

High-Order Multidomain Spectral Difference Method for the Navier-Stokes Equations

Yuzhi Sun* and Z.J. Wang†

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

Yen Liu‡

NASA Ames Research Center, Moffett Field, CA 94035

A high order multidomain spectral difference (SD) method is developed for the three dimensional Navier-Stokes equations on unstructured hexahedral grids. The method is easy to implement since it involves one-dimensional operations only, and does not involve surface or volume integrals. Universal reconstructions are obtained by distributing solution and flux points in a geometrically similar manner in a unit cube. The concepts of the Riemann solver and high-order local representations are applied to achieve conservation and high order accuracy. In this paper, accuracy studies are performed to numerically verify the order of accuracy using flow problems with analytical solutions. High order of accuracy and spectral convergence are obtained for the propagation of an isotropic vortex and the Couette flow. The capability of the method for more complex, inviscid and viscous flow problems with curved boundaries is also demonstrated.

I. Introduction

It is well known that the computation of the aerodynamic flow field around a helicopter is a considerable challenge¹ because of the following difficulties: the strong interaction between the moving blades, vortices and wakes; the disparate length scales in the flow turbulence under flight conditions; the complex and moving geometries. Although low order (first and second order) finite volume methods have become the main choice of many commercial Computational Fluid Dynamics (CFD) codes, and proven successful in tackling a wide variety of flow problems²⁻⁴ in engineering design, vortex-dominated flows require high-order methods. This is mainly because lower order methods usually dissipate propagating vortices too quickly. For example, to study the classical blade-vortex interaction problem, it is estimated that the blade vortex should be well preserved for several revolutions. However, low order methods may completely “consume” the vortex after one or two revolutions, therefore making the computation results very inaccurate.

Many high-order methods (order > 2) have been developed in CFD for a wide range of applications, such as large eddy simulation, computational aeroacoustics, etc. Most high-order methods were developed for structured grids, e.g., ENO/WENO methods⁵, compact methods⁶⁻⁷, optimized methods⁸. In particular, high-order compact methods have been successfully employed to tackle vortex dominated problems, including vortex breakdown over a delta wing⁹. For complex configurations, it may be too time-consuming to generate smooth structured grids required by these high-order methods. Our focus in this study is therefore on high-order methods for unstructured grids.

There have been intensive research efforts on high-order methods for flow simulation on unstructured grids in the last two decades. An incomplete list of notable examples includes the spectral element method¹⁰, multi-domain spectral method¹¹⁻¹², k-exact finite volume method¹³, WENO methods¹⁴, discontinuous Galerkin method¹⁵⁻¹⁷, high-order residual distribution methods¹⁸, spectral volume¹⁹⁻²² and spectral difference methods²³⁻²⁶. Among those methods, some are based on the weighted residual form of the governing equations, for instance the discontinuous Galerkin (DG)¹⁵⁻¹⁷ method. Some are based on the integral form of the governing equations, e.g., the k-exact finite volume method¹³ and the spectral volume (SV)¹⁹⁻²² methods. Others, such as staggered grid multi-domain spectral method¹¹⁻¹² and spectral difference (SD) method²³⁻²⁶, are based on the differential form.

* Postdoc Research Associate of Aerospace Engineering, 2271 Howe Hall, sunyuzhi@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.

When selecting a method to implement for three-dimensional problems, the cost or complexity of the method is often an important factor. It is obvious that methods based on the differential form are the easiest to implement since they do not involve surface or volume integrals. This is particularly true when high-order curved boundaries need to be dealt with. Based on our experiences with the DG, SV and SD methods on 2D triangular meshes, the SD method is the most efficient for the 2D Euler equations. Therefore the SD method is selected to solve the 3D Navier-Stokes equations on unstructured hexahedral grids. The use of hexahedral grids is again a compromise between flexibility and efficiency. Although tetrahedral grids are easier to generate for complex 3D configurations, hexahedral grids have been shown to possess higher efficiency and accuracy for viscous boundary layers.

The SD method²³⁻²⁶ and the staggered-grid multi-domain spectral method¹¹⁻¹² on hexahedral grids actually converge to the same method. The solution unknowns or degrees-of-freedom (DOFs) are the conserved variables at the Gauss points, while fluxes are evaluated at Lobatto points to generate the flux derivatives to update the DOFs. On a hexahedral element, all the operations can be performed in a one-dimensional manner, resulting in higher efficiency and less implementation cost. As in the Godunov-type finite volume method, Riemann solvers are used at element interfaces to couple the discontinuous elements together, and provide the necessary numerical dissipation which makes the method conservative and stable.

The paper is organized as follows. In the next section, the formulation of the 3D spectral difference method is described for a hexahedral element. Numerical results including accuracy studies are presented in Section 3, together with results for several often used demonstration cases. Conclusions and possible future work are outlined in Section 4.

II. Formulation of the 3D Spectral Difference Method on Hexahedral Grids

A. Governing equation

Consider the unsteady compressible 3D Navier-Stokes equations in conservative form written as

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z} = 0, \quad (1a)$$

where Q is the vector of conserved variables, and F , G , H are the total fluxes including both the inviscid and viscous flux vectors, i.e., $F = F^i - F^v$, $G = G^i - G^v$, $H = H^i - H^v$. Expressed in vector forms,

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{Bmatrix}, \quad F^i = \begin{Bmatrix} \rho u \\ p + \rho u^2 \\ \rho uv \\ \rho uw \\ u(E + p) \end{Bmatrix}, \quad G^i = \begin{Bmatrix} \rho v \\ \rho uv \\ p + \rho v^2 \\ \rho vw \\ v(E + p) \end{Bmatrix}, \quad H^i = \begin{Bmatrix} \rho w \\ \rho uw \\ \rho vw \\ p + \rho w^2 \\ w(E + p) \end{Bmatrix} \quad (1b)$$

and

$$F^v = \begin{Bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \\ u\tau_{xx} + v\tau_{yx} + w\tau_{zx} + \frac{\mu C_p}{P_r} T_x \end{Bmatrix}, \quad G^v = \begin{Bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \\ u\tau_{xy} + v\tau_{yy} + w\tau_{zy} + \frac{\mu C_p}{P_r} T_y \end{Bmatrix}, \quad H^v = \begin{Bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{xz} + v\tau_{yz} + w\tau_{zz} + \frac{\mu C_p}{P_r} T_z \end{Bmatrix} \quad (1c)$$

