denotes the $[q_{i,1}, \ldots, q_{i,Nk}]^T$ column vector, Eq. 23 can be rewritten in matrix form as:

$$Sb = \overline{q} \ , \tag{24}$$

where the S reconstruction matrix is given by

$$S = \begin{bmatrix} \frac{1}{V_{i,1}} \int_{CV_{i,1}} e_1(x,y)dS & \cdots \\ \vdots & \cdots \\ \frac{1}{V_{i,N_k}} \int_{CV_{i,N_k}} e_1(x,y)dS & \cdots \end{bmatrix}$$

$$\begin{bmatrix} \cdots & \frac{1}{V_{i,1}} \int_{CV_{i,1}} e_{N_k}(x,y)dS \\ \cdots & \vdots \\ \cdots & \frac{1}{V_{i,N_k}} \int_{CV_{i,N_k}} e_{N_k}(x,y)dS \end{bmatrix} \ . \tag{25}$$

Hence, the reconstruction coefficients, b, can be obtained as:

$$b = S^{-1}\overline{q} \ , \tag{26}$$

provided that S is non-singular. Substituting Eq. 26 into Eq. 22, p_{k-1} is, then, expressed in terms of shape functions, $L = [L_1, \ldots, L_{Nk}]$, defined as $L = eS^{-1}$, such that one could write:

$$q(x,y) \ = \ p_{k-1}(x,y) \ = \ \sum_{j=1}^{N_k} L_j(x,y) q_{i,j} \ = \ L\overline{q} \ . \tag{27}$$

Equation 27 gives the value of the conserved state variable, $q(x, y)$, at any point within the SV and its boundaries, including the quadrature points. Note that $\overline{q}$ in the equation takes the form as a column vector, as presented in Eq. 24. The previous equation can be interpreted as an interpolation of a property at a point using a set of cell averaged values and the respective weights, which are set equal to the corresponding cardinal base value evaluated at that point.

Once the polynomial base functions, e_l, are chosen, the L shape functions are uniquely defined by the partition of the spectral volume. The shape and partition of the SV can be arbitrary, as long as the S matrix is non-singular. A detailed discussion of quality and stability analysis of the SFV method partitions can be found in Breviglieri *et al.* (2008). The partitions used in the present paper follow the orientations given in van den Abeele and Lacor (2007) and they are shown in Fig. 1.

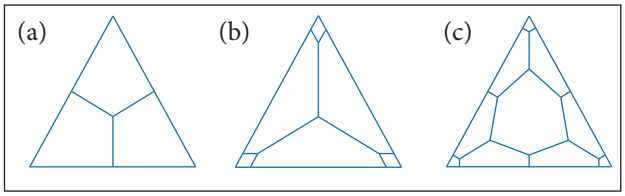

Figure 1. Triangular spectral volume partitions for (a) linear, (b) quadratic and (c) cubic reconstructions.

HIGH-ORDER BOUNDARY REPRESENTATION

From the formulation described so far, it is clear that any input mesh will be divided into a finer mesh and, in principle, render the computation more costly. In the standard 2nd-order finite volume scheme, the mesh boundaries are represented as line segments. This coarse approximation of the geometry results in a cluster of mesh nodes into highly-curved boundaries simply to represent the curved nature of it, for instance, in regions such as the leading edge of an airfoil.

If such approach is carried over to the SFV method, there is no gain in computational performance. The solution is to use high-order geometric elements in the mesh, effectively curving the edges of cells along the domain boundaries. For the present study, quadratic and cubic boundary representations are addressed, respectively, for the 3rd- and 4th-order SFV schemes and only for the elements located at wall boundaries.

In order to perform this representation, one can adopt isoparametric SV cells and map them to the boundary data. However, this particular SV will differ in the partition design from the other SVs. Thus, it will require a dedicated reconstruction and shape function values for property interpolations.

For a quadratic SV, one needs to specify 6 nodes, while, for a cubic SV, 10 nodes, as shown in Fig. 2. In order to perform the transformation, one can use the following relation to map a particular triangle of the mesh to the standard element, partition it there, and map it back to the physical domain to get the new nodes of the CV faces,

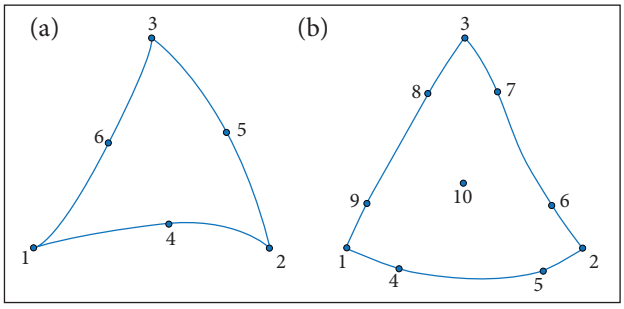

Figure 2. Quadratic (a) and cubic (b) isoparametric SV elements.

$$\vec{r} = \sum_{j=1}^{m} M_j(\xi,\eta)\vec{r}_j \tag{28}$$

where $\vec{r} = (x, y)$ and M_j are the shape functions for the transformation.

Once the "curved" SV is partitioned, the interpolation shape functions and the CV edge normals must be recalculated. Note that, typically, only 1 edge of the SV stands at a boundary, as depicted in Fig. 3.

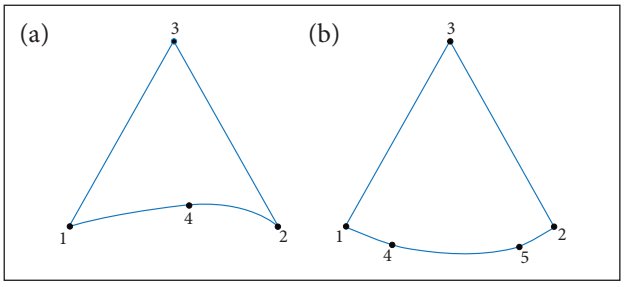

Figure 3. Simplified quadratic (a) and cubic (b) SVs with one curved boundary edge.

Therefore, one could use a simplified formulation for this specific edge. In such case, the mapping becomes

$$\begin{aligned} M_1(\xi,\eta) &= 1-3\xi+2\xi(\xi+\eta)-\eta\ , \\ M_2(\xi,\eta) &= -\xi+2\xi(\xi+\eta)\ , \\ M_3(\xi,\eta) &= \eta\ , \\ M_4(\xi,\eta) &= 4\xi(1-\xi-\eta)\ , \end{aligned} \tag{29}$$

for the quadratic SV, and

$$\begin{aligned} M_1(\xi,\eta) &= 1-\frac{9}{2}\xi-\eta+9\xi(\xi+\eta)-\frac{9}{2}\xi(\xi+\eta)^2\ , \\ M_2(\xi,\eta) &= \xi\left[1-\frac{9}{2}(\xi+\eta)+\frac{9}{2}(\xi+\eta)^2\right]\ , \\ M_3(\xi,\eta) &= \eta\ , \\ M_4(\xi,\eta) &= 9\xi\left[1-\frac{5}{2}(\xi+\eta)+\frac{3}{2}(\xi+\eta)^2\right]\ , \\ M_5(\xi,\eta) &= \xi\left[-\frac{9}{2}+18(\xi+\eta)-\frac{27}{2}(\xi+\eta)^2\right]\ , \end{aligned} \tag{30}$$

for the cubic SV. The simplified formulation is utilized throughout this study.

One particular challenge is to precisely obtain the mid-face node at the curved edge. The authors previously tried to work with neighborhood interpolation to determine its position, but such approach is not generally applied to "difficult" geometries. The interpolation could render wrong values if one of the neighbors is close to a sharp corner or if mirrored meshes are used, with the same x-coordinates for some mesh nodes.

A solution to this problem was to implement a geometry interpreter inside the solver. Within every simulation, the user provides the geometry prescribed by splines in the IGES format (Gruttke 1995). No matter how complicated the geometric construct is for a particular configuration, the present approach always obtains the correct node correspondence, as illustrated in Fig. 4.

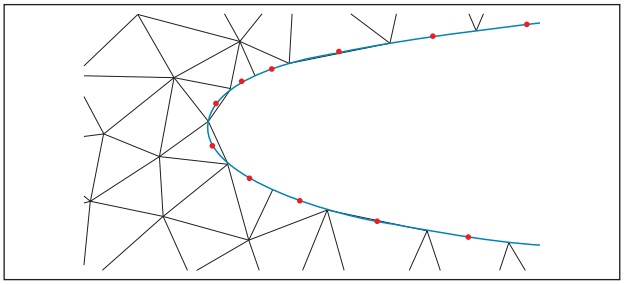

Figure 4. Mid-face nodes, in red, for quadratic boundary reconstruction computed from actual geometry definition.

The surface integral in the physical domain, Eq. 12, is also transformed to a surface integral for the standard element in the computational domain. Let the equation of the r-th face of $C_{i,j}$ in the standard SV be

$$\eta = \eta_r(\xi)\ , \quad \xi_{r1} < \xi < \xi_{r2}\ . \tag{31}$$

Then, along this face,

$$d\eta = \eta_r' d\xi\ . \tag{32}$$

Since $\eta = \eta_r(\xi)$ is a line segment in the standard element, η_r' is a constant, evaluated as $(\eta_2 - \eta_1)/(\xi_2 - \xi_1)$. Furthermore, along such a face,

$$\begin{aligned} dx &= \frac{\partial x}{\partial \xi}d\xi+\frac{\partial x}{\partial \eta}d\eta=\left(\frac{\partial x}{\partial \xi}+\frac{\partial x}{\partial \eta}\eta_r'\right)d\xi \\ dy &= \frac{\partial y}{\partial \xi}d\xi+\frac{\partial y}{\partial \eta}d\eta=\left(\frac{\partial y}{\partial \xi}+\frac{\partial y}{\partial \eta}\eta_r'\right)d\xi \\ dS &= (dx^2+dy^2)^{\frac{1}{2}}=\left[\left(\frac{dx}{d\xi}\right)^2+\left(\frac{dy}{d\xi}\right)^2\right]^{\frac{1}{2}}d\xi\ , \end{aligned} \tag{33}$$

where

$$\begin{aligned} \frac{dx}{d\xi} &= \frac{\partial x}{\partial \xi} + \frac{\partial x}{\partial \eta}\eta_r' \\ \frac{dy}{d\xi} &= \frac{\partial y}{\partial \xi} + \frac{\partial y}{\partial \eta}\eta_r' \ . \end{aligned} \tag{34}$$

The face unit normal vector, $\vec{n} = (n_x, n_y)$, is defined as:

$$\begin{aligned} n_x &= \frac{dy}{dS} \\ n_y &= -\frac{dx}{dS} \ . \end{aligned} \tag{35}$$

Hence, it is recomputed for the curved faces. It easily follows from Eq. 28, for instance, that

$$\begin{aligned} \frac{\partial x}{\partial \xi} &= [-3 + 2(\xi + \eta) + 2\xi]x_1 + [-1 + \\ &\quad + 2(\xi + \eta) + 2\xi]x_2 + [4(1 - \xi - \eta) - 4\xi]x_4 \\ \frac{\partial x}{\partial \eta} &= (-1 + 2\xi)x_1 + (2\xi)x_2 + x_3 - (4\xi)x_4 \end{aligned} \tag{36}$$

for the simplified quadratic SV. Analogous formulation is applied for the *y* derivatives and, also, for the simplified cubic SV. The surface integral, then, becomes

$$\begin{aligned} \int_S (\vec{P} \cdot \vec{n}) dS &= \int_S \vec{P} \cdot \left(\frac{dy}{dS}, -\frac{dx}{dS} \right) dS = \\ &= \int_{\xi_{r1}}^{\xi_{r2}} \vec{P} \cdot \left(\frac{dy}{d\xi}, -\frac{dx}{d\xi} \right) d\xi \ . \end{aligned} \tag{37}$$

This line integral in the standard element can be evaluated using the standard Gauss quadrature formulation:

$$\begin{aligned} &\int_{\xi_{r1}}^{\xi_{r2}} \vec{P} \cdot \left(\frac{dy}{d\xi}, -\frac{dx}{d\xi} \right) d\xi = \\ &= (\xi_{r2} - \xi_{r1}) \sum_{q=1}^{nq} w_q \, f_{Roe}(\xi_{rq}) \ , \end{aligned} \tag{38}$$

where w_q represents the Gauss quadrature weights; f_{Roe} is the numerical flux function.

The numerical integration, then, follows the discussion presented in section "Spatial Discretization".

LIMITED RECONSTRUCTION

For the Euler equations, in the presence of flow discontinuities, it is necessary to limit reconstructed properties at flux integration points to produce a converged simulation. The present limiter technique involves 2 stages. First, the solver must find out and mark "troubled cells" which are, in the 2nd stage, limited.

For the detection and limiting process, the limiter employs a Taylor series expansion for the reconstruction (van Altena 1999) with regard to the cell-averaged derivatives. The troubled cells are, then, limited in a hierarchical manner, *i.e.*, from the highest-order derivative to the lowest-order one. If the highest derivative is not limited, the original polynomial is preserved and so is the order of the method at the element level.

This limiter technique is capable of suppressing spurious oscillations near solution discontinuities without loss of accuracy at local extrema in smooth regions. Originally, this limiter methodology was developed for the spectral difference method in Yang and Wang (2009). In the present study, the formulation is extended for the SFV method. The extension here reported applies some ideas from previous research on ENO and WENO schemes (Wolf and Azevedo 2006) in the calculation of the limited properties. It should be emphasized that there are other approaches which implement a similar hierarchical moment limiter formulation for the SFV scheme (Xu *et al.* 2009), but with slight differences, in comparison to the present approach, on the calculation of the derivatives for the limited reconstruction.

Several markers (or sensors) were developed and employed for unstructured meshes over the past decades. For an in-depth review, the interested reader is referred to Qiu and Shu (2006). The limiter marker used in the present paper is termed Accuracy-Preserving TVD (AP-TVD) marker (Yang and Wang 2009). One important aspect is that the troubled-cell and limited properties are inherent to the SV element, and not to the CVs in which the flux calculations are performed. Once the SV element is confirmed as a troubled cell, its polynomial, based on CV cell-averaged variables, can no longer be used in any flux integration point, because the property function is no longer smooth within such SV. Hence, it is of utmost importance to limit as few SVs as possible.

