

SPECTRAL FINITE VOLUME SCHEME APPLIED TO INVISCID COMPRESSIBLE FLOWS

Carlos Breviglieri, carbrevi@yahoo.com.br

Instituto Tecnológico de Aeronáutica, CTA/ITA/IEC, 12228-900, São José dos Campos, SP, Brazil

João Luiz F. Azevedo, azevedo@iae.cta.br

Edson Basso, basso@iae.cta.br

Instituto de Aeronáutica e Espaço, CTA/IAE/ALA, 12228-903, São José dos Campos, SP, Brazil

Abstract. *This work describes the implementation and analysis of a high-order accurate spatial discretization Spectral Finite Volume scheme applied to high-speed flows on unstructured grids. The reconstruction process of such schemes is described for linear, quadratic and cubic polynomials. In the paper, the high-order scheme is presented for the solution of the dimensionless, 2-D Euler equations in a cell centered finite volume context. High-order flux integration is achieved using Gaussian quadratures. An approximate Riemann solver is used to evaluate the fluxes on the interfaces of the control volumes and an implicit LU-SGS scheme provides the time integration of the equations. The features of the present methodology also include an exact high-order domain boundary representation and a hierarchical moment limiter to treat flow solution discontinuities. Results for high-speed flow simulations are presented with the objective of assessing the implemented capabilities.*

Keywords: *High Order Method, Spectral Finite Volume, Implicit Method, 2D Euler Equations, Unstructured Meshes*

1. INTRODUCTION

The Computational Aerodynamics group of Instituto de Aeronáutica e Espaço (IAE) has been developing CFD solvers for two and three dimensional systems, considering both structured and unstructured meshes for over a decade [Basso et al., 2003]. One research area of the development effort is aimed at the implementation of high-order methods suitable for problems of interest to the Institute, i.e., external high-speed aerodynamics. Some upwind schemes such as the van Leer flux vector splitting scheme [van Leer, 1982], the Liou AUSM⁺ flux vector splitting scheme [Liou, 1996] and the Roe flux difference splitting scheme [Roe, 1981] were implemented and tested for second-order accuracy with a MUSCL reconstruction [Anderson et al., 1986]. However, the nominally second-order schemes presented results with an order of accuracy smaller than expected in the solutions for unstructured grids. Aside from this fact, it is well known that total variation diminishing (TVD) schemes have their order of accuracy reduced to first order in the presence of shocks due to the effect of limiters.

This observation has motivated the group to study and to implement essentially non-oscillatory (ENO) and weighted essentially non-oscillatory (WENO) schemes in the past [Wolf and Azevedo, 2006]. However, as the intrinsic reconstruction model of these schemes relies on gathering neighbouring cells for polynomial reconstructions for each cell at each time step, both schemes were found to be very demanding on computer resources for resolution orders greater than three, in 2-D, or anything greater than 2nd order, in 3-D. This fact motivated the consideration of the spectral finite volume method, as proposed by Wang and co-workers [Wang, 2002, Wang and Liu, 2002, Wang and Liu, 2004, Wang et al., 2004, Liu et al., 2006, Sun et al., 2006], as a more efficient alternative. The numerical solver is currently implemented for the solution of the 2-D Euler equations in a cell centered finite volume context for triangular meshes, with an implicit LU-SGS scheme for time integration.

The remainder of the paper is organized as follows. In Section 2, the theoretical formulation is detailed, including spatial and time integration methods. Details are given for linear, quadratic and cubic polynomial reconstructions for triangular mesh elements. A future Section will present the formulation for high-order boundary representation and for the limiter formulation. In Section 4, preliminary numerical results are presented.

2. THEORETICAL FORMULATION

2.1 Governing Equations

In the present work, the 2-D Euler equations are solved in integral form as

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

where $\vec{P} = E\hat{i} + F\hat{j}$. The application of the divergence theorem to Eq. (1) yields

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

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

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

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

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

where e_t is the total energy per unit volume, and the ratio of specific heats, γ , is set as 1.4 for all computations in this work. In the finite volume context, for stationary meshes, Eq. (2) can be rewritten for the i -th mesh element as

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

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

2.2 Spatial Discretization

The spatial discretization process determines a k -th order discrete approximation to the integral in the right-hand side of Eq. (5). In order to solve it numerically, the computational domain, Ω , with proper initial and boundary conditions, is discretized into N non-overlapping triangles, the spectral volumes (SVs), such that

$$\Omega = \bigcup_{i=1}^N SV_i. \quad (6)$$

One should observe that the spectral volumes could be composed by any type of polygon, given that it is possible to decompose its bounding edges into a finite number of line segments Γ_K , such that

$$SV_i = \bigcup \Gamma_K. \quad (7)$$

In the present paper, however, the authors assume that the computational mesh is always composed of triangular elements. Hence, although the theoretical formulation is presented for the general case, the actual SV partition schemes are only implemented for triangular grids.