In (1a-1c), ρ is the density, u , v and w are the velocity components in x , y and z directions, p is the pressure, and E is the total energy, μ is dynamic viscosity, C_p is the specific heat at constant pressure, P_r is the Prandle number, and T is the temperature. For a perfect gas, the pressure is related to the total energy by

$$E = \frac{P}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2 + w^2), \quad (1d)$$

with a constant ratio of specific heats $\gamma = 1.4$ for air. The stress tensor in (1c) takes the following form

$$\tau_{xx} = 2\mu(u_x - (u_x + v_y + w_z)/3) \quad (1e.1)$$

$$\tau_{yy} = 2\mu(v_y - (u_x + v_y + w_z)/3) \quad (1e.2)$$

$$\tau_{zz} = 2\mu(w_z - (u_x + v_y + w_z)/3) \quad (1e.3)$$

$$\tau_{xy} = \tau_{yx} = \mu(v_x + u_y) \quad (1e.4)$$

$$\tau_{yz} = \tau_{zy} = \mu(w_y + v_z) \quad (1e.5)$$

$$\tau_{zx} = \tau_{xz} = \mu(u_z + w_x). \quad (1e.6)$$

B. Coordinate transformation

It is assumed that the computational domain is divided into non-overlapping unstructured hexahedral cells or elements. The use of hexahedral cells for viscous boundary layers is preferred over tetrahedral cells because of their efficiency and accuracy. In order to handle curved boundaries, both linear and quadratic isoparametric elements are employed, with linear elements used in the interior domain and quadratic elements used near high-order curved boundaries. In order to achieve an efficient implementation, all physical elements (x, y, z) are transformed into a standard cubic element $(\xi, \eta, \zeta) \in [0, 1] \times [0, 1] \times [0, 1]$ as shown in Figure 1. The transformation can be written as

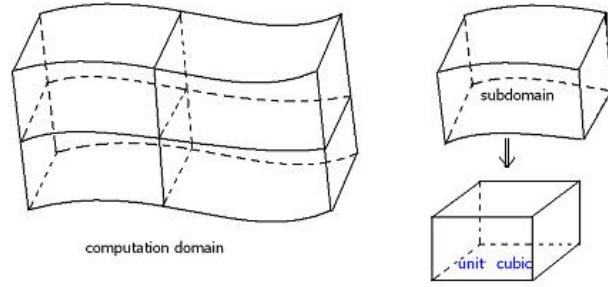


Figure 1 Transformation from a physical element to a standard element

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \sum_{i=1}^K M_i(\xi, \eta, \zeta) \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}, \quad (2)$$

where K is the number of points used to define the physical element, (x_i, y_i, z_i) are the Cartesian coordinates of those points, and $M_i(\xi, \eta, \zeta)$ are the shape functions. For the transformation given in (2), the Jacobian matrix J takes the following form

$$J = \frac{\partial(x, y, z)}{\partial(\xi, \eta, \zeta)} = \begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix}.$$

For a non-singular transformation, its inverse transformation must also exist, and the Jacobian matrices are related to each other according to

$$\frac{\partial(\xi, \eta, \zeta)}{\partial(x, y, z)} = \begin{bmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{bmatrix} = J^{-1}.$$

Therefore the metrics can be computed according to

$$\begin{aligned} \xi_x &= (y_\eta z_\zeta - y_\zeta z_\eta) / |J|, & \xi_y &= (x_\zeta z_\eta - x_\eta z_\zeta) / |J|, & \xi_z &= (x_\eta y_\zeta - x_\zeta y_\eta) / |J|, \\ \eta_x &= (y_\zeta z_\xi - y_\xi z_\zeta) / |J|, & \eta_y &= (x_\xi z_\zeta - x_\zeta z_\xi) / |J|, & \eta_z &= (x_\zeta y_\xi - x_\xi y_\zeta) / |J|, \\ \zeta_x &= (y_\xi z_\eta - y_\eta z_\xi) / |J|, & \zeta_y &= (x_\eta z_\xi - x_\xi z_\eta) / |J|, & \zeta_z &= (x_\xi y_\eta - x_\eta y_\xi) / |J|. \end{aligned}$$

The governing equations in the physical domain are then transformed into the computational domain (standard element), and the transformed equations take the following form

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial \tilde{F}}{\partial \xi} + \frac{\partial \tilde{G}}{\partial \eta} + \frac{\partial \tilde{H}}{\partial \zeta} = 0. \quad (3)$$

where

$$\begin{aligned} \tilde{Q} &= |J| \cdot Q \\ \tilde{F} &= \tilde{F}^i - \tilde{F}^v, \tilde{G} = \tilde{G}^i - \tilde{G}^v, \tilde{H} = \tilde{H}^i - \tilde{H}^v. \\ \begin{bmatrix} \tilde{F}^i \\ \tilde{G}^i \\ \tilde{H}^i \end{bmatrix} &= |J| \begin{bmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{bmatrix} \cdot \begin{bmatrix} F^i \\ G^i \\ H^i \end{bmatrix}, \quad \begin{bmatrix} \tilde{F}^v \\ \tilde{G}^v \\ \tilde{H}^v \end{bmatrix} = |J| \begin{bmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \zeta_x & \zeta_y & \zeta_z \end{bmatrix} \cdot \begin{bmatrix} F^v \\ G^v \\ H^v \end{bmatrix}. \end{aligned}$$

Let $\vec{S}_\xi = |J|(\xi_x, \xi_y, \xi_z)$, $\vec{S}_\eta = |J|(\eta_x, \eta_y, \eta_z)$, $\vec{S}_\zeta = |J|(\zeta_x, \zeta_y, \zeta_z)$. Then we have $\tilde{F} = \vec{f} \cdot \vec{S}_\xi$, $\tilde{G} = \vec{f} \cdot \vec{S}_\eta$, $\tilde{H} = \vec{f} \cdot \vec{S}_\zeta$, with $\vec{f} = (F, G, H)$.