To that end, the marker is designed to first check for the flux integration points in each SV and mark those that do not satisfy the monotonicity criterion. However, if an extremum is smooth, the first derivative of the solution, at such point, should be locally monotonic. Hence, in a second moment, and using derivative information, as described in the forthcoming discussion, the limiter sensor unmarks those SVs that have smooth local extrema and were unnecessarily marked as troubled cells. Therefore, for instance, for a quadratic reconstruction, the limiting process can be summarized in the following stages:

1. For a given spectral volume, SV_i, compute the minimum and maximum cell averages using a local stencil which includes its immediate face neighbors, *i.e.*,

$$Q_{min,i} = \min\left(Q_i\ ,\ \min_{1\leqslant r\leqslant nf} Q_r\right)$$
$$Q_{max,i} = \max\left(Q_i\ ,\ \max_{1\leqslant r\leqslant nf} Q_r\right) \tag{39}$$

2. The *i*-th cell is considered as a possible troubled cell if

$$p_i(x_{rq}, y_{rq}) > 1.001\, Q_{max,i} \quad \text{or} \quad p_i(x_{rq}, y_{rq}) < 0.999\, Q_{min,i} \tag{40}$$

where (x_{rq}, y_{rq}) identifies a quadrature point in the outer edges of SV_i. Note that p_i here denotes the reconstruction polynomial of order $k-1$ of the *i*-th SV cell, in the sense of Eq. 22. The 1.001 and 0.999 constants are not problem-dependent. They are simply used to overcome machine error, when comparing 2 real numbers, and to avoid the trivial case of when the solution is constant in the neighborhood of the spectral volume considered. This step is performed in order to check the monotonicity criterion over the SV cells for which a troubled reconstruction might exist.

3. Since the previous steps may flag more SVs than strictly necessary, the next operations attempt to unmark SVs in smooth regions of the flow. Hence, for a given marked spectral volume, a minmod TVD function is applied to verify whether the cell-averaged 2nd derivative is bounded by an estimate of the 2nd derivative obtained using cell-averaged 1st derivatives of the neighboring spectral volumes. Such test is performed as:
   - If the unit vector in the direction connecting the centroids of the *i*-th and *nb*-th cells is denoted $\vec{l} = l_x\hat{i} + l_y\hat{j}$, where *nb* indicates the face-neighbor of a marked SV_i, the 2nd derivative in such direction is defined as:

$$Q_{ll,i} = Q_{xx,i}l_x^2 + 2Q_{xy,i}l_xl_y + Q_{yy,i}l_y^2 \tag{41}$$

   - In a similar fashion, the first derivatives in the same $\vec{l}$ direction, for both *i*-th and *nb*-th cells, can be computed as:

$$Q_{l,i} = Q_{x,i}l_x + Q_{y,i}l_y\ ,$$
$$Q_{l,nb} = Q_{x,nb}l_x + Q_{y,nb}l_y\ ; \tag{42}$$

   - Another estimate of the second derivative, in the $\vec{l}$ direction, can be obtained as:

$$\tilde{Q}_{ll,i} = \frac{Q_{l,nb} - Q_{l,i}}{|\vec{r}_i - \vec{r}_{nb}|}\ , \tag{43}$$

   where $\vec{r}$ is the centroid coordinate vector.
   - A scalar limiter for this face is computed according to

$$\phi_{i,nb}^{(2)} = \text{minmod}\left(1, \frac{\tilde{Q}_{ll,i}}{Q_{ll,i}}\right)\ ; \tag{44}$$

   - The process is repeated for the other faces of SV_i, and the scalar limiter for the SV is the minimum among those computed for the faces, *i.e.*,

$$\phi_i^{(2)} = \min_{nb}\left(\phi_{i,nb}^{(2)}\right)\ ; \tag{45}$$

   - If $\phi_i^{(2)} = 1$, the second derivatives are bounded, as previously defined, and, hence, SV_i is actually in a smooth region of the flow. Therefore, SV_i is unmarked.

4. If the previous test is not satisfied, this means that the particular SV_i spectral volume should indeed be limited. In this case, the limiter for the first derivative reconstruction must also be computed. The calculation procedure follows the same approach as for the second derivatives, and it can be summarized as:
   - An estimate of the first derivative in the $\vec{l}$ direction is calculated as:

$$\tilde{Q}_{l,i} = \frac{Q_{nb} - Q_i}{|\vec{r}_i - \vec{r}_{nb}|}\ ; \tag{46}$$

   - Such estimate is compared to the cell-average first derivative in the $\vec{l}$ direction, computed according to Eq. 42, in order to obtain the scalar limiter for the face as:

$$\phi_{i,nb}^{(1)} = \text{minmod}\left(1, \frac{\tilde{Q}_{l,i}}{Q_{l,i}}\right)\ ; \tag{47}$$

   - As before, the scalar limiter for the cell is the minimum of those limiters computed for the faces, *i.e.*,

$$\phi_i^{(1)} = \min_{nb}\left(\phi_{i,nb}^{(1)}\right)\ . \tag{48}$$

The cell-averaged derivatives for the *i*-th cell, necessary to perform the previous calculations, are obtained by solving a quadratic least-squares reconstruction problem, for a 3rd-order scheme, or a cubic least-squares reconstruction problem, for a 4th-order scheme. For the quadratic reconstruction presented here, one would impose the mean conservation constraint in the first row of the least-square system and solve the following linear system:

$$\begin{bmatrix} M_{x^1y^0}|_{nb1} & M_{x^0y^1}|_{nb1} & M_{x^2y^0}|_{nb1} & M_{x^1y^1}|_{nb1} & M_{x^0y^2}|_{nb1} \\ M_{x^1y^0}|_{nb2} & M_{x^0y^1}|_{nb2} & M_{x^2y^0}|_{nb2} & M_{x^1y^1}|_{nb2} & M_{x^0y^2}|_{nb2} \\ M_{x^1y^0}|_{nb3} & M_{x^0y^1}|_{nb3} & M_{x^2y^0}|_{nb3} & M_{x^1y^1}|_{nb3} & M_{x^0y^2}|_{nb3} \\ M_{x^1y^0}|_{nb4} & M_{x^0y^1}|_{nb4} & M_{x^2y^0}|_{nb4} & M_{x^1y^1}|_{nb4} & M_{x^0y^2}|_{nb4} \\ M_{x^1y^0}|_{nb5} & M_{x^0y^1}|_{nb5} & M_{x^2y^0}|_{nb5} & M_{x^1y^1}|_{nb5} & M_{x^0y^2}|_{nb5} \end{bmatrix} \begin{Bmatrix} Q_x \\ Q_y \\ Q_{xx} \\ Q_{xy} \\ Q_{yy} \end{Bmatrix} = \begin{Bmatrix} Q_{nb1} - Q_i \\ Q_{nb2} - Q_i \\ Q_{nb3} - Q_i \\ Q_{nb4} - Q_i \\ Q_{nb5} - Q_i \end{Bmatrix} . \tag{49}$$

The matrix terms are the SV area moments and they can be computed, up to the desired order of accuracy, by numerical integration as:

$$M_{x^m y^n}|_i = \int_{SV} (x_{rq} - x_i)^m (y_{rq} - y_i)^n dV \ . \tag{50}$$

The SV area moments are computed during an initial stage of the numerical solver and kept in memory for efficiency. The *nb*1 to *nb*5 subscripts represent the neighbors of SV_i that form the computational stencil to compute the averaged derivatives. This stencil is determined by an ENO-based search, as in Wolf and Azevedo (2006), for the smoothest SV set. Since only a small number of SVs is selected for limited reconstruction, the overhead of this search does not adversely affect the overall performance of the scheme.

It is also important to observe that, considering all the information already available in a SFV method implementation, there are other possible approaches to compute the averaged derivatives. Actually, such approaches can be more computationally efficient than the one here adopted. The interested reader is referred, for instance, to the study presented in Yang and Wang (2009) for further details of one of such alternative approaches.

Finally, the quadratic limited polynomial, which is used in order to obtain property values at the quadrature points for a troubled SV_i spectral volume, is given by:

$$p_i^{\text{limited}}(x_{rq}, y_{rq}) = Q_i + \phi_i^{(1)} \left[\frac{1}{V_i} (Q_x M_x + Q_y M_y)_i \right] + \phi_i^{(2)} \left[\frac{1}{V_i} \left(\frac{1}{2} Q_{xx} M_{x^2} + Q_{xy} M_{xy} + \frac{1}{2} Q_{yy} M_{y^2} \right)_i \right] . \tag{51}$$

The area moments terms, *M*... in the previous equation, are computed as in Eq. 50 by replacing the *m* and *n* exponents with the appropriate order. The limited reconstruction is based on primitive variables $\{\rho, u, v, p\}^T$, instead of conserved variables, as one can readily check for non-physical values as, for instance, negative pressures (Bigarella and Azevedo 2007, 2012). Once these properties are available from the limited reconstruction, the vector of conserved variables is easily obtained to resume the numerical flux integration.

RESULTS

The results presented here attempt to validate the implementation of the data structure, temporal integration, numerical convergence stability and resolution of the SFV method. The overall performance of the method is compared with that of a 2nd-order monotonic upstream-centered scheme for conservation laws (MUSCL) scheme and a 3rd-order WENO scheme implementations. The latter uses an oscillation indicator proposed by Jiang and Shu (1996), with the modification of Friedrich (1998). The Roe numerical flux is also used with both schemes. Moreover, the geometric coefficients for the WENO reconstructions are computed in a pre-processing step and kept in memory during the computation. For more details on the MUSCL and WENO scheme formulations, the interested reader is referred to the study in Barth and Jespersen (1989) as well as Wolf and Azevedo (2006, 2007).

For the results here reported, density is made dimensionless with respect to the freestream condition, and pressure is made dimensionless with respect to the freestream density times the freestream speed of sound squared. All numerical simulations are carried out on a 16-core 3.2 GHz PC Intel64 architecture, with Linux OS. The code is written in Fortran 95 language, and the Intel Fortran Compiler® with optimization flags is used. For the performance comparisons, which are presented in this section, all residuals are normalized by the first iteration residue.

SCALAR CONSERVATION LAWS

The accuracy of the SFV method is tested for the linear scalar advection equation:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} = 0 \ , \tag{52}$$

for $-1 \le x \le 1$ and $u(x, 0) = u_0(x)$with periodic boundary condition at the domain extremes. For this setup, the initial condition is $u_0(x) = \sin(\pi x)$. The previously described 3rd-order TVD Runge-Kutta method is employed for time integration

with a Δt value of 10^{-4}, in order to make the discretization error time-step independent. Furthermore, the hierarchical limiter is considered in this test to verify if its marker formulation is able to ignore smooth solutions.

Table 1 shows the L_1 and L_∞ error norms produced using the SFV method with equidistant CVs for $t = 1$. One can observe that the 2nd-, 3rd- and 4th-order schemes are capable of achieving the expected order of accuracy even on coarse grids. However, the performance of the 5th-order method is not as good.

This behavior is related to the oscillatory pattern of the polynomial interpolation, due to the equidistant distribution, as previously observed by Wang and Liu (2004). As the grid is refined, the errors actually increase in both norms, which give negative orders of accuracy.

The same problem is simulated with the Gauss-Lobatto point distributions and the results are presented in Table 2. For the 2nd-order scheme, the Gauss-Lobatto point is actually in the middle of the domain, resulting in an equidistant CV partition, and therefore the corresponding results are not shown in Table 2.

One can note that all schemes are, now, able to achieve their expected order of accuracy. A comparison of the data in the tables indicates that the Gauss-Lobatto partitions yield smaller errors in both norms, when compared to the equidistant CV distributions. In the present study, *n*DOF represents the number of degrees of freedom for the problem, which, in this case, is given by $n\text{DOF} = N_k \times N$, where N_k is given by Eq. 7 and N is the number of SVs. The results shown in Tables 1 and 2 can

Table 1. Accuracy assessment of the 1-D SFV method for the linear scalar advection equation. Equidistant CV partition.

Order	N	$1/n$ DOF	L_∞ error	L_∞ order	L_1 error	L_1 order
2	40	1.25×10^{-2}	5.34×10^{-2}	–	3.29×10^{-2}	–
	80	6.25×10^{-3}	1.41×10^{-2}	1.92	8.72×10^{-3}	1.92
	160	3.13×10^{-3}	3.56×10^{-3}	1.98	2.24×10^{-3}	1.96
	320	1.56×10^{-3}	8.94×10^{-4}	1.99	5.65×10^{-4}	1.99
	640	7.81×10^{-4}	2.24×10^{-4}	2.00	1.42×10^{-4}	1.99
	1,280	3.91×10^{-4}	5.59×10^{-5}	2.00	3.55×10^{-5}	2.00
3	60	5.56×10^{-3}	4.12×10^{-3}	–	2.40×10^{-3}	–
	120	2.78×10^{-3}	5.31×10^{-4}	2.96	3.08×10^{-4}	2.96
	240	1.39×10^{-3}	6.67×10^{-5}	2.99	3.90×10^{-5}	2.98
	480	6.94×10^{-4}	8.34×10^{-6}	3.00	4.90×10^{-6}	2.99
	960	3.47×10^{-4}	1.04×10^{-6}	3.00	6.14×10^{-7}	3.00
	1,920	1.74×10^{-4}	1.30×10^{-7}	3.00	7.69×10^{-8}	3.00
4	40	6.25×10^{-3}	3.24×10^{-3}	–	1.64×10^{-3}	–
	80	3.13×10^{-3}	1.91×10^{-4}	4.08	1.13×10^{-4}	3.86
	160	1.56×10^{-3}	1.47×10^{-5}	3.70	7.04×10^{-6}	4.01
	320	7.81×10^{-4}	8.44×10^{-7}	4.12	4.30×10^{-7}	4.03
	640	3.91×10^{-4}	5.39×10^{-8}	3.97	2.71×10^{-8}	3.99
	1,280	1.95×10^{-4}	3.50×10^{-9}	3.95	1.69×10^{-9}	4.00
5	40	5.00×10^{-3}	1.27×10^{-3}	–	4.42×10^{-4}	–
	80	2.50×10^{-3}	7.55×10^{-5}	4.07	2.19×10^{-5}	4.33
	160	1.25×10^{-3}	8.15×10^{-6}	3.21	3.98×10^{-6}	2.46
	320	6.25×10^{-4}	1.06×10^{-5}	–0.38	4.58×10^{-6}	–0.20
	640	3.13×10^{-4}	4.61×10^{-4}	–5.44	1.80×10^{-4}	–5.29

be visualized in graphical form in Fig. 5. The results shown in this figure refer to error measured in the L_∞ norm.

