Development of a 3D Viscous Compressible Flow Solver using Spectral Difference Method on Unstructured Hexahedral Grids

Report ACL 2008-3

by

Chunlei Liang, Antony Jameson, Sachin Premasuthan

Aerospace Computing Laboratory

Aeronautics and Astronautics

Stanford University

- April 2008 -

Abstract

This report presents a three-dimensional high-order Spectral Difference solver suitable for Large Eddy Simulation. The solver is based on the formulation of Sun et al. (2007) implemented on unstructured hexahedral grid elements. It is firstly validated using a testing problem of subsonic inviscid flow past a circular cylinder. We demonstrate the spatial accuracy up to four-order using a viscous compressible Couette flow with analytical solution.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The increase in computational power is enabling three-dimensional simulations of various research problems previously deemed intractable such as unsteady Large Eddy Simulation and Aero-acoustics. These simulations, however, require highly accurate numerical methods which are efficient and yet robust to solve real-world problems. Clearly, the need for highly accurate methods in applications such as large eddy simulation, direct numerical simulation, computational aero-acoustics etc., has been driving forces for the development of higher order schemes for unstructured meshes such as the Discontinuous Galerkin (DG) Method (Bassi and Rebay (1997)), Spectral Volume (SV) method (Liu et al. (2006*b*); Wang and Liu (2006)) and Spectral Difference (SD) Method (Liu et al. (2006*a*); Wang et al. (2007)). The SD method is a newly developed efficient high-order approach based on differential form of the governing equation. It was originally proposed by Liu et al. (2006*a*) and developed for wave equations in their paper on triangular grids. Wang et al. (2007) extended it to 2D Euler equations on triangular grids and Sun et al. (2007) further developed it for three-dimensional Navier-Stokes equations on hexahedral unstructured meshes. The SD method combines elements from finite-volume and finite-difference techniques, and is particularly attractive because it is conservative, and has a simple formulation and easy to implement.

Chapter 2

Numerical formulation

The formulation of the equations is similiar to the formulation of Sun et al. (2007) for unstructured hexahedral grids. Consider the unsteady compressible 3D Navier Stokes equations in conservative form

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} + \frac{\partial H}{\partial z} = 0 \tag{2.1}$$

where Q is the vector of conserved variables; F, G and H are the total fluxes including both inviscid and viscous flux vectors. To achieve an efficient implementation, all elements in the physical domain (x, y, z) are transformed into a standard square element ($0 \leq \xi \leq 1$, $0 \leq \eta \leq 1$ and $0 \leq \beta \leq 1$).

The transformation can be written as:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \sum_{i=1}^{K} M_i(\xi, \eta, \beta) \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} \tag{2.2}$$

where K is the number of points used to define the physical element, (x_i, y_i, z_i) are the cartesian coordinates at those points, and $M_i(\xi, \eta, \beta)$ are the shape functions. For present implementation, we define K as 8 nodal points illustrated in figure 2.1. Nodal point 1 location is ($\xi = 0$, $\eta = 0$, $\beta = 0$). The metrics and the Jacobian of the transformation can be computed. The governing equations in the physical domain are then transferred into the computational domain, and the transformed equations take the following form:

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial \tilde{F}}{\partial \xi} + \frac{\partial \tilde{G}}{\partial \eta} + \frac{\partial \tilde{H}}{\partial \beta} = 0 \tag{2.3}$$

