18th AIAA Computational Fluid Dynamics Conference
25 - 28 June 2007, Miami, FL

AIAA 2007-4325

Efficient Quadrature-Free 3D High-Order Spectral Volume Method on Unstructured Grids

Michael Yang[1], Rob Harris[2] and Z.J. Wang[3]
Department of Aerospace Engineering, Iowa State University, Ames, IA 50011

and

Yen Liu[4]
NASA Ames Research Center, Moffett Field CA 94035

The high-order spectral volume (SV) method has been extended for solving 3D hyperbolic conservation laws, and its implementation using an efficient quadrature-free approach has been performed to achieve high efficiency while maintaining accuracy. In the SV method, in order to perform a high-order polynomial reconstruction, each simplex cell – called a spectral volume (SV) – is partitioned into a "structured" set of sub-cells called control volumes (CVs) in a geometrically similar manner, thus a universal reconstruction formula can be obtained for all SVs from the cell-averaged solutions on the CVs. The SV method avoids the volume integral required in the DG method, but it does introduce more cell faces where surface integrals are needed. In this paper, the reconstructions for the fluxes are built based on the nodal values on a selected set of optimized and geometrically similar nodes within each SV. The most important advantage of this new approach is to use a set of universal shapefunctions for face integrals, which avoids the use of quadrature formulas without losing the properties of compactness and robustness that are inherent to the SV method. In high-order computations for many practical 3D problems, this new approach greatly reduces the number of computer operations and the required storage as compared to the implementation that uses quadrature formulas. In this paper, accuracy studies are performed on the 3D advection equations, and the 3D Euler equation for vortex evolution problems and flows around a sphere. The designed orders of accuracy have been achieved for all the corresponding orders of polynomial reconstruction.

I. Introduction

THE need for high-order methods for conservation laws on unstructured grids has been widely recognized in many engineering areas, for example, vortex-dominated flows, rotorcraft flow/blade-vortex interaction problems, aero-acoustic noise predictions, LES/DNS for complex configurations, computational electromagnetics (CEM), etc. In order to make CFD a useful tool for the real-world problems, the criteria for a high-order algorithm to meet should at least be 1) accurate; 2) conservative; 3) geometrically flexible; 4) computationally efficient; 5) easy to implement. The current leader of high-order methods on unstructured grids is the discontinuous Galerkin (DG) method[1-5]. The spectral volume (SV) method[6-10] and the spectral difference (SD) method[11-13] are two recently developed high-order methods which share similarities with the DG method while having differences on how degrees-of-freedom are updated. In this paper, we focus on extending the quadrature-free SV method[14] to three dimensions.

The SV method is a conservative Godunov-type finite volume (FV) method capable of capturing discontinuities. The SV method combines two key ideas which are the basis of the finite volume and finite element methods. These are the Riemann solver, which accounts for the physics of wave propagation, and the high-order polynomial reconstruction. However, instead of using a (large) stencil of neighboring cells to perform the reconstruction as in the *k*-exact finite volume methods, here in the Spectral Volume methods a simplex unstructured

[1]Graduate Research Assistant, 0233 Howe Hall, mhyang@iastate.edu, AIAA member.
[2]Graduate Research Assistant, 0237 Howe Hall, rharris@iastate.edu, AIAA member.
[3]Associate Professor of Aerospace Engineering, 2271 Howe Hall, zjw@iastate.edu, Associate Fellow of AIAA.
[4]Research Scientist, Yen.Liu@nasa.gov, Mail Stop T27B-1.

grid cell – called a spectral volume – is partitioned into a "structured" set of sub-cells called control volumes (CVs), and cell-averaged solutions on these sub-cells are then the degrees-of-freedom (DOFs). These DOFs are used to reconstruct a high-order polynomial inside the SV. If all the spectral volumes are partitioned in a geometrically similar manner, a universal reconstruction formula can be obtained for all simplexes. With reconstructed solutions at both sides of an interface, the numerical flux can be computed using an approximate Riemann solver. Then the DOFs can be updated to high-order accuracy using the usual Godunov-type finite volume procedure. Numerical tests in both 1D and 2D have verified that the SV method is accurate, conservative, efficient, geometrically flexible, and easy to implement. Recently the SV method has been extended to 3D conservation laws[10,15].

Here we want to point out that in a DG method the DOFs are updated by using a weighted residual approach including both face and volume integrals, while in a SV method the DOFs are updated by using a finite volume approach including only face integrals. However, the SV method does introduce more interfaces where more Riemann problems are solved. For two-dimensional Euler equations, both methods seem to achieve similar efficiency[16]. For 3D flow problems, the partition of a SV cell (tetrahedron) can be complicated and the number of the aforementioned sub-faces can be large (for example, roughly 130 for a cubic partition). Therefore quite a number of Gauss quadrature points might be needed (for example, roughly 800 for a cubic partition) per SV for the traditional implementation of a SV method to compute the face integrals to the desired precision, making the 3D SV method expensive.

A quadrature-free method was proposed by Atkins and Shu[17] for DG method, and was tested in 1D scalar advection and 2D scalar advection and linear Euler equations. In this paper we extend a quadrature-free 2D SV method[14] to three dimensions, which handles the face integrals more efficiently. In this approach, the SV method is implemented using a set of geometrical-similarly nodes within each SV. The set of nodes was chosen to be near optimal for each SV to reconstruct a degree *k+1* polynomial approximation for the flux vector. To avoid the quadrature formulas, a universal reconstruction is utilized for the face integrals of numerical flux on all faces to save computer resources. The flux integrals are computed analytically after the shape functions defined by this nodal set are integrated analytically for a standard element during preprocessing. This reconstruction of the integral shape functions is universal for all SVs if a nodal set is distributed in a geometrically similar manner for all SVs.

The paper is organized as follows. In Section 2, we review the traditional formulation of 3D SV method. In section 3, the quadrature-free approach is described in detail. Numerical results including accuracy studies for 3D advection equations, as well as Euler equations for 3D vortex evolution and sphere are presented in Section 4. Finally, conclusions and some possibilities for future work are given in Section 5.

