47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition
5 - 8 January 2009, Orlando, Florida

AIAA 2009-601

Progress in High-Order Discontinuous Galerkin Methods for Aerospace Applications

Dimitri Mavriplis *
Cristian Nastase † Khosro Shahbazi ‡
Li Wang §
Nicholas Burgess ¶
Department of Mechanical Engineering
University of Wyoming, Laramie, Wyoming, 82072-3295

This paper covers progress in developing and applying high-order accurate discretizations based on the Discontinuous Galerkin approach to aerodynamic applications. The paper concentrates on various specific areas including discretization of diffusion terms, shock capturing, and multigrid solution methods. The development of a fully conservative Arbitrary Lagrangian Eulerian formulation which obeys the Geometric Conservation Law while maintaining the design order of accuracy of the static mesh discretization is also presented. Additionally, adjoint-based sensitivity analysis for design optimization as well as error estimation and adaptive control is demonstrated. Finally, prospects for future advancements and efficiency gains over traditional finite volume methods are discussed in a concluding section.

I. Introduction

While numerical simulation capabilities for engineering analysis and design have been constantly advancing over the last 50 years, there is a perception that over the last decade these capabilities have been maturing, particularly in the areas of high-fidelity multi-physics problems of relevance to the aerospace industry. Most recently, engineering simulation capabilities have advanced principally as a result of improvements in computer hardware capabilities, and the increased problem sizes (grid resolution, temporal resolution, additional physics) these enable, as opposed to fundamental advances in algorithmic capabilities. However, it is generally acknowledged that discretization errors constitute one of the dominant error sources in most current-day simulations, and adequate resolution of these error sources often remains largely intractable due to the wide range of spatial and temporal scales which must be resolved in most complex engineering simulations. As an example, the AIAA sponsored Drag Prediction Workshop series has identified spatial discretization error as a dominant error source for steady-state aerodynamic drag calculations, in spite of an order of magnitude growth in the sizes of the grids employed in the workshop series over the last decade.[1–4] Furthermore, the generation of ever finer meshes, with some calculations using from 10^8 to 10^9 grid points,

*Professor, email: mavripl@uwyo.edu
†Post Doctoral Researcher, email: nastase@uwyo.edu
‡Post-doctoral Researcher, Current address: Division of Applied Mathematics, Brown University, email: shahbazi@brown.edu
§Ph.D. Candidate, email: wangli@uwyo.edu
¶Ph.D. Candidate, email: nburges2@uwyo.edu

is taxing the capability of current mesh generation tools, while at the same time, the drive to massively parallel multi-core computer architectures (with the advent of large installations comprising in excess of 100,000 cores) is slowing the adoption of larger problem sizes, due to scalability issues.

The difficulties in reducing discretization errors through extensive grid refinement with current technology can be seen by examining the asymptotic properties of second-order accurate discretizations. For such schemes, doubling the mesh resolution in each direction in three dimensions results in an eight-fold increase in computational work, but only delivers a four-fold reduction in error. While adaptive meshing techniques can be used to enhance the accuracy of a low-order scheme, these methods can only yield a constant factor of improvement over the non-adaptive case, and once the optimal mesh resolution has been achieved, further increases in accuracy can only be obtained through global refinement with the same asymptotic properties as those discussed above.

On the other hand, the use of high-order methods offers the possibility of changing the asymptotic relation between error and resolution. As illustrated in Figure 1, the more rapid decrease in discretization error with mesh resolution for a fourth-order method compared to a second-order method ensures that the higher-order method will ultimately become the method of choice as the accuracy requirements are made more stringent. Furthermore, additional increases in accuracy and efficiency can be obtained through the use of adaptive techniques applied to high order methods, and these have been shown to deliver optimal (exponential) error reduction when used in the context of h-p refinement techniques.[5–7] Thus, the use of high-order discretizations with h-p adaptive techniques offers the possibility of delivering revolutionary improvements in simulation fidelity. These advantages are compounded by the high degree of scalability on massively parallel computers achieved by high-order methods with compact stencils, such as the Discontinuous Galerkin approach.[8–11]