C. Space discretization

In the standard element, two sets of points are defined, namely the solution points and the flux points, illustrated in Figure 2 for 2D. The solution unknowns or degrees-of-freedom (DOFs) are the conserved variables at the solution points, while fluxes are computed at the flux points in order to update the solution unknowns. In order to construct a degree $(N-1)$ polynomial in any direction, solutions at N points are required. The solution points in 1D are chosen to be the Gauss quadrature points defined by

$$X_s = \frac{1}{2} \left[1 - \cos \left(\frac{2s-1}{2N} \cdot \pi \right) \right], \quad s = 1, 2, \dots, N. \quad (4)$$

The flux points are the Gauss-Lobatto points given by

$$X_{s+1/2} = \frac{1}{2} \left[1 - \cos \left(\frac{s}{N} \cdot \pi \right) \right], \quad s = 0, 1, \dots, N. \quad (5)$$

For the DOFs at the solution points, we employ degree $N-1$ Lagrange interpolation polynomials in each coordinate direction defined as

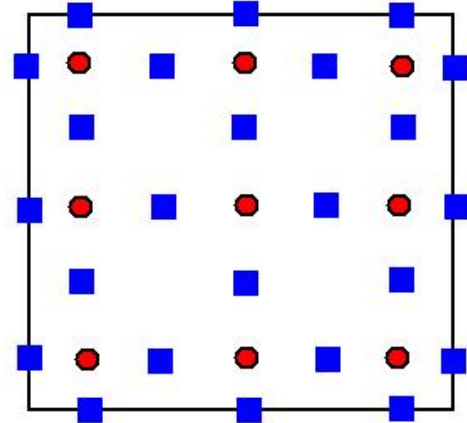


Figure 2 Distribution of solution points (circles) and flux points (squares) in a standard element for a third-order SD scheme

$$h_i(X) = \prod_{s=1, s \neq i}^N \left(\frac{X - X_s}{X_i - X_s} \right). \quad (6a)$$

Similarly, for the fluxes at the flux points, we use degree N Lagrange interpolation polynomials in each coordinate direction given by

$$l_{i+1/2}(X) = \prod_{s=0, s \neq i}^N \left(\frac{X - X_{s+1/2}}{X_{i+1/2} - X_{s+1/2}} \right). \quad (6b)$$

The reconstructed solution for the conserved variables in the standard element is just the tensor products of the three one-dimensional polynomials, i.e.,

$$Q(\xi, \eta, \varsigma) = \sum_{k=1}^N \sum_{j=1}^N \sum_{i=1}^N \frac{\tilde{Q}_{i,j,k}}{|J_{i,j,k}|} h_i(\xi) \cdot h_j(\eta) \cdot h_k(\varsigma). \quad (7)$$

Similarly, the reconstructed fluxes in each direction take the following form:

$$\tilde{F}(\xi, \eta, \varsigma) = \sum_{k=1}^N \sum_{j=1}^N \sum_{i=0}^N \tilde{F}_{i+1/2,j,k} l_{i+1/2}(\xi) \cdot h_j(\eta) \cdot h_k(\varsigma), \quad (8a)$$

$$\tilde{G}(\xi, \eta, \varsigma) = \sum_{k=1}^N \sum_{j=0}^N \sum_{i=i}^N \tilde{G}_{i,j+1/2,k} h_i(\xi) \cdot l_{j+1/2}(\eta) \cdot h_k(\varsigma), \quad (8b)$$

$$\tilde{H}(\xi, \eta, \varsigma) = \sum_{k=0}^N \sum_{j=1}^N \sum_{i=1}^N \tilde{H}_{i,j,k+1/2} h_i(\xi) \cdot h_j(\eta) \cdot l_{k+1/2}(\varsigma). \quad (8c)$$

The reconstructed fluxes are only element-wise continuous, but discontinuous across cell interfaces. For the inviscid flux, a Riemann solver, such as the Rusanov²⁷ or Roe flux²⁸, is employed to compute a common flux at interfaces to ensure conservation and stability. For the viscous flux, the approach in Reference 12 is employed. The viscous flux is a function of both the conserved variables and their gradients, i.e., $\tilde{F}_{i+1/2,j,k}^v = \tilde{F}^v(Q_{i+1/2,j,k}, \nabla Q_{i+1/2,j,k})$, $\tilde{G}_{i,j+1/2,k}^v = \tilde{G}^v(Q_{i,j+1/2,k}, \nabla Q_{i,j+1/2,k})$, and $\tilde{H}_{i,j,k+1/2}^v = \tilde{H}^v(Q_{i,j,k+1/2}, \nabla Q_{i,j,k+1/2})$. The gradient of the conserved variables in the physical domain can be easily computed using

$$\nabla Q = \frac{1}{|J|} \left[\frac{\partial Q \bar{s}_\xi}{\partial \xi} + \frac{\partial Q \bar{s}_\eta}{\partial \eta} + \frac{\partial Q \bar{s}_\varsigma}{\partial \varsigma} \right]. \quad (9)$$

Here the derivatives along each coordinate line

$$\left(\frac{\partial Q \bar{s}_\xi}{\partial \xi} \right)_{j,k} = \sum_{r=0}^N (Q \bar{s}_\xi)_{r+1/2,j,k} \cdot l'_{r+1/2}(\xi) \quad (10a)$$

$$\left(\frac{\partial Q \bar{s}_\eta}{\partial \eta} \right)_{i,k} = \sum_{r=0}^N (Q \bar{s}_\eta)_{i,r+1/2,k} \cdot l'_{r+1/2}(\eta) \quad (10a)$$

$$\left(\frac{\partial \bar{Q} \bar{S}_\zeta}{\partial \zeta} \right)_{i,j} = \sum_{r=0}^N (\bar{Q} \bar{S}_\zeta)_{i,j,r+1/2} \cdot l'_{r+1/2}(\zeta). \quad (10a)$$

