Efficient Implementation of High-Order Spectral Volume Method for Multidimensional Conservation Laws on Unstructured Grids

Rob Harris[*] and Z. J. Wang[†]

Department of Aerospace Engineering, Iowa State University, Ames, IA 50011

Yen Liu[‡]

NASA Ames Research Center, Moffett Field, CA 94035

An efficient implementation of the high-order spectral volume (SV) method is presented for multidimensional conservation laws on unstructured grids. In the SV method, each simplex grid cell is called a spectral volume (SV), and the SV is further subdivided into polygonal (2D), or polyhedral (3D) control volumes (CVs) to support high-order data reconstructions. In the traditional implementation, Gauss quadrature formulas are used to approximate the flux integrals on all faces. In the new approach, a near optimal nodal set is selected and used to reconstruct a high-order polynomial approximation for the flux vector, and then the flux integrals on the internal faces are computed analytically, without the need for Gauss quadrature formulas. This gives a great advantage over the traditional SV method in efficiency and ease of implementation. For SV interfaces, a quadrature free approach is compared with the Gauss quadrature approach to further evaluate the accuracy and efficiency. A simplified treatment of curved boundaries is also presented that avoids the need to store a separate reconstruction for each boundary cell. Fundamental properties of the new SV implementation are studied and high-order accuracy is demonstrated for linear and nonlinear advection equations, and the Euler equations. Several well-known inviscid flow test cases are utilized to show the effectiveness of the simplified curved boundary representation.

I.   Introduction

The spectral volume (SV) method is a recently developed finite volume method for hyperbolic conservation laws on unstructured grids.[1-6] The SV method belongs to a general class of Godunov-type finite volume method[7-8], which has been under development for several decades, and is considered to be the current state-of-the-art for the numerical solution of hyperbolic conservation laws. For a more detailed review of the literature on the Godunov-type method, refer to Wang[1], and the references therein. Many of the most popular numerical methods, such as the k-exact finite volume[9-10], the essentially non-oscillatory (ENO)[11-12], and weighted ENO[13] methods are also Godunov-type methods. The SV method is also closely related to the discontinuous Galerkin (DG)[14-17] method, a popular finite-element method for conservation laws. Both the SV and DG methods employ multiple degrees of freedom within a single element, but the SV method avoids the volume integral required in the DG method. Each simplex in the SV method utilizes a "structured" set of sub-cells, thus resulting in more cell faces where Riemann problems must be solved; however this inherent property of sub-cell resolution gives the SV method the ability to capture discontinuities with higher resolution than the DG method. For a more thorough comparison of the SV and DG methods, refer to Wang.[1]

The aforementioned additional cell faces present in the SV method number in the dozens per SV for 2D and in the hundreds per SV for 3D. In the traditional implementation of the SV method, all face integrals are computed by means of Gauss quadrature formulas, which are appropriate to the shape and dimension of the face. For example, a

[*]Graduate Research Assistant, 0237 Howe Hall, rharris@iastate.edu, AIAA Member.
[†]Associate Professor of Aerospace Engineering, 2271 Howe Hall, zjw@iastate.edu, Associate Fellow of AIAA.
[‡]Research Scientist, Yen.Liu@nasa.gov, Mail Stop T27B-1.

pentagonal face existing in the partition of a tetrahedral SV (3D) is split into 3 triangles. To carry out the integration, a Gauss quadrature formula of appropriate precision is then employed for each triangle. This procedure can be very efficient in 2D, where the required Gauss quadrature points number in the dozens per SV. However in 3D, the partition of a tetrahedron can be so complicated that thousands of Gauss quadrature points per SV may be necessary to compute the face integrals to the desired precision.

In this paper, we present two different approaches to handle the face integrals more efficiently. In each approach, a nodal set is defined within each SV to support a polynomial reconstruction for the flux vector. The shape functions defined by this nodal set are then integrated analytically over a standard element, resulting in an analytical representation for the flux integral on all faces. This reconstruction is universal for all SVs if a nodal set is distributed in a geometrically similar manner for all SVs. The first approach employs the above methodology for the internal faces only, while using the Gauss quadrature formulas for faces bounding an SV. This approach will be referred to as the partial quadrature (PQ) approach. The second approach employs the above methodology for all faces, and is thus referred to as the quadrature free (QF) approach. Both approaches are shown to be stable and convergent.

It is well known that high-order methods are very sensitive to the treatment of curved wall boundaries. It has been shown recently[18] that in the high-order SV method, unless care is taken to represent curved boundaries with high-fidelity, the solution accuracy may be reduced to low-order. The customary approach is to introduce higher order elements for cells on curved wall boundaries. This approach has been shown to produce desired results[18], but this necessitates storage of a separate reconstruction for each cell bounding a curved wall. For large problems, and especially in 3D, the memory requirement for this approach may be large. Recently, Krivodonova and Berger[19], and Luo et al.[32] have presented a simplified treatment for curved boundaries which utilizes a high-order approximation of the normal vector to the physical rather than computational geometry. This eliminates the need to store a separate reconstruction for wall boundary cells. We implement a similar technique in this paper, while maintaining the conservative property.

The paper is organized as follows. In Section 2, we review the basic formulation of the SV method. After that, the partial quadrature (PQ) and quadrature free (QF) approaches are described in detail in Section 3. Section 4 outlines the methodology for a simplified curved boundary representation. Numerical results including accuracy studies for linear and nonlinear advection, as well as the Euler equations are presented in Section 5. In addition, computations of inviscid flows over a circular cylinder, and a NACA 0012 airfoil are carried out to demonstrate the effectiveness of the curved boundary representation. Finally, conclusions and some possibilities for future work are summarized in Section 6.

II.   Review of the Basic Spectral Volume Method

Consider the multidimensional conservation law

$$\frac{\partial Q}{\partial t} + \frac{\partial f(Q)}{\partial x} + \frac{\partial g(Q)}{\partial y} + \frac{\partial h(Q)}{\partial z} = 0, \tag{1a}$$

on domain $\Omega \times [0,T]$ and $\Omega \subset R^3$ with the initial condition

$$Q(x,y,z,0) = Q_0(x,y,z), \tag{1b}$$

and appropriate boundary conditions on $\partial\Omega$. In (1), x,y, and z are the Cartesian coordinates and $(x,y,z) \in \Omega$, $t \in [0,T]$ denotes time, Q is the vector of conserved variables, and f, g and h are the fluxes in the x, y and z directions, respectively. Domain Ω is discretized into I nonoverlapping triangular (2D), or tetrahedral (3D) cells. In the SV method, the simplex grid cells are called SVs, denoted S_i, which are further partitioned into CVs, denoted $C_{i,j}$, which depend on the degree of the polynomial reconstruction. Examples of partitions supporting quadratic and cubic reconstructions are shown in Figure 1 (2D), and in Figure 2 (3D). Volume-averaged conserved variables on the CVs are then used to reconstruct a high-order polynomial inside the SV. To represent the solution as a polynomial of degree m, we need N pieces of independent information, or degrees of freedom (DOFs). Where N is calculated as follows:

$$N = \frac{(m+1)(m+2)\cdots(m+d)}{d!}, \tag{2}$$

where d is the spatial dimension of the problem. The DOFs in the SV method are the volume-averaged conserved variables at the N CVs. Define the CV-averaged conserved variable for $C_{i,j}$ as

$$\overline{Q}_{i,j} = \frac{1}{V_{i,j}} \int_{C_{i,j}} Q dV, \quad j=1,\ldots,N, \quad i=1,\ldots,I, \tag{3}$$

where $V_{i,j}$ is the volume of $C_{i,j}$. Given the CV-averaged conserved variables for all CVs in S_i, a polynomial $p_i(x,y,z)$ $\in P^m$ (the space of polynomials of at most degree m) can be reconstructed such that it is a $(m+1)^{th}$ order accurate approximation to $Q(x,y,z)$ inside S_i.

$$p_i(x,y,z) = Q(x,y,z) + O(h^{m+1}), \quad (x,y,z) \in S_i, \quad i=1,\ldots,I, \tag{4}$$

where h is the maximum edge length of all the CVs. This reconstruction can be solved analytically by satisfying the following conditions:

$$\frac{1}{V_{i,j}} \int_{C_{i,j}} p_i(x,y,z)dV = \overline{Q}_{i,j}, \quad j=1,\ldots,N. \tag{5}$$

This polynomial $p_i(x,y,z)$ is the $(m+1)^{th}$ order approximation we are looking for as long as the solution $Q(x,y,z)$ is smooth in the region covered by S_i. The reconstruction is expressed more conveniently as

$$p_i(x,y,z) = \sum_{j=1}^{N} L_j(x,y,z)\overline{Q}_{i,j}, \tag{6}$$

where $L_j(x,y,z) \in P^m$ are the shape functions which satisfy

$$\frac{1}{V_{i,j}} \int_{C_{i,j}} L_n(x,y,z)dV = \delta_{jn}. \tag{7}$$

Integrating (1) in $C_{i,j}$, we obtain

$$\frac{d\overline{Q}_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \int_{A_r} (F \cdot \hat{n})dA = 0, \quad j=1,\ldots,N, \quad i=1,\ldots,I, \tag{8}$$

where $F = (f,g,h)$, A_r represents the rth face of $C_{i,j}$, $\hat{n}$ is the outward unit normal vector of A_r, and K is the number of faces in $C_{i,j}$. More details of this, including representative plots of the shape functions can be found in Wang and Liu.[2] If F is a nonlinear function of the conserved variables, then the surface integration on each face is performed with a $(m+1)^{th}$-order-accurate Gauss quadrature formula; i.e.,

$$\int_{A_r} (F \cdot \hat{n})dA \cong \sum_{q=1}^{J} w_{rq}\hat{F}(Q(x_{rq},y_{rq},z_{rq})) \cdot \hat{n}_r A_r, \tag{9}$$