On the other hand, high-order methods have not made significant in-roads for complex engineering simulation problems. This may be partly attributed to the difficulties associated with treating complex geometries, issues involved in resolving discontinuities, robustness of the methods, and efficiency of the solvers for steady-state and time-implicit problems. However, recent developments in high-order methods have brought these techniques to the level where we believe they may be applied effectively to complex engineering problems in the near future. While early high-order spectral methods were only applicable to simple configurations, the growing popularity of spectral element[7] and more recently discontinuous Galerkin[11,12] and spectral volume and difference methods[13,14] has shown how high-order methods can be applied to complex configurations. However, many of these methods were originally conceived for hyperbolic problems, and their extension to diffusion problems has proved to be non-trivial. For solutions involving discontinuities such as shock waves or slip lines, both limiting procedures as well as artificial dissipation approaches have been developed to enable smooth capturing of discontinuous profiles, although it remains unclear how robust and efficient these different approaches may be for complex geometries. While the initial development of high-order methods has concentrated on the solution of wave-like phenomena where spatial and temporal scales are closely related, and explicit time-stepping schemes remain the method of choice,[9–11] for many engineering applications implicit time-stepping schemes are required, which in turn require the efficient solution of complex non-linear problems at each time step. Recent advances in this area have been demonstrated, for example using h-p multigrid schemes for steady-state,[15–17] and time-implicit problems.[18,19] The formulation of high-order methods in arbitrary Lagrangian-Eulerian (ALE) form has also received relatively little attention in the literature[20–22] although this will become an important consideration for important applications such as aeroelasticity. Finally, the use of adaptive methods, for low or high order methods, has traditionally been hampered by the availability of reliable error estimation criteria to drive the adaptive process. Recent developments in goal-oriented error estimation procedures using adjoint techniques have shown promise in this area[23–28] although the practical application of these methods to high-order discretizations using combined h-p adaptation on complex three dimensional geometries remains relatively unexplored.

In this paper, we outline some of the advances made in these various areas in our research group, which over time, are contributing to the emergence of a practical high-order DG methodology applicable to industrial aerospace problems. This paper provides a broad overview of various of these techniques, and the

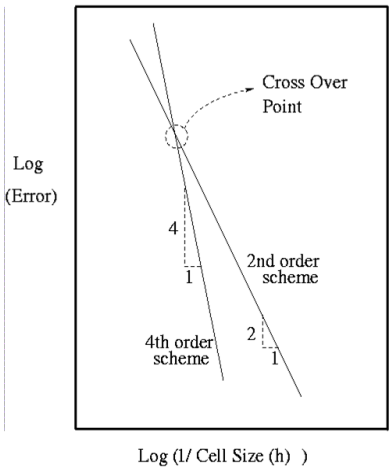


Figure 1. Illustration of asymptotic behavior of numerical error for a second-order and fourth-order accurate scheme as a function of grid resolution. The higher-order scheme may incur more error on coarse grids (and be more computationally expensive) but eventually surpasses the low order scheme as the mesh resolution is increased. The location of this cross over point determines which scheme is most suitable for the desired accuracy level.

reader is referred to the literature for a more in-depth description of each particular approach.

II. Discretization

A. Euler equations

Discontinuous Galerkin discretizations were originally devised for hyperbolic equations[8–11] and their application to the Euler equations is now relatively straight forward. The principal choices to be made include the selection of a set of basis functions which spans the polynomial space required for high-order accuracy, the specification of appropriate quadrature rules for numerical integration, and the selection of a numerical flux function to be used for resolving the discontinuous solution at inter-element boundaries.

The conservative form of the compressible Euler equations describing the conservation of mass, momentum and total energy are given in vectorial form

$$\frac{\partial \mathbf{U}(\mathbf{x},t)}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = 0 \tag{1}$$

subject to appropriate boundary and initial conditions within a three-dimensional domain Ω. Explicitly, the state vector $\mathbf{U}$ of the conservative variables and the Cartesian components of the inviscid flux $\mathbf{F} = (\mathbf{F}^x, \mathbf{F}^y, \mathbf{F}^z)$ are given as:

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E_t \end{pmatrix}, \quad \mathbf{F}^x = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(E_t + p) \end{pmatrix}, \quad \mathbf{F}^y = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ \rho vw \\ v(E_t + p) \end{pmatrix}, \quad \mathbf{F}^z = \begin{pmatrix} \rho w \\ \rho uw \\ \rho vw \\ \rho w^2 + p \\ w(E_t + p) \end{pmatrix}, \tag{2}$$

where ρ is the fluid density, (u, v, w) are the fluid velocity Cartesian components, p is the pressure and E_t is the total energy. For an ideal gas, the equation of state relates the pressure to total energy by:

$$p = (\gamma - 1)\left[E_t - \frac{1}{2}\rho(u^2 + v^2 + w^2)\right] \tag{3}$$

where $\gamma = 1.4$ is the ratio of specific heats.