For a given order of spatial accuracy using the SFV method, each SV_i element must be partitioned in

$$N_m = \frac{k(k+1)}{2} \quad (8)$$

sub-elements or control volumes (CVs). If one denotes by $CV_{i,j}$ the j -th control volume of SV_i , the cell-averaged conserved variables, q , at time t , for $CV_{i,j}$ are computed as

$$q_{i,j} = \frac{1}{V_{i,j}} \int_{CV_{i,j}} q(x,y) dV, \quad (9)$$

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, $p_i(x,y) \in P^{k-1}$, with degree $k-1$, can be reconstructed to approximate each component of the CV conserved variable vector, q , inside SV_i , *i.e.*,

$$p_i(x,y) = q(x,y) + O(h^{k-1}), \quad (x,y) \in SV_i, \quad (10)$$

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

$$\frac{dq_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{nf} \int_{A_r} (\vec{f} \cdot \vec{n}) dS = 0, \quad (11)$$

where $\vec{f} = E\hat{i} + F\hat{j}$, at the CV level, and nf is the number of edges of $CV_{i,j}$.

The boundary integral in Eq. (11) can be further discretized into the convective operator form

$$C(q_{i,j}) \equiv \int_S (\vec{f} \cdot \vec{n}) dS = \sum_{r=1}^{nf} \int_{A_r} (\vec{f} \cdot \vec{n}) dS, \quad (12)$$

where nf is the number of faces, or edges in 2-D, of $CV_{i,j}$, and A_r represents the area, or the length in 2-D, of the r -th edge of the CV. Given the fact that \vec{n} is constant for each line segment, the integration on the right side of Eq. (12) can be performed numerically with a k -th order accurate Gaussian quadrature formula

$$\int_S (\vec{f} \cdot \vec{n}) dS = \sum_{r=1}^{nf} \sum_{q=1}^{nq} w_{rq} \vec{f}(q(x_{rq}, y_{rq})) \cdot \vec{n}_r A_r + O(A_r h^k), \quad (13)$$

where (x_{rq}, y_{rq}) and w_{rq} are, respectively, the Gaussian quadrature point coordinates and the weights on the r -th edge of SV_i and $nq = \text{integer}[(k+1)/2]$ is the number of quadrature points required on the r -th edge. For the second-order schemes, one Gaussian quadrature point is used in the integration. Given the coordinates of the end points of the element edge, z_1 and z_2 , one can obtain the quadrature point as the middle point of the segment connecting the two end points, $G_1 = \frac{1}{2}(z_1 + z_2)$. For the third and fourth order schemes, two Gaussian points are necessary along each line segment. Using the method described above, one can compute values of $q_{i,j}$ for instant t for each CV. 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 cell boundaries. At any moment during the simulation, one can compute the SV-averaged conserved variable vector, Q_i , for the i -th spectral volume, as

$$Q_i = \frac{1}{V_i} \sum_{j=1}^{N_m} q_{i,j} V_{i,j}. \quad (14)$$

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 Section 3.

As previously stated, the flux integration across CV boundaries that lies on the SV edges involves two discontinuous states, to the left and to the right of the edge. This flux computation can be carried out using a flux vector splitting procedure or an exact or approximate Riemann solver. The latter procedure is adopted in the present work and the numerical flux can be written as

$$\vec{f}(q(x_{rq}, y_{rq})) \cdot \vec{n}_r \approx f_{Riemann}(q_L(x_{rq}, y_{rq}), q_R(x_{rq}, y_{rq}), \vec{n}_r), \quad (15)$$

where q_L is the conserved variable vector obtained by the p_i polynomial applied at the (x_{rq}, y_{rq}) coordinates and q_R is the same vector obtained with the p_{nb} polynomial in the same coordinates of the edge. Note that the nb subscript represents the element to the right of the edge, whereas the i subscript denotes the CV to its left. As the numerical flux integration in the present paper is based on one of the forms of a Riemann solver, this is the mechanism which introduces the upwind and artificial dissipation effects into the method, making it stable and accurate. In the present work, the authors have used the Roe flux difference splitting method [Roe, 1981] to compute the numerical flux, *i.e.*,

$$f_{Riemann} = f_{roe}(q_L, q_R, \vec{n}) = \frac{1}{2} [\vec{f}(q_L) + \vec{f}(q_R)] \cdot \vec{n} - \frac{1}{2} |\overline{B}| (q_R - q_L). \quad (16)$$

Here, $|\overline{B}|$ is the Roe matrix in the direction normal to the edge.

