INL/CON-11-22526
PREPRINT

A Reconstructed Discontinuous Galerkin Method for the Compressible Flows on Unstructured Tetrahedral Grids

AIAA CFD Conference

Hong Luo
Yidong Xia
Robert Nourgaliev
Chunpei Cai

June 2011

The INL is a
U.S. Department of Energy
National Laboratory
operated by
Battelle Energy Alliance

A Reconstructed Discontinuous Galerkin Method for the Compressible Flows on Unstructured Tetrahedral Grids

Hong Luo[1] and Yidong Xia[2]
North Carolina State University, Raleigh, NC, 27695

Robert Nourgaliev[3],
Idaho National Laboratory, Idaho Falls, ID, 83415

Chunpei Cai[4]
New Mexico State University, Las Cruces, NM, 88001

A reconstruction-based discontinuous Galerkin (RDG) method is presented for the solution of the compressible Navier-Stokes equations on unstructured tetrahedral grids. The RDG method, originally developed for the compressible Euler equations, is extended to discretize viscous and heat fluxes in the Navier-Stokes equations using a so-called inter-cell reconstruction, where a smooth solution is locally reconstructed using a least-squares method from the underlying discontinuous DG solution. Similar to the recovery-based DG (rDG) methods, this reconstructed DG method eliminates the introduction of ad hoc penalty or coupling terms commonly found in traditional DG methods. Unlike rDG methods, this RDG method does not need to judiciously choose a proper form of a recovered polynomial, thus is simple, flexible, and robust, and can be used on unstructured grids. The preliminary results indicate that this RDG method is stable on unstructured tetrahedral grids, and provides a viable and attractive alternative for the discretization of the viscous and heat fluxes in the Navier-Stokes equations.

I. Introduction

The discontinuous Galerkin methods[1-31](DGM) have recently become popular for the solution of systems of conservation laws. Nowadays, they are widely used in computational fluid dynamics, computational acoustics, and computational magneto-hydrodynamics. The discontinuous Galerkin methods combine two advantageous features commonly associated to finite element and finite volume methods. As in classical finite element methods, accuracy is obtained by means of high-order polynomial approximation within an element rather than by wide stencils as in the case of finite volume methods. The physics of wave propagation is, however, accounted for by solving the Riemann problems that arise from the discontinuous representation of the solution at element interfaces. In this respect, the methods are therefore similar to finite volume methods. The discontinuous Galerkin methods have many attractive features:1) They have several useful mathematical properties with respect to conservation, stability, and convergence; 2) The method can be easily extended to higher-order (>2nd) approximation; 3) The methods are well suited for complex geometries since they can be applied on unstructured grids. In addition, the methods can also handle non-conforming elements, where the grids are allowed to have hanging nodes; 4) The methods are highly parallelizable, as they are compact and each element is independent. Since the elements are discontinuous, and the inter-element communications are minimal, domain decomposition can be efficiently employed. The compactness also allows for structured and simplified coding for the methods; 5) They can easily handle adaptive strategies, since refining or coarsening a grid can be achieved without considering the continuity restriction commonly associated with the conforming elements. The methods allow easy implementation of *hp*-refinement, for example, the order of accuracy, or shape, can vary from element to element; 6) They have the ability

[1]Associate Professor, Department of Mechanical and Aerospace Engineering, Senior Member AIAA.
[2]Ph.D. student, Department of Mechanical and Aerospace Engineering, Student Member AIAA.
[2]Senior Scientist, Multiphysics Methods Group, Member AIAA.
[4]Assistant Professor, Department of Mechanical and Aerospace Engineering, Senior Member AIAA.

to compute low Mach number flow problems without recourse to the time-preconditioning techniques normally required for the finite volume methods. However, DGM have a number of weaknesses that have yet to be addressed, before they can become a viable, attractive, competitive, and ultimately superior numerical method over more mature and well-established second order finite volume methods for flow problems of practical interest in a complex configuration environment. In particular, there are three most challenging and unresolved issues in the DGM: a) how to efficiently discretize diffusion terms required for the Navier-Stokes equations, b) how to effectively control spurious oscillations in the presence of strong discontinuities, and c) how to develop efficient time integration schemes for time accurate and steady-state solutions. Indeed, compared to the finite element methods and finite volume methods, the DG methods require solutions of systems of equations with more unknowns for the same grids. Consequently, these methods have been recognized as expensive in terms of both computational costs and storage requirements.

DG methods are indeed a natural choice for the solution of the hyperbolic equations, such as the compressible Euler equations. However, the DG formulation is far less certain and advantageous for the compressible Navier-Stokes equations, where viscous and heat fluxes exist. A severe difficulty raised by the application of the DG methods to the Navier-Stokes equations is the approximation of the numerical fluxes for the viscous fluxes, that has to properly resolve the discontinuities at the interfaces. Taking a simple arithmetic mean of the solution derivatives from the left and right is inconsistent, because the arithmetic mean of the solution derivatives does not take into account a possible jump of the solutions. A number of numerical methods have been proposed in the literature, such as those by Bassi and Rebay[21,22], Cockburn and Shu[23], Baumann and Oden[24], Peraire and Persson[25], and many others. Arnold et al.[26] have analyzed a large class of discontinuous Galerkin methods for second-order elliptic problems in a unified formulation. All these methods have introduced in some way the influence of the discontinuities in order to define correct and consistent diffusive fluxes. Lately, Gassner et al[27] introduced a numerical scheme based on the exact solution of the diffusive generalized Riemann problem for the discontinuous Galerkin methods. Liu et al[28], and Luo et al[29] used a BGK-based DG method to compute numerical fluxes at the interface for the Navier-Stokes equations, which has the ability to include both convection and dissipation effects. Unfortunately, all these methods seem to require substantially more computational effort than the classical continuous finite element methods, which are naturally suited for the discretization of elliptic problems. More recently, van Leer et al[30-32] proposed a recovery-based DG (rDG) method for the diffusion equation using the recovery principle, that recovers a smooth continuous solution that in the weak sense is indistinguishable from the discontinuous discrete solution. The rDG method is further developed by Nourgaliev et al[33] to solve the compressible Navier-Stokes equations on structured grids. The most attractive feature of rDG is the simplicity and generality of the recovery principle, which immediately allows one to create and further develop DG diffusion schemes. These schemes are of higher accuracy than any other DG diffusion schemes currently in use, while boasting the smallest eigenvalues, hence the largest stability range for explicit time-marching schemes, as shown by Huynh[34]. However, one significant weakness of rDG methods is a lack of flexibility. For instance in the case of DG(P1), due to the embedded 1-D interpolation problem, a cubic basis at an interface is required in the direction ζ connecting two centroids of the two cells adjacent to that interface. In 2D, two more basis (out of six degrees of freedom) is needed in the direction η normal to ζ. In 3D, four more basis (out of eight degrees of freedom) is required in two directions normal to ζ. This makes the choice of the recovery basis nontrivial. In addition, the basis of the recovered solution is twice as large as the basis of the underlying DG solution. This higher order representation of the recovered solution indeed yields higher order accuracy for the diffusion equations on regular grids, and however is unnecessary for the Navier-Stokes equations, as the overall order of accuracy is determined not only by diffusive fluxes but also advective fluxes. The fundamental issue is how to judiciously choose a proper form of a recovered polynomial in such a way that the resulting recovered linear system is well conditioned, and thus can be inverted. Clearly, this lack of flexibility makes rDG methods less appealing for the multidimensional problems. Fortunately, recovery is not the only way to obtain a locally smooth polynomial solution at the interface from the underlying discontinuous Galerkin solutions. Rather, reconstruction widely used in the finite volume methods provides an alternative, probably a better choice to obtain a higher-order polynomial representation.