The next test case addressed considers the linear advection of a Gaussian pulse. Hence, a pulse with a half width equal to 0.05 dimensionless units is used as initial condition:

$$u(x,0) = \exp\left[-\left(\frac{x-0.5}{0.05}\right)^2\right] . \quad (53)$$

The linear advection problem again assumes periodic boundary conditions. The simulation is carried out for various orders of accuracy, k = 1, 2, 3, 4 and 6, and up to a final time t = 2 dimensionless units. The Gauss-Lobatto point distribution is used for CV partitioning.

Figure 6 shows the analytical and numerical solution profiles for a grid with 100 SVs. The 1st-order simulation smeared the pulse so much that it is hardly recognizable. This is due to the extra amount of dissipation associated with such low-order scheme. The 2nd-order simulation retained the shape of the initial condition, but also smeared the pulse and produced an oscillatory behavior, which, in turn, is associated with the dispersion properties of the method. The 3rd-,

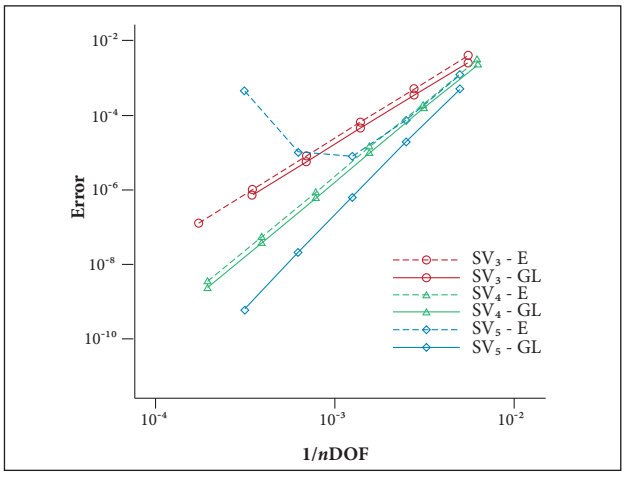

Figure 5. Accuracy assessment of the 1-D SFV method for the linear scalar advection equation. Effect of equidistant (E) and Gauss-Lobatto (GL) CV partition. Error shown in the L_∞ norm.

Table 2. Accuracy assessment of the 1-D SFV method for the linear scalar advection equation. Gauss-Lobatto CV partition.

Order	N	1/n DOF	L_∞ error	L_∞ order	L_1 error	L_1 order
3	60	5.56×10^{-3}	2.67×10^{-3}	–	1.24×10^{-3}	–
	120	2.78×10^{-3}	3.65×10^{-4}	2.87	1.61×10^{-4}	2.95
	240	1.39×10^{-3}	4.67×10^{-5}	2.97	2.05×10^{-5}	2.97
	480	6.94×10^{-4}	5.90×10^{-6}	2.98	2.59×10^{-6}	2.98
	960	3.47×10^{-4}	7.41×10^{-7}	2.99	3.23×10^{-7}	3.00
4	40	6.25×10^{-3}	2.26×10^{-3}	–	7.30×10^{-4}	–
	80	3.13×10^{-3}	1.60×10^{-4}	3.82	5.07×10^{-5}	3.85
	160	1.56×10^{-3}	9.72×10^{-6}	4.04	3.18×10^{-6}	3.99
	320	7.81×10^{-4}	6.15×10^{-7}	3.98	2.00×10^{-7}	3.99
	640	3.91×10^{-4}	3.85×10^{-8}	4.00	1.26×10^{-8}	3.99
	1,280	1.95×10^{-4}	2.41×10^{-9}	4.00	7.87×10^{-10}	4.00
5	40	5.00×10^{-3}	5.30×10^{-4}	–	1.46×10^{-4}	–
	80	2.50×10^{-3}	1.96×10^{-5}	4.76	4.58×10^{-6}	4.99
	160	1.25×10^{-3}	6.50×10^{-7}	4.91	1.49×10^{-7}	4.94
	320	6.25×10^{-4}	2.13×10^{-8}	4.93	4.91×10^{-9}	4.92
	640	3.13×10^{-4}	6.13×10^{-10}	5.12	1.57×10^{-10}	4.97
6	60	2.78×10^{-3}	1.28×10^{-5}	–	2.57×10^{-6}	–
	120	1.39×10^{-3}	1.88×10^{-7}	6.09	4.08×10^{-8}	5.98
	240	6.94×10^{-4}	2.98×10^{-9}	5.98	6.49×10^{-10}	5.97
	480	3.47×10^{-4}	4.51×10^{-11}	6.05	1.04×10^{-11}	5.96

4th- and 6th-order simulations yield good results, which are very similar to the analytical solution shown in the figure.

Table 3 shows the error and order measured from the numerical experiment. Figure 7 shows graphically the table data. One can observe that the design order is obtained considering the Gauss-Lobatto CV partition scheme. As the order of the method increases, the error is consistently reduced to machine zero. It is worth mentioning that the limiter is enabled for these simulations and it can be observed that it is not activated for such smooth solutions.

Another qualitative test case, which addresses a combination of smooth and discontinuous profiles, as in Krivodonova (2007), is also performed to check the SFV method capabilities. The advected signal consists of a smooth Gaussian pulse, a square pulse, a triangle and half an ellipse, which are defined in the domain $-1 \leq x \leq 1$.

Hence, the problem initial condition is defined as:

$$u(x,0) = \begin{cases} [G(x,\beta,z-\delta) + G(x,\beta,z+\delta) \\ \qquad +4G(x,\beta,z)]/6, & -0.8 \leq x \leq -0.6\ , \\ 1, & -0.4 \leq x \leq -0.2\ , \\ 1 - |10(x-0.1)|, & 0 \leq x \leq 0.2\ , \\ [F(x,\alpha,a-\delta) + F(x,\alpha,a+\delta) \\ \qquad +4F(x,\alpha,a)]/6, & 0.4 \leq x \leq 0.6\ , \\ 0, & \text{otherwise}\ , \end{cases} \tag{54}$$

with

$$G(x,\beta,z) = e^{-\beta(x-z)^2}\ ,$$
$$F(x,\alpha,a) = \sqrt{\max(1-\alpha^2(x-a)^2, 0)}\ ,$$

where $a = 0.5$, $z = 0.7$, $\delta = 0.005$, $\alpha = 10$ and $\beta = \log 2/36\ \delta^2$.

Once more, the Gauss-Lobatto distribution is used for CV partitioning and, for these tests, there are 100 SVs in the mesh. The same time discretization algorithm and Δt are used as in the previous case. The simulation is carried out with various orders of accuracy, for $k = 2$, 3, 4 and 6, up to a final time $t = 2$ dimensionless time units. The limiter

Table 3. Accuracy assessment of the 1-D SFV method for the Gaussian pulse problem.

Order	N	$1/n$ DOF	L_∞ error	L_∞ order
2	100	5.00×10^{-3}	3.44×10^{-1}	–
	200	2.50×10^{-3}	1.45×10^{-1}	1.25
	400	1.25×10^{-3}	4.42×10^{-2}	1.71
	800	6.25×10^{-4}	9.81×10^{-3}	2.17
3	100	3.33×10^{-3}	1.08×10^{-2}	–
	200	1.67×10^{-3}	9.17×10^{-4}	3.56
	400	8.33×10^{-4}	6.50×10^{-5}	3.82
	800	4.71×10^{-4}	5.31×10^{-6}	3.61
4	100	2.50×10^{-3}	2.48×10^{-4}	–
	200	1.25×10^{-3}	9.24×10^{-6}	4.74
	400	6.25×10^{-4}	5.29×10^{-7}	4.13
	800	3.13×10^{-4}	3.25×10^{-8}	4.02
5	100	2.00×10^{-3}	6.39×10^{-5}	–
	200	1.00×10^{-3}	1.93×10^{-6}	5.05
	400	5.00×10^{-4}	6.23×10^{-8}	4.95
	800	2.50×10^{-4}	1.88×10^{-9}	5.05
6	100	1.43×10^{-3}	3.14×10^{-7}	–
	200	7.14×10^{-4}	2.32×10^{-9}	7.08
	400	3.57×10^{-4}	5.44×10^{-11}	5.41
	800	1.79×10^{-4}	1.99×10^{-11}	1.45

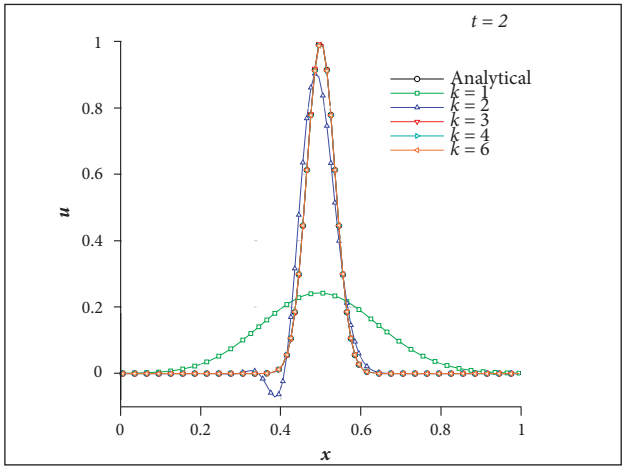

Figure 6. Simulation of a travelling Gaussian pulse with SFV schemes of various orders at $t = 2$ dimensionless units.

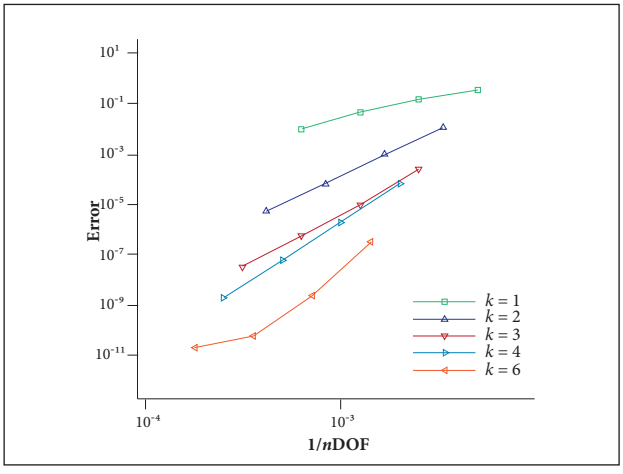

Figure 7. Accuracy assessment of the 1-D SFV Gaussian pulse. Error shown in the L_∞ norm.

is enabled to test its ability to mark non-smooth states of the solution and reduce the oscillations observed in high-order interpolations. The results can be seen in Figs. 8 and 9, plotted for the CV mesh.

The 2nd-order case retains some of the initial features, but it significantly smears the profiles. No oscillations are noticeable due to the use of the limiter formulation. The 3rd-order solution resolves the Gaussian pulse with improved accuracy but there is a noticeable difference in the base of the discontinuous profiles. The same behavior is observed for the 4th-order solution with a tendency to better approach the peak numerical values. However, the result distances itself from the analytical one near steep gradients in the solution. Once again, no oscillations are observed in the numerical solution.

The 6th-order result achieves the better approximation with the analytical data. The smooth profiles are hardly distinguishable from the real solution. The reconstruction is not able to reproduce the triangle and square wave profiles exactly

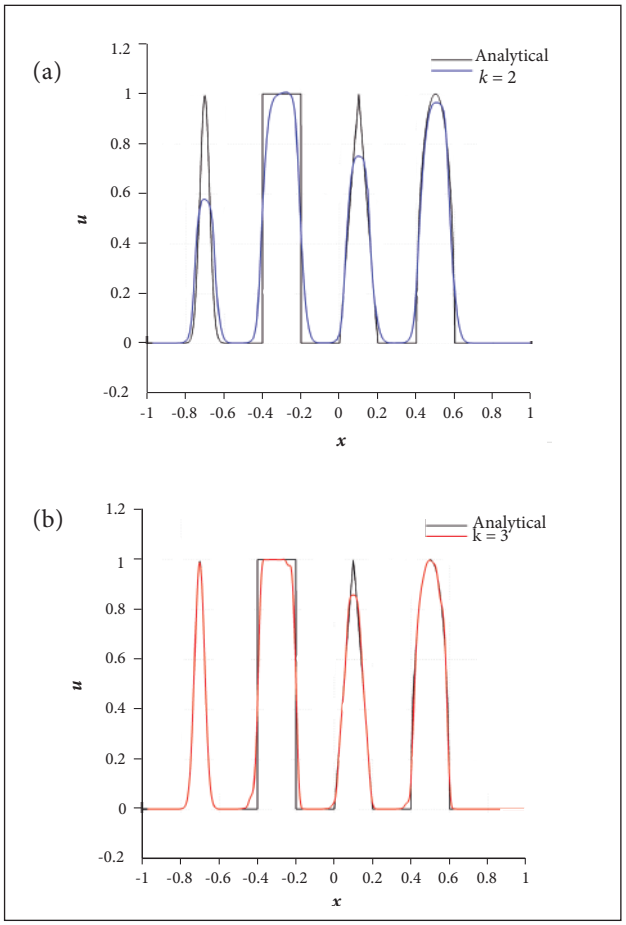

Figure 8. Simulation of travelling discontinuous profiles with (a) 2nd- and (b) 3rd-order SFV schemes at $t = 2$ dimensionless time units.

but yields an approximation with no apparent oscillations and a better resolution before and after such profiles. These results indicate that high-order schemes tend to better resolve the analytical profile without numerical instabilities.

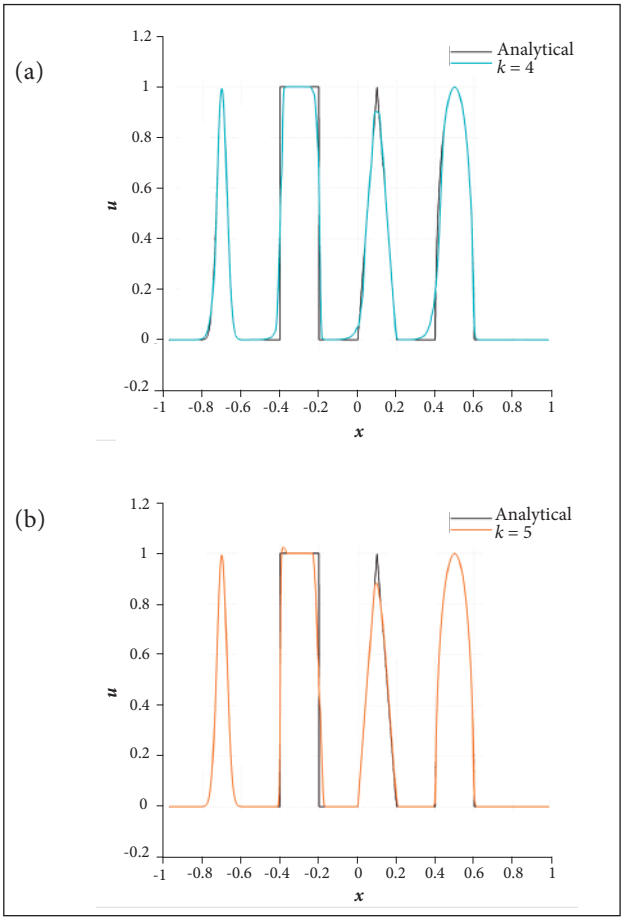

Figure 9. Simulation of travelling discontinuous profiles with (a) 4th- and (b) 6th-order SFV schemes at $t = 2$ dimensionless time units.