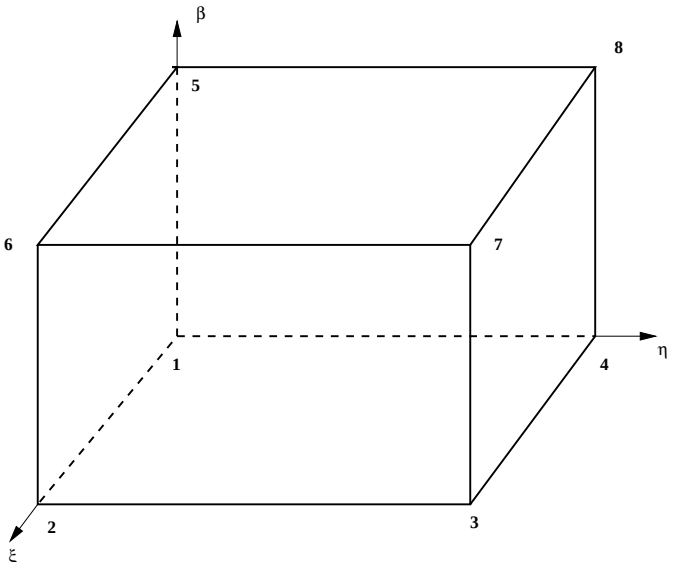


Figure 2.1: The nodal points and standard 3D cell orientation

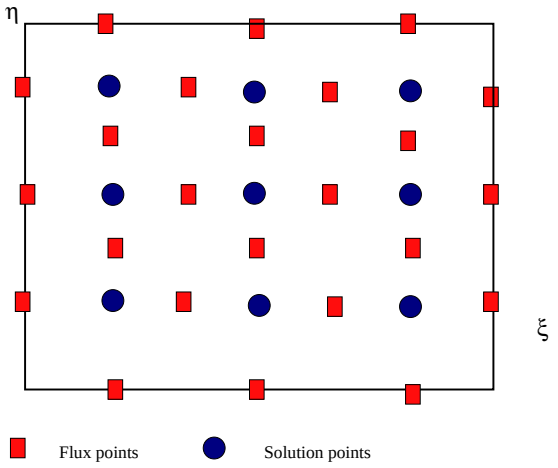


Figure 2.2: Distribution of flux and solution points for the third order SD scheme

where $\tilde{Q} = |J| \cdot Q$ and

$$\begin{pmatrix} \tilde{F} \\ \tilde{G} \\ \tilde{H} \end{pmatrix} = |J| \begin{pmatrix} \xi_x & \xi_y & \xi_z \\ \eta_x & \eta_y & \eta_z \\ \beta_x & \beta_y & \beta_z \end{pmatrix} \begin{pmatrix} F \\ G \\ H \end{pmatrix} \tag{2.4}$$

In the standard element, two sets of points are defined, namely the solution points and the flux points, illustrated in figure 2.2.

In order to construct a degree $(N-1)$ polynomial in each coordinate direction, solution at N points are required. The solution points in 1D are chosen to be the Gauss points defined by:

$$X_s = \frac{1}{2}\left[1 - cos\left(\frac{2s-1}{2N} \cdot \pi\right)\right], s = 1, 2, \cdots, N. \tag{2.5}$$

The flux points are selected to be the Gauss-Lobatto points given by

$$X_{s+1/2} = \frac{1}{2}\left[1 - cos\left(\frac{s}{N}\cdot\pi\right)\right], s = 0, 1, \cdots, N. \tag{2.6}$$

Using the solutions at N solution points, a degree $(N-1)$ polynomial can be built using the following Lagrange basis defined as:

$$h_i(X) = \prod_{s=0, s\neq i}^{N}\left(\frac{X - X_s}{X_i - X_s}\right) \tag{2.7}$$

Similarly, using the fluxes at $(N+1)$ flux points, a degree N polynomial can be built for the flux using a similar Lagrange basis defined as:

$$l_{i+1/2}(X) = \prod_{s=0, s\neq i}^{N}\left(\frac{X - X_{s+1/2}}{X_{i+1/2} - X_{s+1/2}}\right) \tag{2.8}$$

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

$$Q(\xi, \eta, \beta) = \sum_{k=1}^{N}\sum_{j=1}^{N}\sum_{i=1}^{N}\frac{\tilde{Q}_{i,j,k}}{|J_{i,j,k}|}h_i(\xi)\cdot h_j(\eta)\cdot h_k(\beta) \tag{2.9}$$