Dumbser et al[18-20] have introduced a new family of in-cell recovery DG methods, termed PnPm schemes, where Pn indicates that a piecewise polynomial of degree of n is used to represent a DG solution, and Pm represents a reconstructed polynomial solution of degree of m ($m \geq n$) that is used to compute the fluxes. The PnPm schemes are designed to enhance the accuracy of the discontinuous Galerkin method by increasing the order of the underlying polynomial solution. The beauty of PnPm schemes is that they provide a unified formulation for both finite volume and DG methods, and contain both classical finite volume and standard DG methods as two special cases of PnPm schemes, and thus allow for a direct efficiency comparison. When n=0, i.e. a piecewise constant polynomial is used to represent a numerical solution, P0Pm is nothing but classical high order finite volume schemes, where a

polynomial solution of degree m (m ≥1) is reconstructed from a piecewise constant solution. When m=n, the reconstruction reduces to the identity operator, and PnPn scheme yields a standard DG method. Obviously, the construction of an accurate and efficient reconstruction operator is crucial to the success of the PnPm schemes. This is achieved using a so-called in-cell recovery similar to the inter-cell recovery originally proposed by Van Leer et al.[30-32], where recovered equations are obtained using a L_2 projection, i.e., the recovered polynomial solution is uniquely determined by making it indistinguishable from the underlying DG solutions in the contributing cells in the weak sense. Dumbser[20] has applied this PnPm method to solve the compressible Navier-Stokes equations on unstructured grids, where the discretization of the viscous fluxes is based on the exact solution of the diffusive generalized Riemann problem developed by Gassner et al[27].

A reconstructed discontinuous Galerkin method, originally introduced for the compressible Euler equations using an in-cell reconstruction[18,35], has been developed for the solution of the two dimensional compressible Navier-Stokes equations using an inter-cell reconstruction on arbitrary grids[36]. The objective of the effort discussed in this paper is to extend this reconstructed discontinuous Galerkin method for the solution of the compressible Navier-Stokes equations on unstructured tetrahedral grids. Starting from an underlying discontinuous DG solution, a smooth solution is locally reconstructed on the union of two cells adjacent to an interface to compute the viscous and heat fluxes in the Navier-Stokes equations at the interface. The reconstructed continuous solution is required to be conservative and to match the point values of the underlying DG solution and its derivatives at the two neighboring cells. The resultant over-determined system is then solved using a least-squares method. Similar to the recovery-based DG (rDG) methods, this reconstructed DG method automatically generates penalty or coupling terms found in traditional DG methods, and thus is stable, and compact. Unlike rDG methods, this RDG does not need to judiciously choose a proper form of a recovered polynomial, thus is simple, flexible, and robust, and can be implemented on arbitrary grids. The preliminary results indicates that the RDG method is stable, accurate, and efficient, demonstrating its potential to become a viable, attractive, competitive, and perhaps superior DG method over existing DG methods for the solution of the compressible Navier-Stokes equations on unstructured tetrahedral grids. The remainder of this paper is structured as follows. The governing equations are listed in Section 2. The underlying reconstructed discontinuous Galerkin method is presented in Section 3. Numerical experiments are reported in Section 4. Concluding remarks are given in Section 5.

II. Governing Equations

The Navier-Stokes equations governing unsteady compressible viscous flows can be expressed as

$$\frac{\partial \mathbf{U}(x,t)}{\partial t} + \frac{\partial \mathbf{F}_k(\mathbf{U}(x,t))}{\partial x_k} = \frac{\partial \mathbf{G}_k(\mathbf{U}(x,t))}{\partial x_k} \tag{2.1}$$

where the summation convention has been used. The conservative variable vector **U**, advective (inviscid) flux vector **F**, and viscous flux vector **G** are defined by

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ \rho e \end{pmatrix} \qquad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p\delta_{ij} \\ u_j(\rho e + p) \end{pmatrix} \qquad \mathbf{G}_j = \begin{pmatrix} 0 \\ \sigma_{ij} \\ u_l \sigma_{lj} + q_j \end{pmatrix} \tag{2.2}$$

Here ρ, p, and e denote the density, pressure, and specific total energy of the fluid, respectively, and u_i is the velocity of the flow in the coordinate direction x_i. The pressure can be computed from the equation of state

$$p = (\gamma - 1)\rho(e - \frac{1}{2}u_j u_j) \tag{2.3}$$

which is valid for perfect gas, where γ is the ratio of the specific heats. The components of the viscous stress tensor σ_{ij} and the heat flux vector are given by

$$\sigma_{ij} = \mu(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) - \frac{2}{3}\mu\frac{\partial u_k}{\partial x_k}\delta_{ij} \qquad q_j = \frac{1}{\gamma - 1}\frac{\mu}{\mathrm{Pr}}\frac{\partial T}{\partial x_j} \tag{2.4}$$

In the above equations, T is the temperature of the fluid, Pr the laminar Prandtl number, which is taken as 0.7 for air. μ represents the molecular viscosity, which can be determined through Sutherland's law

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{\frac{3}{2}} \frac{T_0 + S}{T + S} \tag{2.5}$$

μ_0 denotes the viscosity at the reference temperature T_0, and S is a constant which for are assumes the value S = 110°K. The temperature of the fluid T is determined by

$$T=\gamma \frac{p}{\rho} \tag{2.6}$$

Neglecting viscous effects, the left-hand side of Eq. (2.1) represents the Euler equations governing unsteady compressible inviscid flows.