II. Review of the General 3D Spectral Volume Method

Consider the 3D conservation law in the following form,

$$\frac{\partial Q}{\partial t} + \nabla \bullet F = 0 \tag{1}$$

on domain $\Omega \times [0,T]$ and $\Omega \subset R^3$ with the initial conditions within Ω and appropriate boundary conditions on $\partial\Omega$. The conservative solution variable *Q* can be a scalar or a vector, and the generalized flux *F* can be a scalar, vector, or even tensor. Domain Ω is discretized into *I* nonoverlapping tetrahedral cells (SVs) which are further partitioned into CVs in a geometrically similar manner, as shown in Figure 1. For a complete 3D polynomial basis, a reconstruction of degree of precision *p* requires at least *N* CVs ($C_{i,j}, i = 1,...,I;\ j = 1,...,N(p)$), where

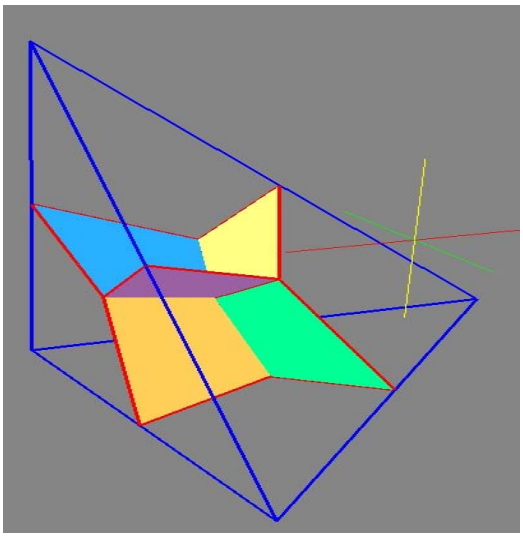
$$N(p) = (p+1)(p+2)(p+3)/6 \tag{2}$$

From the point of view of best interpolation polynomial, the optimal partition should make the Lebesgue constant minimum[7]. Therefore we use the following partitions which have been optimized for a minimal Lebesque constant. Figure 1 shows the linear, quadratic, and cubic partitions of a tetrahedral SV given by Chen[18], where each CV is enclosed by planar polygonal faces for ease of computation. The Lebesque constants are 5.08 and 6.87 for quadratic and cubic partitions, respectively.

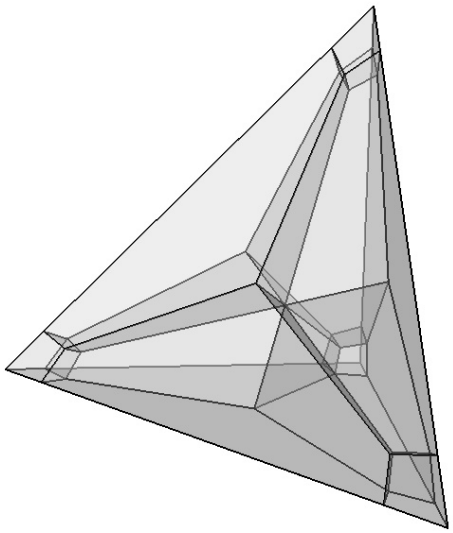
Integrating (1) over each CV, we obtain

$$\frac{d\bar{Q}_{i,j}}{dt} + \frac{1}{V_{i,j}} \sum_{r=1}^{K} \int_{A_r} (F \bullet \vec{n}_{A_r}) dA = 0 \tag{3}$$

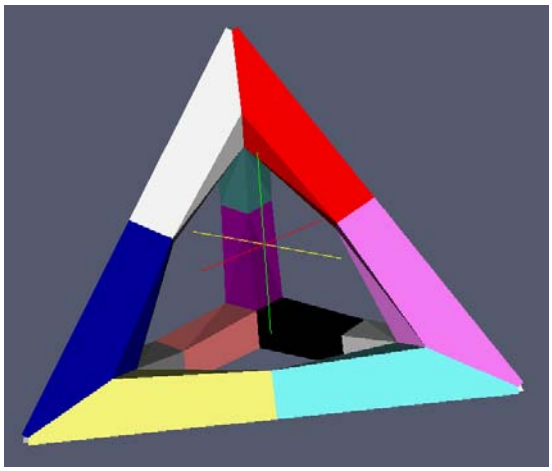
where $\bar{Q}_{i,j}$ is the cell-averaged solution on $C_{i,j}$, and A_r represents the faces (with normal $\vec{n}_{A_r}$) that enclose $C_{i,j}$.