where $J=\text{integer}[(m+2)/2]$ is the number of quadrature points on the rth face in 2D, w_{rq} are the Gauss quadrature weights, $\hat{F}$ is a numerical flux, and (x_{rq},y_{rq},z_{rq}) are the Gauss quadrature points. For time integration, we use the third-order TVD Runge-Kutta scheme.[24]
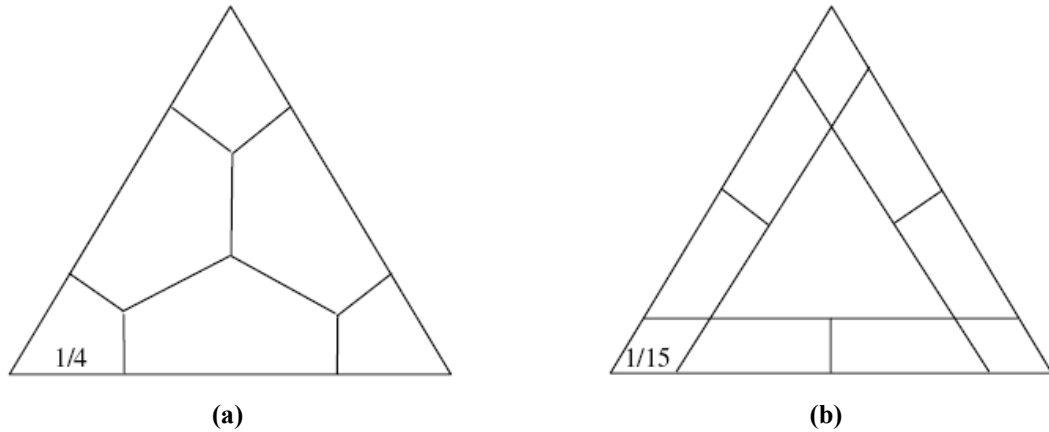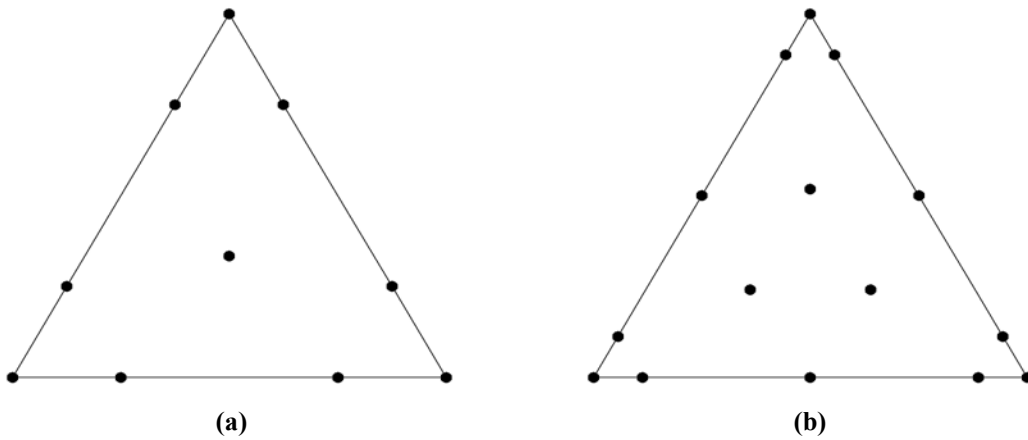

Figure 1. Partitions of a triangular SV supporting quadratic and cubic data reconstructions, shown in (a) and (b), respectively.

(a) **(b)**
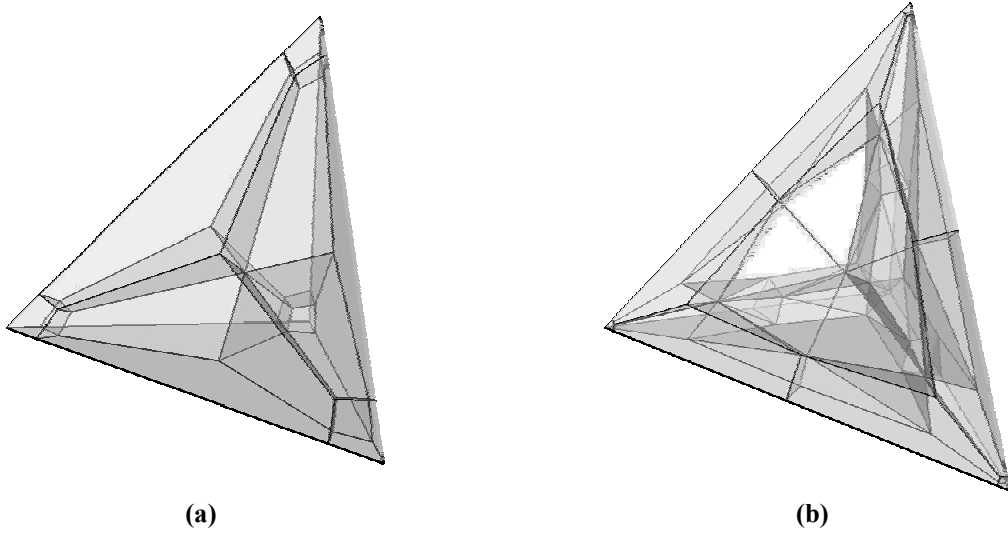
Figure 2. Partitions of a tetrahedral SV supporting quadratic and cubic data reconstructions, shown in (a) and (b), respectively.

III. Formulation of the Quadrature-Free Approach

In the new approach, a near optimal nodal set, such as those shown in Figure 3, is selected from Hesthaven.[21-22] This nodal set is then used to reconstruct a high-order polynomial approximation for the flux vector, and then the flux integrals are computed analytically, without the need for Gauss quadrature formulas. If we assume Q, f, g and h to be all polynomials in (1), obviously f, g and h should be one degree higher than Q. Therefore, a polynomial of degree $m+1$ is employed to reconstruct the flux vector.

(a) **(b)**

Figure 3. Nodal sets in a triangular SV supporting cubic and quartic data reconstructions for the flux vector, shown in (a) and (b), respectively.

The flux vector F can be computed at any point (x, y, z) by the following

$$F(x,y,z) = \sum_{i=1}^{N_S} M_i(x,y,z)F_i ,$$ (10)

where N_s is the number of nodes in the nodal set, calculated from (2), F_i is the flux vector evaluated at node i, and $M_i(x,y,z)$ are the shape functions defined by the nodal set which satisfy:

$$M_n\left(x_j, y_j, z_j\right) = \delta_{jn}.$$ (11)

Some representative 2D examples of the shape functions are shown in Figures 4 and 5.