Finally, one ends up with the semi-discrete SFV scheme for updating the CVs, which can be written as

$$\frac{dq_{i,j}}{dt} = -\frac{1}{V_{i,j}} \sum_{r=1}^{nf} \sum_{q=1}^{nq} w_{rq} f_{Riemann}(q_L(x_{rq}, y_{rq}), q_R(x_{rq}, y_{rq}), \vec{n}_r) A_r, \quad (17)$$

where the right hand side of Eq. (17) is the equivalent convective operator, $C(q_{i,j})$, for the j -th control volume of SV_i . It is important to emphasize that some edges of the CVs, resulting from the partition of the SVs, lie inside the SV element in the region where the polynomial is continuous. For such edges, there is no need to compute the numerical flux, as described above. Instead, one uses analytical formulas for the flux computation, *i.e.*, without numerical dissipation.

2.3 Temporal Discretization

In order to obtain the steady state solution of the flow from an initial condition, a relaxation scheme is necessary. The convergence behavior to steady state of high-order methods, such as the SFV method, is generally poor with explicit time marching approaches. The approach typically used in the present research group has been to resort to explicit, multi-stage, Runge-Kutta time-stepping methods. However, adequate solution convergence rate, especially for the higher-order implementations, dictate that an implicit time integrator should be employed. Therefore, an implicit LU-SGS scheme is also implemented in the context of the present work. Equation (17) can be rewritten as

$$V_{i,j} \frac{dq_{i,j}}{dt} = -R_{i,j} \quad (18)$$

where $R_{i,j}$ is the right-hand side residual for the j -th CV of the i -th SV and it tends to zero as the simulation converges to a steady-state solution. Using the implicit Euler method for time integration, Eq. (18) can be written in discrete form as

$$V_{i,j} \frac{\Delta q_{i,j}^n}{\Delta t} = -R_{i,j}^{n+1} \quad (19)$$

where Δt is the time increment and $\Delta q^n = q^{n+1} - q^n$. The previous equation can be linearized in time as

$$V_{i,j} \frac{\Delta q_{i,j}^n}{\Delta t} = -R_{i,j}^n - \frac{\partial R_{i,j}^n}{\partial q} \Delta q_{i,j}^n. \quad (20)$$

The term $\partial R/\partial q$ represents the Jacobian matrix. Writing the equation for all CV's, one obtains the delta form of the backward Euler scheme

$$A\Delta q = -R \quad (21)$$

where

$$A = \frac{V}{\Delta t} \mathbf{I} + \frac{\partial R^n}{\partial q} \quad (22)$$

and \mathbf{I} is the identity matrix.

In order to reduce the number of non-zero entries in the Jacobian matrix and to simplify the linearization process, only a first-order representation of the numerical fluxes, \bar{f} , is linearized. The simplified numerical flux function is defined as

$$\bar{f}_i(q_i, q_{nb}, \vec{n}_r) = \frac{1}{2} \left[\bar{f}(q_i, \vec{n}_r) + \bar{f}(q_{nb}, \vec{n}_r) - |\lambda|(q_{nb} - q_i) \right] \quad (23)$$

for the r -th edge that shares the i and nb control volumes. A scalar dissipation model is used, where

$$|\lambda| = |v_n| + a. \quad (24)$$

One should note that the dissipation in the flux function is approximated by the Jacobian matrix spectral radius. The simplified numerical flux function is used to obtain the left-hand side Jacobian matrices. Hence, the linearization of Eqs. (22) and (23) yields

$$\begin{aligned} \frac{\partial R_i}{\partial q_i} &= \frac{1}{2} [J(q_i) + |\lambda| \mathbf{I}] \\ \frac{\partial R_i}{\partial q_{nb}} &= \frac{1}{2} [J(q_{nb}) - |\lambda| \mathbf{I}] \end{aligned} \quad (25)$$

where J is the Jacobian of the inviscid flux vectors in the direction normal to the edge.