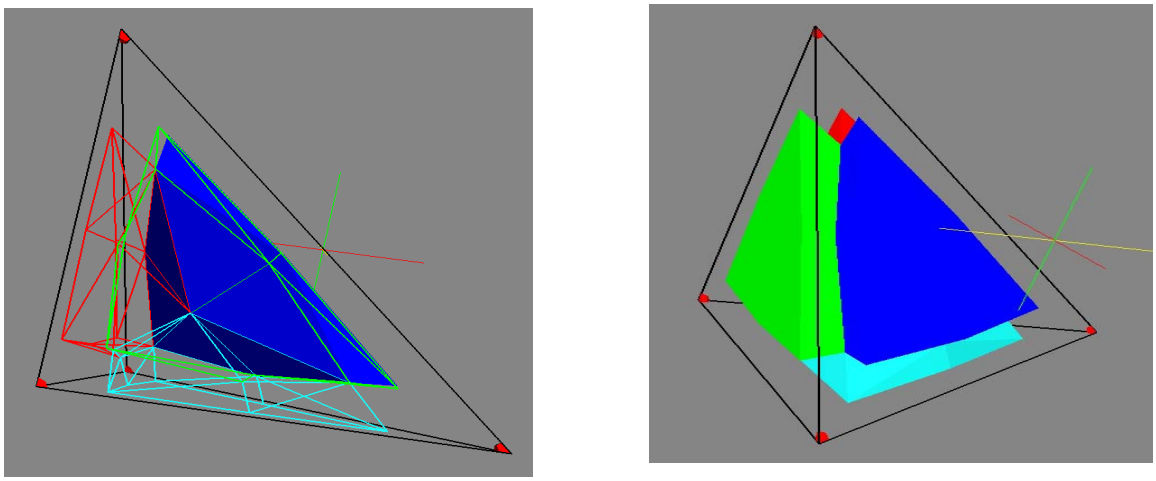
(a) Linear partition



(b) Quadratic partition



(c) Cubic partition: 10-sided sub-cells



(d) Cubic partition: 6-sided and 19-sided sub-cells

Figure 1. Partition of a tetrahedron

A high-order polynomial is then reconstructed within each SV such that

$$\frac{1}{V_{ij}}\int_{C_{i,j}} P_i^p(x,y,z)dxdydz = \bar{Q}_{i,j} \tag{4}$$

where P_i^p is a polynomial (or vector polynomial) of degree *p* for the *i*th SV. Let D_p denote the space of degree *p* polynomials in three dimensions. To actually solve the reconstruction problem, we introduce the complete polynomial basis, $e_l(x,y,z) \in D_p$, where $D_p = span\{e_l(x,y,z)\}_{l=1}^{N(p)}$. Therefore P_i^p can be expressed as

$$P_i^p = \sum_{l=1}^{N(p)} a_i^l e_l \tag{5}$$

or in the matrix form

$$P_i^p = ea \tag{6}$$

where *e* is the basis function vector [$e_1,...,e_N$] and *a* is the reconstruction coefficient vector [$[a_1,...,a_N]^T$. Substituting (5) into (4), we then obtain

$$\frac{1}{V_{ij}}\sum_{l=1}^{N(p)} a_i^l \int_{C_{i,j}} e_l(x,y,z)dxdydz = \bar{Q}_{i,j}, \qquad j = 1,...,N(p). \tag{7}$$

Let $\bar{Q}$ denote the column vector $[Q_{i,1},...,Q_{i,N}]^T$. Equation (7) can be rewritten in the matrix form

$$Ra = \bar{Q} \tag{8}$$

where the reconstruction matrix

$$R = \begin{bmatrix} \frac{1}{V_{i1}}\int_{C_{i,1}} e_1(x,y,z)dV & \ldots & \frac{1}{V_{i1}}\int_{C_{i,1}} e_N(x,y,z)dV \\ \ldots\ldots & & \ldots. \\ \frac{1}{V_{iN}}\int_{C_{i,N}} e_1(x,y,z)dV & \ldots & \frac{1}{V_{iN}}\int_{C_{i,N}} e_N(x,y,z)dV \end{bmatrix} \tag{9}$$

The reconstruction coefficients a can be solved as

$$a = R^{-1}\bar{Q} \tag{10}$$

provided that the reconstruction matrix *R* is nonsingular. Substituting (10) into (5) or (6), P_i^p is then expressed in terms of *cardinal* basis functions or shape functions $L = [L_1,...L_N]$:

$$P_i^p = \sum_{j=1}^{N} L_j(x,y,z)\bar{Q}_{i,j} = L\bar{Q} \tag{11}$$

Here *L* is defined as

$$L \equiv eR^{-1} \tag{12}$$

which satisfies

$$\frac{1}{V_{ij}}\int_{C_{i,j}} L_l dV = \delta_{jl} \tag{13}$$

Equation (11) gives the functional representation of the state variable *Q* within the SV. Therefore the function value of *Q* at a quadrature point or any point (x_{rq}, y_{rq}, z_{rq}) within the *i*th-SV is thus simply

$$P_i^p(x_{rq}, y_{rq}, z_{rq}) = \sum_{j=1}^{N(p)} L_j(x_{rq}, y_{rq}, z_{rq})\bar{Q}_{i,j} \tag{14}$$

Note that once the polynomial basis functions e_l are chosen, the shapefunctions L_j are solely determined by the

partition of a SV cell. The shape and the partition of a SV cell, in general, can be arbitrary as long as the reconstruction matrix *R* is nonsingular. However, different shapes of SV cells can result in the same expression of the shapefunctions (in terms of a few geometric parameters) if a geometrically similar partition can be applied to them. Since the volume integral of polynomial basis in (9) can be carried out easily over a transformed standard tetrahedron, the shape functions *L*, which are universal for all SVs, can be calculated analytically and stored as a preprocessing step.

The flux integration over a face is performed using the Gauss quadrature formula

$$\int_{A_r} (F \cdot \vec{n}) dA = \sum_{q=1}^{J} w_{rq} F(P(x_{rq}, y_{rq}, z_{rq})) \cdot \vec{n}_r A_r + O(h^k) \tag{15}$$

where J is the number of quadrature points on the rth face, and w_{rq} are the Gauss quadrature weights, and (x_{rq}, y_{rq}, z_{rq}) are the Gauss quadrature points. Since a discontinuous solution can exist between SVs, an approximate Riemann solver is used to find fluxes for faces on SV boundaries. The 3rd-order TVD Runge-Kutta scheme is employed for time integration.

III. Formulation of Quadrature-Free Approach for 3D SV method

In all the previous implementations of SV method, the surface integrals are evaluated with quadrature formulas that are appropriate to the face shape and to the required degree of accuracy. In 1D or 2D cases, Gauss quadrature formulas are very efficient. But in the 3D case, the number of quadrature points can be very large for high-order SVs due to the large number of interior sub-faces within each SV, making the 3D SV method very expensive. To overcome this deficiency, we propose an efficient quadrature-free approach for the SV method.