III. Reconstructed Discontinuous Galerkin Method

The governing equation (2.1) is discretized using a discontinuous Galerkin finite element formulation. To formulate the discontinuous Galerkin method, we first introduce the following weak formulation, which is obtained by multiplying the above conservation law by a test function W, integrating over the domain Ω, and then performing an integration by parts,

$$\int_{\Omega} \frac{\partial \mathbf{U}}{\partial t} \mathbf{W} d\Omega + \int_{\Gamma} \mathbf{F}_k \mathbf{n}_k d\Gamma - \int_{\Omega} \mathbf{F}_k \frac{\partial \mathbf{W}}{\partial x_k} d\Omega = \int_{\Gamma} \mathbf{G}_k \mathbf{n}_k d\Gamma - \int_{\Omega} \mathbf{G}_k \frac{\partial \mathbf{W}}{\partial x_k} d\Omega, \qquad \forall \mathbf{W} \in V \tag{3.1}$$

where $\Gamma(=\partial\Omega)$ denotes the boundary of Ω, and $\mathbf{n}_j$ the unit outward normal vector to the boundary. We assume that the domain Ω is subdivided into a collection of non-overlapping tetrahedral elements Ω_e. We introduce the following broken Sobolev space V_h^p

$$V_h^p = \left\{ v_h \in \left[L_2(\Omega)\right]^m : v_h \big|_{\Omega_e} \in \left[V_p^m\right] \forall \Omega_e \in \Omega \right\}, \tag{3.2}$$

which consists of discontinuous vector-values polynomial functions of degree p, and where m is the dimension of the unknown vector and

$$V_p^m = \operatorname{span}\left\{ \prod x_i^{\alpha_i} : 0 \le \alpha_i \le p, 0 \le i \le d \right\}, \tag{3.3}$$

where α denotes a multi-index and d is the dimension of space. Then, we can obtain the following semi-discrete form by applying weak formulation on each element Ω_e

$$\begin{aligned} &\text{Find} \quad \mathbf{U}_h \in V_h^p \quad \text{such as} \\ &\frac{d}{dt} \int_{\Omega_e} \mathbf{U}_h \mathbf{W}_h d\Omega + \int_{\Gamma_e} \mathbf{F}_k(\mathbf{U}_h) \mathbf{n}_k \mathbf{W}_h d\Gamma - \int_{\Omega_e} \mathbf{F}_k(\mathbf{U}_h) \frac{\partial \mathbf{W}_h}{\partial x_k} d\Omega = \\ &\int_{\Gamma_e} \mathbf{G}_k(\mathbf{U}_h) \mathbf{n}_k \mathbf{W}_h d\Gamma - \int_{\Omega_e} \mathbf{G}_k(\mathbf{U}_h) \frac{\partial \mathbf{W}_h}{\partial x_k} d\Omega, \quad \forall \mathbf{W}_h \in V_h^p \end{aligned} \tag{3.4}$$

where $\mathbf{U}_h$ and $\mathbf{W}_h$ represent the finite element approximations to the analytical solution **U** and the test function **W** respectively, and they are approximated by a piecewise polynomial function of degrees p, which are discontinuous between the cell interfaces. Assume that B is the basis of polynomial function of degrees p, this is then equivalent to the following system of N equations,

$$\begin{aligned} &\frac{d}{dt} \int_{\Omega_e} \mathbf{U}_h B_i d\Omega + \int_{\Gamma_e} \mathbf{F}_k(\mathbf{U}_h) \mathbf{n}_k B_i d\Gamma - \int_{\Omega_e} \mathbf{F}_k(\mathbf{U}_h) \frac{\partial B_i}{\partial x_k} d\Omega = \\ &\int_{\Gamma_e} \mathbf{G}_k(\mathbf{U}_h) \mathbf{n}_k B_i d\Gamma - \int_{\Omega_e} \mathbf{G}_k(\mathbf{U}_h) \frac{\partial B_i}{\partial x_k} d\Omega, \quad 1 \le i \le \mathrm{N} \end{aligned} \tag{3.5}$$

where N is the dimension of the polynomial space. Since the numerical solution $\mathbf{U}_h$ is discontinuous between element interfaces, the interface fluxes are not uniquely defined. The choice of these fluxes is crucial for the DG formulation. Like in the finite volume methods, the inviscid flux function $\mathbf{F}_k(\mathbf{U}_h)\mathbf{n}_k$ appearing in the boundary integral can be replaced by a numerical Riemann flux function $\mathbf{H}_k(\mathbf{U}^L_h, \mathbf{U}^R_h, \mathbf{n}_k)$ where $\mathbf{U}_h^L$ and $\mathbf{U}_h^R$ are the conservative state vector at the left and right side of the element boundary. The computation of the viscous fluxes in the boundary integral has to properly resolve the discontinuities at the interfaces. This scheme is called discontinuous Galerkin method of degree p, or in short notation DG(P) method. Note that discontinuous Galerkin formulations are very similar to finite volume schemes, especially in their use of numerical fluxes. Indeed, the classical first-order cell-centered finite volume scheme exactly corresponds to the DG(P_0) method, i.e., to the discontinuous Galerkin method using a piecewise constant polynomial. Consequently, the DG(P_k) methods with k>0 can be regarded as a natural generalization of finite volume methods to higher order methods. By simply increasing the degree P of the polynomials, the DG methods of corresponding higher order can be obtained.

The domain and boundary integrals in Eq. (3.5) are calculated using Gauss quadrature formulas. The number of quadrature points used is chosen to integrate exactly polynomials of order of $2p$ and $2p+1$ in the reference element for the domain and surface integrals, respectively. For the DG(P_1) method considered in this study, three points are used in the boundary integrals and the domain integrals are evaluated using four points.

In the traditional DGM, numerical polynomial solutions $\mathbf{U}_h$ in each element are expressed using either standard Lagrange finite element or hierarchical node-based basis as following

$$\mathbf{U}_h = \sum_{i=1}^{N} \mathbf{U}_i(t) B_i(x), \tag{3.6}$$

where B_i are the finite element basis functions. As a result, the unknowns to be solved are the variables at the nodes $\mathbf{U}_i$, However, numerical polynomial solutions $\mathbf{U}$ can be expressed in other forms as well. In the present work, the numerical polynomial solutions are represented using a Taylor series expansion at the center of the cell. For example, if we do a Taylor series expansion at the cell centroid, a linear polynomial solution can be expressed as follows

$$\mathbf{U}_h = \mathbf{U}_c + \frac{\partial \mathbf{U}}{\partial x}\Big|_c (x - x_c) + \frac{\partial \mathbf{U}}{\partial y}\Big|_c (y - y_c) + \frac{\partial \mathbf{U}}{\partial z}\Big|_c (z - z_c) \tag{3.7}$$

where $\mathbf{U}_c$ is the mean value of $\mathbf{U}$ in this cell. The unknowns to be solved in this formulation are the cell-averaged variables and their derivatives at the center of the cells regardless of element shapes. In this case, the dimension of the polynomial space is four and the four basis functions are

$$B_1 = 1 \qquad B_2 = x - x_c \qquad B_3 = y - y_c \qquad B_4 = z - z_c \tag{3.8}$$