The weak formulation for Eq. (1) is obtained by minimizing the residual with respect to the expansion function in an integral sense:

$$\int_{\Omega_k} \phi_i \left[\frac{\partial \mathbf{U}_p(\mathbf{x},t)}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}_p)\right]_k d\Omega_k = 0 \tag{4}$$

After integrating by parts the weak statement of the problem becomes:

$$\int_{\Omega_k} \phi_i \frac{\partial \mathbf{U}_p}{\partial t}\, d\Omega_k - \int_{\Omega_k} \nabla\phi_i \cdot \mathbf{F}(\mathbf{U}_p)\, d\Omega_k + \int_{\partial\Omega_k} \phi_i \mathbf{F}^*(\mathbf{U}_p)\cdot \mathbf{n}\, d(\partial\Omega_k) = 0 \tag{5}$$

The computational domain Ω is partitioned into an ensemble of non-overlapping elements and within each element the solution is approximated by a truncated polynomial expansion

$$\mathbf{U}(\mathbf{x},t) \approx \mathbf{U}_p(\mathbf{x},t) = \sum_{j=1}^{M} \mathbf{u}_j(t)\phi_j(\mathbf{x}) \tag{6}$$

where M is the number of modes defining the truncation level. This semi-discrete formulation (*i.e.* continuous in time) employs a discontinuous Galerkin formulation in spatial variables within each element Ω_k.

The local discontinuous Galerkin approach makes use of element-based basis functions, which results in solution approximations which are local, discontinuous, and doubled valued on each elemental interface. Monotone numerical fluxes are used to resolve the discontinuity, providing the means of communication between adjacent elements and specification of the boundary conditions. The numerical flux, $\mathbf{F}^*(\mathbf{U}_p)\cdot\mathbf{n}$, is obtained as a solution of a local one-dimensional Riemann problem and depends on the internal interface state, $\mathbf{U}_p^+$, the adjacent element interface state, $\mathbf{U}_p^-$ and the orientation as defined by the normal vector, $\mathbf{n}$, of the interface. An approximate Riemann solver is used to compute the flux at inter-element boundaries. Current implementations include the flux difference splitting schemes of Rusanov,[29] Roe,[30] HLL[31] and HLLC.[32–34]

The discrete form of the local discontinuous Galerkin formulation is defined by the particular choice of the set of basis functions, $\{\phi_i, i = 1\ldots M\}$. The basis set is defined on the master element $\widehat{\Omega}(\xi_j, j = 1\ldots 3)$ spanning between $\{-1 < \xi_j < 1\}$. We seek a set of hierarchical basis functions in order to simplify our subsequent spectral multigrid implementation. The basis set contains *vertex*, *edge* and *bubble* functions[6,35] based on Jacobi polynomials of variable weights. Since the basis set is defined in the master element, a coordinate transformation, $\mathbf{x}_p = \mathbf{x}_p(\xi_1,\xi_2,\xi_3)$, is required to compute the derivatives and the integrals in physical space $\Omega_k(x,y,z)$. For iso-parametric elements, the basis functions are expressed as functions of ξ_1, ξ_2 and ξ_3, and the coordinate transformation, and its Jacobian are given by:

$$\mathbf{x}_p = \sum_{j=1}^{M} \widehat{\mathbf{x}}_j \phi_j(\xi_1,\xi_2,\xi_3), \qquad J_k(\xi_1,\xi_2,\xi_3) = \left|\frac{\partial(x,y,z)}{\partial(\xi_1,\xi_2,\xi_3)}\right| \tag{7}$$

In the simple case of straight-sided or -faced elements the mapping is linear and its Jacobian, J_k, and its metrics are constant within each element, and can be evaluated just by using the element vertex coordinates. In the case of elements with curved faces, the Jacobian J_k varies within the element, and must be evaluated explicitly at the individual quadrature points in the numerical integration process. While straight-sided/faced elements are used in the interior of the domain, curved boundary elements which conform to the original description of the boundary geometry must be used in order to retain the $p+1$ accuracy property of the discretization scheme. Since most mesh generation packages produce a list of elements, with coordinates

given only for the corner points of the elements, additional information is required in order to create curved boundary elements. This is achieved by projecting additionally created surface points onto the original boundary geometry description, as depicted in Figure 2 for the two-dimensional case.