Using an edge-based data structure, the Jacobian matrix is stored in lower, upper and diagonal components, which are computed as

$$\begin{aligned} L &= \frac{1}{2} [-J(q_{nb}, \vec{n}_r) - |\lambda| \mathbf{I}] \\ U &= \frac{1}{2} [J(q_{nb}, \vec{n}_r) - |\lambda| \mathbf{I}] \\ D &= \frac{V}{\Delta t} \mathbf{I} + \sum_{nb} \frac{1}{2} [J(q_i, \vec{n}_r) + |\lambda| \mathbf{I}]. \end{aligned} \quad (26)$$

Note that L , U and D represent the strict lower, upper and diagonal matrices, respectively. Equation (21) represents a system of linear simultaneous algebraic equations that needs to be solved at each time step. The iterative LU-SGS solution method is employed, along with a mesh renumbering algorithm [Cuthill and McKee, 1969], and the system is solved in two steps, a forward and backward sweep,

$$\begin{aligned} (D + L)\Delta q^* &= R \\ (D + U)\Delta q &= D\Delta q^*. \end{aligned} \quad (27)$$

3. NUMERICAL FORMULATION

3.1 Spectral Finite Volume Formulation

The evaluation of the conserved variables at the quadrature points is necessary in order to perform the flux integration over the mesh element faces. These evaluations can be achieved by reconstructing conserved variables in terms of some base functions using the DOFs within a SV. The present work has carried out such reconstructions using polynomial base functions, although one can choose any linearly independent set of functions. Let P_m denote the space of m -th degree polynomials in two dimensions. Then, the minimum dimension of the approximation space that allows P_m to be complete is

$$N_m = \frac{(m+1)(m+2)}{2}. \quad (28)$$

In order to reconstruct q in P_m , it is necessary to partition the SV into N_m non-overlapping CVs, such that

$$SV_i = \bigcup_{j=1}^{N_m} CV_{i,j}. \quad (29)$$

The reconstruction problem, for a given continuous function in SV_i and a suitable partition, can be stated as finding $p_m \in P_m$ such that

$$\int_{CV_{i,j}} p_m(x, y) dS = \int_{CV_{i,j}} q(x, y) dS. \quad (30)$$

With a complete polynomial basis, $e_\ell(x, y) \in P_m$, it is possible to satisfy Eq. (30). Hence, p_m can be expressed as

$$p_m = \sum_{\ell=1}^{N_m} b_\ell e_\ell(x, y), \quad (31)$$

where e is the base function vector, $[e_1, \dots, e_{N_m}]$, and b is the reconstruction coefficient vector, $[b_1, \dots, b_{N_m}]^T$. The substitution of Eq. (31) into Eq. (30) yields

$$\frac{1}{V_{i,j}} \sum_{\ell=1}^{N_m} b_\ell \int_{CV_{i,j}} e_\ell(x, y) dS = q_{i,j}. \quad (32)$$

If q denotes the $[q_{i,1}, \dots, q_{i,N_m}]^T$ column vector, Eq. (32) can be rewritten in matrix form as

$$Sb = q, \quad (33)$$

where the S reconstruction matrix is given by

$$S = \begin{bmatrix} \frac{1}{V_{i,1}} \int_{CV_{i,1}} e_1(x, y) dS & \cdots & \frac{1}{V_{i,1}} \int_{CV_{i,1}} e_{N_m}(x, y) dS \\ \vdots & \cdots & \vdots \\ \frac{1}{V_{i,N_m}} \int_{CV_{i,N_m}} e_1(x, y) dS & \cdots & \frac{1}{V_{i,N_m}} \int_{CV_{i,N_m}} e_{N_m}(x, y) dS \end{bmatrix} \quad (34)$$

Table 1. Polynomial base functions.

Reconstruction Order	e
linear	$[1 \ x \ y]$
quadratic	$[1 \ x \ y \ x^2 \ xy \ y^2]$
cubic	$[1 \ x \ y \ x^2 \ xy \ y^2 \ x^3 \ x^2y \ xy^2 \ y^3]$

and, then, the reconstruction coefficients b can be obtained as

$$b = S^{-1}q, \quad (35)$$

provided that S is non-singular. Substituting Eq. (35) into Eq. (31), p_m is, then, expressed in terms of shape functions $L = [L_1, \dots, L_{N_m}]$, defined as $L = eS^{-1}$, such that one could write

$$p_m = \sum_{j=1}^{N_m} L_j(x, y)q_{i,j} = Lq. \quad (36)$$

Equation (36) gives the value of the conserved state variable, q , at any point within the SV and its boundaries, including the quadrature points, (x_{rq}, y_{rq}) . Note that q in the equation takes the form as a column vector, as presented in Eq.(33). The above 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.

3.2 Partition Schemes

For the linear SFV method reconstruction, $m = 1$, one needs to partition a SV into three sub-elements, as in Eq. (28) and use the appropriate base vector as defined in Table 1. The partition scheme is given for a standard element and it is uniquely defined for this case. The structured aspect of the CVs within the SVs is used to determine neighborhood information and generate the global connectivity data considering a hash table search algorithm [Knuth, 1998].