The discontinuous Galerkin formulation then leads to the following four equations

$$\begin{gathered} \frac{d}{dt}\int_{\Omega_e} \tilde{\mathbf{U}} d\Omega + \int_{\Gamma_e} \mathbf{F}_k(\mathbf{U}_h)\mathbf{n}_k d\Gamma = \int_{\Gamma_e} \mathbf{G}_k(\mathbf{U}_h)\mathbf{n}_k d\Gamma, \quad i = 1 \\ M_{3x3} \frac{d}{dt}\left(\frac{\partial \mathbf{U}}{\partial x}\Big|_c \quad \frac{\partial \mathbf{U}}{\partial y}\Big|_c \quad \frac{\partial \mathbf{U}}{\partial z}\Big|_c \right)^T + R_{3x1} = 0 \end{gathered} \tag{3.9}$$

where, the element of the mass matrix $m_{ij} = \int_{\Omega_e} B_{i+1} B_{j+1} d\Omega$. Note that in this formulation, equations for the cell-averaged variables are decoupled from equations for their derivatives due to the judicial choice of the basis functions and the fact that

$$\int_{\Omega_e} B_1 B_i d\Omega = 0, \qquad 2 \le i \le 4 \tag{3.10}$$

This set of basis functions has a considerable disadvantage, especially when higher-order polynomial solutions are considered. Occasionally an inverse of the mass matrix M may not exist. In the implementation of this DG method, the basis functions are actually normalized in order to improve the conditioning of the system matrix (3.5) as follows:

$$B_1 = 1 \qquad B_2 = \frac{x - x_c}{\Delta x} \qquad B_3 = \frac{y - y_c}{\Delta y} \qquad B_4 = \frac{z - z_c}{\Delta z} \tag{3.11}$$

here $\Delta x=0.5(x_{max}-x_{min})$, $\Delta y=0.5(y_{max}-y_{min})$, and $\Delta z=0.5(z_{max}-z_{min})$, and x_{ma} ,x_{min}, y_{max}, z_{min} y_{max}, and z_{min} are the maximum and minimum coordinates in the cell Ω_e in x-, y-, and z-directions, respectively. A linear polynomial solution can then be rewritten as

$$\mathbf{U}_h = \widetilde{\mathbf{U}} + \frac{\partial \mathbf{U}}{\partial x}\Big|_c \Delta x B_2 + \frac{\partial \mathbf{U}}{\partial y}\Big|_c \Delta y B_3 + \frac{\partial \mathbf{U}}{\partial z}\Big|_c \Delta z B_4 \tag{3.12}$$

The above normalization is especially important to alleviate the stiffness of the system matrix for higher-order DG approximations.

This Taylor-basis DG method has a number of attractive features. Theoretically, this formulation allows us to clearly see the similarity and difference between DG and FV methods. In fact, the discretized governing equations for the cell-averaged variables and the assumption of polynomial solutions on each cell are exactly the same for both finite volume and DG methods. The only difference between them is the way how they obtain high-order (>1) polynomial solutions. In the finite volume methods, the polynomial solution of degrees *p* are reconstructed using the mean values of the neighboring cells, which can be obtained using either TVD/MUSCL or ENO/WENO reconstruction schemes. Unlike the FV methods, the DG methods compute the derivatives in a manner similar to the mean variables. This is compact, rigorous, and elegant mathematically in contrast with arbitrariness characterizing the reconstruction schemes with respect how to compute the derivatives and how to choose the stencils in the FV methods. Furthermore, the higher order DG methods can be easily constructed by simply increasing the degree *p* of the polynomials locally, in contrast to the finite volume methods which use the extended stencils to achieve higher order of accuracy. In addition, the Taylor-basis DG method makes the implementation of both in-cell and inter-cell reconstruction schemes straightforward and simple[35].

The discretization of the Navier-Stokes equations requires the evaluation of the viscous fluxes at a cell interface, which has to properly resolve the discontinuities at the interfaces. Taking a simple arithmetic mean of the viscous fluxes from the left and right cells is inconsistent, because the arithmetic mean of the solution derivatives does not take into account a possible jump of the solutions. In the reconstructed DG method, a continuous solution $\mathbf{U}^R$ is locally reconstructed on the union of two cells $\Omega_{ij}(=\Omega_i \cup \Omega_j)$ adjacent to the interface based on the underlying discontinuous Galerkin solution in the two abutting elements. This reconstructed smooth solution is then used to compute the viscous fluxes at the interface. Without lose of generality, let us consider the case of DG(P1) method, where the reconstructed solution $\mathbf{U}^R$, similar to the underlying DG solution on Ω_i, can be expressed in Ω_{ij} using a Taylor basis as follows:

$$\mathbf{U}^R = \widetilde{\mathbf{U}}_{ij} + \mathbf{U}_x^{ij} B_2(\mathbf{x}) + \mathbf{U}_y^{ij} B_3(\mathbf{x}) + \mathbf{U}_z^{ij} B_4(\mathbf{x}) \tag{3.13}$$

where $\tilde{\mathbf{U}}_{ij}$ is the mean value of $\mathbf{U}^R$ on Ω_{ij}, and the derivatives are the point-wise value at the center of Ω_{ij}. There are four degrees of freedom, and therefore four unknowns to be determined. However, the cell-average value $\tilde{\mathbf{U}}_{ij}$ can be trivially obtained, by requiring the reconstruction scheme to be conservative, a fundamental requirement. Due to the judicious choice of Taylor basis in our DG formulation, this leads to

$$\widetilde{\mathbf{U}}_{ij} = \frac{\widetilde{\mathbf{U}}_i \Omega_i + \widetilde{\mathbf{U}}_j \Omega_j}{\Omega_i + \Omega_j} \tag{3.15}$$

The remaining three degrees of freedom can be determined by imposing that the reconstructed solution and its derivatives are equal to the underlying DG solution and its derivatives at cells i and j. Consider cell i, one obtains

$$\begin{aligned}
\mathbf{U}_i &= \widetilde{\mathbf{U}}_{ij} + \mathbf{U}_x^{ij} B_{2i} + \mathbf{U}_y^{ij} B_{3i} + \mathbf{U}_z^{ij} B_{4i} \\
\frac{\partial \mathbf{U}}{\partial x}\Big|_i \Delta x_i &= \mathbf{U}_x^{ij} \frac{\partial B_{2i}}{\partial x} \Delta x_i \\
\frac{\partial \mathbf{U}}{\partial y}\Big|_i \Delta y_i &= \mathbf{U}_y^{ij} \frac{\partial B_{3i}}{\partial y} \Delta y_i \\
\frac{\partial \mathbf{U}}{\partial z}\Big|_i \Delta z_i &= \mathbf{U}_z^{ij} \frac{\partial B_{4i}}{\partial z} \Delta z_i
\end{aligned} \tag{3.16}$$

where the basis function B is evaluated at the center of cell i, i.e., $B_i=B(x_j,y_j)$. This can be written in a matrix form as follows:

$$\begin{pmatrix} B_{2i} & B_{3i} & B_{4i} \\ \dfrac{\partial B_{2i}}{\partial x}\Delta x_i & 0 & 0 \\ 0 & \dfrac{\partial B_{3i}}{\partial y}\Delta y_i & 0 \\ 0 & 0 & \dfrac{\partial B_{4i}}{\partial z}\Delta z_i \end{pmatrix} \begin{pmatrix} \mathbf{U}_x^{ij} \\ \mathbf{U}_y^{ij} \\ \mathbf{U}_z^{ij} \end{pmatrix} = \begin{pmatrix} \mathbf{U}_i - \widetilde{\mathbf{U}}_{ij} \\ \dfrac{\partial \mathbf{U}}{\partial x}\Big|_i \Delta x_i \\ \dfrac{\partial \mathbf{U}}{\partial y}\Big|_i \Delta y_i \\ \dfrac{\partial \mathbf{U}}{\partial z}\Big|_i \Delta z_i \end{pmatrix} \tag{3.17}$$

Similar equations can be derived for the cell j. As a result, a non-square matrix of 8x3 is obtained. This over-determined linear system of 8 equations for 3 unknowns can then be solved in the least-squares sense. In the present work, it is solved using a normal equation approach, which, by pre-multiplying through by matrix transpose, yields a symmetric linear system of equations 3x3. This linear system of equations can be then trivially solved to obtain the three first derivatives of the reconstructed continuous linear polynomial solution. This reconstructed smooth linear polynomial solution is then used to compute the viscous and heat fluxes in the Navier-Stokes equations at the interfaces. Similar to the recovered DG methods, the inter-cell reconstruction is compact, as it only involves two cells adjacent to the interfaces. Unlike the recovery-based DG methods, the reconstructed DG method only reconstructs a smooth polynomial solution of the same order as the underlying DG solution, thus there is no need to judiciously choose a proper form of a recovered polynomial and ensure that the recovered system is well conditioned and can be inverted. The resulting DG method for the discretization of the viscous and heat fluxes is termed a reconstructed DG method (RDG(P1) in short notation).

The inter-cell reconstruction described above can be applied to the boundary cells in a straightforward manner with the aid of a so-called ghost cell approach, where the computational domain is extended to a set of ghost cells. The values of the flow variables in the ghost cells are determined by the desired boundary conditions. The compactness of the DG methods makes the imposition of boundary conditions simple and easy, which only requires one layer of ghost cells, regardless of the order of the DG methods.

It is worth to note that the application of this inter-cell reconstruction to DG(P0) method where the first derivative is approximated using a second-order central differencing method demonstrates that this reconstruction DG method automatically provides the coupling terms required for the stability and leads to a 5-point second-order scheme for the diffusive operator (second derivative) in 1D on a uniform grid, contrary to most of discretization methods that lead to a 3-point stencil second-order method. This analysis indicates the potential of this reconstruction method for the accurate and robust discretization of the viscous fluxes on highly non-uniform, highly stretched, and highly distorted grids, as it is practically impossible to obtain a second-order accurate and compact cell-centered finite volume method for multi-dimensional problems on such grids.

This reconstructed DG method has been implemented in a well-tested 3D DG code[13-17,29]. In this code, a fast, low-storage *p*-multigrid method[16,17] is developed to obtain steady state solutions, and an explicit three-stage third-order TVD Runge-Kutta scheme is used to advance solution in time for the unsteady flow problems. Many upwind schemes have been implemented for the discretization of the inviscid fluxes, although HLLC scheme is exclusively used for the approximate solution of the Riemann problem in this work. Our numerical results in 2D have shown that the computing costs required by this RDG method is about half of the computing costs by BR2 scheme for the discreitization of the viscous fluxes in the Navier-Stokes equations. This is mainly due the fact that the BR2 scheme requires to solve a linear system. The formation of this linear system involves the boundary integrals using Gauss quadrature formulas, and thus is computationally intensive. The RDG method only involves the solution of a least-squares problem at an inter-cell interface. The resulting reconstruction matrix can be pre-computed, inverted, and stored at the start of the computation, thus leading to a very efficient discretization method for the viscous fluxes. Note that this is achieved at the expense of an increasing storage requirement. In the present work, the reconstruction matrix is not pre-computed and stored in order to avoid the significant increase in memory requirements.

IV. Numerical Examples

A. Blasius Boundary Layer

A laminar boundary layer over an adiabatic flat plate at a free-stream Mach number of 0.2 and a Reynolds number of 100,000 based on the freestream velocity and the length of the flat plate is considered in this test case, where the computational domain is bounded from -0.5 to 1 in the x-direction, 0 to 0.5 in the y-direction, and 0 to 0.1 in the z-direction, and the flat plate starts at point (0,0) and extends to (1,0). This problem is chosen to illustrate that RDG method is able to obtain highly accurate solution for the Navier-Stokes equations, as the Blasius solution can be used to measure accuracy of the numerical solutions. The grid used in this test case has 106,819 elements, 20,856 grid points, and 5,824 boundary points, as shown in Figure 1, where the computed velocity contours in the flow field are also displayed. Figure 2 compares the logarithmic plot of the computed skin friction coefficient obtained by RDG solutions with the Blasius solution.

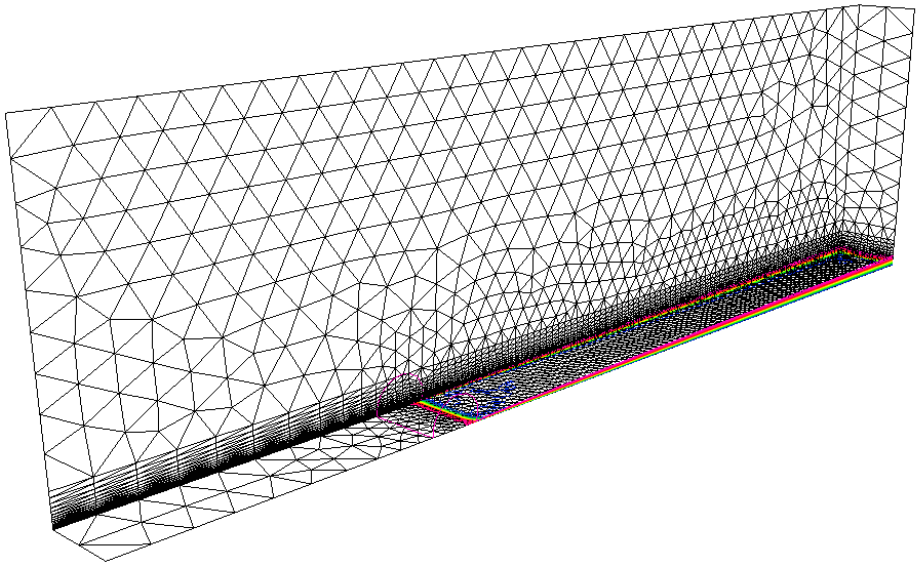

Figure 1. Unstructured mesh (nelem=106,819, npoin=20,856, nboun=5,824) and computed velocity contours by the RDG(P1)2) for subsonic flow past a flat plate at M_∞= 0.2, Re=100,000, $\alpha=0^o$.