Similarly, the reconstructed flux polynomials take the following form:

$$\begin{aligned}
\tilde{F}(\xi, \eta, \beta) &= \sum_{k=1}^{N}\sum_{j=1}^{N}\sum_{i=0}^{N}\tilde{F}_{i+1/2,j,k}l_{i+1/2}(\xi)\cdot h_j(\eta)\cdot h_k(\beta), \\
\tilde{G}(\xi, \eta, \beta) &= \sum_{k=1}^{N}\sum_{j=0}^{N}\sum_{i=1}^{N}\tilde{G}_{i,j+1/2,k}h_i(\xi)\cdot l_{j+1/2}(\eta)\cdot h_k(\beta), \\
\tilde{H}(\xi, \eta, \beta) &= \sum_{k=0}^{N}\sum_{j=1}^{N}\sum_{i=1}^{N}\tilde{H}_{i,j,k+1/2}h_i(\xi)\cdot h_j(\eta)\cdot l_{k+1/2}(\beta)
\end{aligned} \tag{2.10}$$

The reconstructed fluxes are only element-wise continuous, but discontinuous across cell interfaces. For the inviscid flux, a Riemann solver is employed to compute a common flux at interfaces to ensure conservation amd stability. In our case, we have used the Rusanov solver to compute the interface fluxes.

In summary, the algorithm to compute the inviscid flux derivatives consists of the following steps:

1. Given the conservative variables at the solution points, the conservative variables are computed at the flux points

2. The inviscid fluxes at the interior flux points are computed using the solutions computed at Step (1)

3. The inviscid fluxes at the element interfaces are computed using the Rusanov solver. Given the normal direction of the interface n, and the averaged normal velocity component V_n and the sound speed c, the Rusanov flux on the interface can be computed.

4. The derivative of the fluxes are computed at the solution points using the derivatives of Lagrange operators l

$$\begin{aligned}
\left(\frac{\partial \tilde{F}}{\partial \xi}\right)_{i,j,k} &= \sum_{r=0}^{N} \tilde{F}_{r+1/2,j,k} \cdot l'_{r+1/2}(\xi_i), \\
\left(\frac{\partial \tilde{G}}{\partial \eta}\right)_{i,j,k} &= \sum_{r=0}^{N} \tilde{G}_{i,r+1/2,k} \cdot l'_{r+1/2}(\eta_j), \\
\left(\frac{\partial \tilde{H}}{\partial \beta}\right)_{i,j,k} &= \sum_{r=0}^{N} \tilde{H}_{i,j,r+1/2} \cdot l'_{r+1/2}(\beta_k)
\end{aligned} \tag{2.11}$$

Firstly, we write three-dimensional Euler equation in conservation form on Cartesian coordinates below

$$\frac{\partial Q}{\partial t} + \nabla F_e(Q) = 0 \tag{2.12}$$

where the conservative variables Q and Cartesian components $f_e(Q)$, $g_e(Q)$ and $h_e(Q)$ of the inviscid flux vector $F_e(Q)$ are given by

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{Bmatrix}, \quad f_e(Q) = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E+p) \end{Bmatrix} \tag{2.13}$$

$$g_e(Q) = \begin{Bmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(E+p) \end{Bmatrix}, \quad h_e(Q) = \begin{Bmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(E+p) \end{Bmatrix} \tag{2.14}$$

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

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2 + w^2) \tag{2.15}$$

with a constant ratio of specific heat γ. For all test cases in the present study, γ is going to be 1.4 for air.

Three-dimensional Navier-Stokes equations have an additional viscous term dependent on not only conservative variables but also their gradients.

$$\frac{\partial Q}{\partial t} + \nabla F_e(Q) - \nabla F_v(Q, \nabla Q) = 0 \tag{2.16}$$