The linear partition is presented in Fig. 1(a). The integrals of the reconstruction matrix in Eq. (34) are obtained analytically for the triangular mesh elements [Liu and Vinokur, 1998].

For the quadratic reconstruction, $m = 2$, one needs to partition a SV into six sub-elements and use the base vector as defined in the second line of Table 1. The partition scheme is also given in this work for a right triangle. The nodes of the partition are obtained in terms of barycentric coordinates of the SV element nodes in the same manner as the linear partition. The structured aspect of the CVs within the SVs is used to determine neighborhood information and generate the connectivity table. The ghost creation process and edge-based data structure is the same as for the linear reconstruction case. The partition used in the present work follows the orientations given in Ref. [van den Abeele and Lacor, 2007] and it is shown in Fig. 1(b).

For the cubic reconstruction, $m = 3$, one needs to partition the SV into ten sub-elements and to use the base vector as defined in Table 1. The ghost creation process and edge-based data structure is the same as for the linear and quadratic reconstruction cases. As a matter of fact, the same algorithm used to perform these tasks can be applied to higher order reconstructions. The partition used in this work is the improved cubic partition [van den Abeele and Lacor, 2007], presented in Fig. 1(c). As with the quadratic reconstruction, each CV edge has two quadrature points for numerical flux integration.

3.3 High-Order Boundary Treatment

From the formulation described thus 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 MUSCL finite volume scheme [Bigarella, 2007], 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, in regions such as the leading edge of an airfoil, for instance.

If such approach is carried over to the SFV method, there is no gain in computational performance. As the literature presents for high-order schemes, such as DG and SFV methods, one solution is to treat these boundary faces as curved from within the solver. For the present work, a quadratic and cubic boundary representation is performed for the 3rd and 4th-order SFV schemes. Although it is suggested to also perform a quadratic boundary representation to the 2nd-order SFV scheme [Wang and Liu, 2006] such approach is not followed here. Therefore, the 2nd order scheme has no special boundary treatment.

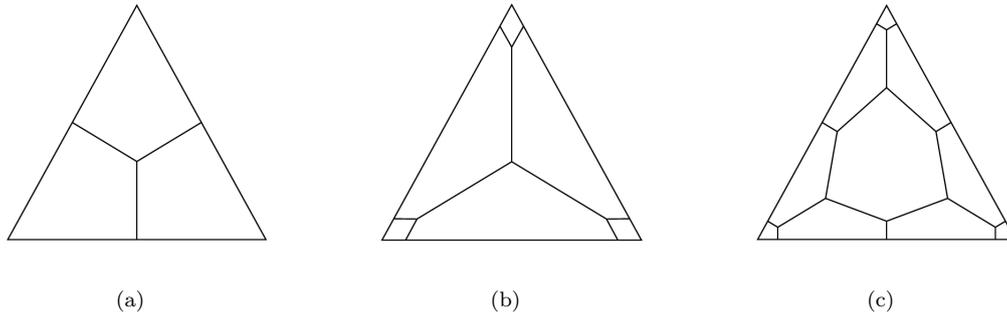


Figure 1. Triangular spectral volume partitions for (a) linear, (b) quadratic and (c) cubic reconstructions.

In order to perform this representation, one can adopt isoparametric SV elements 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 properties interpolation. In this work, only SV that are adjacent to wall boundaries receive this treatment.

Once the “curved” SV is partitioned, the interpolation shape functions and the CV face normals must be recalculated. Note that typically only one face of the SV stands at a boundary and one could use a simplified formulation for this specific face. For more information, the interested reader is referred to Ref. [Wang and Liu, 2006] and references therein.

4. NUMERICAL RESULTS

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. For the steady case simulations, the CFL number is set as a constant value and the local time step is computed using the local grid spacing and characteristic speeds. For all test cases, the CFL number is set to 10^6 .

All numerical simulations are carried out on a dual-core 3.0 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.