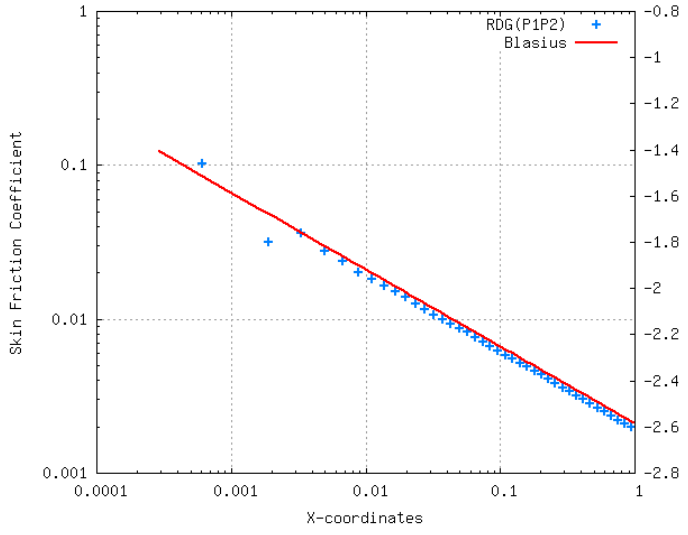

Figure 2. Logarithmic plot of the computed skin friction coefficient distribution along the center line of the flat plate obtained by the RDG(P1P2) solutions.

B. Subsonic viscous flow past a Sphere

This test case involves a subsonic viscous flow past a sphere at a Mach number of 0.3, and an angle of attack 0^o, and a Reynolds number of 118 based on the freestream velocity and the diameter of the sphere, which has been studied both experimentally[37] and numerically[38]. An adiabatic wall is assumed in this test case. Note that only a half of the configuration is modeled, due the symmetry of the problem. Figure 3 shows the computational grid used in this test case, consisting of 119,390 tetrahedral elements, 22,530 grid points, and 4,511 boundary points, and the computed velocity contours in the flow field obtained by the RDG method. The computational streamlines are compared with experimental streamlines in Figure 4, where steady separation bubble is readily observed in both plots and the size of the seperation region in the computation agrees very well with that of the experiment.

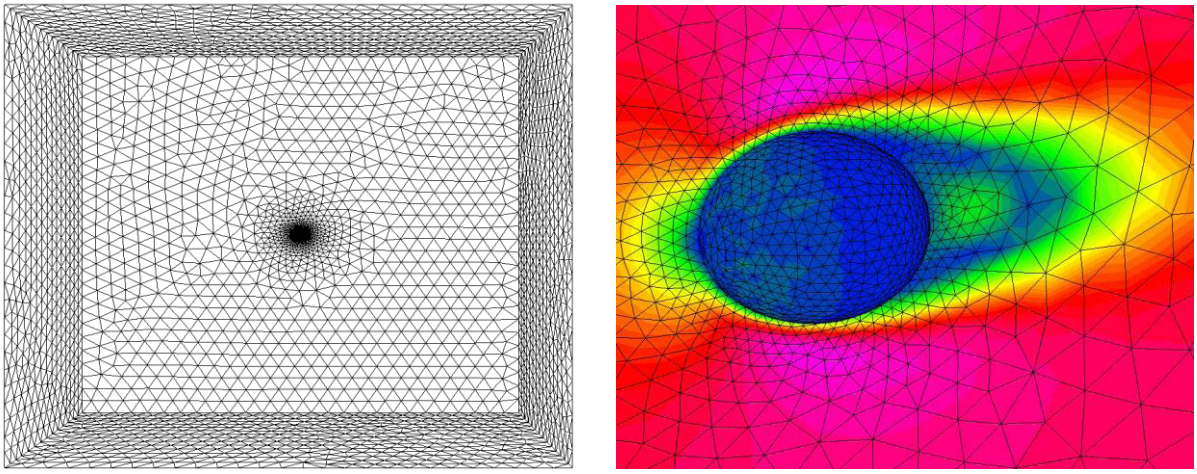

Figure 3. Unstructured mesh (left) (nelem=119,390, npoin=22,530, nboun=4,511) and computed Mach number contours by the RDG(P1P2) (right) for subsonic flow past a sphere at M_∞= 0.3, Re=118, $\alpha=0^o$.

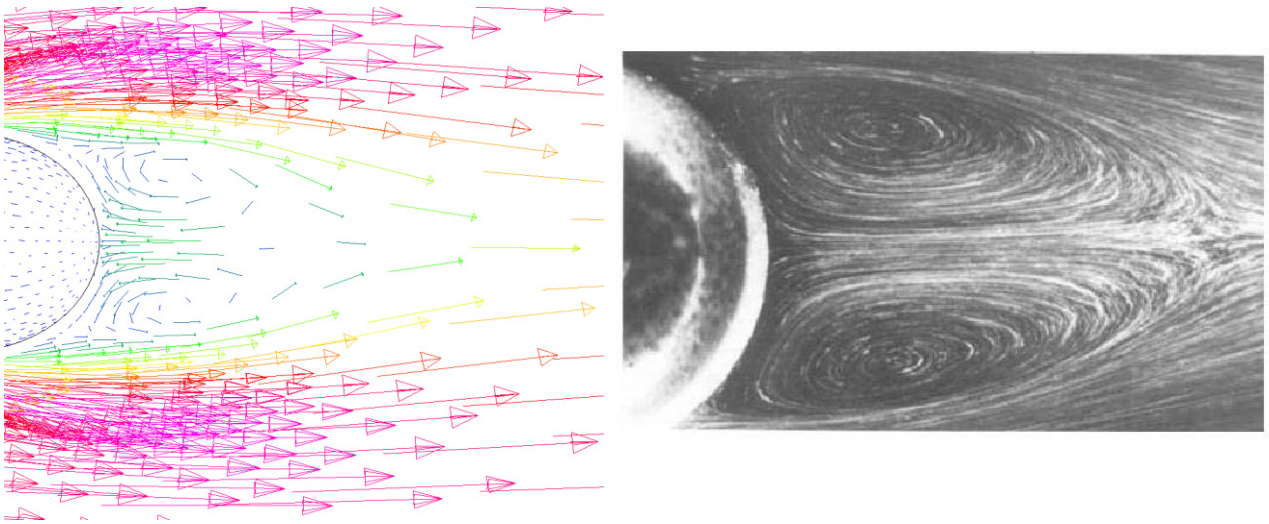

Figure 4. Velocity vector plot in the back of the sphere from the computation (left) and streamlines of the flow filed from experiment (right) for subsonic flow past a sphere at M_∞= 0.3, Re=118, $\alpha=0^o$.