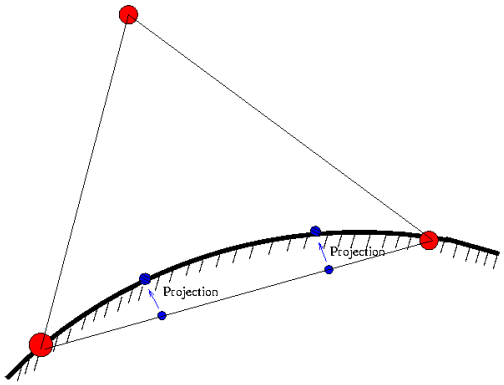


Figure 2. Two-dimensional illustration of projection of additional surface points for creation of curved surface element.

The number of additional surface points to be created on each face is a function of the p-order of the discretization, and generally super-parametric elements of order p+1 are used. The correspondence between the coordinates of the new projected surface points in physical space and in isoparametric space is then used to compute the exact Jacobian for the curved elements.[36]

For the general case (*i.e.*curved elements), using Eq. (7), the solution expansion and the weak statement within each element, $\widehat{\Omega}_k$, becomes:

$$\mathbf{U}_p(\xi,\eta,t) = \sum_{j=1}^{M} \widehat{\mathbf{u}}_j(t)\phi_j(\xi_1,\xi_2,\xi_3) \tag{8}$$

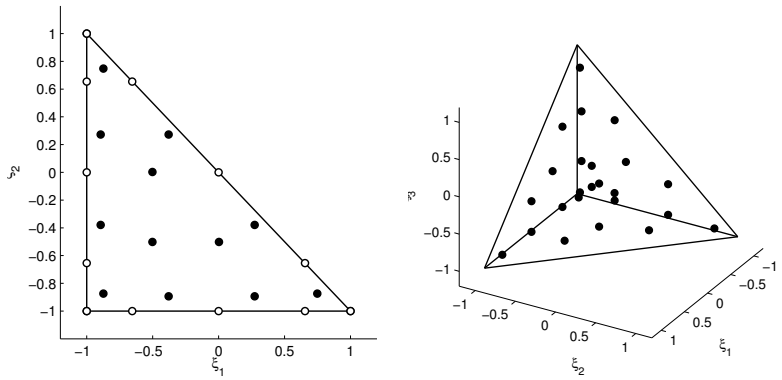
$$\int_{\widehat{\Omega}_k} \phi_i \frac{\partial \mathbf{U}_p}{\partial t} |J_k| d\widehat{\Omega}_k - \int_{\widehat{\Omega}_k} \nabla\phi_i J_k^{-1} \cdot \mathbf{F}(\mathbf{U}_p) |J_k| d\widehat{\Omega}_k + \int_{\partial\widehat{\Omega}_k} \phi_i \mathbf{F}^*(\mathbf{U}_p) \cdot \mathbf{n} |J_k| (\partial\widehat{\Omega}_k) = 0 \tag{9}$$

The resulting semi-discrete form, Eq. (9), can be further simplified as:

$$\mathbf{M}\frac{d\mathbf{U}_p}{dt} + \mathbf{R}(\mathbf{U}_p) = 0 \tag{10}$$

where $\mathbf{M}$ and $\mathbf{R}(\mathbf{U}_p)$ represent the mass matrix and the non-linear residual of the spatial discretization terms, respectively. This system of ordinary equations, Eq. (10), is solved in the modal space and the integrals are evaluated by economical Gaussian quadrature rules,[6,37,38] which requires a projection of the solution values to the quadrature points used in the numerical integration. In order to preserve the $p+1$ accuracy order of the numerical approximation, the element integral uses quadrature rules which are exact for polynomial degree $2p$ within the master element, while the boundary integral uses quadrature rules which are exact for polynomial degree $2p+1$.[9] For boundary elements with curved edges or faces, the Jacobians must be evaluated at the integration quadrature points, whereas for interior elements with straight edges or faces, these are constant and need only be evaluated once for each element. Figure 3 illustrates the set of quadrature points used on triangular faces and for tetrahedral elements for a p=3 discretization.