4.1 Wedge Flow

The first test case is the computation of the supersonic flow field past a wedge with half-angle $\theta = 15$ deg, based on data from Ref. [Slater, 2009]. The computational mesh has 2409 nodes and 4613 volumes. Numerical pressure contours, obtained with the 3rd-order SFV method, are shown in Fig. 2. For comparison purposes, the 3rd-order SFV and WENO methods are used for this simulation. The leading edge of the wedge is located at coordinates $(x, y) = (0, 0)$. The computational domain is bounded along the bottom by the symmetry plane of the wedge and the wedge surface. The inflow boundary is located at $x = -0.5$. The outflow boundary is placed at $x = 1.0$. The farfield boundary is placed at $y = 1.0$, in order to guarantee that it is located above the oblique shock. The analytical solution [Anderson, 1982, Staff, 1953] gives the change in properties across the oblique shock as a function of the freestream Mach number and shock angle, which is obtained from the $\theta - \beta - Mach$ relation. For this case, a freestream Mach number of $M_1 = 2.5$ is used, and the oblique shock angle β is obtained as 36.94 deg. For the present simulations, the limiter is turned on, and only elements in the shock wave region are marked for limited reconstruction.

The numerical solution of the SFV method is in good agreement with the analytical result for this case, as observed in Table 2, which presents the analytical and numerical ratio of properties before (subscript 1) and after (subscript 2) the oblique shock wave. The numerical results reported in Table 2 are obtained by performing some averaging of the computational results right after the shock in the interior of the domain.

4.2 RAE2822 Airfoil

The transonic flow over a RAE 2822 airfoil with 2.31 deg. angle-of-attack and freestream Mach number $M_\infty = 0.729$ is also considered. In order to evaluate the present implementation against computations performed with a WENO scheme [Wolf and Azevedo, 2006], two different simulations are performed for this test case. Initially, a simulation with a second-order WENO scheme, on a standard finite volume mesh which uses linear

Table 2. Comparison of analytical and computational results for the oblique shock wave properties.

Property	Analytical	Numerical
M_2	1.8735	1.88
ρ_2/ρ_1	1.8663	1.87
p_2/p_1	2.4675	2.47
T_2/T_1	1.3218	1.32

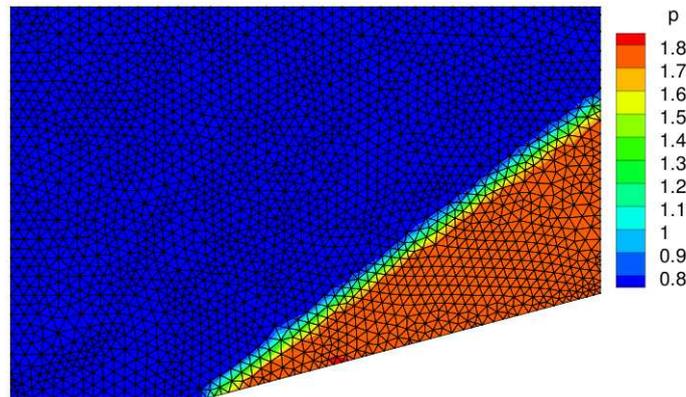


Figure 2. Supersonic wedge flow unstructured mesh with pressure contours for the 3rd-order SFV method.

boundary representation, is performed. The mesh for such calculation has 16,383 nodes and 32,399 elements, of which 340 elements lie on the airfoil surface, as necessary to properly describe the airfoil geometry, especially on the leading edge. The second simulation is carried out for the third-order SFV method with quadratic boundary representation. The approach does not require a very fine mesh, as in the first test case, because the geometry definition file is directly available to the solver. Moreover, only the edges of SVs that are on the airfoil surface are mapped and stored for high-order boundary computation. As discussed, only the boundary spectral volumes with high-order curved edges need a special treatment and require the additional storage. Spectral volumes in the interior of the domain are always treated through the use of the standard partition scheme. The coarse mesh, used for this second test case, has 2,700 nodes and 5,265 elements. This mesh has only 85 spectral volume elements along the airfoil surface. It should be noted, however, that, although the coarse mesh has much fewer elements, the computation is actually carried over on 31,590 CV elements for the third order SFV method. Therefore, the two test cases are comparable.

The present test cases evidence the large benefit of using a high-order method for spatial discretization. The ability of obtaining an equivalent numerical solution, with much less stringent mesh requirements than would be necessary to standard second order methods, is an important advantage. The numerical C_p results for the two test cases are presented in Fig. 3, together with the corresponding experimental data [Cook et al., 1979]. The numerical solution of the SFV method is in overall good agreement with the experimental data and with the calculations from the WENO scheme with the finer mesh, as shown in Fig. 3. Clearly, the experimental data shows a smoothed transonic shock, due to shock wave–boundary layer interaction, which is an important phenomenon for such supercritical airfoils. The present computations cannot reproduce such physics, since the Euler equations are used. Furthermore, Fig. 4 presents the L_2 norm of the density residuals as a function of the number of iterations and of the actual wall-clock time. Such results indicate that the third-order SFV method, with proper boundary treatment, outperforms the second-order simulation, especially in terms of the number of iterations.