RINGLEB FLOW

The Ringleb flow simulation consists of an external flow, which has an analytical solution for the Euler equations derived with the hodograph transformation (Shapiro 1953). The analytical solution is used as initial condition for all simulations here discussed. The flow depends on the inverse of the stream function, κ, and the velocity magnitude, v_t. In the present simulations, these parameters are chosen as κ = 0.4 and 0.6, in order to define the lateral boundaries of the domain, and $v_t = 0.35$ to define the inlet and outlet boundaries. For such configuration, the test case represents an irrotational and isentropic flow around a symmetric blunt obstacle. An interesting property of the Ringleb test case is that transition

of flow regime, from supersonic to subsonic, for example, is shockless (Wang and Liu 2006).

In order to measure the order of the implemented SFV method, 4 meshes are considered for the grid refinement study, corresponding to 128; 512; 2,048 and 8,192 spectral volume cells. The analytical solution is computed for all meshes in order to measure how close the numerical results are to the exact solution. The error with respect to the analytical solution is computed using the L_1 and L_∞ norms of the density. Figure 10 shows the 2,048-cell grid and the Mach number contours computed in this grid with the 4th-order SFV method, using the corresponding high-order boundary representation.

The L_∞ norms of the error for the density values obtained in the converged solutions with the 3rd- and 4th-order SFV method are shown in Fig. 11 for the 4 meshes considered. The figure indicates that the theoretical orders of accuracy are actually recovered by the simulations with the high-order boundary representation. It should be pointed out that the same numerical test case was studied in Breviglieri *et al.* (2008), considering only a linear boundary representation. It was observed in this effort that the low-order boundary treatment causes a shock wave to develop close to the inner boundary, which, then, makes the limiter active. Eventually, the shock wave propagates and it causes the simulation to diverge.

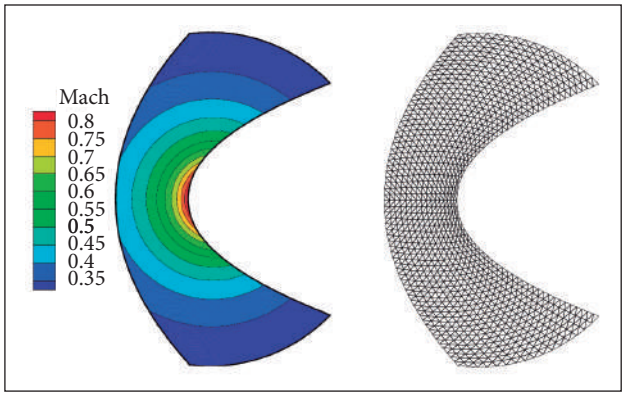

Figure 10. Computational mesh and Mach number contours calculated with the 4th-order SFV method for the 2,048-cell grid.

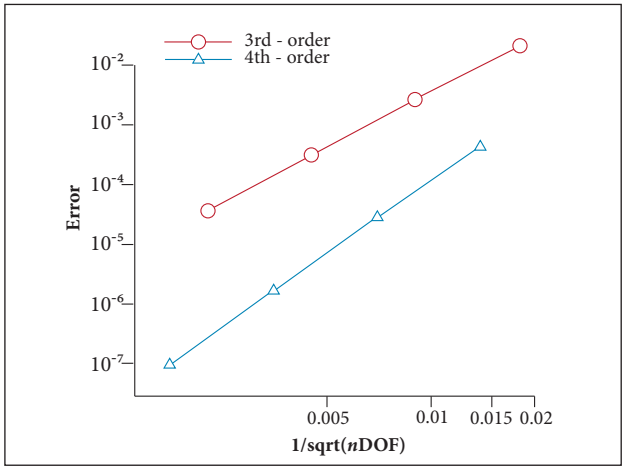

Figure 11. Ringleb flow error measurement with the 3rd- and 4th-order SFV method. Density error shown in the L_∞ norm.

In the present paper, however, which considers the higher-order boundary representation, reasonable results are always obtained for this test case, including the simulations with the 4th-order SFV method. As previously discussed, for the 3rd-order scheme, a quadratic polynomial is used to represent the SV edges which lie along the domain boundaries. In a similar fashion, for the 4th-order scheme, a cubic polynomial is employed instead, which is compatible with the internal polynomial order of each SV. Table 4 presents the L_1 and L_∞ error norms of the density for the present calculations with the high-order boundary representation. The table also shows the actual measured order of accuracy for the 3rd- and 4th-order SFV methods. The orders of accuracy in the results shown in Table 4 are calculated as indicated in Wang and Liu (2006). The actual orders of accuracy here obtained are in good agreement with those shown in the cited reference.

Table 4. Accuracy assessment of SFV method for the Ringleb flow test case.

Order	SV cells	L_1 error	L_1 order	L_∞ error	L_∞ order
3	128	2.41×10^{-2}	–	2.11×10^{-2}	–
	512	4.14×10^{-3}	2.54	2.65×10^{-3}	2.99
	2,048	6.27×10^{-4}	2.72	3.13×10^{-4}	3.08
	8,192	8.67×10^{-5}	2.85	3.60×10^{-5}	3.12
4	128	5.77×10^{-4}	–	4.37×10^{-4}	–
	512	6.48×10^{-5}	3.16	2.82×10^{-5}	3.95
	2,048	6.15×10^{-6}	3.39	1.68×10^{-6}	4.07
	8,192	6.87×10^{-7}	3.16	9.75×10^{-8}	4.11

NACA 0012 AIRFOIL

For the NACA 0012 airfoil simulations, 2 meshes are considered. The 1st simulations are performed on a mesh with 716 cells and 358 nodes, of which 40 nodes define the airfoil wall. This mesh is denoted as the coarse grid. On the other end, there is a fine mesh with 7,117 cells and 3,555 nodes, of which 116 nodes represent the

airfoil surface. Both of these meshes are *O*-type grids and they are presented in Fig. 12. The airfoil geometry is collapsed at the trailing edge. The far field boundary radius is 50 chord units. Differently from all other simulations in the present paper, the LU-SGS implicit time marching scheme, as discussed in Breviglieri *et al.* (2010b) and Parsani *et al.* (2010), is used for this test case. A CFL value of 1.0×10^6 is considered. The main objectives of the test case are to assess the SFV method accuracy and convergence for a transonic steady-state flow regime, as well as to provide further insight into the effects of a high-order boundary treatment.

The freestream flow replicates the conditions of the experimental data (McDevitt and Okuno 1985), that is, freestream Mach number of $M_\infty = 0.8$ and 0 deg. angle-of-attack. Simulations with the 2nd-, 3rd- and 4th-order SFV schemes are performed, along with the 1st-order Roe scheme. Figure 13 presents density contours for the coarse mesh, for the 3rd-order SFV method solution and its comparison to the 1st-order Roe scheme. The 3rd-order flow solution considers quadratic curved boundary reconstruction. The figure shows 30 evenly-spaced density contours, with values ranging from 0.8 to 1.6 in dimensionless density. A high-order post-processor tool would be necessary in order to accurately see and analyze these results. Nevertheless, one can already see that the 1st-order solution has clearly smeared out the shock wave that occurs in this flow.

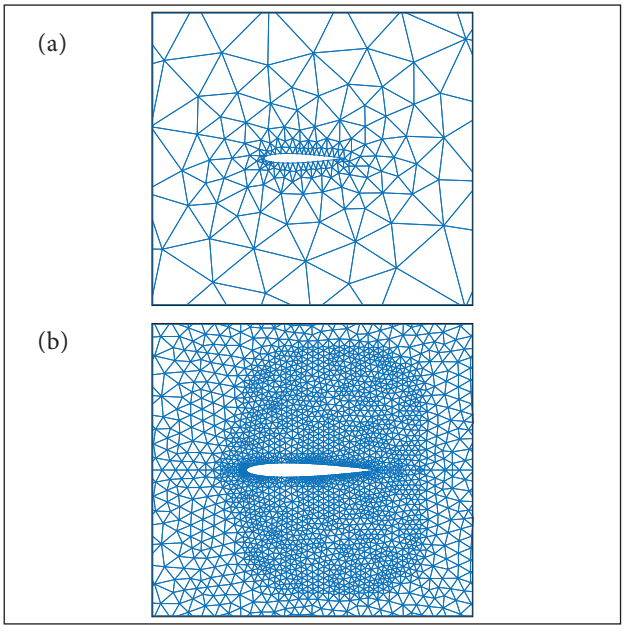

Figure 12. Computational meshes used in the NACA 0012 simulations (a) Coarse mesh and (b) Fine mesh.

Figure 14 shows the corresponding pressure coefficient (*Cp*) distributions for the same mesh and for the same methods, as compared to the experimental data for this flight condition. It is clear from the *Cp* distributions that the 1st-order scheme introduces too much dissipation and, essentially, smears out the shock wave, as already pointed out. The high-order *Cp* distribution, on the other hand, is remarkably close to the experimental results, particularly for the shock position and considering the crude geometric discretization.

These results illustrate the potential of high-order methods to ease the burden on the mesh generation process for

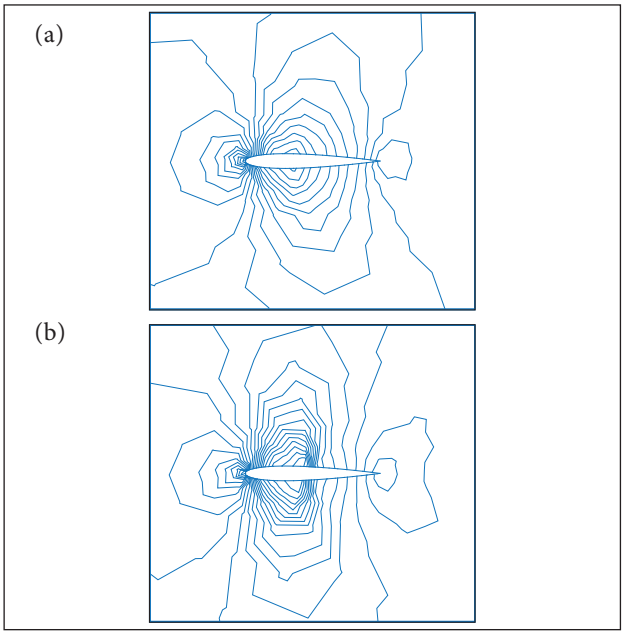

Figure 13. Evenly-spaced density contours on the coarse mesh: 30 contour lines are shown for dimensionless density values ranging from 0.8 to 1.6. (a) 1st-order Roe scheme; (b) 3rd-order SFV scheme

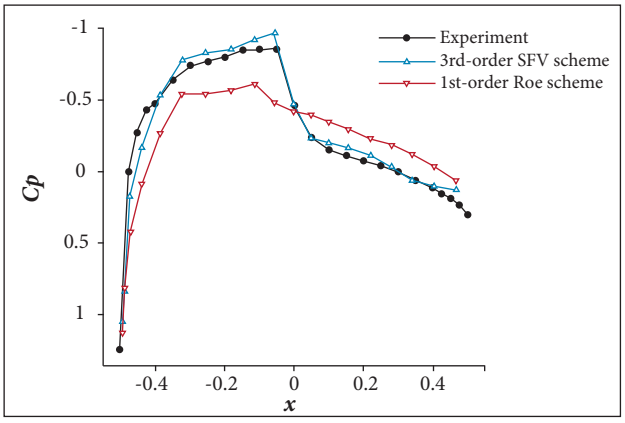

Figure 14. *Cp* distributions for the coarse mesh solutions.

aeronautical applications. One should observe, however, that the experimental results consider the presence of the boundary layer and the consequent shock-boundary layer interaction that necessarily occurs in the experiment. For the numerical solution, the shock is typically captured as a sharper discontinuity, as one should expect for an Euler simulation. In this case, however, the high-order solution seems to follow exactly the experimental data due to the extremely coarse mesh used.

The other set of results considers the fine mesh. Figure 15 presents density contours for the same range of dimensionless density variation, as in the coarse grid simulations, for the 1st-order Roe method and for the 2nd- and 3rd-order SFV schemes. One can observe that the high-order solutions present much more features and sharper flow gradients when compared to the 1st-order method, mainly in the vicinity of the shock wave. These results confirm the higher accuracy and resolution of the SFV methods, even though a limiter is used.

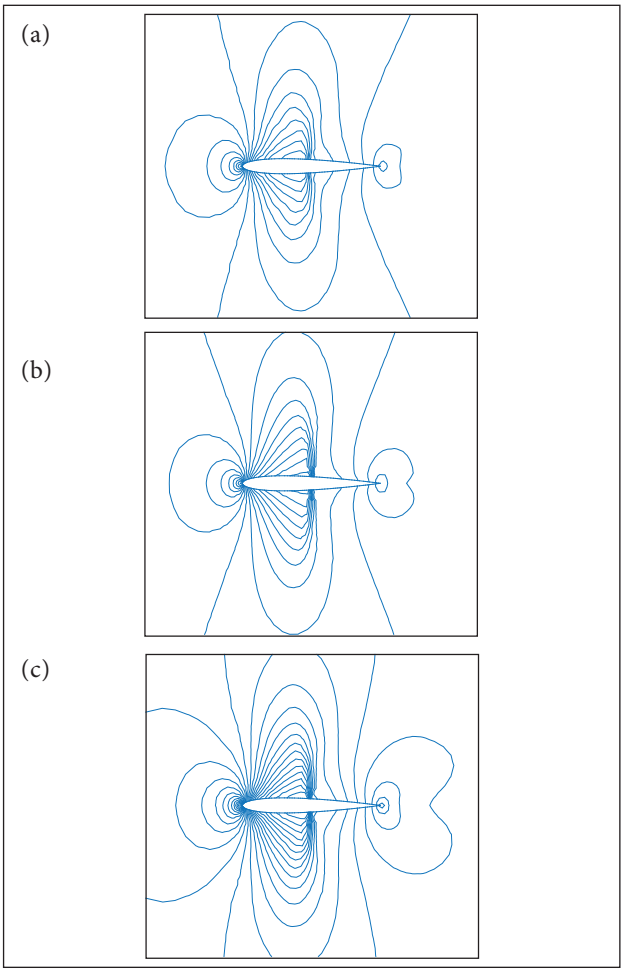

Figure 15. Evenly-spaced density contours on the fine mesh: 30 contours are shown for dimensionless density varying from 0.8 to ranging from 0.8 to 1.6. (a) 1st-order Roe scheme; (b) 2nd-order SVF scheme; (c) 3rd-order SVF scheme

Another relevant simulation is performed to assess the benefits of the curved boundary implementation, namely, the measure of entropy error ϵ_s levels at the airfoil boundary. Since the diffusive flux vectors are 0, there is no physical dissipation mechanism that produces heat in regions of smooth flow, away from shocks. If no external heat is added into the flow, then it is adiabatic. Hence, from the first law of thermodynamics, it follows that entropy, given by

$$s = C_v \ln\left(\frac{p}{\rho^\gamma}\right) \ , \tag{55}$$

is constant throughout the field if no shocks are present. Therefore, the entropy error ϵ_s, defined as

$$\epsilon_s = \frac{p}{p_\infty}\left(\frac{\rho_\infty}{\rho}\right)^\gamma - 1 \ , \tag{56}$$

is a good measure of the accuracy of a numerical solution of the Euler equations.