Once all fluxes are computed at the flux points, the flux polynomials are built according (8), and the derivatives of the fluxes are then evaluated at each solution point to update the DOFs, i.e.,

$$\frac{\partial \tilde{Q}_{i,j,k}}{\partial t} = - \left(\frac{\partial \tilde{F}}{\partial \xi} + \frac{\partial \tilde{G}}{\partial \eta} + \frac{\partial \tilde{H}}{\partial \zeta} \right)_{i,j,k}. \quad (11)$$

For time integration, we employ a multistage SSP or TVD Runge-Kutta scheme¹⁶. In summary, the algorithm to compute the inviscid flux derivatives employs the following steps:

1. Given the conserved variables at the solution points $\{\tilde{Q}_{i,j,k}\}$, compute the conserved variables at the flux points $\{Q_{i+1/2,j,k}, Q_{i,j+1/2,k}, Q_{i,j,k+1/2}\}$ using (7) (Note that $h_m(X_n) = \delta_{mn}$);
2. Compute the inviscid fluxes at the interior flux points using the solutions computed at Step 1, i.e., $\{\tilde{F}_{i+1/2,j,k}^i, i = 1, \dots, N-1\}$, $\{\tilde{G}_{i,j+1/2,k}^i, j = 1, \dots, N-1\}$, $\{\tilde{H}_{i,j,k+1/2}^i, k = 1, \dots, N-1\}$;
3. Compute the inviscid flux at element interfaces using a Riemann solver, such as the Rusanov solver. In terms of the left and right conserved variables and fluxes (as shown in Figure 3), the normal direction of the interface \bar{n} , and the averaged normal velocity component \bar{V}_n and sound speed \bar{c} , the interface fluxes for the Rusanov solver are given by

$$\begin{aligned} \tilde{F}^i &= \frac{1}{2} (\tilde{F}_L^i + \tilde{F}_R^i - (|\bar{V}_n| + \bar{c}) \cdot (Q_R - Q_L) \cdot |\bar{S}_\xi| \cdot \text{sign}(\bar{n} \cdot \bar{S}_\xi)) \\ \tilde{G}^i &= \frac{1}{2} (\tilde{G}_L^i + \tilde{G}_R^i - (|\bar{V}_n| + \bar{c}) \cdot (Q_R - Q_L) \cdot |\bar{S}_\eta| \cdot \text{sign}(\bar{n} \cdot \bar{S}_\eta)) \\ \tilde{H}^i &= \frac{1}{2} (\tilde{H}_L^i + \tilde{H}_R^i - (|\bar{V}_n| + \bar{c}) \cdot (Q_R - Q_L) \cdot |\bar{S}_\zeta| \cdot \text{sign}(\bar{n} \cdot \bar{S}_\zeta)) \end{aligned}$$

4. Compute the derivatives of the fluxes at all the solution points according to

$$\left(\frac{\partial \tilde{F}}{\partial \xi} \right)_{i,j,k} = \sum_{r=0}^N \tilde{F}_{r+1/2,j,k} \cdot l'_{r+1/2}(\xi_i) \quad (12a)$$

$$\left(\frac{\partial \tilde{G}}{\partial \eta} \right)_{i,j,k} = \sum_{r=0}^N \tilde{G}_{i,r+1/2,k} \cdot l'_{r+1/2}(\eta_j) \quad (12b)$$

$$\left(\frac{\partial \tilde{H}}{\partial \zeta} \right)_{i,j,k} = \sum_{r=0}^N \tilde{H}_{i,j,r+1/2} \cdot l'_{r+1/2}(\zeta_k). \quad (12c)$$

To compute the viscous flux derivatives at the flux points, we follow the algorithm given in Reference 12:

1. Same as Step 1 for the inviscid flux computations;
2. Compute the average solutions at cell interface flux points using the solutions at both sides of the interface

$$\bar{Q} = \frac{1}{2} (Q_L + Q_R);$$

3. Compute the gradients of the solutions at the flux points using the computed solutions at the flux points. One can compute the gradients at the solution points first using Eqs. (9) and (10) with the same set of derivative coefficients given in Eq. (12) and then interpolate these gradients to the flux points with the same set of coefficients used in Step 1, or one can compute the gradients at the flux points directly by evaluating the derivative coefficients at those points;

4. Compute the average gradients at cell interface flux points using the gradients from both sides of the interface

$$\nabla \bar{Q} = \frac{1}{2}(\nabla Q_L + \nabla Q_R);$$

5. Compute the viscous fluxes at all flux points using the solutions computed at Step 2 and the gradients computed at Step 5.

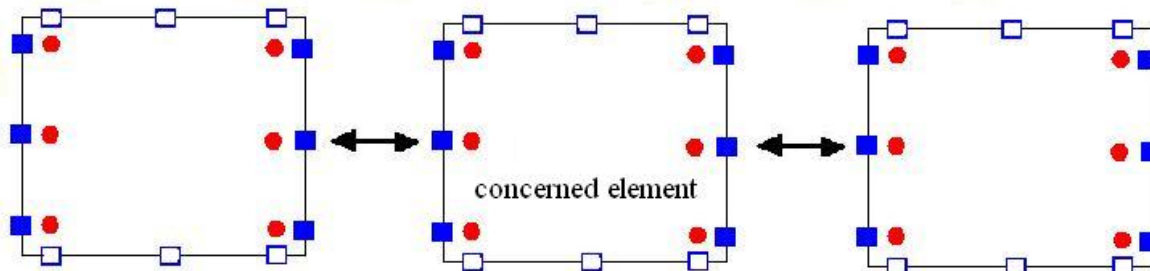


Figure 3 The connectivity of an element with its two horizontal neighbors showing points where the solutions (red circles) and horizontal fluxes (solid blue squares) are located.