American Institute of Aeronautics and Astronautics

Figure 4. Shape functions in a triangular SV supporting a cubic data reconstruction for the flux vector. The node which each function corresponds to is shown in bold.
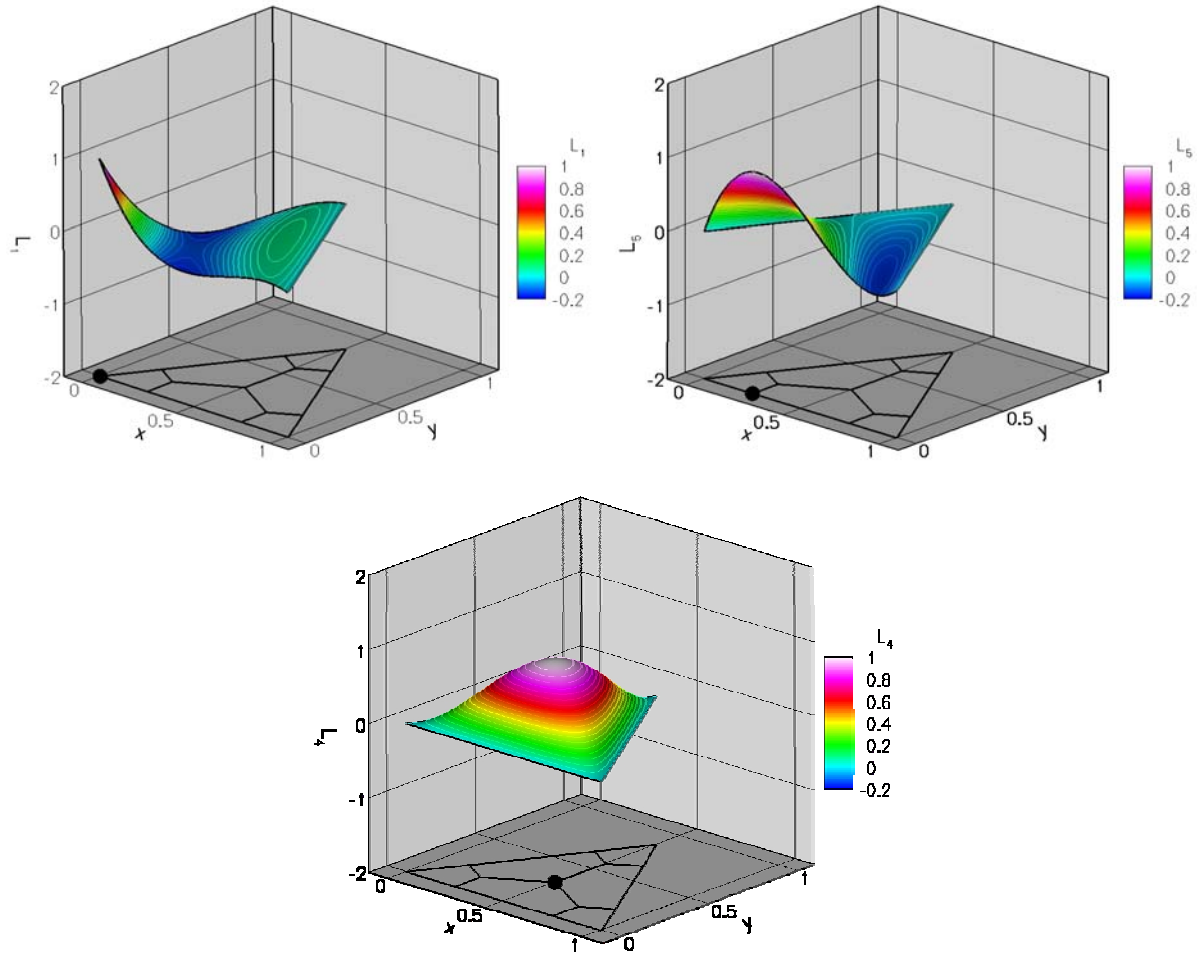
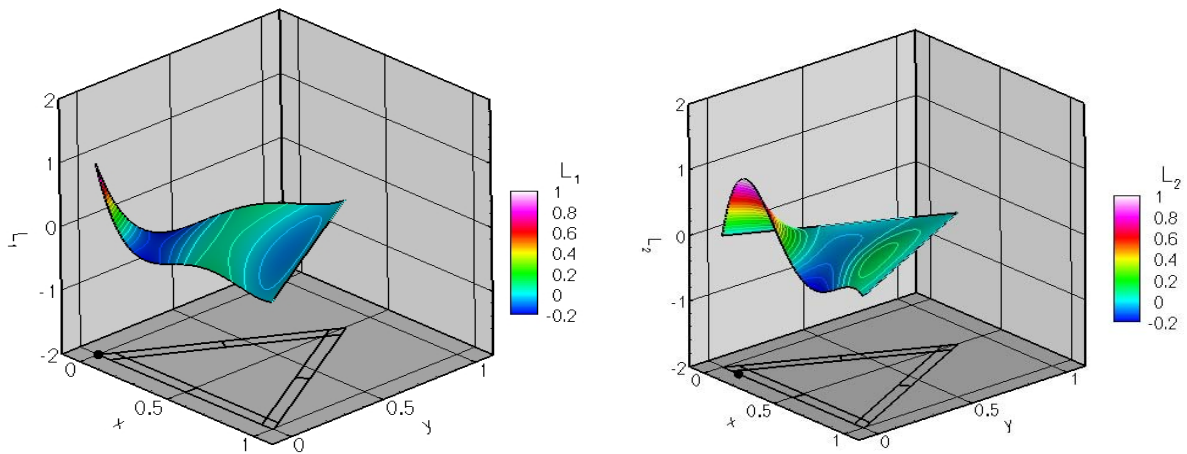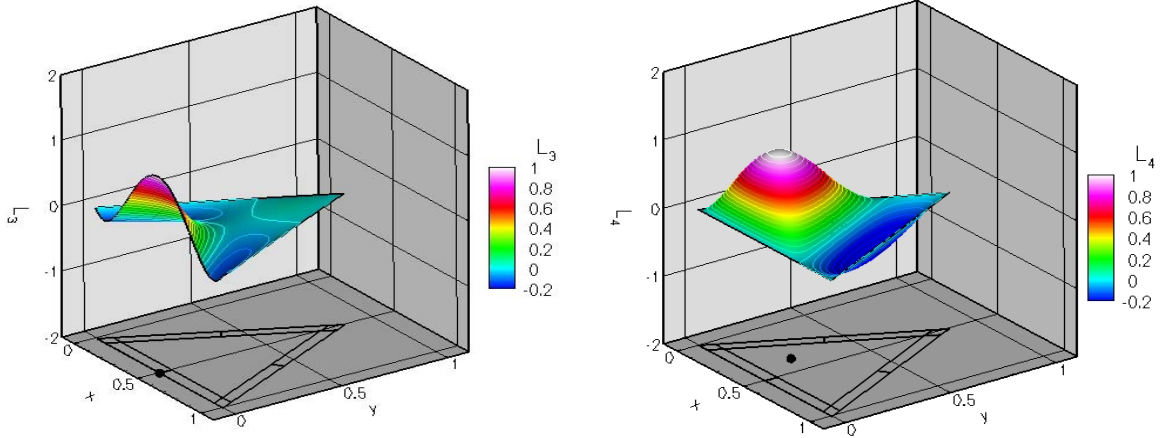American Institute of Aeronautics and Astronautics

Figure 5. Shape functions in a triangular SV supporting a quartic data reconstruction for the flux vector. The node which each function corresponds to is shown in bold.

In the PQ approach, (10) is integrated over each internal face, resulting in an analytical representation for the flux integral on all internal faces. This is done using a program capable of symbolic calculus, such as Mathematica.[25] This allows for the flux integral on each internal face to be computed as a weighted average of the flux evaluated at the nodal set. The face integrals on SV-bounding faces are then computed from (9), via the traditional approach. In the QF approach, the face integrals on internal faces are evaluated as in the PQ approach, but the SV-bounding faces require some care. Here, (6) and (10) are integrated over each SV-bounding CV face, and the resulting face integral can be expressed as the integral of a Riemann flux as follows:

$$\int_{A_r} (F \cdot \hat{n}) dA \cong \frac{A_r}{2} \left[\overline{F}_{n,L} + \overline{F}_{n,R} - \alpha_c \left(\overline{Q}_R - \overline{Q}_L \right) \right] \tag{12}$$

where $\overline{F}_{n,L}$ and $\overline{F}_{n,R}$ denote the face-averaged normal component of the flux vector due to the SV to the left and right of the interface, respectively, and α_c is taken as the maximum absolute eigenvalue as in the Rusanov Flux[26], which is evaluated at the face center. $\overline{Q}_R$ and $\overline{Q}_L$ are the face-averaged conserved variables due to the SV to the right and left of the interface, respectively. Eq. (12) can be deduced from the following. Since $p_i(x,y,z) = Q(x,y,z) + O(h^{m+1})$, the following is also true,

$$Q_R - Q = O(h^{m+1}),$$
$$Q_L - Q = O(h^{m+1}), \tag{13a}$$
$$Q_R - Q_L = O(h^{m+1}),$$

and the maximum absolute eigenvalue $\alpha \sim O(1)$, so we can say that

$$\alpha(Q_R - Q_L) \sim O(h^{m+1}) \tag{13b}$$

Integrating (13b) over a face, we obtain

$$\int_{A_r} \alpha(Q_R - Q_L) dA = A_r \alpha_c \left(\overline{Q}_R - \overline{Q}_L \right) + O\left(A_r h^{m+1} \right) \tag{14}$$

At each node shown in Figure 3, the flux is computed based on the reconstructed solution polynomial. Therefore, the flux error at each node of the nodal set is of order $O(h^{m+1})$, i.e.,

$$F_{n,L} = F_n + O(h^{m+1}),$$
$$F_{n,R} = F_n + O(h^{m+1}), \tag{15}$$
$$(F_{n,L} + F_{n,R})/2 = F_n + O(h^{m+1}),$$

where F_n is the (unknown) "exact" normal flux at the face. Integrating the 3rd equation in (15) over a face, we obtain

$$\int_{A_r} \tfrac{1}{2} \left(F_{n,L} + F_{n,R} \right) dA = \tfrac{1}{2} A_r \left(\overline{F}_{n,L} + \overline{F}_{n,R} \right) = F_n A_r + O\left(A_r h^{m+1} \right) \tag{16}$$

So, the surface integral in (8) can be expressed by adding together (14) and (16) to get

$$\int_{A_r} (F \cdot \hat{n}) dA = \frac{A_r}{2} \left[\overline{F}_{n,L} + \overline{F}_{n,R} - \alpha_c (\overline{Q}_R - \overline{Q}_L) \right]_r + O(A_r h^{m+1}) \tag{17}$$

If F=constant, the following identity exists:

$$\sum_{r=1}^{K} \int_{A_r} (F \cdot \hat{n}) dA = 0. \tag{18}$$

Therefore, we will gain an extra order of accuracy if we sum up the surface integrals for the faces of $C_{i,j}$; i.e.,

$$\sum_{r=1}^{K} \int_{A_r} (F \cdot \hat{n}) dA = \sum_{r=1}^{K} \frac{A_r}{2} \left[\overline{F}_{n,L} + \overline{F}_{n,R} - \alpha_c (\overline{Q}_R - \overline{Q}_L) \right]_r + O(A_r h^{m+2}). \tag{19}$$

Since $O(V_{i,j}) = O(A_r h)$, we have

$$\frac{d\overline{Q}_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \frac{A_r}{2} \left[\overline{F}_{n,L} + \overline{F}_{n,R} - \alpha_c (\overline{Q}_R - \overline{Q}_L) \right]_r = O(h^{m+1}). \tag{20}$$

Thus, if the surface integral in (8) is evaluated using (12), spatial accuracy of order $m+1$ is assured.

To summarize, the first and second terms in (12) are computed as a weighted average of the flux evaluated at the nodal set of the left, and right SV, respectively, and the third and fourth terms are computed as a weighted average of the cell-averaged conserved variables of the right and left SV, respectively. In practice, face-averaged shape functions are computed during preprocessing for a standard element and then multiplied by the physical face area for computation. This eliminates the need to store all of the Jacobians.

In a quick comparison of the approaches, a 2D 3rd order SV method using the traditional "full quadrature" approach with 2 quadrature points per face necessitates 36 flux calculations per SV, the PQ approach requires 28 flux calculations per SV, and the QF approach requires only 10 flux calculations per SV. Similarly, for a 4th order SV method, the number of flux calculations per SV are 54, 39, and 15 for the "full quadrature", PQ, and QF approaches, respectively. So, the PQ and QF approaches are slightly more efficient than the "full quadrature" approach in 2D, but are much less costly to evaluate in 3D where the faces number in the hundreds.