The Cartesian components $f_v(Q, \nabla Q)$, $g_v(Q, \nabla Q)$ and $h_v(Q, \nabla Q)$ of viscous flux vector $F_v(Q, \nabla Q)$ are given by

$$f_v(Q, \nabla Q) = \begin{Bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \\ u\tau_{xx} + v\tau_{yx} + w\tau_{zx} + \frac{\mu C_p}{P_r} T_x \end{Bmatrix},$$

$$g_v(Q, \nabla Q) = \begin{Bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \\ u\tau_{xy} + v\tau_{yy} + w\tau_{zy} + \frac{\mu C_p}{P_r} T_y \end{Bmatrix},$$

$$h_v(Q, \nabla Q) = \begin{Bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{xz} + v\tau_{yz} + w\tau_{zz} + \frac{\mu C_p}{P_r} T_z \end{Bmatrix} \tag{2.17}$$

where μ is the dynamic viscosity, C_p is the specific heat and P_r stands for Prandtl number. T is temperature which can be derived from the perfect gas assumption. λ is set to $-2/3$

according to the Stokes hypothesis.

$$\tau_{xy} \equiv \tau_{yx} = \mu(v_x + u_y), \quad \tau_{xz} \equiv \tau_{zx} = \mu(w_x + u_z),$$
$$\tau_{yz} \equiv \tau_{zy} = \mu(w_y + v_z), \quad \tau_{xx} = 2\mu\left\{u_x - \tfrac{u_x+v_y+w_z}{3}\right\},$$
$$\tau_{yy} = 2\mu\left\{v_y - \frac{u_x + v_y + w_z}{3}\right\}, \quad \tau_{zz} = 2\mu\left\{w_z - \tfrac{u_x+v_y+w_z}{3}\right\}$$

The solution procedures to get viscous fluxes can be described as the following steps.

- reconstruct $Q_{k,i}$ at the flux points from the conservative variables at the solution points using equation 2.9.
- average the field of $Q_{k,i}$ on the element interfaces as $\overline{Q_f} = \frac{1}{2}(Q^L_{k,i}+Q^R_{k,i})$. Meanwhile, boundary conditions shall be applied for u, v, w and T.
- ∇u, ∇v, ∇w and ∇T can be evaluated from $\overline{Q_f}$ using equation (2.11) where $\nabla Q = \begin{Bmatrix} Q_x \\ Q_y \\ Q_z \end{Bmatrix}$ and $Q_x = \frac{\partial Q}{\partial \xi}\xi_x + \frac{\partial Q}{\partial \eta}\eta_x + \frac{\partial Q}{\partial \beta}\beta_x$, etc.
- reconstruct ∇u, ∇v, ∇w and ∇T from equation (2.9) and average them on the element interfaces as $\overline{\nabla Q_f} = \frac{1}{2}(\nabla Q^L_{k,i} + \nabla Q^R_{k,i})$
- with $\overline{Q_f}$ and $\overline{\nabla Q_f}$, we are ready to compute viscous flux vectors described in equation (2.17) at the element interfaces.

Flows with either steady or unsteady solutions are considered in this report. At present time, the time iterative schemes implemented in this solver consists of two Runge-Kutta type explicit schemes for the following system.

$$\frac{\partial Q}{\partial t} = R^c_i(Q) \tag{2.18}$$

A Jameson-style four-stage Runge-Kutta scheme is described as the following.

$$Q^{(1)}_i = Q^n_i - \alpha_1 \Delta t R_i(Q^n);$$
$$Q^{(2)}_i = Q^n_i - \alpha_2 \Delta t R_i(Q^1);$$
$$Q^{(3)}_i = Q^n_i - \alpha_3 \Delta t R_i(Q^2);$$
$$Q^{(n+1)}_i = Q^n_i - \alpha_3 \Delta t R_i(Q^3); \tag{2.19}$$

where $\alpha_l = 1/(5-l)$. Unlike the standard multi-stage Runge-Kutta scheme, the four-stage Runge-Kutta scheme is at best second-order accurate in time. Unless otherwise stated, the computations performed in this report employ this time advancement scheme.

Most of computations in this report use a slower but with four-order accuracy, i.e. a strong-stability-preserving five-stage Runge-Kutta scheme (Spiteri and Ruuth (2002)).