III. Numerical Results

A. Accuracy study using the vortex propagating problem

In order to quantify the numerical order of accuracy of the SD method for the Euler equations with both mesh-refinement (h-refinement) and order refinement (p-refinement), the vortex propagation problem is chosen since it has an analytical solution. This is an idealized problem for the Euler equations, which was used by Shu⁵.

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 $T = p / \rho$, and no perturbation in velocity w and entropy $S = p / \rho^\gamma$:

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

where $r^2 = x^2 + y^2$, and the vortex strength $\varepsilon = 5$. In the numerical simulation, the computational domain is taken to be $[-5, 5] \times [-5, 5] \times [-5, 5]$, with characteristic inflow and outflow boundary conditions imposed on the boundaries in x and y directions, and extrapolation in the z direction. It can be readily verified that the Euler equations with the above initial conditions admit an exact solution that moves with the speed $(1, 1, 0)$ in the diagonal direction on the x - y plane.

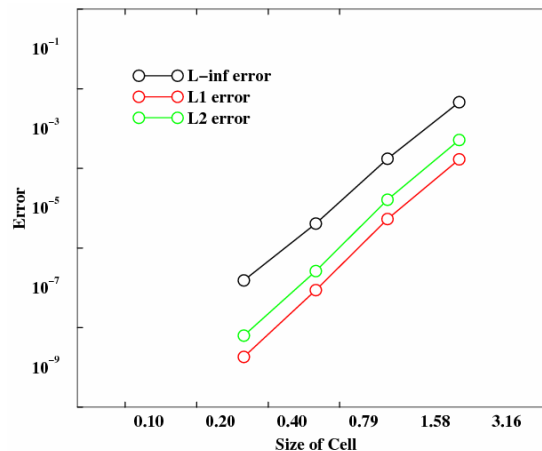


Figure 4 Time-independent errors for the vortex propagation problem with h-refinement

For the h -refinement study, the order of accuracy (np) was set to 6. Four meshes were employed. Figure 4 shows the time-independent errors between the numerical solution and the analytical solution in L_∞ , L_1 and L_2 norms at $t=0.1$. The three-stage Runge-Kutta scheme was employed for the time integration. Note that the average slope is about 6, indicating that the numerical order of accuracy is 6th-order.

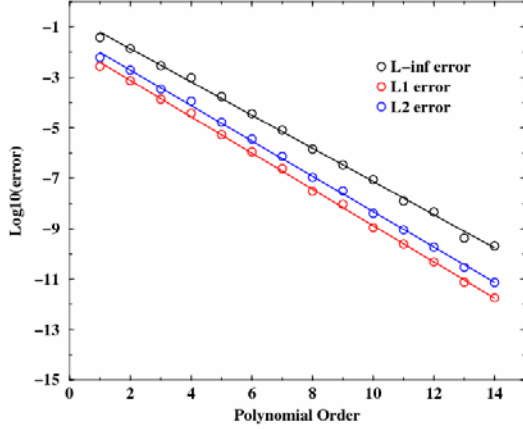


Figure 5 Convergence of density error for the vortex propagation problem with p-refinement

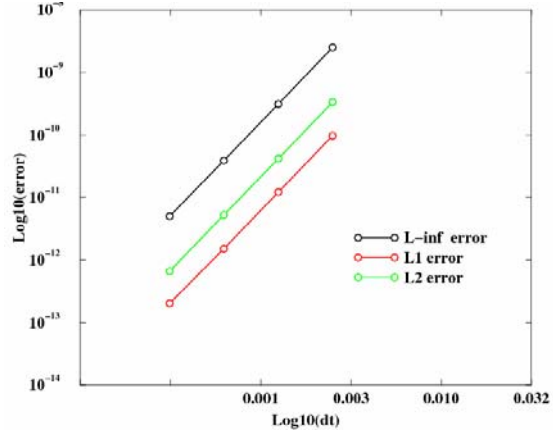


Figure 6 Temporal error of the vortex problem on a fine mesh

For the p -refinement study, a coarse grid with 100 (10x10x1) elements was used. Then the order of the polynomial basis was increased. The numerical errors are displayed at $t=0.1$ in Figure 5. Note that the exponential decay of error with respect to the order of accuracy is achieved.

Lastly, the time integration accuracy was also studied on a fine mesh (25 elements, $np = 25$). The numerical errors with different time-step sizes are displayed at $t=0.1$ in Figure 6. Note that the order of accuracy of the time integration scheme is 3rd-order.

B. Accuracy study using the Couette flow

The Couette flow is an analytical solution of the Navier-Stokes equations, and was selected to study the accuracy for the 3D Navier-Stokes solver. This problem models the viscous flow between a stationary, fixed temperature (T_0) front plate, and a moving, fixed temperature (T_1) rear plate at speed of U . The distance between the two plates is H . It has an exact solution under the simplification that the viscosity coefficient μ is a constant.

The steady analytic solution is

$$u = \frac{U}{H} y, \quad v = 0, \quad w = 0$$

$$T = T_0 + \frac{y}{H} (T_1 - T_0) + \frac{\mu U^2}{2k} \cdot \frac{y}{H} \left(1 - \frac{y}{H}\right)$$

$$p = \text{constant}, \quad \rho = \frac{p}{R \cdot T},$$

where k is the thermal conductivity, and R is the gas constant.

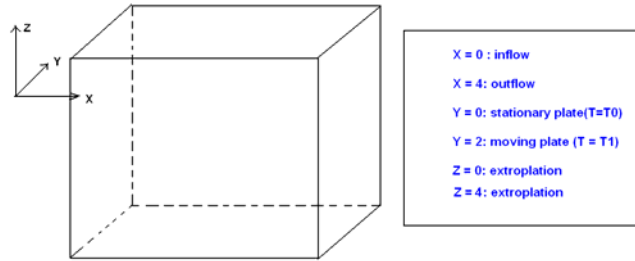


Figure 7 Computational domain and boundary conditions

In our simulations, we chose $U = 1.0, H = 2.0, T_0 = 0.8, T_1 = 0.85, \mu = 0.01$. The computational domain of $[0,4] \times [0,2] \times [0,4]$ is displayed as Figure 7.