IV.   Formulation of Curved Boundary Representation

One of the biggest advantages of high-order methods over low-order methods is their ability to achieve better solution accuracy using far fewer degrees-of-freedom. However, for problems involving curved wall boundaries which are represented by line-segments or planar-facets, an inordinate amount of elements may be necessary just to preserve the geometry with adequate precision. A much more desirable approach is to represent curved boundaries with high-order polynomials that are compatible with the order of the data reconstruction. Favorable results have been obtained[18] when applying this technique to the SV method. One drawback of this approach is that it necessitates computation and storage of a separate reconstruction for each and every wall-bounding cell. For large problems, and especially in 3D, the memory requirement can become large. Here we advocate an approach in which the geometry is represented with high fidelity, but a separate reconstruction for every wall-bounding cell is avoided. Similar approaches have been recently presented by Krivodonova and Berger[19], and Luo et al.[32] The basic idea employed here is that the flux integrals are carried out on the physically curved faces using the polynomial reconstruction from a linear SV, as opposed to a curved SV. This allows us to employ the same reconstruction for the SVs on curved boundaries that is already in use for all other SVs. Since the curvature will usually be very small, this is a reasonable approximation.

This procedure involves a number of steps. First, a set of points must be defined for each SV face that will be used to construct a polynomial representation for that face. Wang and Liu[18] showed that a quadratic boundary representation is sufficient for both 3rd and 4th order simulations, so a quadratic boundary is also utilized in this study. Thus, 3 points are needed to define a quadratic polynomial for each SV face in 2D. Two of the points are taken as the SV face endpoints, while the 3rd is computed from a cubic polynomial fit between the SV face endpoints and the opposite endpoints of both neighboring SV faces. If the SV face in question is at the end of the curved wall, or at a sharp trailing edge, then the 3rd point is computed from a quadratic fit between the two SV face endpoints and the opposite endpoint of the adjacent SV face. A schematic of this arrangement is shown in Figure 6 for clarity.
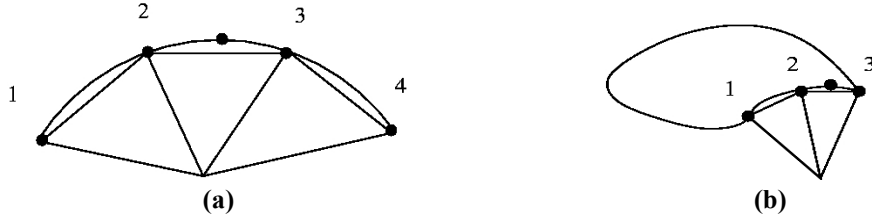
Figure 6. Stencil for computation of 3rd point for quadratic polynomial: (a) 4 points used to fit cubic polynomial; (b) 3 points used to fit quadratic polynomial.

Following the usual practice[18], isoparametric SVs can be used to map SVs with curved boundaries into the standard SV. Assume that a one-to-one transformation exists between a general SV in the physical space (x, y) and the standard triangle in the computational domain (ξ, η) as shown in Figure 7, i.e.,

$$\xi = \xi(x, y),$$
$$\eta = \eta(x, y).$$

$$(21)$$

The partition of the SV is performed in the standard triangle, and the partition in the physical domain is the result of the inverse transformation from the computational domain back to the physical domain as shown in Figure 8.

$$x = x(\xi, \eta),$$
$$y = y(\xi, \eta).$$

$$(22)$$

Therefore, a necessary condition for a valid transformation is that the inverse transformation exists.
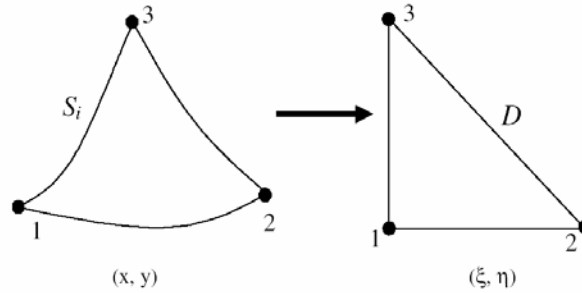
Figure 7. Transformation of a general SV in the physical domain to the standard triangle in the computational domain.
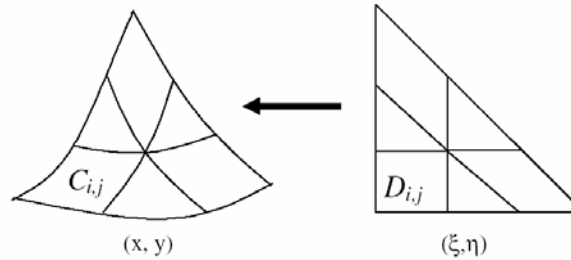
Figure 8. Transformation of the partition from the standard triangle in the computational domain to the general SV in the physical domain.

In the most general case, a quadratic isoparametric SV requires the specification of 6 nodes, as shown in Figure 9a. The transformation can be expressed as

$$r = \sum_{j=1}^{m} C_j(\xi, \eta) r_j,$$

$$(23)$$

where $r=(x, y)$. The shape functions for a quadratic SV in 2D can be written as

$$C_1(\xi,\eta) = (1 - \xi - \eta)(1 - 2\xi - 2\eta),$$
$$C_2(\xi,\eta) = \xi(2\xi - 1),$$
$$C_3(\xi,\eta) = \eta(2\eta - 1),$$
$$C_4(\xi,\eta) = 4\xi\eta,$$
$$C_5(\xi,\eta) = 4\xi(1 - \xi - \eta),$$
$$C_6(\xi,\eta) = 4\eta(1 - \xi - \eta). \tag{24}$$

In most cases, however, it is not necessary to use the most general isoparametric SV since only one boundary of the SV is usually curved as shown in Figure 9b. As a result, simplified curved SVs (SCSV) can be used. The shape functions for the quadratic SCSV can be easily found to be

$$C_1(\xi,\eta) = 1 - 3\xi + 2\xi(\xi + \eta) - \eta,$$
$$C_2(\xi,\eta) = -\xi + 2\xi(\xi + \eta),$$
$$C_3(\xi,\eta) = \eta,$$
$$C_4(\xi,\eta) = 4\xi(1 - \xi - \eta). \tag{25}$$

Since we are not computing a new reconstruction based on the isoparametric SVs, the purpose of this transformation is to modify the SVs such that the endpoints of the faces of wall-bounding CVs lie on the curved SV face. This is shown in Figure 10 for clarity. The node locations and face endpoints are computed using (23) and (25), and the face endpoints are all connected with straight lines. Although the CV faces themselves remain linear, with the CV face endpoints situated on the curved SV face, the flux integration can be carried out along the curved face and still satisfy conservation. Since the reconstruction is only universal for geometrically similar partitions and node distributions, there is some error inherent to the geometrical dissimilarity of the SVs on curved wall boundaries. In most situations, the SVs on curved wall boundaries will be small compared to all other SVs, and accordingly the degree of nonlinearity of these SV faces will also be small. It is thus reasoned that any error introduced in the reconstruction due to the presence of geometrically dissimilar SVs on curved wall boundaries will likely be small.

(a)   (b)

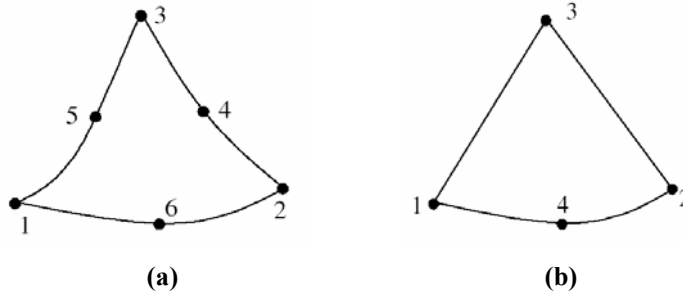**Figure 9. Quadratic (a) and simplified quadratic (b) SVs with three and one curved boundaries, respectively.**

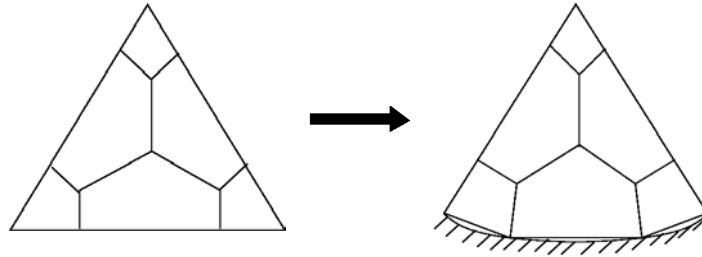**Figure 10. Traditional SV partition transformed to accommodate curved-wall boundary.**

Let the equation of the rth wall-bounding face of $C_{i,j}$ in the standard SV be

$$\eta = 0, \quad \xi_{r1} < \xi < \xi_{r2}. \tag{26}$$

Then the surface integral in (8) can be written as

$$\int_{A_r}(F \cdot \hat{n})dA = \int_{\xi_{r1}}^{\xi_{r2}} F \cdot \left(\frac{dy}{d\xi}, -\frac{dx}{d\xi} \right)d\xi. \tag{27}$$

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

$$\int_{\xi_{r1}}^{\xi_{r2}} F \cdot \left(\frac{dy}{d\xi}, -\frac{dx}{d\xi} \right)d\xi = (\xi_{r2} - \xi_{r1})\sum_{q=1}^{J} w_{rq} F_N(\xi_{rq}), \tag{28}$$

where w_{rq} represent the Gauss quadrature weights, and $F_N = F \cdot \left(\frac{dy}{d\xi}, -\frac{dx}{d\xi} \right)$ denotes the dot product of the flux

vector with the area normal vector. To preserve a uniform free stream (Q = constant), the surface integral must satisfy the following identity

$$\int_{A_r}\hat{n}dA = \int_{\xi_{r1}}^{\xi_{r2}} \left(\frac{dy}{d\xi}, -\frac{dx}{d\xi} \right)d\xi . \tag{29}$$

It is obvious that the unit normal vectors for the curved face are polynomials one order lower than the transformation polynomials (22). Therefore, a one-point Gauss formula is sufficient to preserve a free stream for a quadratic boundary representation. If the flux function is of the same order polynomial as the solution, more quadrature points are required to exactly integrate the flux over curved boundaries. Here, a two-point Gauss formula is employed for all cases.