Figure 16 presents the entropy error generated by the 3rd-order SFV method with linear and curved boundary cells for the coarse mesh, which has only 40 cells to represent the complete airfoil geometry. As expected, the curved boundary approach is able to produce smaller error levels than the linear boundary edges. One can even note, from the figure, that at the $x = 0$ position there is an increase of entropy error due to the presence of the shock wave in this region. This, again, demonstrates that such extension indeed improves the overall accuracy of the high-order SFV method.

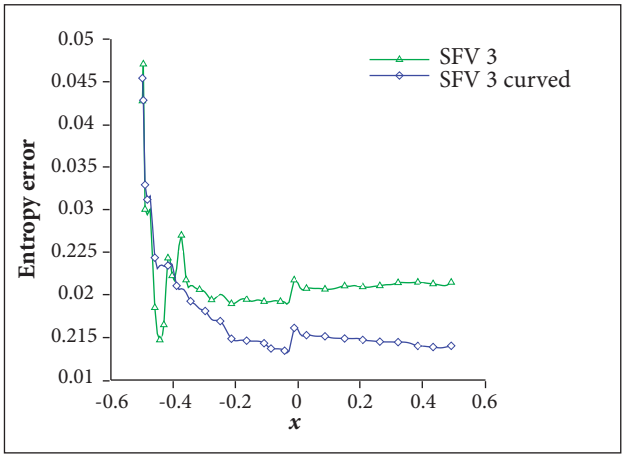

Figure 16. Entropy error for 3rd-order SFV method.

Figure 17 presents the convergence history for the simulations here considered in terms of the L_∞ norm of the continuity equation residue. The convergence history stalls for the SFV method, due to the presence of the limiter. This behavior is typical of solutions which use non-linear limiters (Venkatakrishnan 1995).

Nevertheless, the simulations have reached a steady level, based on the force coefficients. Moreover, Table 5 shows the relative costs of the different methods, normalized by the 2nd-order SFV method. The costs are measured on the fine mesh simulations and provide an overall estimate of the iteration cost associated with high-order solutions. One should observe that the 3rd-order WENO simulation is approximately 4 times more demanding than the 3rd-order SFV scheme. This increase in cost is associated with the dynamic stencil computation characteristic of the WENO solution, whereas the SFV method uses a static computational stencil throughout the simulation.

Table 5. Iteration cost estimates.

Method	Order	Limiter	Time
Roe	1	No	0.09
MUSCL	2	Yes	1.30
SFV	2	Yes	1.00
SFV	3	Yes	3.26
WENO	3	No	12.54

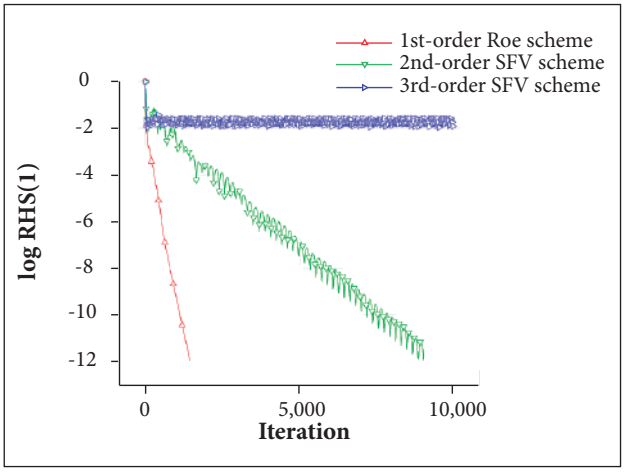

Figure 17. Convergence history for the Roe and SFV schemes.

FORWARD FACING STEP

This test case uses the same geometrical, boundary and flow parameters as the cases studied by Woodward and Colella (1984). The case was first proposed by Emery (1968) for evaluating finite-difference methods, but the results are not comparable to the later studies owing to their low resolution and parametric differences. The forward step test case is designed to resolve complex oblique shock reflections, pertinent to supersonic variable-geometry jet engine intakes. Due to computational constraints faced by Emery (1968), the test case was designed to be numerically simple to set up. The initial conditions are uniform throughout the domain, and the inlet boundary condition is supersonic. Such setup is actually difficult to perform experimentally, due to the unrealistic combination of initial and boundary conditions. However, quantitative results are available for a range of numerical methods (Woodward and Colella 1984), which makes this a good test case for validating the capabilities of the present flow solver, independently of the numerical method used.

The artificial nature of the test setup helps to evaluate the robustness of the spatial discretization algorithm combined with the limiter technique. Due to the strong shock reflection at the lower face of the step during the first few iterations, it is difficult to maintain positivity of pressure and density using various numerical schemes. Furthermore, the edge of the forward step is a singular point of the Prandtl-Meyer expansion fan generated by the flow over the step. The continuity and momentum equations can disobey the second law of thermodynamics through an expansion fan. This introduces numerical difficulty in the form of a nonphysical expansion shock, which at high Mach numbers and small cell sizes can yield negative pressures and densities in the code. The MUSCL reconstruction, for the 2nd-order Roe scheme, for instance, is not able to produce a solution as the simulation diverges after $t = 1.0$ dimensionless time units.

The 2-D configuration is 3 dimensionless length units long and 1 unit wide, with a step of 0.2 units high located at 0.6 units from the configuration inlet. The inflow and outflow boundary conditions are both supersonic, so the solver does not have to account for waves leaving the domain at the entrance boundary, or entering the domain at the exit boundary. Initially, the flow is at $M = 3$ everywhere. As stated in the seminal study of Woodward and Colella (1984), this "admittedly artificial" initial condition makes the problem very easy to set up. Since no physical constants or parameters, such as viscosity, are involved in the Euler equations, space and time units can be eliminated without the need for a dimensionless constant, and the flow is driven by pressure and density ratios. The simulation is run until $t = 4.0$ dimensionless time units, and the resulting shock wave pattern is examined. Two meshes are considered

for this simulation, a coarse and a fine one. These meshes are shown in Fig. 18. The coarse mesh has a characteristic length $h = 1/40$, and it has 8,272 cells and 4,134 nodes. The finer mesh has 49,304 cells and 24,650 nodes, as well as a characteristic length $h = 1/100$. Further visualization of the mesh refinement can be seen in Fig. 19, which shows both the coarse and fine meshes for the region of the step. This last figure allows for a better visualization of the level of mesh refinement in both cases.

For the present simulations, the 1st-order Roe method is considered along with the 2nd- and 3rd-order SFV methods. A total variation bounded (TVB) minmod limiter (Shu 1987) is considered in the present study for the 2nd-order SFV scheme in

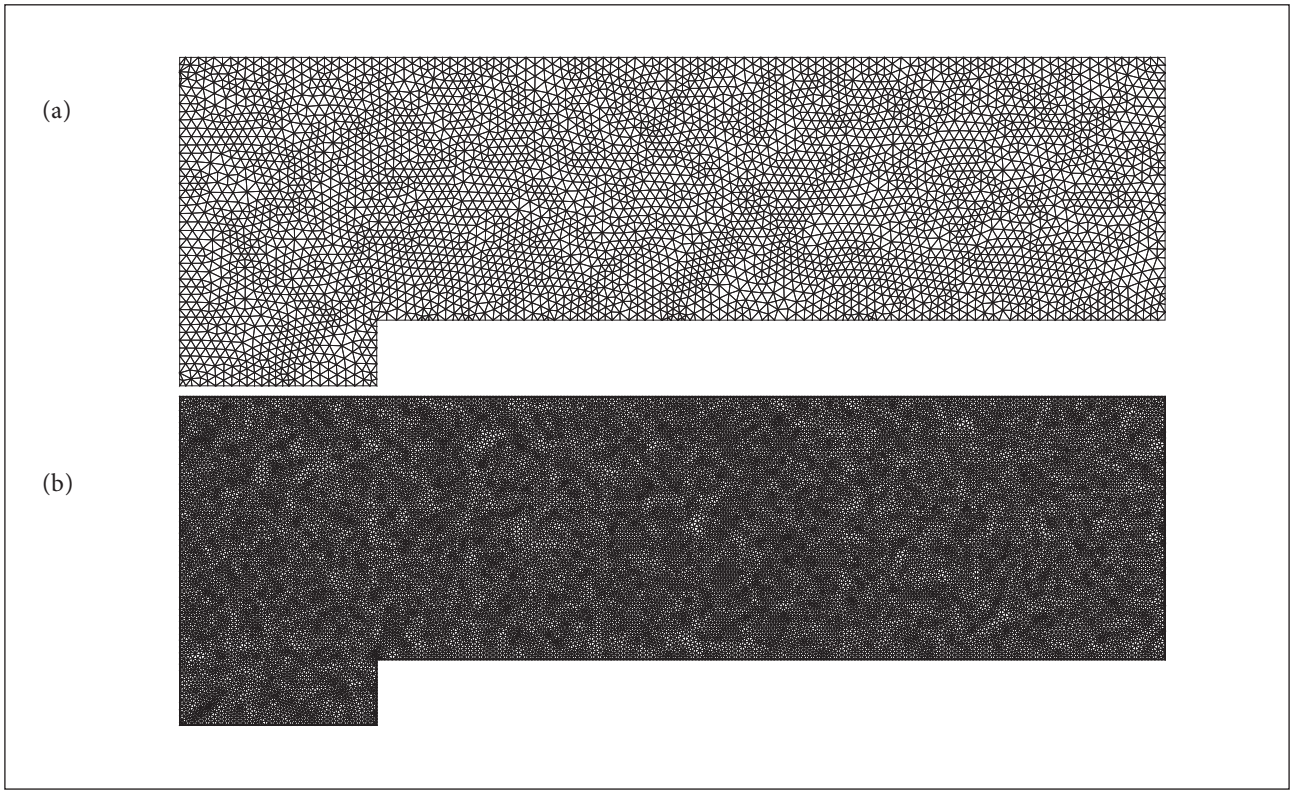

Figure 18. Computational meshes used in the forward facing step configuration simulations. (a) Coarse Mesh; (b) Fine mesh.

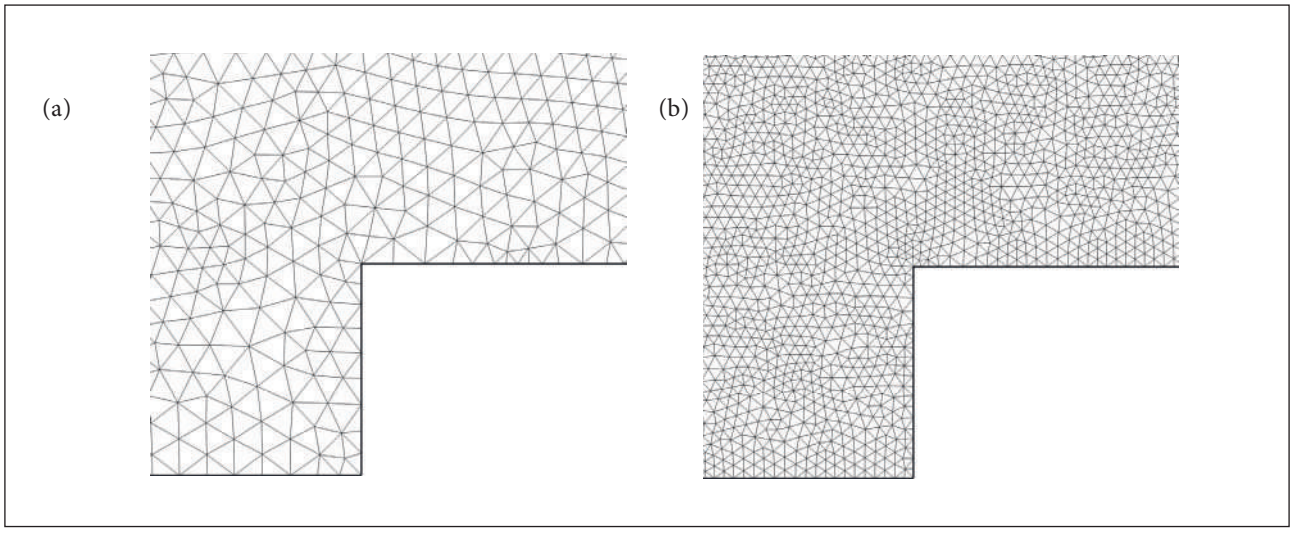

Figure 19. Detail of the domain discretization for the step region. (a) Coarse Mesh; (b) Fine mesh.

order to compare the results with the proposed hierarchical limiter for the 3rd-order SFV method. Note that, for a 2nd-order scheme, the hierarchical limiter would reduce the local reconstruction order to 1st-order accuracy anyway. Density contours for the coarse mesh can be seen in Figs. 20, 21 and 22, respectively, for the cited methods. Thirty evenly-spaced density contour lines are presented, for dimensionless density values ranging from 0.1 to 4.6. One can observe that the SFV schemes present a better resolution of the shear layer compared to the 1st-order method, as well as a slightly improved shock resolution.