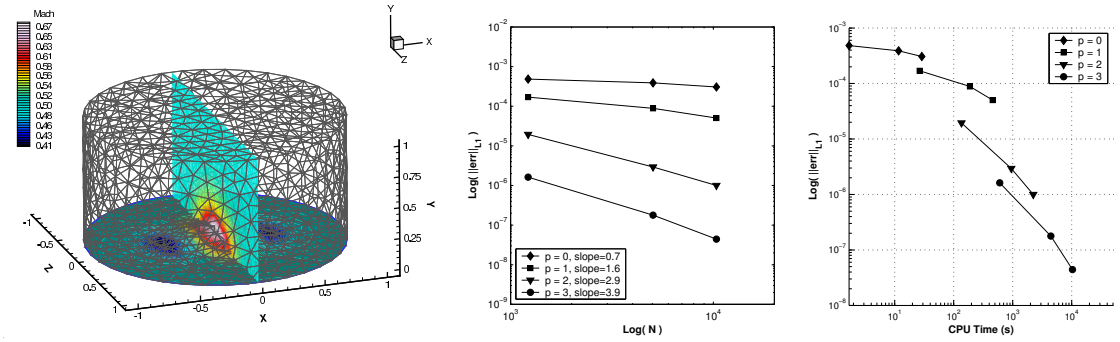
Figure 4(a) illustrates the inviscid flow computed over a three-dimensional bump, which was used to assess the accuracy properties of the DG discretization described above. In Figure 4(b), the error in the solution, as measured by the RMS level of entropy in the flow field is seen to decrease asymptotically at rates close to the design order for discretizations ranging from p=0 to p=3. Perhaps more importantly,



(a) (b)

Figure 3. Quadrature points for p=3 discretization on (a) standard triangle; (b) standard tetrahedron.

Figure 4(c) illustrates the fact that the higher-order discretizations achieve lower error tolerances for similar computational effort, and thus are most suitable for cases where high accuracy is desired.



(a) (b) (c)

Figure 4. (a) Mach contours for DG simulation of flow over a three-dimensional bump at p=4; (b) Discretization error as a function of mesh resolution for various orders of accuracy; (c) Discretization error as a function of CPU time for various orders of accuracy.

B. Navier-Stokes equations

The discretization of diffusive operators such as those found in the viscous terms of the Navier-Stokes equations is less straight-forward for DG discretizations. Over the years, a number of approaches have been proposed, with various degrees of success, and a useful comparison of various approaches can be found in Arnold et al.[39] Our criteria for the choice of a discretization approach for diffusive operators include stability and robustness, lack of tunable parameters, and the retention of the nearest neighbor stencil which forms the basis of the inviscid discretization.

The Local Discontinuous Galerkin (LDG) approach developed by Cockburn and Shu[40] has proved to

be successful for the simulation of viscous flows, although this approach requires the use of a non-nearest neighbor stencil, which has adverse implications for memory and parallelization requirements. More recently, Peraire and Persson[41] have devised a compact stencil LDG variant which retains the desirable nearest neighbor stencil of the purely convective DG discretization. The second method of Bassi and Rebay (BR2)[42] constitutes an alternative which relies on a nearest neighbor stencil, and has been used extensively for the simulation of viscous flows.[15] The recovery approach developed by Van-Leer et al.[43,44] achieves high accuracy with a nearest neighbor stencil, although these methods are still in their infancy, and extensions to three-dimensional problems remain to be demonstrated.

The interior penalty (IP) method constitutes one of the earliest approaches for DG discretizations of diffusive operators.[45] Traditionally, this method has been shunned due to the requirement of specifying a penalty parameter for stability, which has most often been done in an ad-hoc manner. However, recently an explicit expression for the penalty parameter in the original IP method has been obtained.[46] This expression, which is based on local grid metrics and approximation order, can be thought of as a simplified version of the lifting operator in the BR2 discretization. With the ad-hoc nature of the penalty parameter prescription removed, we favor the IP method over other approaches mainly due to the simplicity of the IP discretization, which also retains all the favorable properties of the best of the other approaches. In Figure 5, the accuracy of the IP method[45,46]for a pure diffusion problem (Poisson's equation on a square with a sinusoidal exact solution) is compared with that obtained using the BR2 discretization.[42] The results are shown for different grid sizes and approximation orders $p = 1, \cdots, 5$. As is clear from the figure, almost identical accuracy is achieved for both IP and BR2 methods.