We conducted a p-refinement study on a coarse grid with 16 (4x4x1) elements. The order of the polynomial basis was increased up to 9, and the 3-stage Runge-Kutta scheme was employed for the time integration. The exponential decay of numerical errors in L_∞ , L_1 , and L_2 norms was again observed as shown in Figure 8.

C. Flow over a cylinder

To test the capability of SD method for handling curved boundaries, the flow over a cylinder was selected as a validation case. The simulations were performed for both inviscid and viscous flows. Two computational grids were generated and are shown in Figure 9. The coarse grid has 160 elements and the fine grid has 590 elements.

The coarse grid was employed in the inviscid flow simulation using the 4th order SD scheme with both linear and quadratic boundary representations. It was found that the quadratic boundaries were critical for inviscid flows. This is fully demonstrated in Figure 10, which displays the x velocity contours with the linear and quadratic boundaries. There is a large entropy increase in the flow with the linear boundary representation, and the flow is not symmetric at all. On the other hand, the flow with the quadratic boundary produced a symmetric flow pattern, as shown in the right plot of Figure 10.

The low Reynolds number viscous flow problems were solved on the fine grid again using the 4th order SD scheme. Numerical results show that periodic vortex shedding was predicted very well, as depicted in Figure 11, which displays the instantaneous x velocity contours with a Reynolds number of 150. Several different Reynolds numbers were then selected in the simulations using the coarse grid. The pressure history plots with Reynolds numbers 60 and 120 near

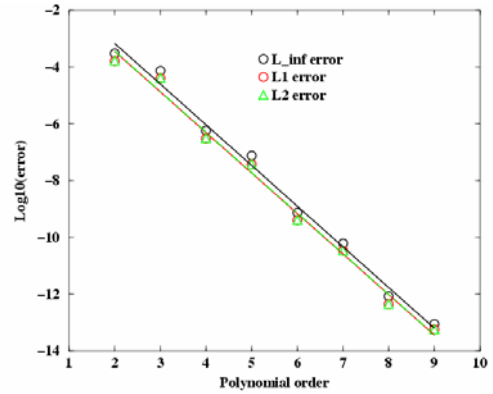


Figure 8 Convergence of the density error for the Couette flow problem

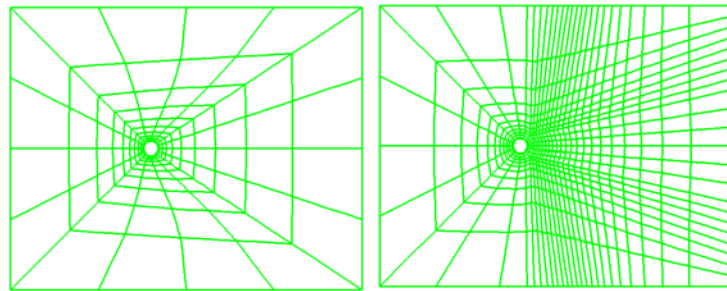


Figure 9 coarse and fine grids for a cylinder

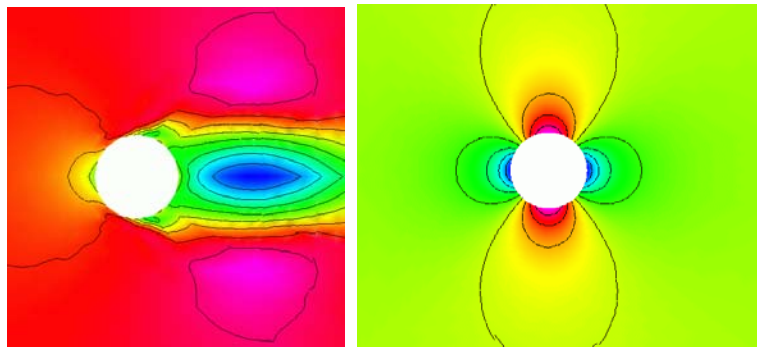


Figure 10. u contours: linear wall (left), quadratic wall (right)

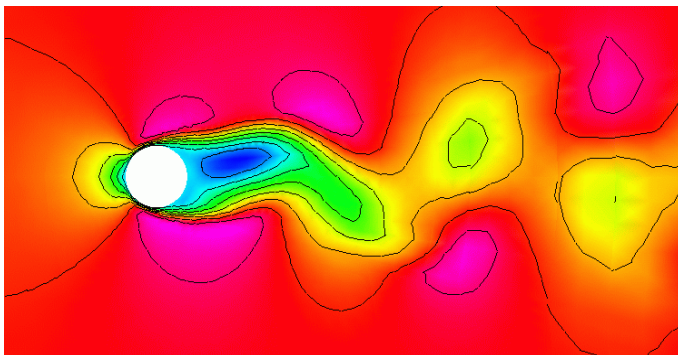


Figure 11 u-velocity contours at z-cutting plane

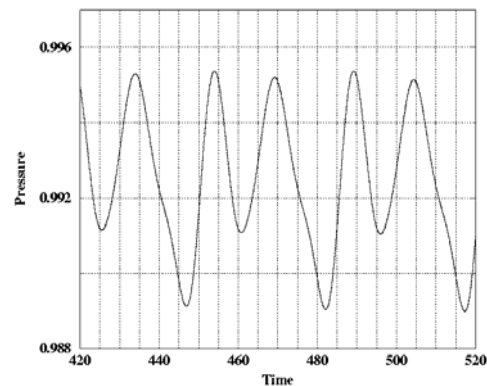


Figure 12 Pressure history (Re=60)