In order to investigate the similarities with the 1st-order simulation, the limited cells for the 2nd- and 3rd-order SFV

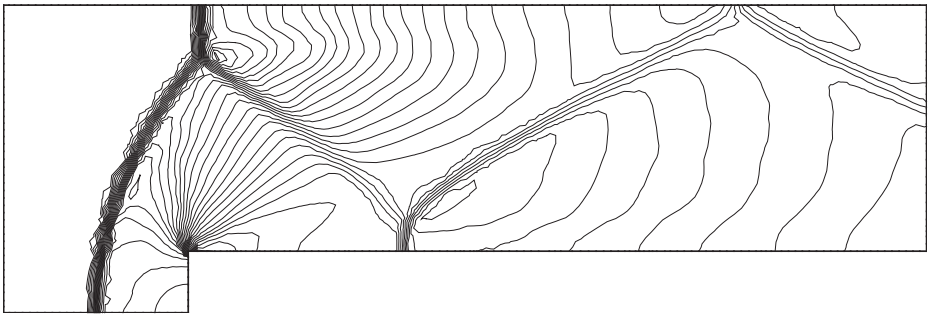

Figure 20. Density contours on the coarse mesh for the 1st-order Roe scheme. Figure shows 30 evenly-spaced dimensionless density contour lines from 0.1 to 4.6.

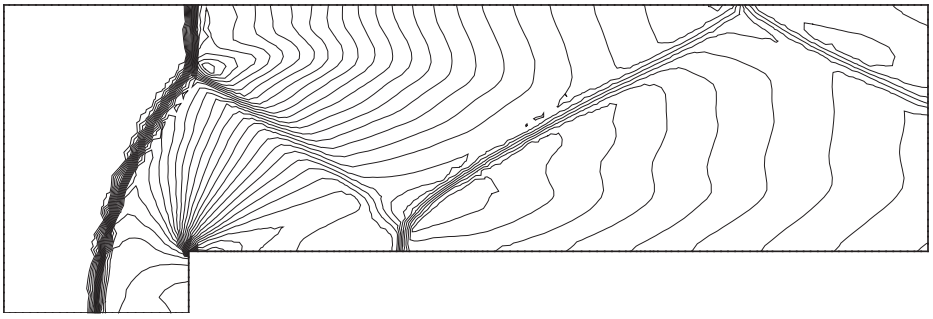

Figure 21. Density contours on the coarse mesh for the 2nd-order SFV scheme with the TVB limiter. Figure shows 30 evenly-spaced dimensionless density contour lines from 0.1 to 4.6.

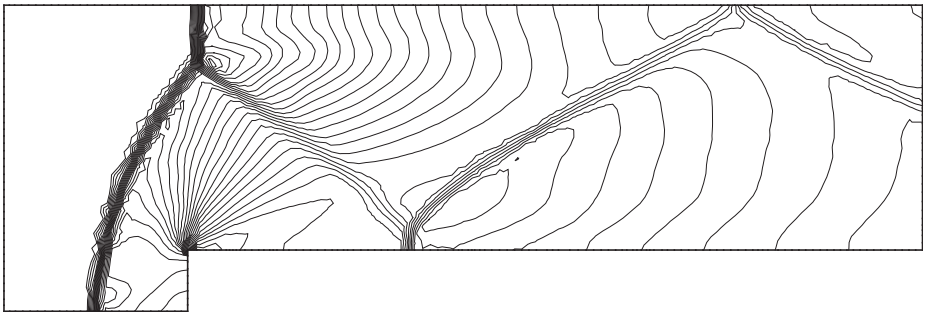

Figure 22. Density contours on the coarse mesh for the 3rd-order SFV scheme with the hierarchical limiter. Figure shows 30 evenly-spaced dimensionless density contour lines from 0.1 to 4.6.

schemes are presented in Figs. 23 and 24. These figures show the cells in which pressure reconstruction was performed at the last time step. For the 2nd-order method, there are too many cells limited by the TVB limiter, drastically reducing the overall solution reconstruction order. For the 3rd-order case, in which the hierarchical limiter is applied, the number of cells marked for limiting is reduced significantly. This confirms the superior behavior of the proposed limiting technique for the SFV method.

The next set of results considers the fine mesh. It is worth mentioning that the actual mesh used in the 3rd-order SFV method simulation has 295,824 control volumes, *i.e.,* 6 times more cells than used in the 1st-order Roe and 3rd-order WENO simulation, because the reconstruction procedure subdivides each original cell, or spectral volume, into 6 new control volumes. This increases the overall simulation costs as reported in Table 6, where the relative computational costs are normalized by those of the 2nd-order SFV scheme. Furthermore, the reader can observe that the 3rd-order WENO simulation is about twice as costly as the 3rd-order SFV scheme. In the present test case, the hierarchical limiter formulation of the SFV method yields a reduction in the cost differences between the present SFV schemes and the WENO solution, in comparison, for instance, with the results observed for the NACA 0012 test case, shown in Table 5. It happens that, due to the unsteady nature of the present problem, the limiter is activated differently at each time step and such behavior causes a penalty in the time step costs of both 2nd- and 3rd-order SFV results.

It is important to observe that, since the 2nd-order SFV method presents a large amount of limited cells due to the TVB limiter formulation, its results are actually representative of a 1st-order scheme. Hence, no results are shown for the fine mesh. Figure 25 shows the dimensionless density contours for the 1st-order Roe and 3rd-order SFV method simulations. Both

Table 6. Cost estimates per time step for the fine mesh.

Method	Order	Limiter	Time
Roe	1	No	0.08
MUSCL	2	Yes	0.25
SFV	2	Yes	1.00
SFV	3	Yes	4.79
WENO	3	No	8.55

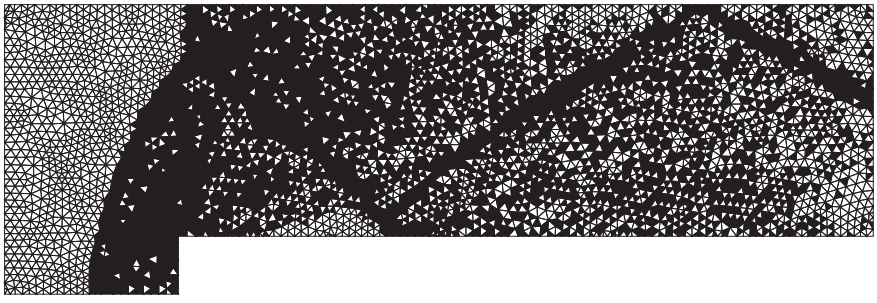

Figure 23. Limited cells for pressure reconstruction at the last time step for the 2nd-order SFV method with the TVB limiter.

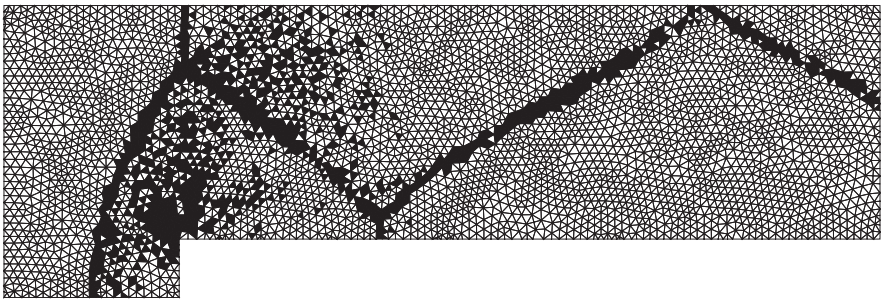

Figure 24. Limited cells for pressure reconstruction at the last time step for the 3rd-order SFV method with the hierarchical limiter.

schemes present a sharp shock resolution, but the SFV method results seem to better capture other flow features such as the shear layer. Again, the limited cells for pressure and internal energy reconstruction at the final time step are shown, respectively, in Figs. 26a and 26b for the 3rd-order SFV method calculations. One can clearly see that the high-order reconstruction is indeed

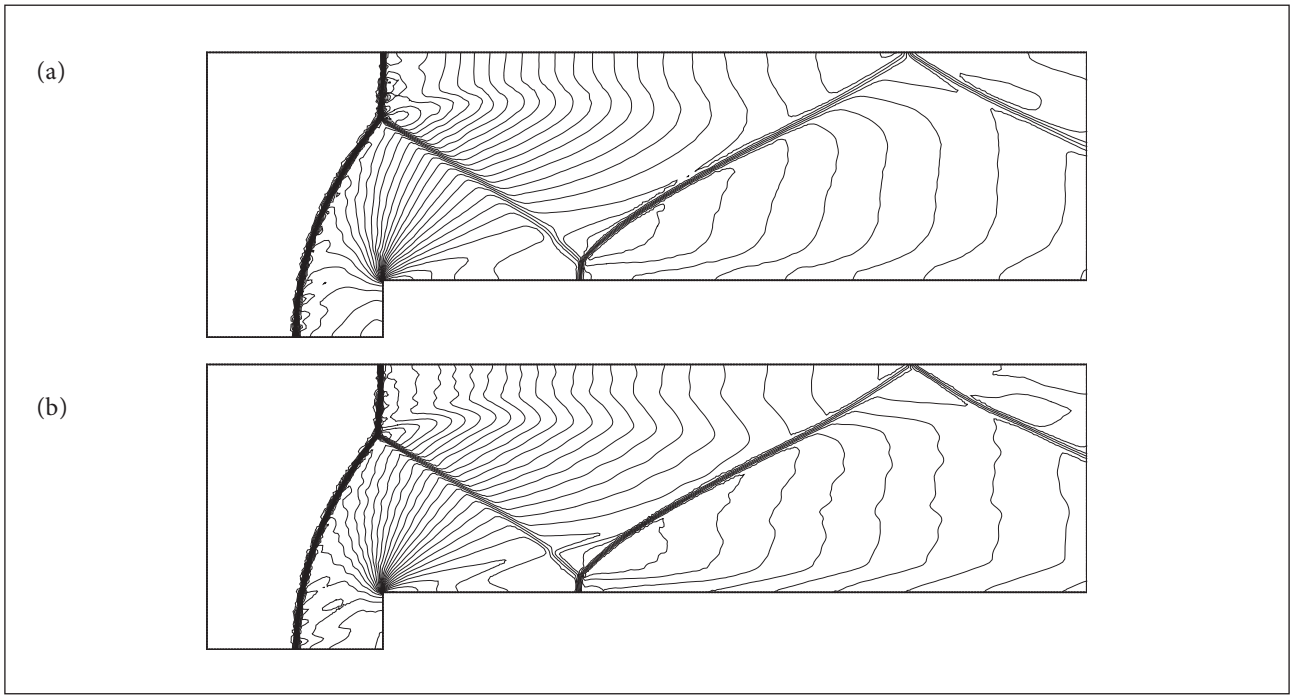

Figure 25. Density contours on the fine mesh for the 1st-order Roe scheme (a) and 3rd-order SFV scheme (b) . Thirty evenly-spaced contour lines for dimensionless density from 0.1 to 4.6.

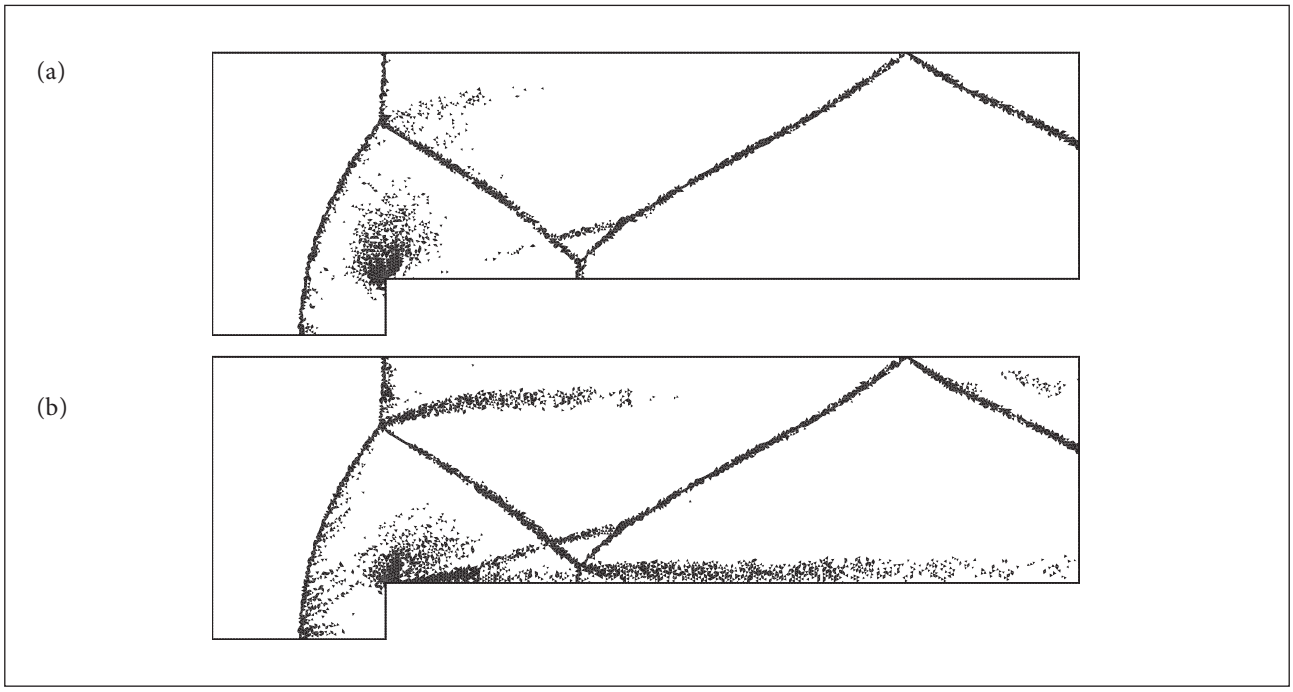

Figure 26 . Limited cells for pressure (a) and internal energy (b) reconstructions at the last time step for the 3rd-order SFV method with the hierarchical limiter on the fine mesh.

happening for the cells away from the discontinuities, since the limiter is only active at the discontinuities themselves.

Finally, the 3rd-order method solution is plotted in Fig. 27 in terms of the density contours at the CV mesh. It is important to understand that this visualization considers data for the CV cells and, therefore, it better represents the actual resolution capabilities of the SFV method. Such definition could be shown in the SV mesh but current visualization tools do not support high-order data visualization. A high-order post-processor would be necessary in order to allow a visualization with this level of resolution at the original SV mesh. Nevertheless, the level of resolution of flow features is improved over the original SV mesh.