V.   Numerical Tests

In this section, the efficient SV implementation is evaluated for 2D scalar advection equations, and for the 2D Euler equations. Both 3rd-order and 4th-order partitions are evaluated. The 3rd-order partition, and the corresponding nodal set to support a cubic reconstruction for the flux vector are shown in Figures 1a, and 3a, respectively. The 4th-order partition, and the corresponding nodal set to support a quartic reconstruction for the flux vector are shown in Figures 1b, and 3b, respectively. The numerical error is assessed using both regular and irregular grids, as shown in Figure 11. The finer grids are generated recursively by cutting each courser grid cell into four finer grid cells. The third-order TVD Runge-Kutta scheme is used for time integration in all cases. Also, all errors presented are time-step independent because the time step Δt was made small enough so that the errors are dominated by the spatial discretization. The initial CV-averaged solutions were computed using the CV-averaged node-based shape functions. In this way, the solution is initialized at the nodal set, and the CV-averaged solutions are computed as a weighted average of these nodal values. This avoids the need for a high-order Gauss quadrature initialization. The solution is taken as converged when the L_2 norm of the residual is reduced by 8 orders of magnitude in all cases. The L_2 norm of the density residual is used for the Euler equation simulations.
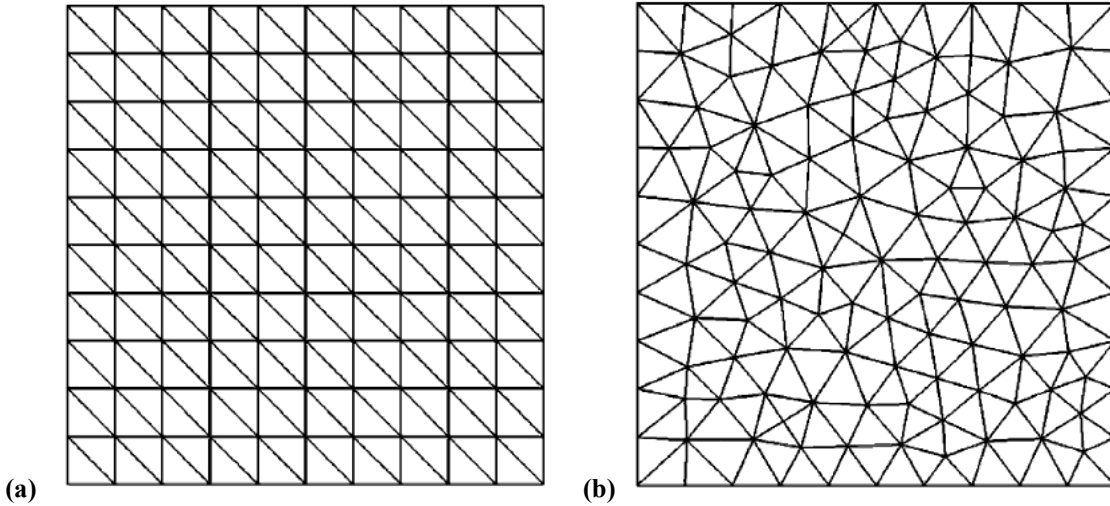

(a)   (b)

Figure 11. Regular and irregular grids for validation cases: (a) regular (10x10x2); (b) irregular (10x10x2).

A. Accuracy study with linear advection equation

In this case, we test the accuracy of the SV method on the 2D linear advection equation:

$$\frac{\partial Q}{\partial t} + \frac{\partial Q}{\partial x} + \frac{\partial Q}{\partial y} = 0, \qquad -1 \le x \le 1, \qquad -1 \le y \le 1, \qquad (30)$$

$$Q(x,y,0) = \sin \pi (x+y), \quad \text{periodic boundary condition.}$$

The numerical solutions were carried out until $t=1s$ for this time-accurate problem. In tables I-II, we present the L_1 and L_∞ errors and orders for regular and irregular grids, respectively. Grid refinement is carried out until both the L_1 and L_∞ orders are sufficiently demonstrated. For this case, the PQ results are identical to the QF results, as expected for a linear equation, so just the QF results are presented. The formal order of accuracy is observed, using both regular and irregular grids.

Table I
Accuracy of 2D linear advection equation at $t=1$ s on regular grids

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
3	10 x 10 x 2	1.33e-3	-	3.29e-3	-
	20 x 20 x 2	1.73e-4	2.94	4.47e-4	2.88
	40 x 40 x 2	2.19e-5	2.98	5.77e-5	2.95
	80 x 80 x 2	2.76e-6	2.99	7.29e-6	2.98
4	10 x 10 x 2	7.10e-5	-	2.44e-4	-
	20 x 20 x 2	4.38e-6	4.02	1.56e-5	3.97
	40 x 40 x 2	2.73e-7	4.00	9.78e-7	3.99
	80 x 80 x 2	1.71e-8	4.00	6.12e-8	4.00

Table II
Accuracy of 2D linear advection equation at $t=1$ s, on irregular grids

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
3	10 x 10 x 2	3.12e-3	-	1.57e-2	-
	20 x 20 x 2	4.09e-4	2.93	1.93e-3	3.02
	40 x 40 x 2	5.36e-5	2.93	3.37e-4	2.52
	80 x 80 x 2	6.99e-6	2.94	6.18e-5	2.45
4	10 x 10 x 2	2.62e-4	-	1.50e-3	-
	20 x 20 x 2	1.56e-5	4.07	1.13e-4	3.73
	40 x 40 x 2	9.67e-7	4.02	8.23e-6	3.77
	80 x 80 x 2	6.03e-8	4.00	5.38e-7	3.94

B. Accuracy study with Burgers equation

In this case, we test the accuracy of the SV method on the 2D Burgers equation:

$$\frac{\partial Q}{\partial t} + \frac{\partial Q^2/2}{\partial x} + \frac{\partial Q^2/2}{\partial y} = 0, \qquad -1 \le x \le 1, \qquad -1 \le y \le 1, \qquad (31)$$

$$Q(x,y,0) = \frac{1}{4} + \frac{1}{2}\sin \pi (x+y), \quad \text{periodic boundary condition.}$$

The numerical solutions were carried out until $t=0.1s$ for this time-accurate problem. At this time the solution is still smooth, so data limiting is not necessary. In tables III-IV, we present the L_1 and L_∞ errors and orders for regular and

irregular grids, respectively. Grid refinement is carried out until both the L_1 and L_∞ orders are sufficiently demonstrated. Results from the PQ and QF approaches are not identical for this case, but they are very similar, so again only the QF results are shown. The formal order of accuracy is again observed, using both regular and irregular grids.

Table III
Accuracy of 2D Burgers equation at t=0.1 s on regular grids

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
3	10 x 10 x 2	3.96e-4	-	1.63e-3	-
	20 x 20 x 2	6.49e-5	2.61	3.97e-4	2.03
	40 x 40 x 2	1.03e-5	2.66	7.01e-5	2.50
	80 x 80 x 2	1.55e-6	2.73	1.12e-5	2.65
4	10 x 10 x 2	3.28e-5	-	3.77e-4	-
	20 x 20 x 2	2.39e-6	3.78	3.26e-5	3.53
	40 x 40 x 2	1.74e-7	3.77	2.89e-6	3.49
	80 x 80 x 2	1.23e-8	3.83	2.08e-7	3.80

Table IV
Accuracy of 2D Burgers equation at t=0.1 s on irregular grids

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
3	10 x 10 x 2	6.42e-4	-	3.35e-3	-
	20 x 20 x 2	1.17e-4	2.45	9.96e-4	1.75
	40 x 40 x 2	1.88e-5	2.65	2.66e-4	1.90
	80 x 80 x 2	2.91e-6	2.69	6.12e-5	2.12
	160 x 160 x 2	4.41e-7	2.72	1.14e-5	2.43
	320 x 320 x 2	6.51e-8	2.76	1.88e-6	2.59
4	10 x 10 x 2	8.35e-5	-	1.16e-3	-
	20 x 20 x 2	6.20e-6	3.75	1.43e-4	3.02
	40 x 40 x 2	4.57e-7	3.76	1.26e-5	3.51
	80 x 80 x 2	3.28e-8	3.80	1.06e-6	3.56

C. Accuracy study with vortex propagation problem

The unsteady 2D Euler equations in conservative form can be written as

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = 0 \ , \tag{32a}$$

where Q is the vector of conserved variables, F and G are the inviscid flux vectors given below:

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ E \end{Bmatrix}, \qquad F = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ u(E+p) \end{Bmatrix}, \qquad G = \begin{Bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ v(E+p) \end{Bmatrix} . \tag{32b}$$

Here ρ is the density, u and v are the velocity components in x and y directions, p is the pressure, and E is the total energy. The pressure is related to the total energy by

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho\left(u^2 + v^2\right) , \tag{32c}$$

with ratio of specific heats γ. In all of the simulations in this paper, γ is taken to be 1.4.

This is an idealized problem for the Euler equations in 2D. the mean flow is $\{\rho, u, v, p\} = \{1,1,1,1\}$. An isotropic vortex is then added to the mean flow, i.e. with perturbations in u, v, and temperature $T = p/\rho$, and no perturbation in entropy $S = p/\rho^\gamma$:

$$(\delta u, \delta v) = \frac{\varepsilon}{2\pi} e^{0.5(1-r^2)}(-\bar{y}, \bar{x}),$$

$$\delta T = -\frac{(\gamma-1)\varepsilon^2}{8\gamma\pi^2} e^{1-r^2},$$

$$\delta S = 0,$$

where $r^2 = \bar{x}^2 + \bar{y}^2$, $\bar{x} = x - 5$, $\bar{y} = y - 5$, and the vortex strength $\varepsilon = 5$. If the computational domain extends to infinity, the exact solution of the Euler equations with the above initial conditions is just the passive convection of the isotropic vortex with mean velocity (1, 1). In the following accuracy study, the computational domain is taken to be [0, 10] x [0, 10], with characteristic inflow and outflow boundary conditions imposed on the boundaries.