C. Laminar Flow in a Square Duct with 90^o Bend

A 3D laminar flow through a duct with 90^o bend is computed in this example. The cross section of the bend is square throughout the bend. The Reynolds number based on the inlet mean velocity and the entrance width H is 790. The mean radius of the bend is 2.3. The inlet and outlet boundaries are located at 5H up- and downstream from the bend. The grid used in the computation is shown in Figure 5. It contains 230,181 elements, 43,964 grid points, and 10,040 boundary points. The computation was performed at a Mach number of 0.2. Figure 6 shows the computed streamwise velocity contours at several streamwise cross sections.

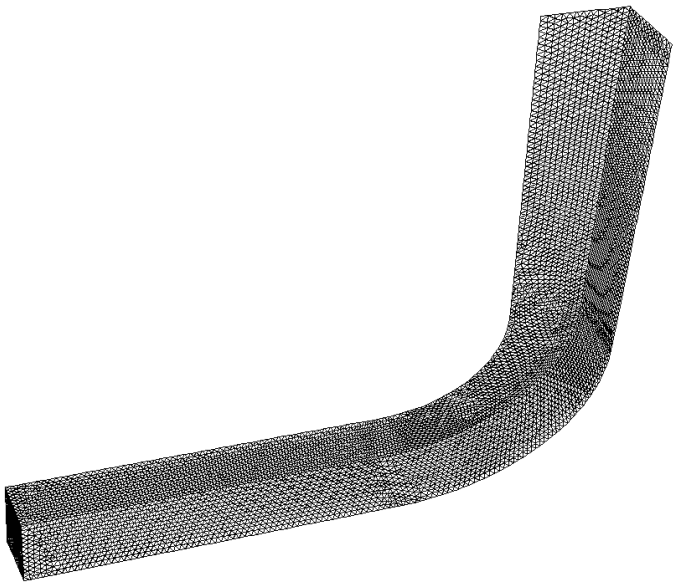

Figure 5. Unstructured mesh used for computing a laminar flow through a square duct with a 90^o bend((nelem=230,181, npoin=43,964, nboun=10,040).

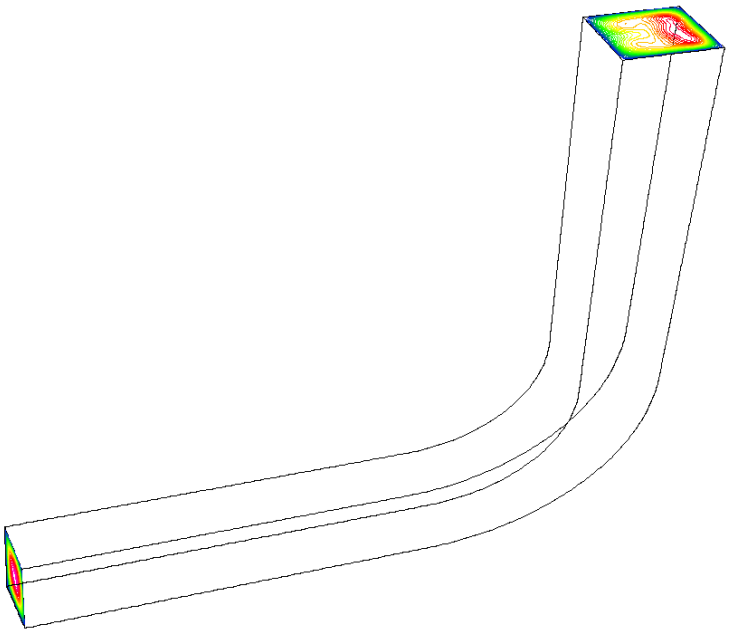

Figure 6. Computed velocity contours for square duct with a 90^o bend at several streamwise sections at M_∞= 0.2, Re=790.

V. Conclusion

A reconstructed discontinuous Galerkin method has been developed for the solution of the compressible Navier-Stokes equations on unstructured tetrahedral grids. A smooth solution is locally reconstructed to discretize viscous and heat fluxes in the Navier-Stokes equations using a least-squares method from the underlying discontinuous discrete solution. The preliminary results are promising, indicating its potential to provide a viable, attractive, competitive, and perhaps superior DG method over existing DG methods for the discretization of the viscous and heat fluxes in the compressible Navier-Stokes equations.

Acknowledgments

This manuscript has been authored by Battelle Energy Alliance, LLC under contract No. DE-AC07-05ID14517 (INL/CON-10-17570) with the U.S. Department of Energy. The United States Government retains and the published, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The authors would like to acknowledge the partial support for this work provided by DOE under Nuclear Engineering University Program. The first and last authors would also like to acknowledge the partial support for this work provided by the NSF under project No. NSF-DMS0914706.

References

[1]Reed, W.H. Reed and T.R. Hill, ``Triangular Mesh Methods for the Neutron Transport Equation," **Los Alamos Scientific Laboratory Report**, LA-UR-73-479, 1973.

[2]B. Cockburn, S. Hou, and C. W. Shu, ``TVD Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for conservation laws IV: the Multidimensional Case," **Mathematics of Computation**, Vol. 55, pp. 545-581, 1990.eorge, P. L., Automatic Mesh Generation, J. Wiley & Sons, 1991.

[3]B. Cockburn, and C. W. Shu, ``The Runge-Kutta Discontinuous Galerkin Method for conservation laws V: Multidimensional System," **Journal of Computational Physics**, Vol. 141, pp. 199-224, 1998.

[4]B. Cockburn, G. Karniadakis, and C. W. Shu, ``The Development of Discontinuous Galerkin Method", in Discontinuous Galerkin Methods, Theory, Computation, and Applications, edited by B. Cockburn, G.E. Karniadakis, and C. W. Shu, Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, Vol. 11 pp. 5-50, 2000.

[5]F. Bassi and S. Rebay, ``High-Order Accurate Discontinuous Finite Element Solution of the 2D Euler Equations," **Journal of Computational Physics**, Vol. 138, pp. 251-285, 1997.

[6]H. L. Atkins and C. W. Shu, ``Quadrature Free Implementation of Discontinuous Galerkin Method for Hyperbolic Equations," **AIAA Journal**, Vol. 36, No. 5, 1998.

[7]F. Bassi and S. Rebay, ``GMRES discontinuous Galerkin solution of the Compressible Navier-Stokes Equations," **Discontinuous Galerkin Methods, Theory, Computation, and Applications**, edited by B. Cockburn, G.E. Karniadakis, and C. W. Shu, Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, Vol. 11 pp. 197-208, 2000.