DOUBLE MACH REFLECTION

This problem is also a standard test case for high-resolution schemes (Woodward and Colella 1984), and it has been extensively studied by many researchers (Cockburn and Shu 1989; Wang *et al.* 2004). The physical problem is that of a right-moving $M_\infty = 10$ shock wave, perpendicular to the axis of a 30 deg. half-angle wedge, which hits the tip of the wedge at time $t = 0$. Hence, the computational domain for the problem is chosen to be a rectangular region in the intervals $[0, 4] \times [0, 1]$, in the x- and y-directions, respectively. Initially, the right-moving $M_\infty = 10$ shock is positioned at $x = 1/6$, $y = 0$ and makes a 60 deg. angle with the x-axis. For the bottom boundary, the exact post-shock conditions are imposed, through the Rankine-Hugoniot relations, for the region from $x = 0$ to $x = 1/6$, and a solid wall boundary condition is used for the rest of the lower domain boundary. For the top boundary, the solution is set to describe the exact motion of the $M_\infty = 10$ shock. The left boundary is again set as the exact post-shock conditions, while the right boundary is set as an outflow boundary.

Two meshes are considered for this study, a coarse and a fine grid. The coarse mesh has 3,619 triangular cells, 1,808 nodes and characteristic dimensionless length $h = 1/20$. The fine mesh has 122,941 cells, 61,469 nodes and characteristic length of $h = 1/120$. Both meshes are shown in Fig. 28. It is interesting to mention that, for the SFV method simulations, the total number of CVs in the fine mesh turn out to be 368,823 and 737,646, respectively, for the 2nd- and 3rd-order methods. For the coarse mesh, the 1st-order Roe method is used, along with the 2nd- and 3rd-order SFV method. For the fine mesh, only the 1st-order Roe and 3rd-order SFV method simulations are reported here. For this test case, for instance, the 2nd-order MUSCL-reconstructed Roe scheme, with TVB minmod limiter, was not able to produce a solution. Negative values of pressure and density are found within the computation domain and the simulation diverges. This test case is a difficult one, in the sense that it requires proper boundary specification, on a per-face basis, and it also features a flow with a high level of kinetic energy.

Figure 29 presents numerical density contours for the coarse mesh, using the 1st-order Roe, 2nd- and 3rd-order SFV schemes, Moreover, 30 evenly-spaced contours are shown in each case and the dimensionless density values range from 1.25 to 21.5. For the mesh considered in this study, there is not much difference between the results. It is possible to note, however, that the Mach stem region, to the right end of the images, seems to be better defined for the SFV methods. For such problem, the use of limiters is obviously necessary. Figure 30 presents the limited cells for pressure reconstruction at the final time step for the 3rd-order SFV method. Clearly, only the shock region is indeed marked for limitation, as expected. Although necessary to obtain a numerical solution, the limiter utilization essentially reduces

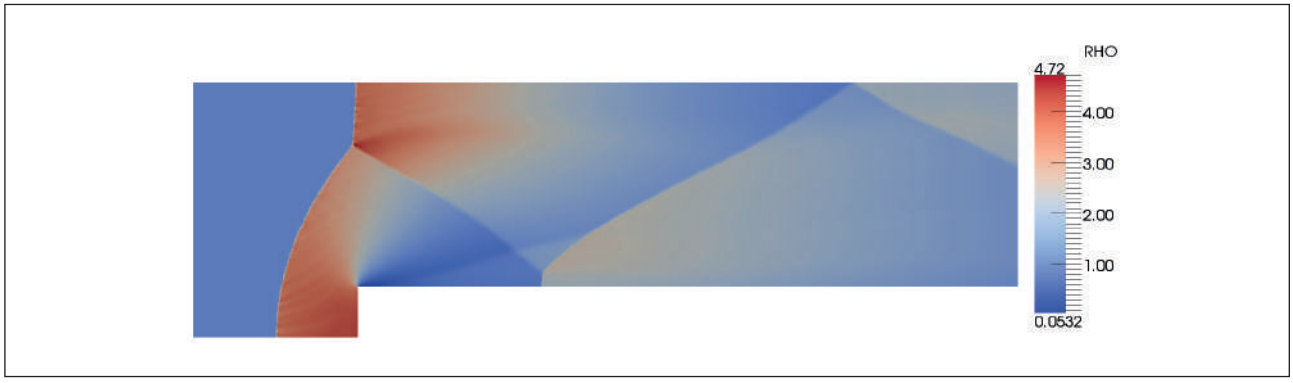

Figure 27. Density contours at the CV mesh level for the 3rd-order SFV method.

the high-order resolution in such regions and it might be responsible for the similarities in the results with the low-order scheme.

Results considering the fine mesh are shown in Figs. 31 which presents dimensionless density contours in the same range used to report the results for the coarse grid. As before, 30 evenly-spaced contours are shown in each case. Clearly, there is much more resolution now, even with the 1st-order method. The shock waves have much improved resolution and the Mach stem corner presents more details than with the coarse mesh simulations. The numerical solution on the fine mesh with the 3rd-order SFV method seems to have spurious oscillations, as one can see in Fig. 31b. The shock waves are clearly better resolved, but the rest of the domain solution seems quite oscillatory and disturbed.

In order to investigate this effect, the limited cells for pressure reconstruction at the last time step are plotted in Fig. 32. One can clearly see that a large number of cells has been selected for pressure limitation. The same behavior is observed for the other variables. This apparent oscillatory behavior has not been previously observed for other test cases and, therefore, the authors suspect that the limiter algorithm has some limitations with regard to simulations with very high Mach numbers and very strong discontinuities.

Further visualization of the results can be seen in Fig. 33, which presents density contours for the solution with the 3rd-order SFV method, but seen at the CV mesh level, that is, with 737,646 cells. As in the previous test case, this visualization considers data for the CV cells across the domain and better represents the actual resolution capabilities of the SFV method. A very sharp main shock wave can be clearly seen in the results, but oscillations in the density distribution are also seen in the post-shock region. Finally, Table 7 shows the relative costs of the different methods, normalized by the 2nd-order SFV method results.

Table 7. Estimates of relative costs per time step.

Method	Order	Limiter	Time
Roe	1	No	0.04
MUSCL	2	Yes	–
SFV	2	Yes	1.00
SFV	3	Yes	3.48
WENO	3	No	5.32

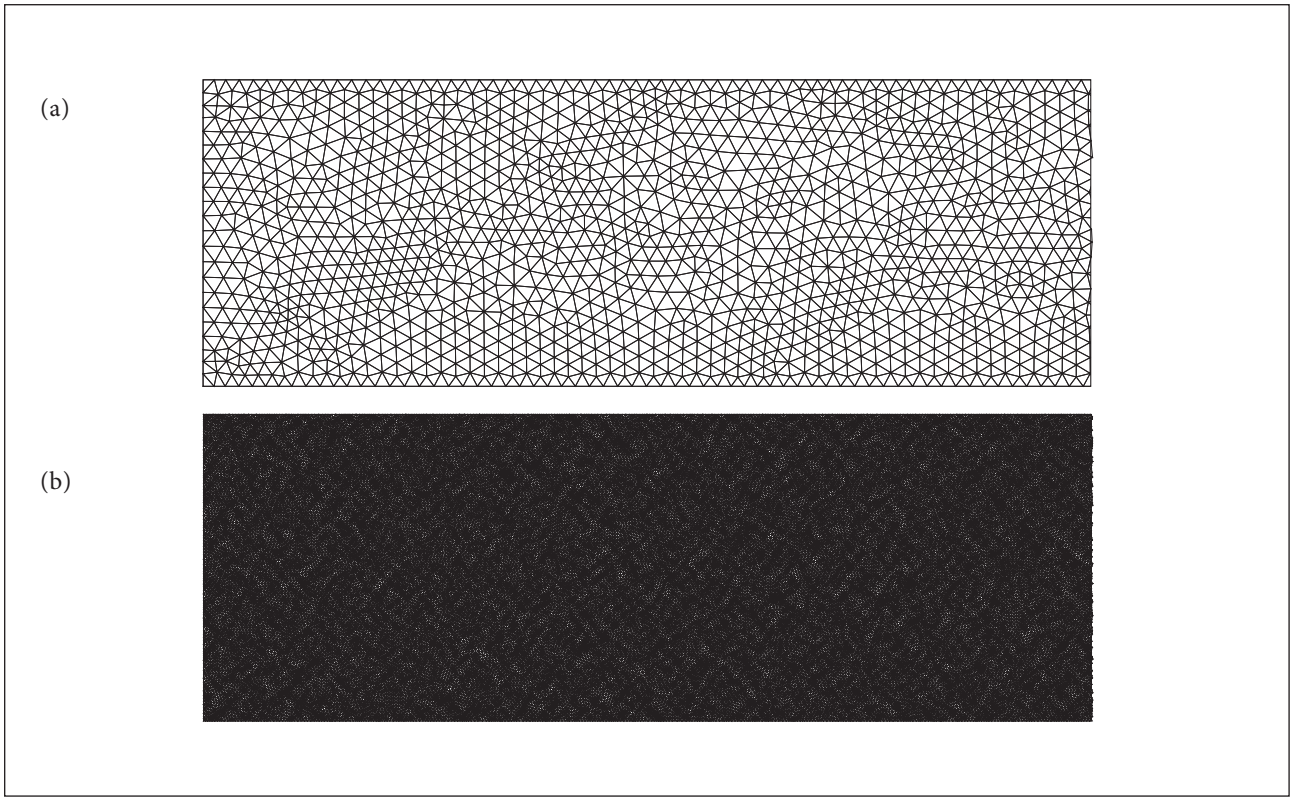

Figure 28. Computational meshes used for the double Mach reflection problem simulations. (a) Coarse mesh; (b) Fine mesh.

Costs are measured on the coarse mesh simulations and provide an overall estimate of the computational requirements associated with such high-order schemes. The 3rd-order WENO solution is about 1.5 times more

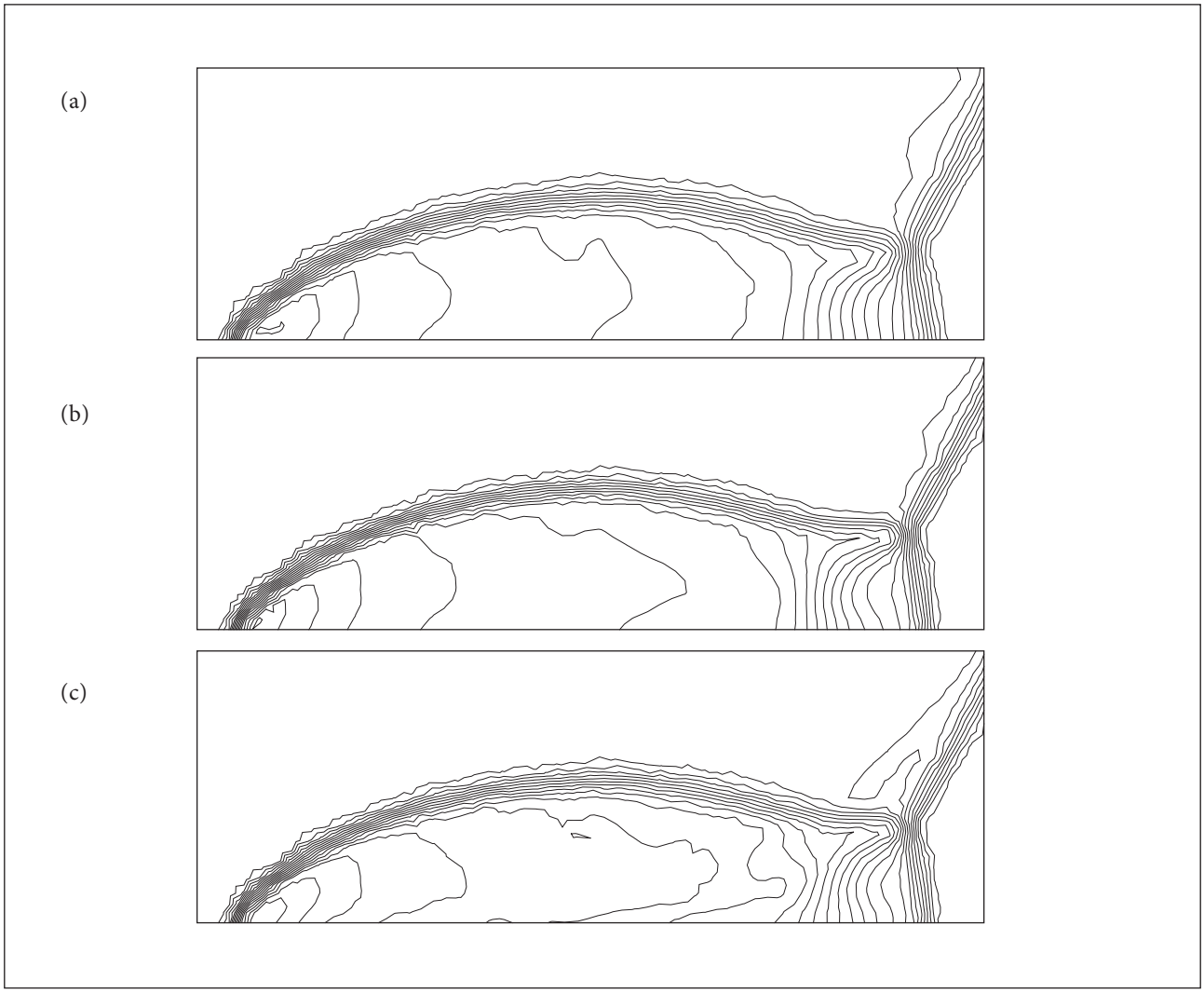

Figure 29. Density contours on the coarse mesh for the 1st-order Roe scheme (a); for 2nd-order SFV scheme(b); and for the 3rd-order SFV scheme (c). Figure shows 30 evenly-spaced dimensionless density contour lines in the range from 1.25 to 21.5.

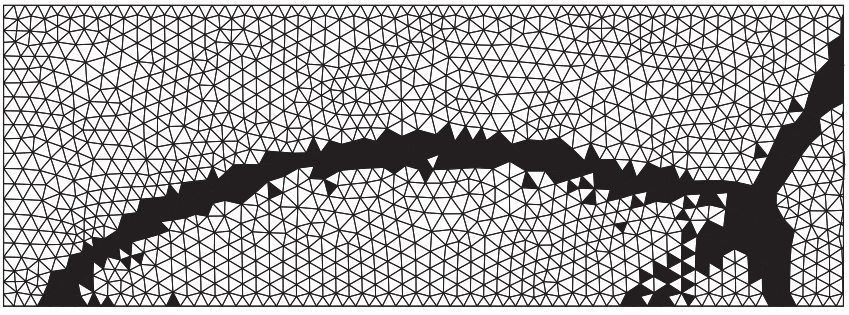

Figure 30. Limited cells for pressure reconstruction at the last time step for the 3rd-order SFV method.

expensive for this problem than the 3rd-order SFV method calculations, considering the same input data. The reader should observe that the hierarchical limiter is active only in the discontinuous region of the solution, as depicted in Fig. 30.