In the new approach, a near optimal nodal set is selected from Ref.[19]. This nodal set is then used to reconstruct a degree *k+1* polynomial approximation for the flux vector, and then the flux integrals are computed analytically, without the need for Gauss quadrature formulas. The flux vector *F* is approximated in terms of the basis set $\{b_l\}$ (constructed from simple monomials),

$$F \approx \sum_{l=1}^{M} b_l e_l \tag{16}$$

If *F(Q)* is linear, then $M = N(p)$; however, when *F*(*Q*) is nonlinear, *M* must be at least $N(p+1)$ to obtain the design accuracy of *p+1*. We prefer to use $M = N(p+1)$ for all the cases, thus $e_l(x, y, z) \in D_{p+1}$, where $D_{p+1} = span\{e_l(x, y, z)\}_{l=1}^{N(p+1)}$ The reconstruction problem reads as follows: Given the nodal values $F_{i,n}$ on a set of nodes within the SV cell *i*, find $P_i^{p+1} \in D_{p+1}$ such that

$$P_i^{p+1}(x_n, y_n, z_n) = F_{i,n} \tag{17}$$

where

$$P_i^{p+1}(x, y, z) = \sum_{l=1}^{M} b_i^l e_l(x, y, z) \tag{18}$$

Therefore a Lagrange shape functions defined by the nodal set can be found from (17) and (18). Substituting (17) into (18) yields

$$F_{i,n} = \sum_{l=1}^{M} b_i^l e_l(x_n, y_n, z_n), \qquad n = 1, 2, ..., M \tag{19}$$

Let $\overline{F}$ denote the column vector $[F_{i,1}, ..., F_{i,M}]^T$. Equation (19) can be rewritten in the matrix form

$$Sb = \overline{F} \tag{20}$$

where the reconstruction matrix

$$S=\begin{bmatrix} e_1(x_1,y_1,z_1) & e_2(x_1,y_1,z_1) & \dots & e_M(x_1,y_1,z_1) \\ & \cdots\cdots & & \\ e_1(x_M,y_M,z_M) & e_2(x_M,y_M,z_M) & \dots & e_M(x_M,y_M,z_M) \end{bmatrix} \tag{21}$$

The reconstruction coefficients b can be solved as

$$b = S^{-1}\overline{F} \tag{22}$$

provided that the reconstruction matrix *S* is nonsingular. Substituting (22) into (18), P_i^{p+1} is then expressed in terms of shape functions $Z=[Z_1,...Z_M]$:

$$P_i^{p+1} = \sum_{n=1}^{M} Z_n(x,y,z)F_{i,n} = Z\overline{F} \tag{23}$$

Here Z is defined as

$$Z \equiv eS^{-1} \tag{24}$$

which satisfies

$$Z_n(x_m,y_m,z_m) = \delta_{mn} \tag{25}$$

Again, this reconstruction is universal for all SVs if a nodal set is distributed in a geometrically similar manner for all SVs. The flux vector *F* can be computed at any point using

$$F_i(x,y,z) = \sum_{n=1}^{M} Z_n(x,y,z)F_{i,n} \tag{26}$$

For the flux on each internal face, the flux integral can be computed as a weighted average of the flux evaluated at the nodal set, i.e.,

$$\int_f F\bullet\vec{n}dS = \sum_{n=1}^{M}(F_{i,n}\bullet\overrightarrow{n_f})\int_f Z_n(x,y,z)dS = \sum_{n=1}^{M}(F_{i,n}\bullet\overrightarrow{n_f})A_f\overline{Z}_{n,f} \tag{27}$$

where $F_{i,n}$ is the flux vector evaluated at node *n* on the *i*-th SV cell; $\overline{Z}_{n,f}$ is the face-averaged value of shape function for face *f*, which is universal for all SVs if a nodal set is distributed in a geometrically similar manner, and thus can be computed during preprocessing for a standard element and then the physical face area A_f is multiplied. Compared with the evaluation of flux integration in (15) by using Gauss quadrature formula, obviously the quadrature-free flux integration given by (27) is much less costly to evaluate for 3D cases.

For the SV-bounding faces, the Riemann flux integral can also be computed without the use of a Gauss quadrature. For example, Rusanov flux gives

$$\hat{F}\bullet\vec{n} = \frac{1}{2}[F^R\bullet\vec{n} + F^L\bullet\vec{n} - \lambda(Q^R - Q^L)] \tag{28}$$

where λ is the local maximum eigenvalue based the right and left cells. Integrating (28) on face f yields

$$\int_f \hat{F}\bullet\vec{n}dS = \frac{1}{2}[\int_f F^R\bullet\vec{n}dS + \int_f F^L\bullet\vec{n}dS - \int_f \lambda(Q^R - Q^L)dS] \tag{29}$$

We notice that

$$\lambda \sim O(1),\ (Q^R - Q^L) \sim O(h^{p+1}),$$

and $$F^R \sim P_R^{p+1} + O(h^{p+1}),\ F^L \sim P_L^{p+1} + O(h^{p+1})$$

Hence $\lambda(Q^R - Q^L) \sim O(h^{p+1})$ is high-order small term compared with F^R and F^L, which allows us to use a face-centered value based on a average state, $\lambda_{c,f}$, to replace the local λ in (29). Then (29) becomes

$$\int_f \hat{F}\bullet\vec{n}dS = \frac{A_f}{2}[\overline{F}^R + \overline{F}^L - \lambda_{c,f}(\overline{Q}^R - \overline{Q}^L)] \tag{30}$$

where

$$\bar{F}^R = \frac{1}{A_f}\int_f F^R \bullet \vec{n} dS, \quad \bar{F}^L = \frac{1}{A_f}\int_f F^L \bullet \vec{n} dS$$
$$\bar{Q}^R = \frac{1}{A_f}\int_f Q^R dS, \qquad \bar{Q}^L = \frac{1}{A_f}\int_f Q^L dS \tag{31}$$

Our numerical tests in the next section have verified the above accuracy analysis, and have shown that this quadrature-free approach preserves the accuracy of the SV method.

IV. Numerical Results

In order to demonstrate the new method, we use the 3D advection equations and 3D Euler equations to analyze the accuracy the present method can achieve by comparing with the exact solutions. For all cases the Rusanov numerical flux is applied. The third-order TVD Runge-Kutta scheme is used for time integration in all cases, and the time step Δt used is small enough so that the numerical errors are dominated by the spatial discretization, independent of the time step. Given the initial values at the nodal set, the CV-averaged solution values were initialized by using the CV-averaged node-based shape functions, without the need of Gauss quadrature for preparing the CV-averaged initial conditions. The partition of the SV cells has been taken from Ref.[10] or [18].