[8]T. C. Warburton, and G. E. Karniadakis, ``A Discontinuous Galerkin Method for the Viscous MHD Equations," **Journal of Computational Physics**, Vol. 152, pp. 608-641, 1999.

[9]J. S. Hesthaven and T. Warburton, ``Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications," Texts in **Applied Mathematics**, Vol. 56, 2008.

[10]P. Rasetarinera and M. Y. Hussaini, ``An Efficient Implicit Discontinuous Spectral Galerkin Method," **Journal of Computational Physics**, Vol. 172, pp. 718-738, 2001.

[11]B. T. Helenbrook, D. Mavriplis, and H. L. Atkins, ``Analysis of *p*-Multigrid for Continuous and Discontinuous Finite Element Discretizations," AIAA Paper 2003-3989, 2003.

[12]K. J. Fidkowski, T. A. Oliver, J. Lu, and D. L. Darmofal, ``*p*-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations," **Journal of Computational Physics**, Vol. 207, No. 1, pp. 92-113, 2005.

[13]H. Luo, J. D. Baum, and R. Löhner, ``A Discontinuous Galerkin Method Using Taylor Basis for Compressible Flows on Arbitrary Grids," **Journal of Computational Physics**, DOI: 210.1016/j.jcp.2008.06.035, Vol. 227, No 20, pp. 8875-8893, October 2008.

[14]H. Luo, J.D. Baum, and R. Löhner, ``On the Computation of Steady-State Compressible Flows Using a Discontinuous Galerkin Method", **International Journal for Numerical Methods in Engineering,** Vol. 73, No. 5, pp. 597-623, 2008.

[15]H. Luo, J. D. Baum, and R. Löhner, ``A Hermite WENO-based Limiter for Discontinuous Galerkin Method on Unstructured Grids," **Journal of Computational Physics**, Vol. 225, No. 1, pp. 686-713, 2007.

[16]H. Luo, J.D. Baum, and R. Löhner, ``A *p*-Multigrid Discontinuous Galerkin Method for the Euler Equations on Unstructured Grids", **Journal of Computational Physics**, Vol. 211, No. 2, pp. 767-783, 2006.

[17]H. Luo, J.D. Baum, and R. Löhner, ``A Fast, *p*-Multigrid Discontinuous Galerkin Method for Compressible Flows at All Speeds", **AIAA Journal** , Vol. 46, No. 3, pp.635-652, 2008.

[18]M. Dumbser, D.S. Balsara, E.F. Toro, C.D. Munz. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. **Journal of Computational Physics**, 227:8209-8253, 2008.

[19]M. Dumbser, O. Zanotti. Very high order PNPM schemes on unstructured meshes for the resistive relativistic MHD equations. **Journal of Computational Physics**, 228:6991-7006, 2009.

[20]M. Dumbser. Arbitrary High Order PNPM Schemes on Unstructured Meshes for the Compressible Navier-Stokes Equations. **Computers & Fluids**, 39: 60-76. 2010.

[21]F. Bassi and S. Rebay, ``A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations," **Journal of Computational Physics**, Vol. 131, pp. 267-279, 1997.

[22]F. Bassi and S. Rebay, ``Discontinuous Galerkin Solution of the Reynolds-Averaged Navier-Stokes and k-ω Turbulence Model Equations, **Journal of Computational Physics**, Vol. 34, pp. 507-540, 2005.

[23]B. Cockburn and C.W. Shu, ``The Local Discontinuous Galerkin Method for Time-dependent Convection-Diffusion System, **SIAM, Journal of Numerical Analysis**, Vo. 16, 2001.

[24]C. E. Baumann and J. T. Oden, ``A Discontinuous *hp* Finite Element Method for the Euler and Navier-Stokes Equations," **International Journal for Numerical Methods in Fluids**, Vol. 31, 1999.

[25]J. Peraire and P. O. Persson, The Compact Discontinuous Galerkin Method for Elliptic Problems. **SIAM Journal on Scientific Computing**, 30: 1806-1824, 2008.

[26]D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini, ``Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems", **SIAM Journal on Numerical Analysis**. Vol. 39, No. 5., pp. 1749-1779, 2002.

[27]G. Gassner, F. Lorcher, and C. D. Munz, ``A Contribution to the Construction of Diffusion Fluxes for Finite Volume and Discontinuous Galerkin Schemes", **Journal of Computational Physics**, Vol. 224, No. 2, pp. 1049-1063, 2007.

[28]H. Liu and K. Xu, ``A Runge-Kutta Discontinuous Galerkin Method for Viscous Flow Equations", **Journal of Computational Physics**, Vol. 224, No. 2, pp. 1223-1242, 2007.

[29]H. Luo, L. Luo and K. Xu - A Discontinuous Galerkin Method Based on a BGK Scheme for the Navier-Stokes Equations on Arbitrary Grids, **Advances in Applied Mathematics and Mechanics**, Vol. 1, No. 3, pp. 301-318, 2009.

[30]B. van Leer and S. Nomura, ``Discontinuous Galerkin Method for Diffusion," AIAA Paper 2005-5108,2005.

[31]B. van Leer and M. Lo, ``A Discontinuous Galerkin Method for Diffusion Based on Recovery," AIAA Paper 2007-4083,2007.

[32]M. Raalte and B. van Leer, ``Bilinear Forms for the Recovery-Based Discontinuous Galerkin Method for Diffusion," Communication of Computational Physics, Vol. 5, No. 2-4, pp. 683-693, 2009.

[33]R. Nourgaliev, H. Park, and V. Mousseau, ``Recovery Discontinuous Galerkin Jacobian-Free Newton-Krylov Method for Multiphysics Problems," To be appeared in Computational Fluid Dynamics Review 2009, 2009.

[34]H.T Huynh, 'A Reconstruction Approach to High-order Schemes Including Discontinuous Galerkin for Diffusion," AIAA Paper 2009-0403, 2009.

[35]H. Luo, L. Luo, R. Nourgaliev, and V. Mousseau, A Reconstructed Discontinuous Galerkin Method for the Compressible Euler Equations on Arbitrary Grids, AIAA-2009-3788, 2009.

[36]H. Luo, L. Luo, R. Nourgaliev, and V. Mousseau, A Reconstructed Discontinuous Galerkin Method for the Compressible Navier-STokes Equations on Arbitrary Grids, **Journal of Computational Physics,** Vol. 229, pp. 6961-6978, doi: 10.1016/j.jcp 2010.05.033, 2010.

[37]S. Taneda, ``Experimental Investigations of the wake behind a sphere at low Reynolds numbers, J. Phys. Soc. Jpn., 11 (1956), pp 1104-1108.

[38]Y. Sun, Z.J. Wang, and Y. Liu, ``High-Order Multidomain Spectral Difference Method for the Navier-Stokes Equations on Unstructured Hexahedral Grids," **Communications in Computational Physics**, Vol. 2, No., 2, pp. 310,333, 2007.