Chapter 3

3D inviscid flow result

Firstly, we compute 3D subsonic inviscid flow past a circular cylinder ($Mach = 0.2$). Two-dimensional flow past a circle has been considered by Bassi and Rebay (1997) and Wang and Liu (2006). The three dimensional grid is obtained after extruding a two-dimensional grid with 32×32 cells shown in figure 3 with three-cell thickness in the spanwise direction. Fixed-value Dirichlet boundary condition is used for the outer boundary of the grid. A quadratic curved wall boundary condition is implemented for the inner circle. In the spanwise direction, a periodic boundary condition is employed. The Mach number contour obtained by 4th order SD scheme for this potential flow problem is shown in figure 3. Nearly ideal symmetric patterns are successfully obtained.

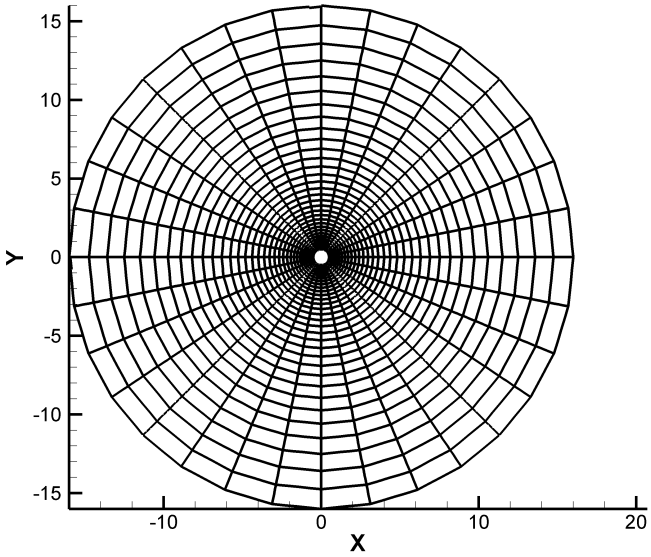


Figure 3.1: Grid $32 \times 32 \times 3$ for inviscid flow past a circular cylinder

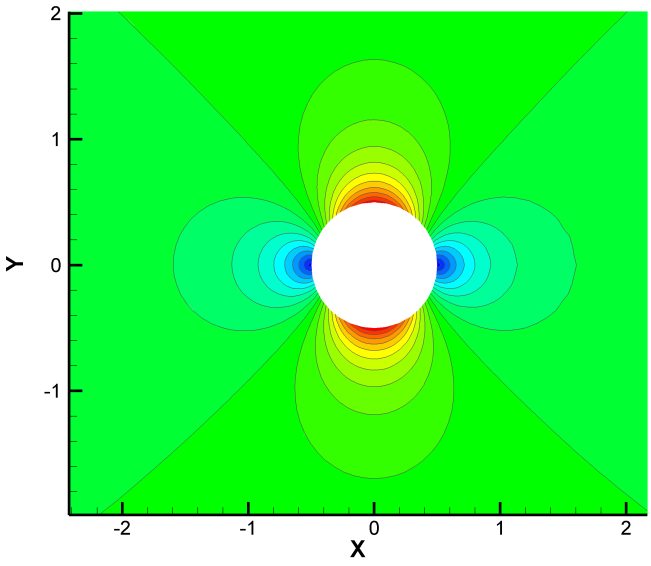


Figure 3.2: Mach number contour obtained for inviscid flow past a circular cylinder using 4th-order SD method and quadratic curved wall boundary

Chapter 4

Validation using 3D viscous compressible Couette flow

The numerical order of accuracy is validated using compressible Couette flow with analytical solution. A grid with $6 \times 3 \times 3$ cells is shown in figure 4.1 (a). A periodic boundary condition is used in the stream-wise direction. A moving wall no-slip boundary with constant temperature and a specified external pressure is used for the top surface ($y = H$). A stationary no-slip wall with constant temperature is used for the bottom surface $y = 0$. In the spanwise direction, periodic boundary condition is again implemented. The exact solutions of the this problem are