5. CONCLUDING REMARKS

The high-order Spectral Finite Volume method is successfully implemented. The method behavior for resolution greater than second order was shown to be in good agreement with both experimental and analytical data. Furthermore, the results obtained are indicative that the current method can yield solutions with similar quality at a much lower computational resource usage than the WENO scheme. The method seems suitable for the aerospace applications in the sense that it is compact, from an implementation point of view, geometry flexible, as it handles unstructured meshes, and computationally efficient.

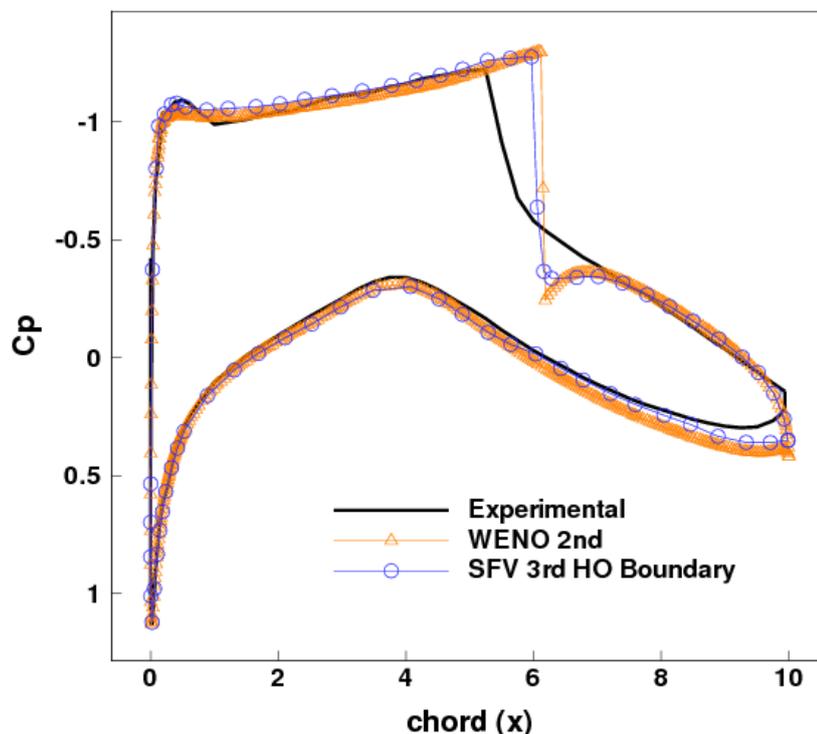


Figure 3. C_p distributions for RAE 2822 airfoil at $M_\infty = 0.729$ and $\alpha = 2.31$ deg.

6. REFERENCES

- Anderson, J. D., 1982, "Modern Compressible Flow, with Historical Perspective - 2nd Edition", McGraw-Hill, Boston.
- Anderson, W. K., Thomas, J. L., and van Leer, B., 1986, Comparison of Finite Volume Flux Vector Splittings for the Euler Equations, "AIAA Journal", Vol. 24, No. 9, pp. 1453–1460.
- Basso, E., Antunes, A. P., and Azevedo, J. L. F., 2003, Chimera Simulations of Supersonic Flows Over a Complex Satellite Launcher Configuration, "Journal of Spacecraft and Rockets", Vol. 40, No. 3, pp. 345–355.
- Bigarella, E., 2007, "Advanced Turbulence Modelling for Complex Aerospace Applications", Ph.D. Thesis, Instituto Tecnológico de Aeronáutica, São José dos Campos, SP, Brazil.
- Cook, P. H., McDonald, M. A., and Firmin, M. C. P., 1979, Aerofoil RAE 2822 – Pressure Distributions and Boundary Layer and Wake Measurements, Experimental Data Base for Computer Program Assessment, AGARD Report AR 138.
- Cuthill, E. and McKee, J., 1969, Reducing the Bandwidth of Sparse Symmetric Matrices, "Proceedings of the 24th National Conference of the ACM", New York. Brandon Systems Press.
- Knuth, D. E., 1998, "The Art of Computer Programming. 3: Sorting and Searching (2nd ed.)", Addison-Wesley, Reading, MA.
- Liou, M. S., 1996, A Sequel to AUSM: AUSM+, "Journal of Computational Physics", Vol. 129, No. 2, pp. 364–382.
- Liu, Y. and Vinokur, M., 1998, Exact Integrations of Polynomials and Symmetric Quadrature Formulas over Arbitrary Polyhedral Grids, "Journal of Computational Physics", Vol. 140, No. 1, pp. 122–147.
- Liu, Y., Vinokur, M., and Wang, Z. J., 2006, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids V: Extension to Three-Dimensional Systems, "Journal of Computational Physics", Vol. 212, No. 2, pp. 454–472.
- Roe, P. L., 1981, Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes, "Journal of Computational Physics", Vol. 43, No. 2, pp. 357–372.
- Slater, J. W., 2009, NPARC Alliance Validation Archive, <http://www.grc.nasa.gov/WWW/wind/valid/wedge/wedge.html>.