The numerical solution is carried out until $t = 2$ on both regular and irregular grids. No limiters are employed in this study since the problem is smooth, and the Rusavov flux is used in all simulations. In tables V-VI, we present

Table V
Accuracy of 2D vortex propagation problem at t=2 s on regular grids using PQ approach

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
3	10 x 10 x 2	1.41e-03	-	4.09e-02	-
	20 x 20 x 2	2.51e-04	2.49	4.88e-03	3.06
	40 x 40 x 2	4.30e-05	2.54	1.05e-03	2.22
	80 x 80 x 2	6.88e-06	2.65	1.95e-04	2.42
	160 x 160 x 2	1.05e-06	2.71	3.16e-05	2.63
4	10 x 10 x 2	6.08e-04	-	9.98e-03	-
	20 x 20 x 2	4.95e-05	3.62	1.12e-03	3.16
	40 x 40 x 2	3.03e-06	4.03	1.14e-04	3.30
	80 x 80 x 2	1.92e-07	3.98	7.28e-06	3.96
	160 x 160 x 2	1.36e-08	3.82	6.22e-07	3.55

Table VI
Accuracy of 2D vortex propagation problem at t=2 s on regular grids using QF approach

Order of accuracy	Grid	L_1 error	L_1 order	L_∞ error	L_∞ order
3	10 x 10 x 2	1.34e-03	-	3.90e-02	-
	20 x 20 x 2	2.50e-04	2.42	5.65e-03	2.79
	40 x 40 x 2	4.28e-05	2.55	1.16e-03	2.29
	80 x 80 x 2	6.70e-06	2.68	2.01e-04	2.52
	160 x 160 x 2	1.01e-06	2.73	3.17e-05	2.67
4	10 x 10 x 2	5.39e-04	-	9.33e-03	-
	20 x 20 x 2	4.64e-05	3.54	9.14e-04	3.35
	40 x 40 x 2	2.80e-06	4.05	8.74e-05	3.39
	80 x 80 x 2	1.71e-07	4.03	5.25e-06	4.06
	160 x 160 x 2	1.19e-08	3.85	4.78e-07	3.46

the L_1 and L_∞ errors and orders in the CV-averaged density for the regular grids using the PQ and QF approaches, respectively. Tables VII-VIII show the L_1 and L_∞ errors and orders for the irregular grids using the PQ and QF approaches, respectively. Comparison of the PQ and QF approaches for this case show very similar behavior. The formal order of accuracy is demonstrated for all cases.

Table VII
Accuracy of 2D vortex propagation problem at t=2 s on irregular grids using PQ approach

Order of accuracy	Grid	L_1 error	L_1 order	$L\infty$ error	$L\infty$ order
3	10 x 10 x 2	1.11e-03	-	2.67e-02	-
	20 x 20 x 2	1.85e-04	2.58	3.23e-03	3.04
	40 x 40 x 2	3.00e-05	2.63	6.91e-04	2.23
	80 x 80 x 2	4.76e-06	2.66	1.32e-04	2.39
	160 x 160 x 2	7.36e-07	2.69	1.95e-05	2.75
4	10 x 10 x 2	3.46e-04	-	8.38e-03	-
	20 x 20 x 2	3.07e-05	3.49	4.52e-04	4.21
	40 x 40 x 2	2.07e-06	3.89	4.87e-05	3.21
	80 x 80 x 2	1.42e-07	3.87	3.89e-06	3.65
	160 x 160 x 2	9.49e-09	3.90	3.66e-07	3.41

Table VIII
Accuracy of 2D vortex propagation problem at t=2 s on irregular grids using QF approach

Order of accuracy	Grid	L_1 error	L_1 order	$L\infty$ error	$L\infty$ order
3	10 x 10 x 2	1.05e-03	-	2.83e-02	-
	20 x 20 x 2	1.86e-04	2.49	3.58e-03	2.98
	40 x 40 x 2	3.01e-05	2.63	7.19e-04	2.32
	80 x 80 x 2	4.67e-06	2.69	1.33e-04	2.44
	160 x 160 x 2	7.15e-07	2.71	2.16e-05	2.62
4	10 x 10 x 2	3.50e-04	-	6.34e-03	-
	20 x 20 x 2	2.92e-05	3.58	4.40e-04	3.85
	40 x 40 x 2	2.03e-06	3.85	4.50e-05	3.29
	80 x 80 x 2	1.36e-07	3.90	3.98e-06	3.50
	160 x 160 x 2	9.09e-09	3.91	4.29e-07	3.22