CONCLUDING REMARKS

The application of the high-order SFV method to steady and unsteady inviscid compressible flow simulations is presented. The paper also addresses the use of high-order boundary treatment, which is relevant because it can allow the usage of coarser meshes without giving up in geometric resolution. The present implementation of the limiter technique, which extends, to SFV methods, ideas that have been previously tested on spectral difference schemes, is an important ingredient of the method efficiency. The present limiter reduces the number of limited spectral volumes to a bare minimum, which reduces computational costs and, at the same time, allows for a more uniform high-order solution. Furthermore, a user-input-free limiter implementation contributes to enhance the robustness of the flow solver. Several classical test cases, both steady

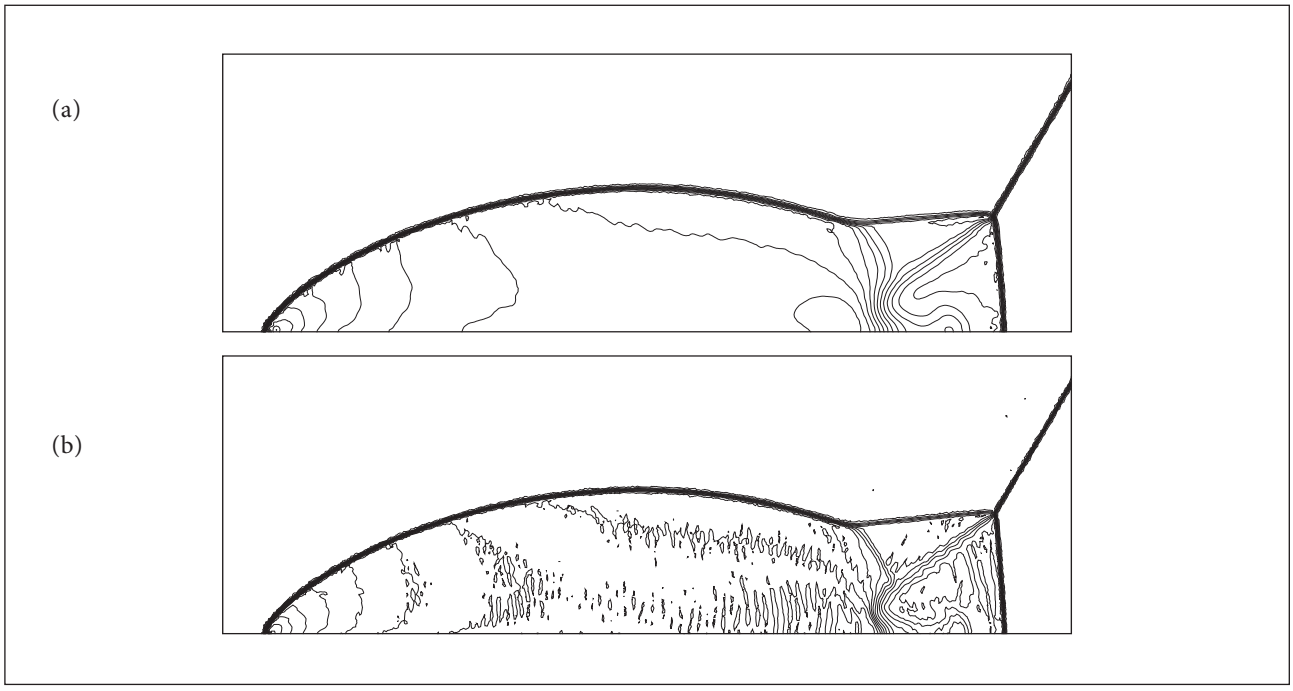

Figure 31. Density contours on the fine mesh for the 1st-order Roe scheme (a) and 3rd-order SFV scheme(b). Figure shows 30 evenly-spaced dimensionless density contour lines in the range from 1.25 to 21.5.

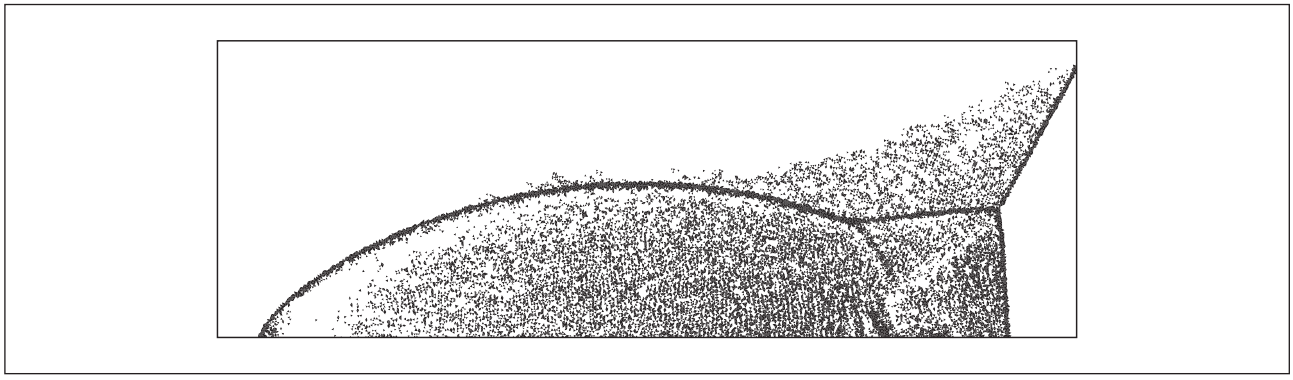

Figure 32. Limited cells for pressure reconstruction at the last time step for the 3rd-order SFV method on the fine mesh.

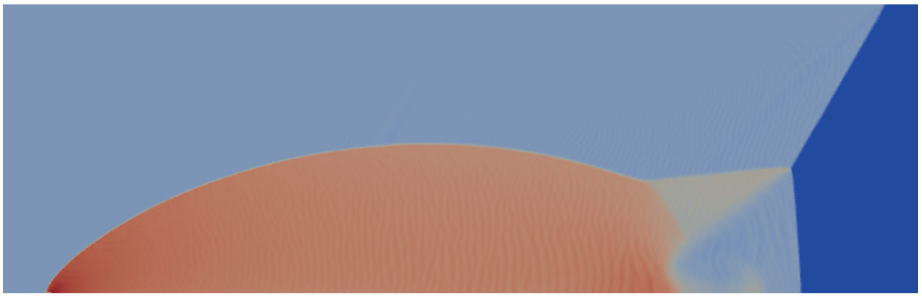

Figure 33. Density contours at the CV mesh level for the 3rd-order SFV method solution in the fine grid.

and unsteady, have been addressed in order to highlight such features.

The main focus of the paper has been on the complex transient problems that stress the numerical method ability to compute flows with strong discontinuities without numerical divergence. The results obtained in the present research for the forward-facing step and the double Mach reflection problems have indicated that the SFV method is able to cope with the strong shock waves and produce good solutions. For both test cases, however, it is clear that the limiter algorithm still has limitations with regard to simulations with very high Mach numbers and very strong discontinuities. It seems that small oscillations are still allowed at the main shock wave, and these are convected downstream by the high-order scheme. Finally, the computational costs of the present high-order implementation are rather modest when compared to the benefits in flow resolution which can be achieved.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the partial support for this research provided by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), under the Research Grants No. 309985/2013-7, No. 400844/2014-1 and No. 443839/2014-0. This study is also supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), through a graduate scholarship for the first author. The authors are also indebted to the partial financial support received from Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), under the Research Grant No. 2013/07375-0.

AUTHOR'S CONTRIBUTION

All authors contributed equally for the development of the work reported in the present paper.

REFERENCES

Abgrall R (1994) On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation. J Comput Phys 114(1):45-58. doi: 10.1006/jcph.1994.1148

Barth T, Frederickson P (1990) Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. Proceedings of the 28th AIAA Aerospace Sciences Meeting; Reno, USA.

Barth T, Jespersen D (1989) The design and application of upwind schemes on unstructured meshes. Proceedings of the 27th Aerospace Sciences Meeting and Exhibit; Reno, USA.

Bigarella EDV, Azevedo JLF (2007) Advanced eddy-viscosity and Reynolds-stress turbulence model simulations of aerospace applications. AIAA J 45(10):2369-2390. doi: 10.2514/1.29332

Bigarella EDV, Azevedo JLF (2012) A study of convective flux schemes for aerospace flows. J Braz Soc Mech Sci & Eng 34(3):314-329. doi: 10.1590/S1678-58782012000300012

Breviglieri C, Azevedo JLF, Basso E (2010a) An unstructured grid implementation of high-order spectral finite volume schemes. J Braz Soc Mech Sci & Eng 32:419-433. doi: 10.1590/S1678-58782010000500001

Breviglieri C, Azevedo JLF, Basso E, Souza MAF (2010b) Implicit high-order spectral finite volume method for inviscid compressible flows. AIAA J 48(10):2365-2376. doi: 10.2514/1.J050395

Breviglieri C, Basso E, Azevedo JLF (2008) High-order unstructured spectral finite volume scheme for aerodynamic applications. Proceedings of the 26th AIAA Applied Aerodynamics Conference; Honolulu, USA.

Cockburn B, Shu CW (1989) TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. II: General framework. Math Comp 52:441-435.

Emery A (1968) An evaluation of several differencing methods for inviscid fluid flow problems. J Comput Phys 2(3):306-331. doi: 10.1016/0021-9991(68)90060-0

Friedrich O (1998) Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids. J Comput Phys 144(1):194-212. doi: 10.1006/jcph.1998.5988

Gruttke WB (1995) The Initial Graphics Exchange Specification (IGES) v. 6.0. Washington: IGES/PDES Organization.

Jiang GS, Shu CW (1996) Efficient implementation of weighted ENO schemes. J Comput Phys 126(1):202-228. doi: 10.1006/jcph.1996.0130

Krivodonova L (2007) Limiters for high-order discontinuous Galerkin methods. J Comput Phys 226(1):879-896. doi: 10.1016/j.jcp.2007.05.011

Liu Y, Vinokur M, Wang ZJ (2006) Spectral (finite) volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems. J Comput Phys 212(2):454-472. doi: 10.1016/j.jcp.2005.06.024

McDevitt J, Okuno AF (1985) Static and dynamic pressure measurements on a NACA 0012 airfoil in the Ames High Reynolds Number Facility. NASA-TP-2485. Moffett Field: NASA Ames Research Center.

Ollivier-Gooch C (1997) Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent least-squares reconstruction. J Comput Phys 133(1):6-17. doi: 10.1006/jcph.1996.5584

Parsani M, van den Abeele K, Lacor C, Turkel E (2010) Implicit LU-SGS algorithm for high-order methods on unstructured grid with *p*-multigrid strategy for solving the steady Navier-Stokes equations. J Comput Phys 229(3):828-850. doi: 10.1016/j.jcp.2009.10.014

Qiu J, Shu CW (2006) A comparison of troubled-cell indicators for Runge-Kutta discontinuous Galerkin methods using weighted essentially nonoscillatory limiters. SIAM J Sci Comput 27(3):995-1013. doi: 10.1137/04061372X

Roe PL (1981) Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 43(2):357-372. doi: 10.1016/0021-9991(81)90128-5

Shapiro AH (1953) The dynamics and thermodynamics of compressible fluid flow. New York: Wiley.

Shu CW (1987) TVB uniformly high-order schemes for conservation laws. Math Comp 49:105-121.

Sun Y, Wang ZJ, Liu Y (2006) Spectral (finite) volume method for conservation laws on unstructured grids VI: Extension to viscous flow. J Comput Phys 215(1):41-58. doi: 10.1016/j.jcp.2005.10.019

Van Altena M (1999) High-order finite-volume discretisations for solving a modified advection-diffusion problem on unstructured triangular meshes (Master's thesis). Vancouver: University of British Columbia.

Van den Abeele K, Lacor C (2007) An accuracy and stability study of the 2D spectral volume method. J Comput Phys 226(1):1007-1026. doi: 10.1016/j.jcp.2007.05.004

Venkatakrishnan V (1995) Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. J Comput Phys 118(1):120-130. doi: 10.1006/jcph.1995.1084

Wang ZJ (2002) Spectral (finite) volume method for conservation laws on unstructured grids: Basic formulation. J Comput Phys 178(1):210-251. doi: 10.1006/jcph.2002.7041

Wang ZJ, Fidkowski K, Abgrall R, Bassi F, Caraeni D, Cary A, Deconinck H, Hartmann R, Hillewaert K, Huynh HT, Kroll N, May G, Persson P, van Leer B, Visbal M (2013) High-order CFD methods: Current status and perspective. Int J Numer Meth Fluid 72(8):811-845. doi: 10.1002/fld.3767

Wang ZJ, Liu Y (2002) Spectral (finite) volume method for conservation laws on unstructured grids II: Extension to two-dimensional scalar equation. J Comput Phys 179(2):665-697. doi: 10.1006/jcph.2002.7082

Wang ZJ, Liu Y (2004) Spectral (finite) volume method for conservation laws on unstructured grids III: One dimensional systems and partition optimization. J Sci Comput 20(1):137-157. doi: 10.1023/A:1025896119548

Wang ZJ, Liu Y (2006) Extension of the spectral volume method to high-order boundary representation. J Comput Phys 211(1):154-178. doi: 10.1016/j.jcp.2005.05.022

Wang ZJ, Liu Y, Zhang L (2004) Spectral (finite) volume method for conservation laws on unstructured grids IV: Extension to two-dimensional systems. J Comput Phys 194(2):716-741. doi: 10.1016/j.jcp.2003.09.012

Wolf WR, Azevedo JLF (2006) High-order unstructured essentially nonoscillatory and weighted essentially nonoscillatory schemes for aerodynamic flows. AIAA J 44(10):2295-2310. doi: 10.2514/1.19373

Wolf WR, Azevedo JLF (2007) High-order ENO and WENO schemes for unstructured grids. Int J Numer Meth Fluid 55(10):917-943. doi: 10.1002/fld.1469

Woodward P, Colella P (1984) The numerical simulation of two-dimensional fluid flow with strong shocks. J Comput Phys 54(1):115-173. doi: 10.1016/0021-9991(84)90142-6

Xu Z, Liu Y, Shu CW (2009) Hierarchical reconstruction for spectral volume method on unstructured grids. J Comput Phys 228(16):5787-5802. doi: 10.1016/j.jcp.2009.05.001

Yang M, Wang ZJ (2009) A parameter-free generalized moment limiter for high-order methods on unstructured grids. Adv Appl Math Mech 1(4):451-480. doi: 10.4208/aamm.09-m0913