1. Accuracy Study with 3D Steady Linear Advection

The governing equation for this problem is

$$\frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} + \frac{\partial u}{\partial z} = 0 \tag{32}$$

The boundary conditions are

$$u = \sin[\pi(2x - y - z)] \text{ for inflow; extrapolation of } u \text{ for outflow.} \tag{33}$$

The above equations describe a steady sinusoidal wave with unit wave speed in all three Cartesian directions. It is obvious that the exact solution for this problem is $u = \sin[\pi(2x - y - z)]$. We solved it numerically by using pseudo-time integration as follows to find its steady-state solution,

$$\frac{\partial u}{\partial \tau} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} + \frac{\partial u}{\partial z} = 0 \tag{34}$$

We generated a sequence of regular unstructured grids (Figure 2) in a cubic domain. The domain size used here is $[0,1]\times[0,1]\times[0,1]$. First the cubic domain is represented by $N\times N\times N$ cubic cells, and then each cell is cut into 6 tetrahedra. Taking $N = 10, 20, 40$, we obtained a sequence of unstructured grids for the accuracy analysis under grid refinement. In the following figures, cell size $I = N\sqrt[3]{6}$. The solution is taken as converged when the L_2 norm of the residual is reduced to machine zero.

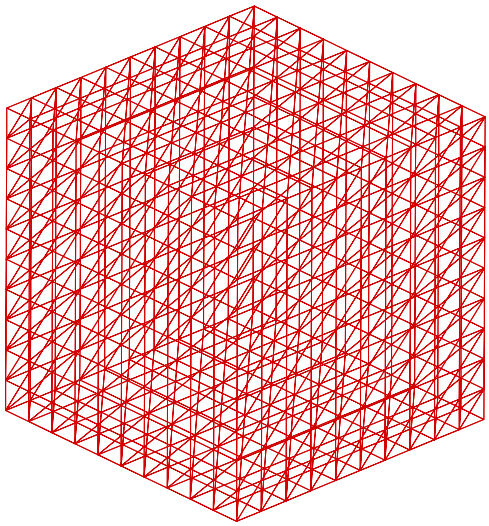
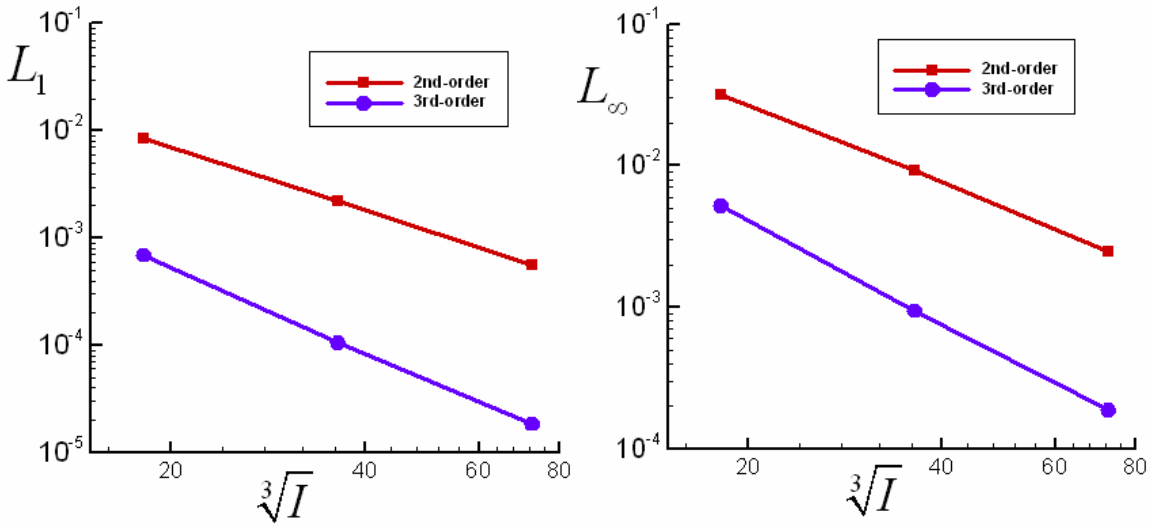


Figure 2. Regular grid for 3D domain (10x10x10x6 tetrahedra)

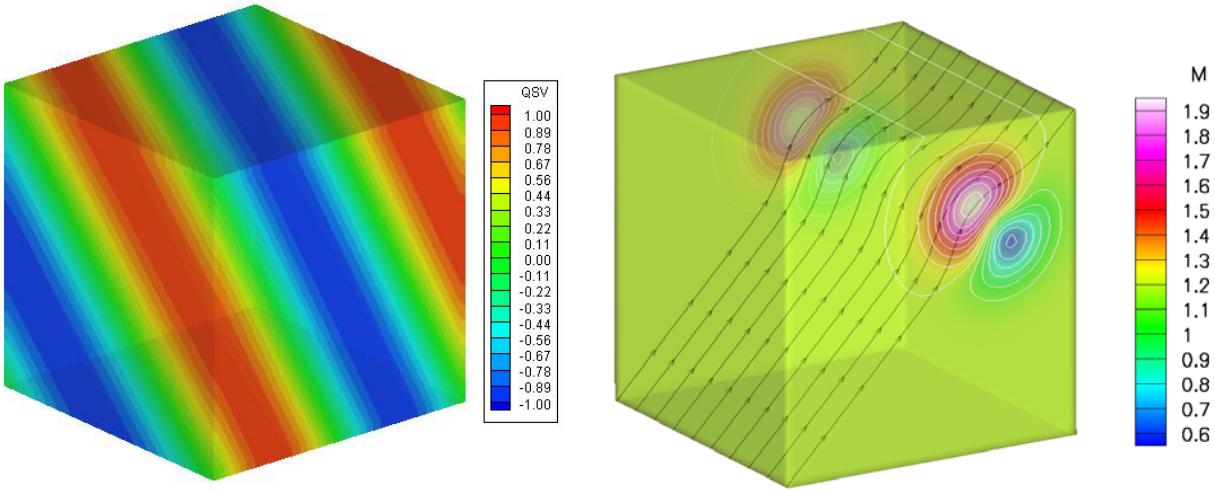
Figure 3 shows the error norms (L_1 and L_∞) of the present numerical solution comparing with the exact solution. The averaged slope is about 1.98 for the 2nd-order cases, and 2.73 for the 3rd-order cases. The results show that the nearly optimum order of accuracy is attained for the 2nd-order and 3rd-order cases, respectively. Based on that we are satisfied with our results of order of accuracy for the 2nd –order and 3rd-order cases.