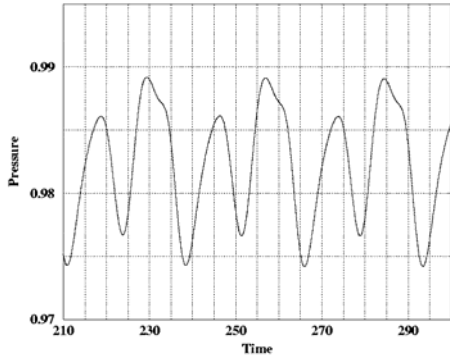


Figure 13 Pressure history (Re=120)

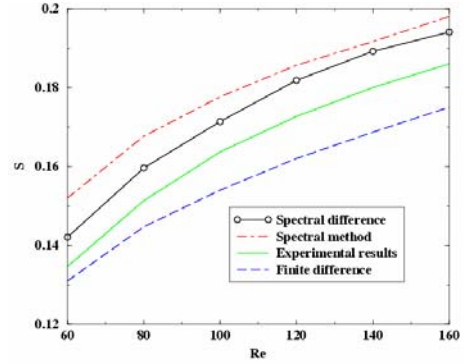


Figure 14 Strouhal-Reynolds relationship

the wake of the cylinder are plotted in Figures 12 and 13. Note that the flow reached a periodic solution for both Reynolds numbers. In addition, the relationship between the Strouhal number and the Reynolds number is compared with experimental results and other numerical simulations in Figure 14. The SD predictions are within the range of other numerical results and close to the experimental results.

D. Flow over a sphere

Finally, inviscid and viscous flows over a sphere were computed to further demonstrate the potential of the SD method for a truly 3D flow problem. Figure 15 shows the computational grid having 768 total cells with quadratic boundaries.

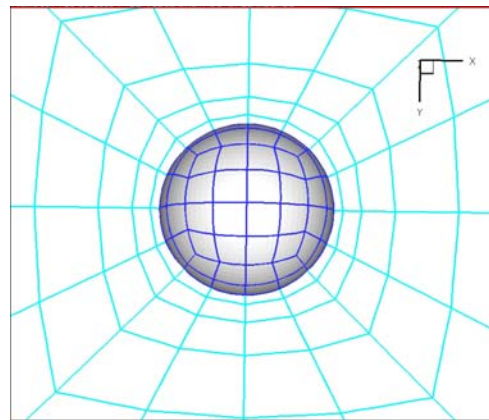


Figure 15 Sphere grids with quadratic boundary (768 elements)

For the inviscid flow simulation, the Mach number is chosen to be 0.3, again using the 4th order of accuracy in space and 3-stage Runge-Kutta scheme in time. The velocity contours are displayed in Figure 16. The flow field is obviously symmetric indicating a low level of entropy generation.

For the viscous flow simulation, the Reynolds number is chosen to be 118. The computational streamlines are compared with the experimental streamlines in Figures 17. In both plots the steady separation bubble is readily observed, and the size of the separation region in the computation agrees very well with that of the experiment.

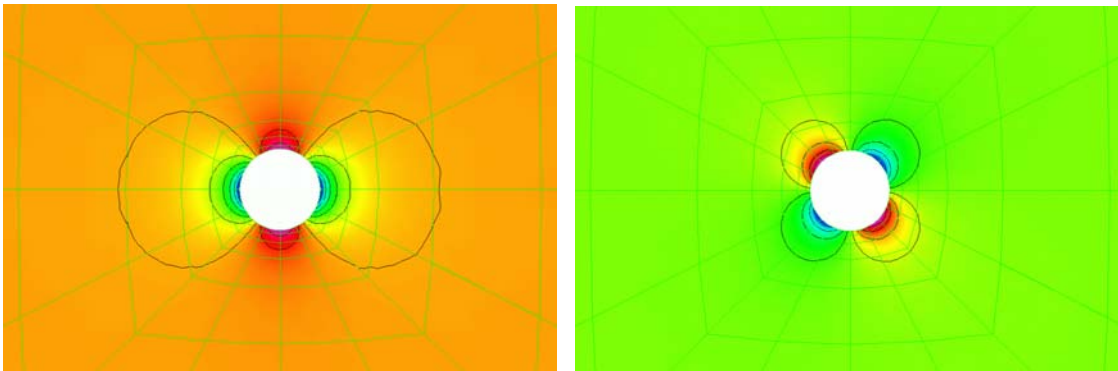


Figure 16 u-velocity (left), v-velocity (right) contours

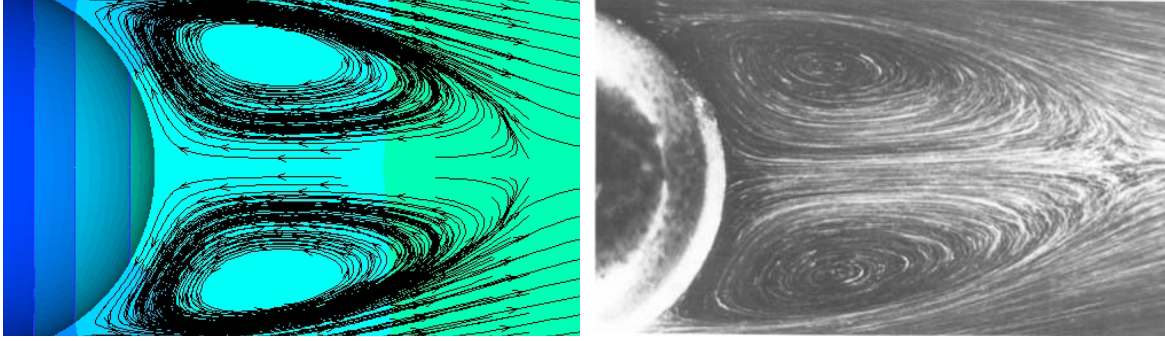


Figure 17 Streamlines of the flow field from computation (left) and experiment (right) results

IV. Conclusions and Future Work

In this paper, the multidomain spectral difference method has been successfully extended to 3D for solving the full Navier-Stokes equations on unstructured hexahedral grids. Because the SD method is based on the differential form of the governing equations, the implementation is straightforward even for high-order curved boundaries. All the operations are basically one-dimensional in each coordinate direction, resulting in improved efficiency. The high order accuracy of the SD method is numerically verified in both space and time, and the exponential decay of the numerical error with respect to p-refinement has been achieved for the vortex propagation problem and the Couette flow. Numerical experiments for flow over a cylinder and a sphere have demonstrated the capability of the SD solver in the treatment of high order curved boundaries. We are currently validating the SD method for more complex flow problems, and working on efficient implicit solution algorithms.