D. Subsonic flow over a circular cylinder

A subsonic inviscid flow around a circular cylinder at Mach = 0.3 is selected to test the new simplified curved boundary representation. Due to flow symmetry, only the top half of the physical domain is selected as the computational domain although the complete physical domain is displayed in all of the figures. The far field boundary is located 20 diameters away from the cylinder center and the characteristic inflow/outflow boundary conditions are used at the far field boundary. In order to perform a grid refinement accuracy study, four different triangular grids were generated from four structured grids with 16x4, 32x8 , 64x16 and 128x32 cells, which are shown in Figure 12. The solution is initialized to a uniform free stream. All simulations employ the simplified curved boundary treatment given in Section IV, and the PQ approach. The simulations were also run using a piece-wise linear boundary representation, but the solution quickly diverged in all cases. Also, even with the curved boundary treatment, the Rusanov flux can become unstable due to spurious entropy production at the trailing edge. This is overcome by the use of either the Roe[27], or HLLC[28-31] flux. Both are convergent and yield similar results. The HLLC flux is advocated here because of its improved efficiency over the Roe flux.
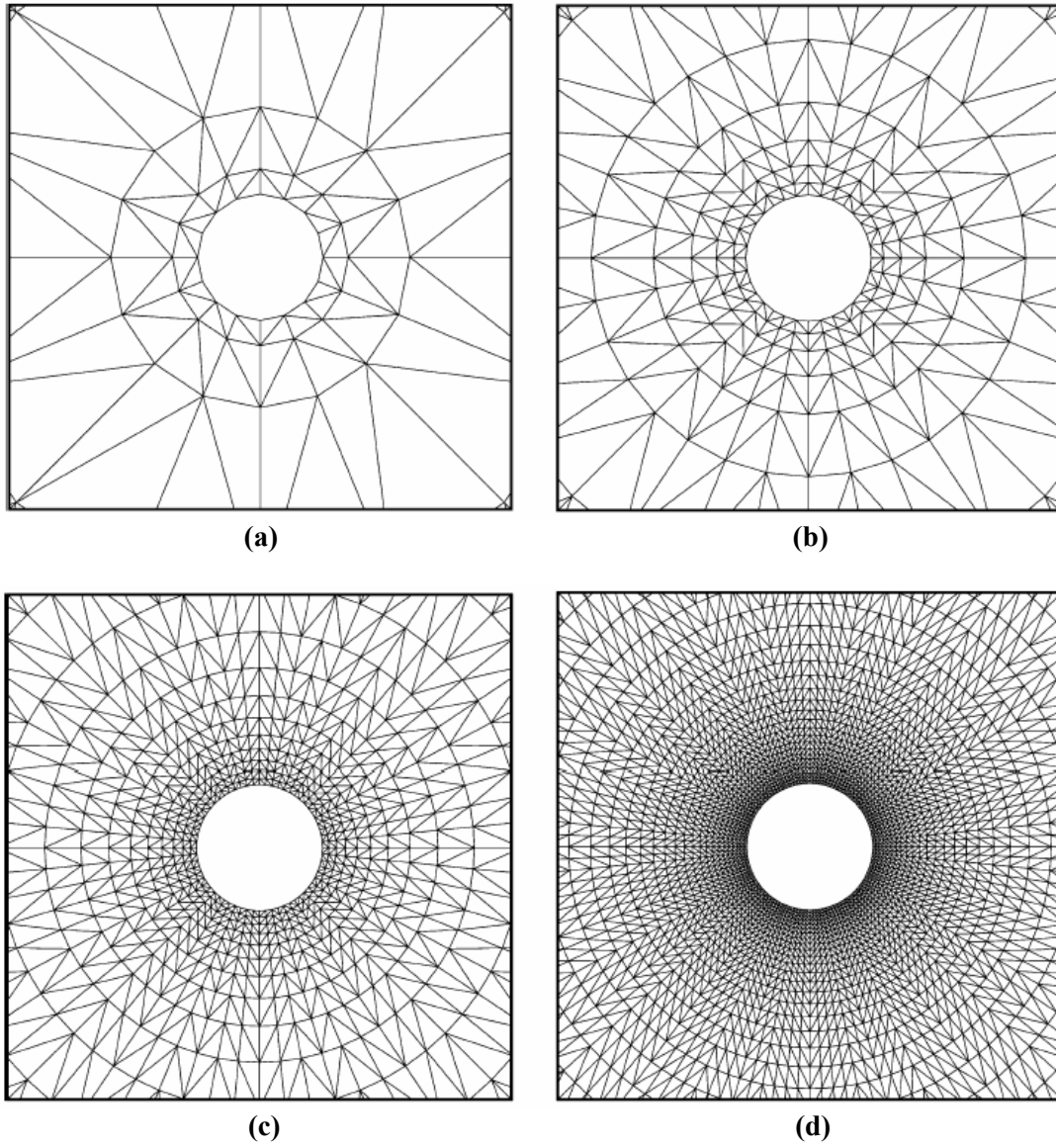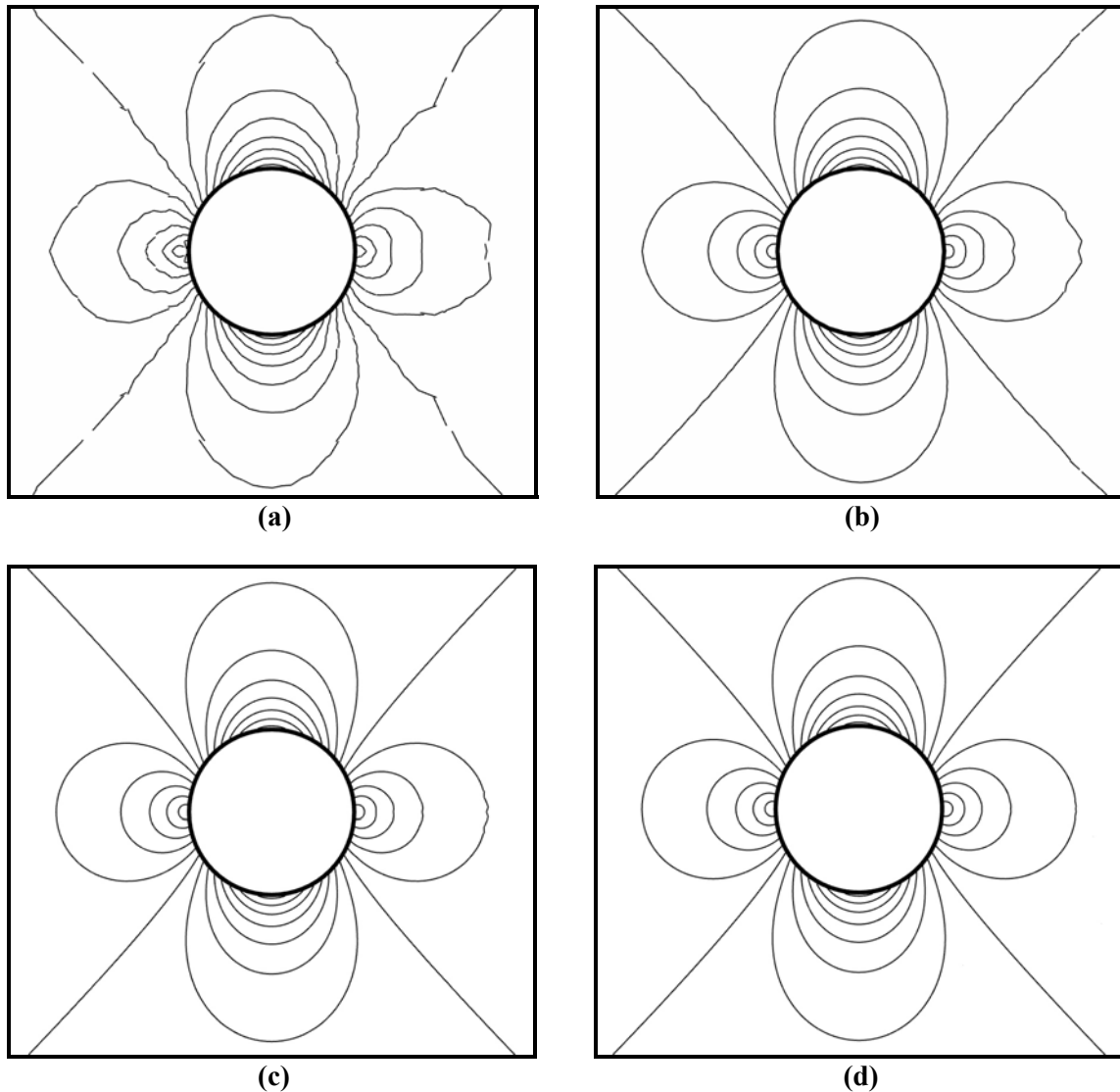
Figure 12. Computational grids for flow over a circular cylinder. (a) 16x4x2 cells. (b) 32x8x2 cells. (c) 64x16x2 cells. (d) 128x32x2 cells.
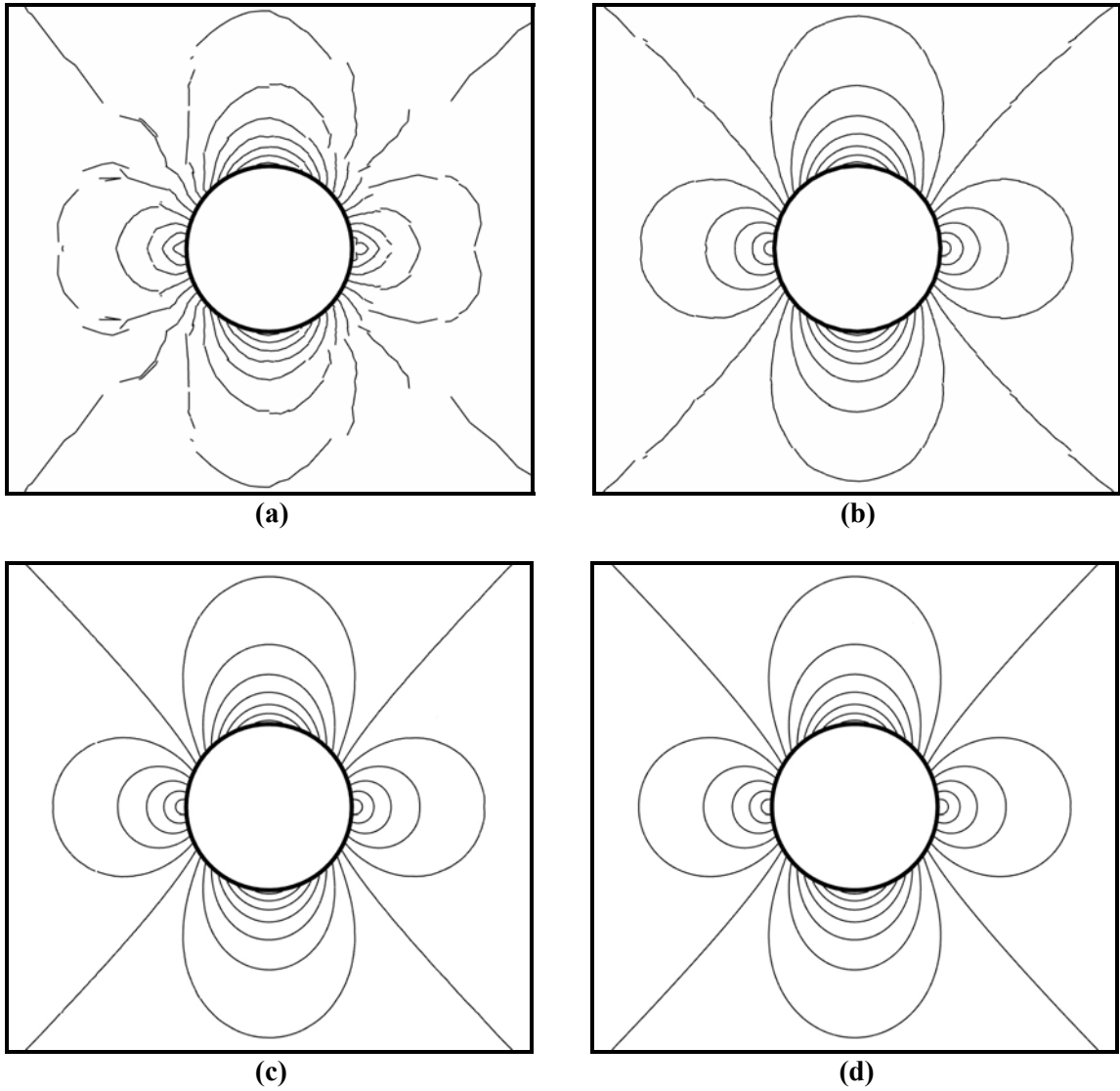
Figure 13. Mach contours computed with 3rd order PQ approach using simplified curved boundary treatment on four different grids. (a) 16x4x2 cells. (b) 32x8x2 cells. (c) 64x16x2 cells. (d) 128x32x2 cells.

Figure 14. Mach contours computed with 4ᵗʰ order PQ approach using simplified curved boundary treatment on four different grids. (a) 16x4x2 cells. (b) 32x8x2 cells. (c) 64x16x2 cells. (d) 128x32x2 cells.

Figures 13 and 14 show fourteen even contours of Mach number between 0 and 0.65. The coarse grids show some discontinuous contours and a numerical wake at the trailing edge. As the grid is refined, the contours become continuous and the flow becomes perfectly symmetric with respect to both axes.

E. Subsonic flow over a NACA 0012 airfoil

As a final demonstration for a more realistic geometry, subsonic flow around a NACA 0012 airfoil at Mach = 0.4, and angle of attack of 5 degrees is simulated. In this simulation, the 3ʳᵈ order SV scheme is tested using a very course mesh with 48x16x2 triangles, as shown in Figure 15, to further demonstrate the effectiveness of the curved boundary treatment. The outer boundary is 20 chords away from the center of the airfoil. This simulation was also carried out using a linear boundary representation. Thirty-three even pressure contours between 0.5 and 0.8 computed using both a linear and quadratic boundary representation are shown in Figures 16a and 16b, respectively. Figures 17a and 17b show thirty-three even contours of Mach number between 0 and 0.8 using a linear and quadratic boundary representation, respectively. Spurious entropy production at the airfoil surface in the linear

boundary representation has caused the pressure and Mach contours to be heavily oscillatory. This effect is dramatically lessened when using a quadratic boundary representation as shown in Figures 16b and 17b.
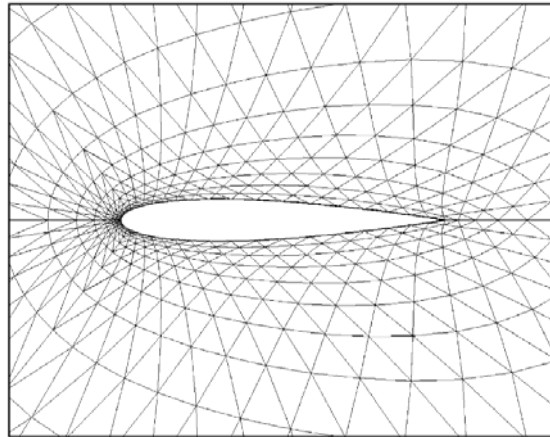