Figure 4(a) gives the contour of the solution on the fine mesh for the 3rd-order method, which shows the advection solution in a clear wave pattern. The number of DOFs used is 3.84×10^6.



(a) L_1 norm of error (b) L_∞ norm of error

Figure 3. Linear advection: Error vs. cell size.



(a) Solution of Linear advection (b) Mach number for vortex evolution

Figure 4. Contours of solution

2. Accuracy Study with 3D Euler Equation

The 3D Euler equation in conservative form can be written as

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}+\frac{\partial H}{\partial z}=0 \tag{35}$$

where Q is the conservative solution variables, F, G, H are the inviscid flux given below,

$$Q=\begin{Bmatrix}\rho\\ \rho u\\ \rho v\\ \rho w\\ E\end{Bmatrix},\quad F=\begin{Bmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ \rho uw\\ u(E+p)\end{Bmatrix},\quad G=\begin{Bmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ \rho vw\\ v(E+p)\end{Bmatrix},\quad H=\begin{Bmatrix}\rho w\\ \rho uw\\ \rho vw\\ \rho w^2+p\\ w(E+p)\end{Bmatrix} \tag{36}$$

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

$$E=\frac{p}{\gamma-1}+\frac{1}{2}\rho(u^2+v^2+w^2) \tag{37}$$

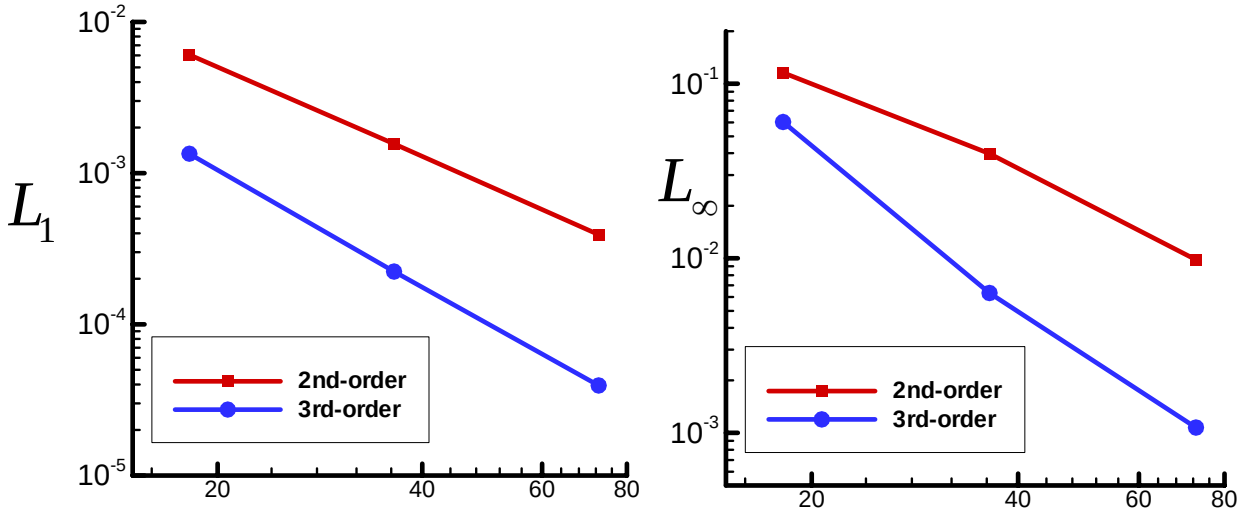
with ratio of specific heat $\gamma=1.4$.

The vortex evolution problem is ideal to test the accuracy of the present method for Euler equations because it has an analytical solution. This testing problem was used by Shu[20] in 2D cases, and Sun[21] in 3D cases. Here the mean flow is $\{\rho,u,v,w,p\}=\{1,1,1,0,1\}$. An isotropic vortex is then added to the mean flow, i.e., with perturbations in u , v , and temperature T , but no perturbation in velocity *w* and entropy $S=p/\rho^{\gamma}$.

$$\begin{aligned}(\delta u,\delta v,\delta w)&=\frac{\varepsilon}{2\pi}e^{0.5(1-r^2)}(-y,x,0)\\ \delta T&=-\frac{(\gamma-1)\varepsilon^2}{8\gamma\pi^2}e^{1-r^2}\\ \delta S&=0\end{aligned} \tag{38}$$

where $r^2=x^2+y^2$, and the vortex strength $\varepsilon=5$. The computational domain is taken to be a cube with size of $[-5,5]\times[-5,5]\times[-5,5]$. A sequence of grids were generated similarly as the previous problem for *N*=10,20,40.

The characteristic inflow and outflow boundary conditions are imposed on the boundaries in *x* and *y* directions, and extrapolation in the *z* direction. It can be easily verified that the exact solution given by the Euler equation with the above specified initial condition and boundary conditions represents the vortex moving from its initial location in the diagonal direction on the x-y plane with the speed (1,1,0).



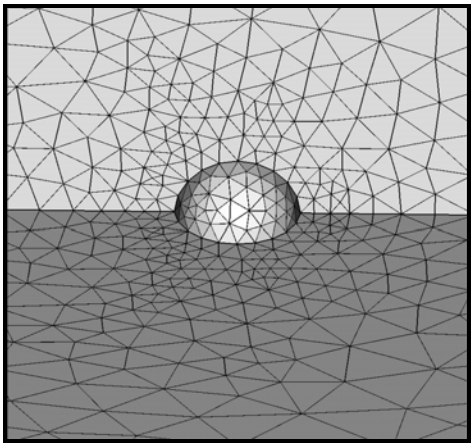
(a) L_1 norm of error. (b) L_∞ norm of error

Figure 5. Vortex evolution problem: Error vs. Cell size

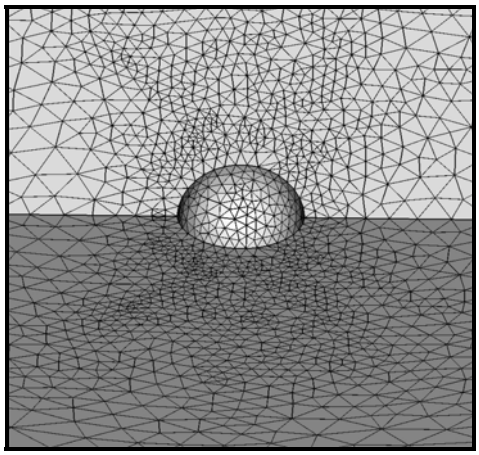
Figure 5 shows the error norms (L_1 and L_∞) of the present numerical solution comparing with the exact solution. The averaged slope is about 1.88 for the 2nd-order cases, and 2.73 for the 3rd-order cases. Again, we see here that the nearly optimum order of accuracy is attained for both 2nd-order and 3rd-order cases, respectively. The Mach number contour on the fine mesh for the 3rd-order method is given in Figure 4(b), which shows the vortex location at t=2. The number of DOFs used is 19.2×10^6.