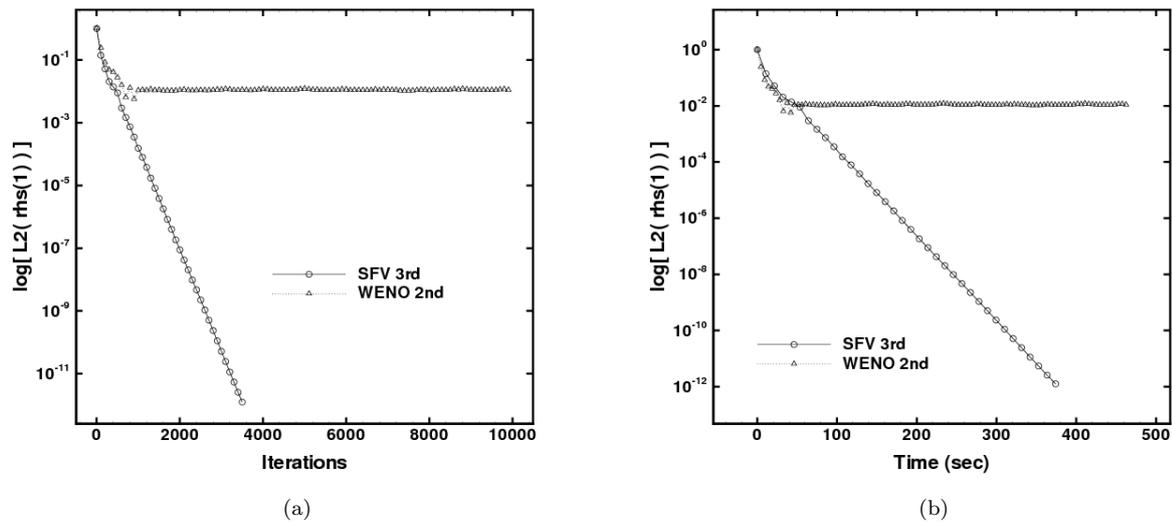


Figure 4. Convergence histories for RAE 2822 airfoil with implicit 3rd-order SFV and 2nd-order WENO schemes.

Staff, A. R., 1953, Equations, Tables, and Charts for Compressible Flow, Report 1135, NACA.

Sun, Y., Wang, Z. J., and Liu, Y., 2006, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids VI: Extension to Viscous Flow, "Journal of Computational Physics", Vol. 215, No. 1, pp. 41–58.

van den Abeele, K. and Lacor, C., 2007, An Accuracy and Stability Study of the 2D Spectral Volume Method, "Journal of Computational Physics", Vol. 226, No. 1, pp. 1007–1026.

van Leer, B., 1982, Flux-Vector Splitting for the Euler Equations, "Lecture Notes in Physics", Vol. 170, pp. 507–512.

Wang, Z. J., 2002, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids: Basic Formulation, "Journal of Computational Physics", Vol. 178, No. 1, pp. 210–251.

Wang, Z. J. and Liu, Y., 2002, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids II: Extension to Two-Dimensional Scalar Equation, "Journal of Computational Physics", Vol. 179, No. 2, pp. 665–698.

Wang, Z. J. and Liu, Y., 2004, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids III: One-Dimensional Systems and Partition Optimization, "Journal of Scientific Computing", Vol. 20, No. 1, pp. 137–157.

Wang, Z. J. and Liu, Y., 2006, Extension of the Spectral Volume Method to High-Order Boundary Representation, "Journal of Computational Physics", Vol. 211, No. 1, pp. 154–178.

Wang, Z. J., Liu, Y., and Zhang, L., 2004, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids IV: Extension to Two-Dimensional Systems, "Journal of Computational Physics", Vol. 194, No. 2, pp. 716–741.

Wolf, W. R. and Azevedo, J. L. F., 2006, High-Order Unstructured Essentially Non Oscillatory and Weighted Essentially Non Oscillatory Schemes for Aerodynamic Flows, "AIAA Journal", Vol. 44, No. 10, pp. 2295–2310.

7. Responsibility notice

The author(s) is (are) the only responsible for the printed material included in this paper