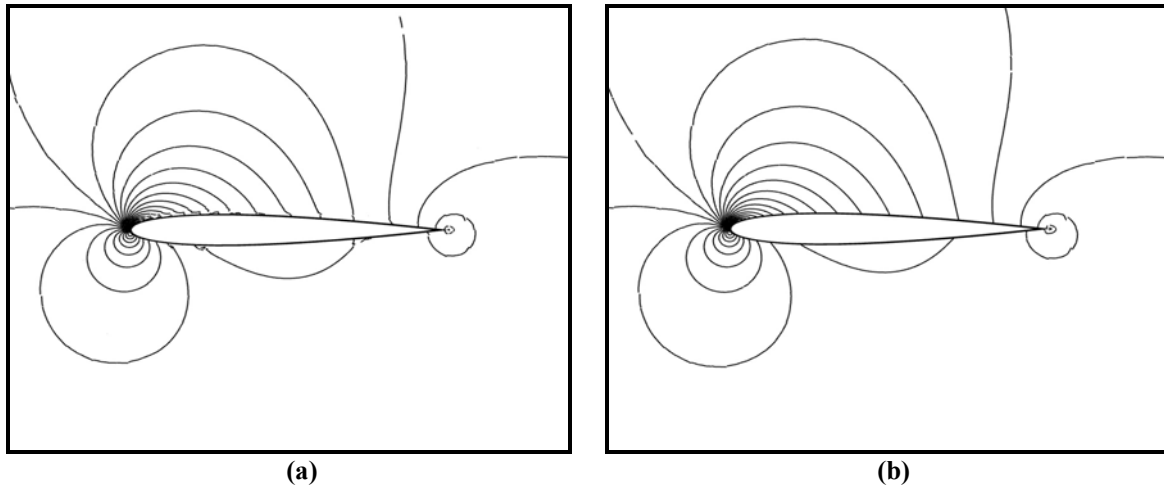**Figure 15. Computational grid for flow over a NACA 0012 airfoil (48x16x2 cells).**

| (a) | (b) |

Figure 16. Pressure contours for 3rd-order PQ approach: (a) linear boundary; (b) quadratic boundary.
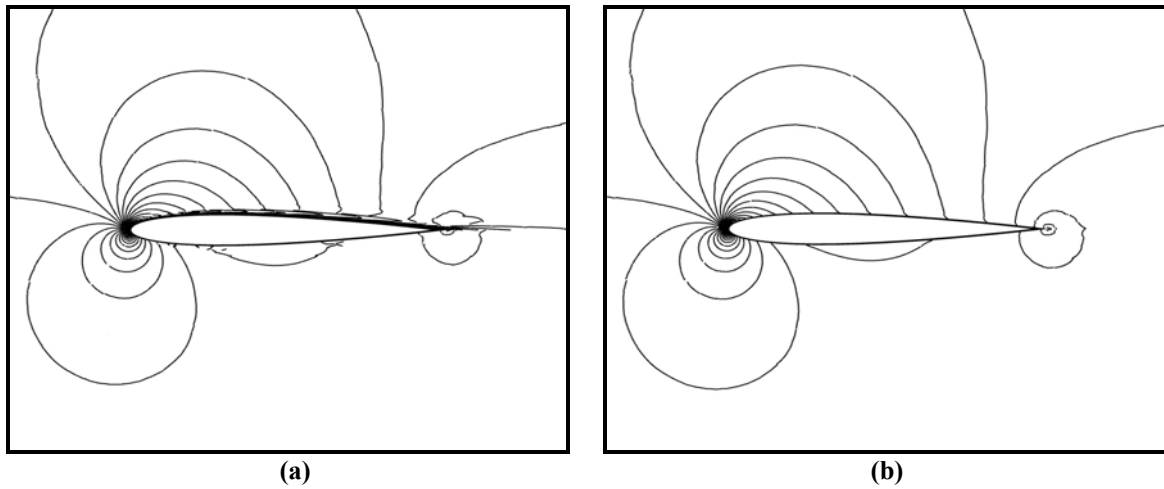

| (a) | (b) |

Figure 17. Mach contours for 3rd-order PQ approach: (a) linear boundary; (b) quadratic boundary.

VI. Conclusions

An efficient implementation of the spectral volume method has been successfully carried out for 2D scalar and the Euler equations. Two different approaches to compute the flux integrals more efficiently, the partial quadrature (PQ) and quadrature free (QF) approaches, have been developed and evaluated for both 3rd and 4th-order SV schemes. For a 3rd order SV scheme, the PQ and QF approaches require 22% and 72% fewer flux calculations than the traditional approach, respectively, and for a 4th order SV scheme, the PQ and QF approaches require 28% and 72% fewer flux calculations than the traditional approach, respectively. The savings is obviously significant in 2D, but it does not end here. In 3D, the thousands of flux calculations per SV required in the traditional approach will literally be reduced to only dozens. For all approaches, it has been found that the nearly optimum order of accuracy can be obtained in both the L_1 and L_∞ norms with respect to density errors. A simplified curved boundary treatment, which significantly reduces complexity of the numerical implementation, has been developed and employed with success. Results for the case of subsonic flow around a NACA 0012 airfoil using both a linear and quadratic boundary representation have been compared, and spurious entropy production at the airfoil surface is much less profound in the quadratic boundary case. The linear boundary representation was also used with 3rd and 4th-order SV schemes for the case of subsonic flow over a cylinder, but we were only able to obtain convergent numerical solutions using a quadratic boundary for this case. The extension of the QF approach to the 3D scalar and Euler equations is now under way, and will be reported in a future publication.

References

1. Z. J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation, J. Comput. Phys. 178 (2002) 210.
2. Z. J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation, J. Comput. Phys. 179 (2002) 665.
3. Z. J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids III: extension to one-dimensional systems, J. Sci. Comput. 20 (2004) 137.
4. Z. J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional Euler equations, J. Comput. Phys. 194 (2004) 716.
5. Y. Liu, M. Vinokur, Z. J. Wang, Spectral (finite) volume method for conservation laws on unstructured grids V: extension to three-dimensional systems, J. Comput. Phys. 212 (2006) 454-472.
6. Y. Sun, Z. J. Wang, Y. Liu, Spectral (finite) volume method for conservation laws on unstructured grids VI: extension to viscous flow, J. Comput. Phys. 215 (2006) 41-58.
7. S. K. Godunov, A finite-difference method for the numerical computation of discontinuous solutions of the equations of fluid dynamics, Mat. Sb. 47 (1959) 271.
8. B. van Leer, Towards the ultimate conservative difference scheme V. a second-order sequel to Godunov's method, J. Comput. Phys. 32 (1979) 101-136.
9. T. J. Barth, P.O. Frederickson, High-order solution of the Euler equations on unstructured grids using quadratic reconstruction, AIAA Paper No. 90-0013, 1990.
10. M. Delanaye, Yen Liu, Quadratic reconstruction finite volume schemes on 3D arbitrary unstructured polyhedral grids, AIAA Paper No. 99-3259-CP, 1999.
11. A. Harten, B. Engquist, S. Osher, S. Chakravarthy, Uniformly high order essentially non-oscillatory schemes III, J. Comput. Phys. 71 (1987) 231.
12. R. Abgrall, On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation, J. Comput. Phys. 114 (1994) 45-58.
13. C. Hu, C.-W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, J. Comput. Phys. 150 (1999) 97-127.
14. B. Cockburn, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, Math. Comput. 52 (1989) 411-435.
15. B. Cockburn, S.-Y. Lin, C.-W. Shu, TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, J. Comput. Phys. 84 (1989) 90-113.
16. B. Cockburn, S. Hou, C.-W. Shu, TVB Runge-Kutta locak projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, Math. Comput. 54 (1990) 545-581.

17. B. Cockburn, C.-W. Shu, The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, J. Comput. Phys. 141 (1998) 199-224.
18. Z. J. Wang, Y. Liu, Extension of the spectral volume method to high-order boundary representation, J. Comput. Phys. 211 (2006) 154-178.
19. L. Krivodonova, M. Berger, High-order accurate implementation of solid wall boundary conditions in curved geometries, J. Comput. Phys. 211 (2006) 492-512.
20. H. L. Atkins, Chi-Wang Shu, Quadrature-free implementation of the discontinuous Galerkin method for hyperbolic equations, AIAA J. 96 (1996) 1683.
21. J. S. Hesthaven, From electrostatics to almost optimal nodal sets for polynomial interpolation in a simplex, SIAM J. Numer. Anal. Vol. 35 No. 2 (1998) 655-676.
22. J. S. Hesthaven, C. H. Teng, Stable spectral methods on tetrahedral elements, SIAM J. Sci. Comput. Vol. 21 No. 6 (2000) 2352-2380.
23. Chen, Q. Y., Partitions for spectral (finite) volume reconstruction in the tetrahedron, SIAM J. Sci. Comput. (2005).
24. C.-W. Shu, Total-Variation-Diminishing time discretizations, SIAM J. Sci. Stat. Comput. 9 (1988) 1073-1084.
25. S. Wolfram, *Mathematica* Book (Wolfram Media and Cambridge Univ. Press, New York, 1999), 4th Ed.
26. V. V. Rusanov, Calculation of interaction of non-steady shock waves with obstacles, J. Comput. Math. Phys. USSR 1 (1961) 267-279.
27. P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (1981) 357-372.
28. A. Harten, P. D. Lax, B. Van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25 (1983) 35-61.
29. E. F. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the HLL Riemann solver, Shock Waves 4 (1994) 25-34.
30. S. Tu, S. Aliabadi, A slope limiting procedure in discontinuous Galerkin finite element method for gasdynamics applications, Int. J. Numer. Anal. & Mod. Vol. 2 No. 2 (2005) 163-178.
31. P. Batten, N. Clarke, C. Lambert, D. M. Causon, On the choice of wavespeeds for the HLLC Riemann solver, SIAM J. Sci. Comput. Vol. 18 No. 6 (1997) 1553-1570.
32. H. Luo, J. Baum, R. Löhner, On the computation of steady-state compressible flows using a discontinuous Galerkin method, presented at the Fourth International Conference on Computational Fluid Dynamics, July 10-14, 2006, Ghent, Belgium.