3. Flow around a Sphere

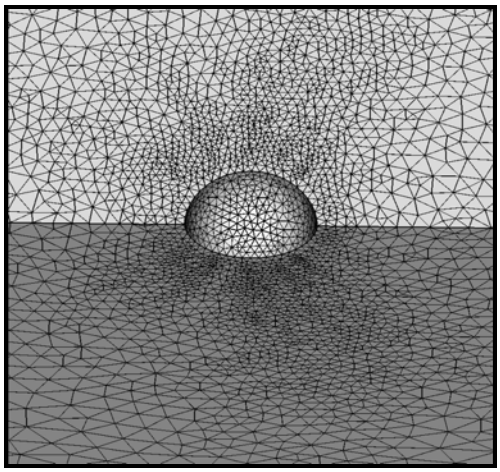
The inviscid flow around a sphere with Mach=0.3 is selected to test the present method for solving Euler equation with curved boundaries. A sequence of irregular unstructured grids were generated as shown in Figure 6. In order to handle the curved boundary, we follow the approach of Krivodonova and Berger[22] originally developed for the DG method. The basic idea employed here is that straight–sided SVs, rather than traditional curved SVs, are employed at wall boundaries, but a zero-flow boundary condition is enforced for the physical boundary as opposed to the computational (polygonal or polyhedral) boundary. This allows us to utilize the same reconstruction for the SVs on curved boundaries that is already in use for all other SVs. The Runge-Kutta scheme with local time stepping at CFL=0.5 is used for time integration. The solution is taken as converged when the L_2 norm of density residual drops 8 orders of magnitude. Figure 7 shows the Mach contours for 2nd-order and 3rd-order solutions. Note that the solution becomes more smooth and symmetric with the increase in the order of accuracy and grid resolution.