Acknowledgements

The study has been funded by Rockwell Scientific/DARPA under contract W911NF-04-C-0102. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, or the U.S. Government.

References

- ¹Conlisk, A.T., Modern Helicopter Aerodynamics, *Annu. Rev. Fluid. Mech.*, vol. 29, pp 515-567, 1997.
- ²Meakin, R.L., Moving body overset grid methods for complete tiltrotor simulations, AIAA paper 93-3350-CP. Presented at the 11th AIAA Computational Fluid Dynamics Conference, Orlando, FL, July 1993.
- ³Duque, E.P.N., Biswas, R., and Strawn, R.C., A solution adaptive structured/unstructured overset grid solver with applications to helicopter rotor flows, AIAA paper 95-1766. Presented at Appl. Aerodyn. Conf., 13th, San Diego, Calif.
- ⁴Biswas, R., Strawn, R. C., A Dynamic Mesh Adaption Procedure for Unstructured Hexahedral Grids, AIAA-96-0027, presented at the 34th Aerospace Sciences Meeting, Reno, NV, Jan. 15-18, 1996.
- ⁵Shu, 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* (Springer-Verlag, Berlin/New York, 1998), Vol. 1697, P. 325.
- ⁶Lele, S.K., "Compact Finite Difference Schemes with Spectral-Like Resolution," *Journal of Computational Physics*, Vol. 103, 1992, pp. 16-42.
- ⁷Visbal, M. and Gaitonde, D., Shock Capturing Using Compact- Differencing- Based Methods, AIAA-2005-1265.
- ⁸Tam, C.K.W., and Webb, J.C., "Dispersion-Relation-Preserving Finite difference Schemes for Computational Acoustics," *Journal of Computational Physics*, Vol. 107, No. 2, 1993, pp. 262-281.
- ⁹Visbal, M.R., Morgan, P.E., and Rizzetta, D.P., An Implicit LES Approach Based on High-Order Compact Differencing and Filtering Schemes, AIAA 2003-4098.
- ¹⁰Patera, A.T., A Spectral element method for fluid dynamics: Laminar flow in a channel expansion, *J. Comput. Phys.* 54, 468(1984).
- ¹¹Kopriva, D.A. and Kalias, J.H., A conservative staggered -grid Chebyshev multidomain method for compressible flows, *J. Comput. Phys.* 125, 244(1996).
- ¹²Kopriva, D.A., A Staggered-Grid Multidomain Spectral Method for the Compressible Navier-Stokes Equations, *Journal of Computational Physics*, Volume 143, pp. 125-158, 1998.
- ¹³Barth, T.J. and Frederickson, P.O., High-Order Solution of the Euler Equations on Unstructured Grids using Quadratic Reconstruction, AIAA Paper No. 90-0013(1990).

- ¹⁴Hu, C. and Shu, C.-W., Weighted essentially non-oscillatory schemes on triangular meshes, *J. Comput. Phys.* 150, 97 (1999).
- ¹⁵Cockburn, B. and Shu, C.-W., TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework, *Math. Comput.* 52,411(1989).
- ¹⁶Cockburn, B. and Shu, C.-W., The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems, *J. Comput. Phys.*, 141, 199 - 224, (1998).
- ¹⁷Bassi, F. and Rebay, S., High-order accurate discontinuous finite element solution of the 2D Euler equations, *J. Comput. Phys.* 138, 251-285 (1997).
- ¹⁸Abgrall, R. and Roe, P.L., High Order Fluctuation Schemes on Triangular Meshes, *Journal of Scientific Computing*, Volume 19, pp. 3 – 36, 2003.
- ¹⁹Wang, Z.J. and Liu, Yen, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids III: Extension to One-Dimensional Systems, *J. Scientific Computing*, Vol. 20 No. 1, pp.137-157 (2004).
- ²⁰Wang, Z.J., Zhang, L., and Liu, Yen, Spectral (finite) volume method for conservation laws on unstructured grids IV: extension to two-dimensional systems, *J. Comput. Phys.* 194, 716-741(2004).
- ²¹Liu, Y., Vinokur, M., and Wang, Z.J., Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids V: Extension to Three-Dimensional Systems, *Journal of Computational Physics* Vol. 212, pp. 454-472 (2006).
- ²²Sun, Yuzhi, Wang, Z.J., and Liu, Yen, Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids VI: Extension to Viscous Flow, *Journal of Computational Physics*, in press.
- ²³Liu, Y., Vinokur, M., and Wang, Z.J., Discontinuous Spectral Difference Method for Conservation Laws on Unstructured Grids, in *Proceeding of the 3rd International Conference in CFD*, Toronto, Canada July 2004.
- ²⁴Liu, Yen, Vinokur, M., and Wang, Z.J., Multi-Dimensional Spectral Difference Method for Unstructured Grids, AIAA-2005-0320.
- ²⁵Wang, Z. J., and Liu, Yen, The Spectral Difference Method for the 2D Euler Equations on Unstructured Grids, AIAA-2005-5112.
- ²⁶Huang, P.G., Wang, Z.J., and Liu, Yen, An Implicit Space-Time Spectral Difference Method for Discontinuity Capturing Using Adaptive Polynomials, AIAA-2005-5255.
- ²⁷Rusanov, V.V., Calculation of interaction of non-steady shock waves with obstacles, *J. Comput. Math. Phys. USSR* 1, 267-279 (1961).
- ²⁸Roe, P.L., Approximate Riemann solvers, parameter vectors, and difference schemes, *J. Comput. Phys.* 43 357-372 (1981).