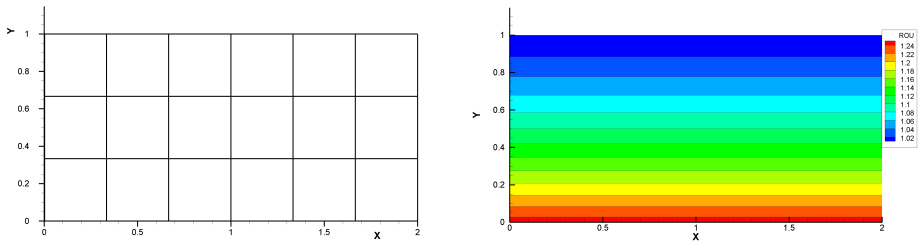
$$u = \frac{U}{H}y, \qquad v = 0, \qquad w = 0;$$
$$e = e_b + \frac{y}{H}(e_t - e_b) + \frac{PrU^2}{2\gamma}\frac{y}{H}(1 - \frac{y}{H}); \quad p = const$$

Where Pr is the Prandtl number and $e = C_v T$ is the internal energy. The density ρ is related to pressure p through $\rho = \frac{p}{(\gamma-1)e}$ The subscripts t and b refer to top and bottom surfaces respectively.

We obtained desired numerical order up to 4th order accuracy using $L2$ error as shown in table 4.1. It demonstrates that the basic implementations of the three-dimensional Spectral Difference method are successful. A typical solution of density ρ is shown in figure 4.1 (b), and it is a quadratic function of y.

No. of elements	No. of DOFs	L2-error	Order
2nd order SD			
2	16	1.25E-02	-
16	128	2.65E-03	2.24
54	432	9.62E-04	2.49
3rd order SD			
2	54	1.46E-03	-
16	432	1.18E-04	3.60
54	864	2.48E-05	3.85
4th order SD			
2	128	1.73E-04	-
16	1024	9.79E-06	4.14
54	3456	1.93E-06	4.01

Table 4.1: L^2 errors and orders of accuracy of viscous Couette flow



(a) Grid $6 \times 3 \times 3$ (b) density contour

Figure 4.1: Compressible Couette flow grid and its solution for density field.

Chapter 5

Concluding Remark

We have developed a three-dimensional spectral difference solver for inviscid and viscous flows from scratch. The 3D solver is successfully applied to compute subsonic inviscid flow past a circular cylinder. A symmetrical Mach number distribution is predicted accurately. We also verified that the solver on hexahedral grids could achieve the desired numerical order on 3D compressible Couette flow.

Chapter 6

Acknowledgement

Chunlei Liang and Antony Jameson would like to thank the grant support from NSF (award no. 0708071). Chunlei Liang also wants to thank Professor Z. J. Wang for discussing about the three-dimensional SD scheme and providing his three-dimensional code on SD method.

Bibliography

Bassi, F., Rebay, S. (1997) High-order accurate discontinuous finite element solution of the 2d Euler equations. Journal of Computational Physics 138: 251–285.

Liu, Y., Vinokur, M., Wang, Z. J. (2006*a*) Spectral difference method for unstructured grids i: Basic formulation. J. of Comput. Phys. 216: 780–801.

Liu, Y., Vinokur, M., Wang, Z. J. (2006*b*) Spectral (finite) volume method for conservation laws on unstructured grids v: Extension to three-dimensional systems. Journal of Computational Physics 212: 454–472.

Spiteri, R. J., Ruuth, S. J. (2002) A new class of optimal high-order strong-stability-preserving time discretization methods. SIAM J. Numer. Anal. 40: 469–491.

Sun, Y., Wang, Z. J., Liu, Y. (2007) High-order multidomain spectral difference method for the navier-stokes equations on unstructured hexahedral grids. Communication in Computational Physics 2: 310–333.

Wang, Z., Liu, Y., May, G., Jameson, A. (2007) Spectral difference method for unstructured grids II: Extension to the Euler equations. Journal of Scientific Computing 32: 45–71.

Wang, Z. J., Liu, Y. (2006) Extension of the spectral volume method to high-order boundary representation. Journal of Computational Physics 211: 154–178.