Coarse grid: 4,856 tetrahedra

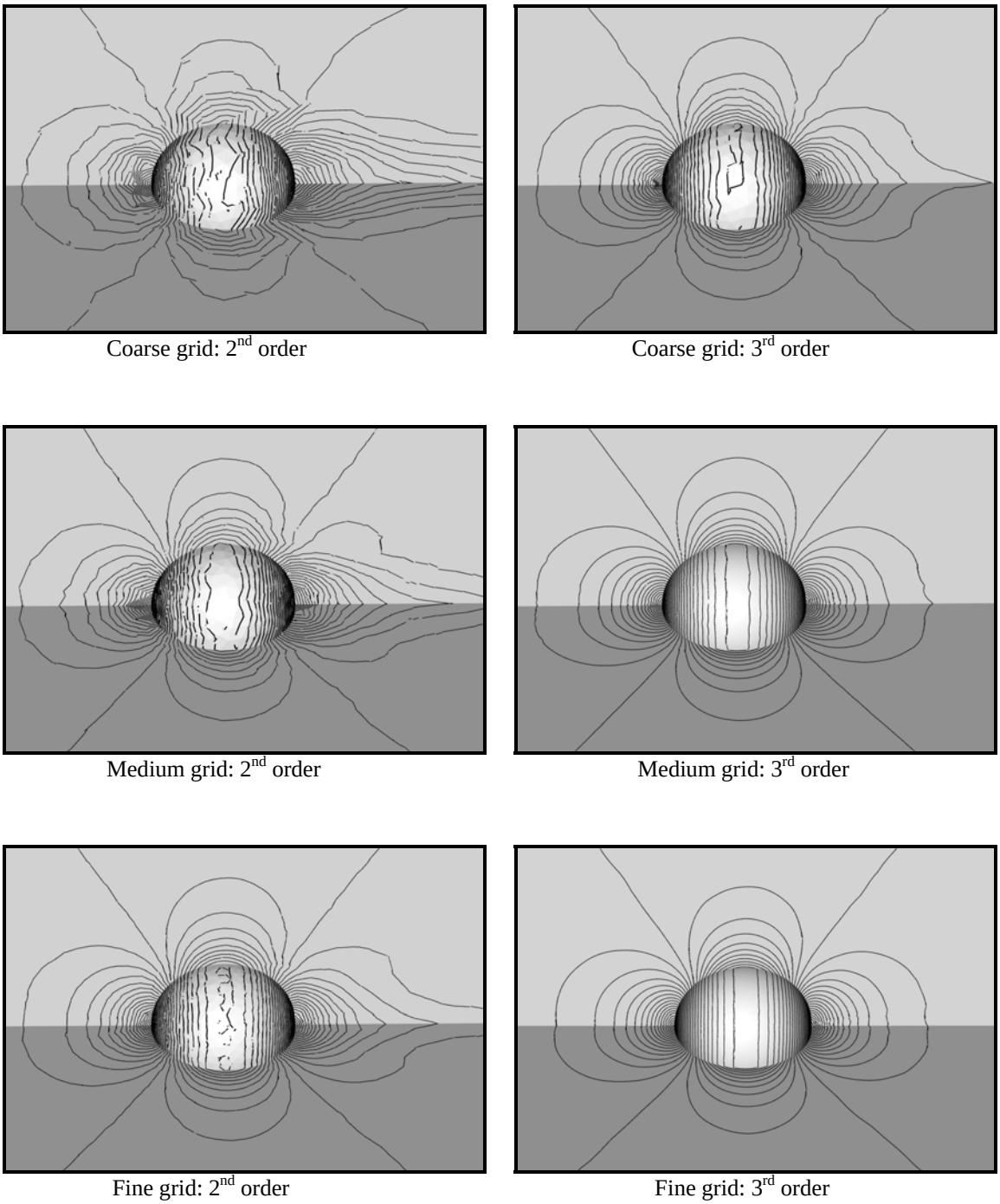


Medium grid: 25,027 tetrahedra



Fine grid: 53,520 tetrahedra

Figure 6. Unstructured meshes for solving the flow field around the sphere



Coarse grid: 2nd order | Coarse grid: 3rd order

Medium grid: 2nd order | Medium grid: 3rd order

Fine grid: 2nd order | Fine grid: 3rd order

Figure 7. Mach contours for flows around the sphere, Mach=0.3

V. Conclusions

The quadrature-free spectral volume method has been extended to 3D conservation laws including linear scalar advection equation and nonlinear Euler equations. From the nodal values on a selected set of optimized and geometrically similar nodes within each SV, we found a set of universal shape functions for face integrals, which avoids the use of quadrature formulas without losing the properties of compactness and robustness that are inherent to the SV method. In high-order computations for 3D problems, it has been shown that this new approach greatly

reduces the number of flux calculations per SV that required in the traditional SV method. Several representative inviscid cases were used to test the new quadrature-free SV method. It has been found that the near optimum order of accuracy can be obtained in both L_1 and L_∞ norms for both 2nd and 3rd-order simulations in 3D. This shows that the new approach preserves the stability and accuracy. In addition, the test case of inviscid flow over a sphere demonstrates the ability of the new approach to effectively handle curved boundaries using a simple curved wall treatment. The extension of the quadrature-free SV method to 4th-order in 3D, development of a quadrature-free implicit solver, and optimization of other SV partitions are now under way, and will be reported in future publications.

Acknowledgements

We would like to thank Mr.Takanori Haga for some helpful discussions. The research is sponsored by DOE. The views and opinions expressed herein do not necessarily represent those of the DOE, or U.S. Government.

References

[1]Cockburn, B. and Shu, C.-W., TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework, *Mathematics of Computation* 52, 411-435 (1989).

[2]Cockburn, B., Lin, S.-Y., and Shu, C.-W., TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one-dimensional systems, *J. Comput. Phys.* 84, 90-113 (1989).

[3]Cockburn, B., Hou, S., and Shu, C.-W., TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case, *Mathematics of Computation* 54, 545-581 (1990).

[4]Bassi, F. and Rebay, S., High-order accurate discontinuous finite element solution of the 2D Euler equations, *J. Comput. Phys*. 138, 251-285 (1997).

[5]Bassi, F. and Rebay, S., A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations, *J. Comput. Phys.* 131, 267-279 (1997).

[6]Wang, Z.J., "Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation," *J. Comput. Phys.* 178, 210(2002).

[7]Wang, Z.J., and Liu, Y., "Spectral (finite) volume method for conservation laws on unstructured grids II: Extension to two-dimensional scalar equation," *J. Comput. Phys.* 179, 665(2002).

[8]Wang, Z.J., and Liu, Y., "Spectral (finite) volume method for conservation laws on unstructured grids III: one dimensional systems and partition optimization," *SIAM J. Sci. Comput.* 20, 1. 137(2004).

[9]Wang, Z.J., and Liu, Y., "Spectral (finite) volume method for conservation laws on unstructured grids IV: Extension to two-dimensional systems," *J. Comput. Phys.* 194, 2, 716(2004).

[10]Liu, Y., Vinokur, M., and Wang, Z.J., "Spectral (finite) volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems," *J. Comput. Phys.* 212, 454(2006).

[11]Liu, Y., Vinokur, M., and Wang, Z.J., "Spectral Difference Method for Unstructured Grids I: Basic Formulation," *J. Comput. Phys.* 216, (2006) 780-801.

[12]Wang, Z.J., Liu, Y., May, G., and Jameson, A., "Spectral Difference Method for Unstructured Grids II: Extension to the Euler Equations," *J. Sci Comp.* 32, 1, (2007).

[13]Sun,Y.Z., Wang, Z.J., Liu, Y., "High-Order Multidomain Spectral Difference Method for the Navier-Stokes Equations on Unstructured Hexahedral Grids," *Comm.Comp. Phy.* Vol. 2, No. 2, 2007, pp. 310-333.

[14]Harris, R., Wang, Z.J. and Liu, Y., Efficient Implementation of High-order Spectral Volume Method for Multidimensional Conservation Laws on Unstructured Grids, AIAA Paper No. 2007-912.

[15]Haga T, Ohnishi N, Sawada K and Masunaga A. Spectral Volume Computation of Flowfield in Aerospace Application Using Earth Simulator, AIAA Paper No. 2006-2823.

[16]Sun,Y.Z., Wang, Z.J., "Evaluation of Discontinuous Galerkin and Spectral Volume Methods for Scalar and System Conservation Laws on Unstructured Grid", *International Journal for Numerical Methods in Fluids* 45 No. 8, 819-838 (2004).

[17]Atkins, H.L. and Shu, C-W., "Quadrature-Free Implementation of Discontinuous Galerkin Method for Hyperbolic Equations", *AIAA J*, 36, 5, (1998)

[18]Chen, Q.Y., "Partitions for spectral (finite) volume reconstruction in the tetrahedron", *SIAM J. Sci. Comput.* (2005).

[19]Hesthaven, J.S. and Teng, C.H., Stable spectral methods on tetrahedral elements, *SIAM J. Sci*. Comput. 21, 2352-2380 (2000).

[20]Shu,C-W., "Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws," In Advanced Numerical Approximation of Nonlinear Hyperbolic Equations, edited by A. Quarteroni, Lecture Notes in Mathematics, Vol.1697, P.325, Springer-Verlag, Berlin/New York, 1998.

[21]Sun, Y.Z., Wang, Z.J. and Liu, Y., "High-Order Multidomain Spectral Difference Method for the Navier-Stokes Equations," AIAA-2006-0301.

[22]Krividonova, L. and Berger, M., High-order accurate implementation of solid wall boundary conditions in curved geometries, Vol.211, *J. Comput. Phys.* p.p.492-512, 2006.