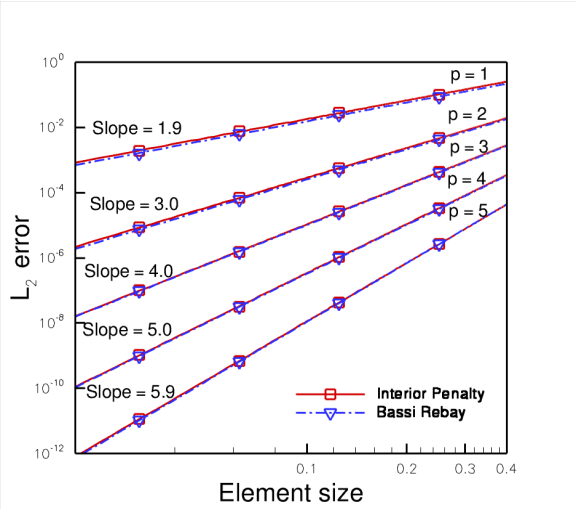
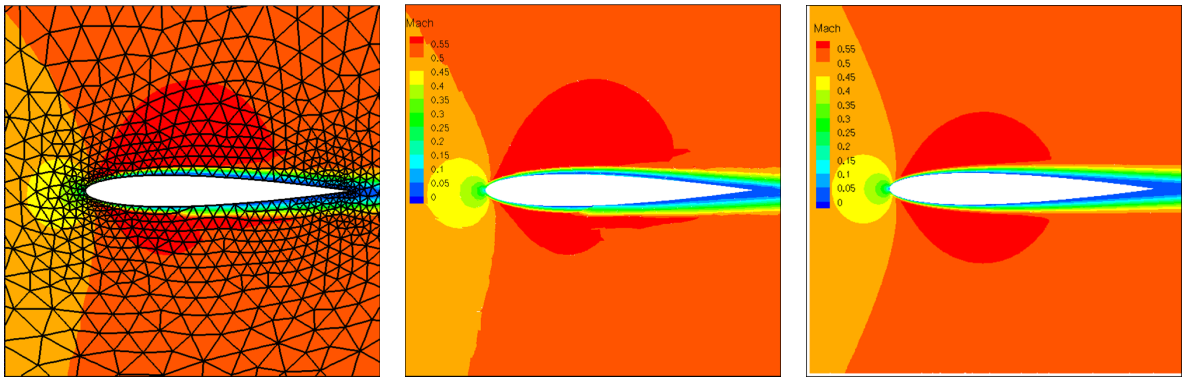


Figure 5. Comparison of performance of the IP and BR2 methods in solving a Poisson equation on a square domain with a sinusoidal exact solution. The L_2 norm of error is plotted vs. element size for $p = 1, \cdots, 5$. For each data set, the best linear fit and its slope are also shown.

Figure 6 depicts the viscous laminar flow computed over a NACA0012 airfoil using the IP method with explicit penalty expression[45,46] using a relatively coarse and non-symmetric grid of 1822 elements. The improvement in accuracy obtained when raising the order of accuracy of the discretization from second (p=1) to sixth (p=5) is clearly evident in these results, and the flow separation point of 81% chord reported by the p=5 solution agrees well with very fine grid finite-volume solutions previously reported.[47]



(a) (b) (c)

Figure 6. (a) Triangular grid containing 1822 element and (b) second-order accurate solution for viscous flow over NACA 0012 airfoil and (c) Sixth-order accurate solution for viscous flow over NACA 0012 airfoil. Mach = 0.5, Incidence = 0 degrees, Reynolds number = 5000

.

III. Solution Methodologies

Because initial applications of DG discretizations often involved wave phenomena, such as acoustics or electromagnetics, the most prevalent approach for integrating these discretizations in time has been through the use of explicit methods, which are well suited for problems where spatial and temporal scales are closely related.[9–11]

For aerospace engineering problems, efficient steady-state and time-implicit solvers are required in order to provide a competitive simulation capability. Ultimately, our interest is in devising optimal solvers, which provide convergence rates which are independent of the order of accuracy of the discretization (p), and the size of the mesh (h), while retaining good parallel efficiency characteristics and low cost per iteration. Considering the non-linear problem

$$\mathbf{R}(\mathbf{U}_p) = 0 \tag{11}$$

where $\mathbf{R}(\mathbf{U}_p)$ now represents the non-linear steady-state residual, or the non-linear problem to be solved at each implicit time step, our approach to devising efficient solution techniques can be best described by first writing out the corresponding Newton scheme given as:

$$\begin{aligned} \left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}_p}\right]^n \Delta \mathbf{U}_p^{n+1} &= -\mathbf{R}(\mathbf{U}_p^n) \\ \mathbf{U}_p^{n+1} &= \mathbf{U}_p^n + \alpha\, \Delta \mathbf{U}_p^{n+1} \end{aligned} \tag{12}$$

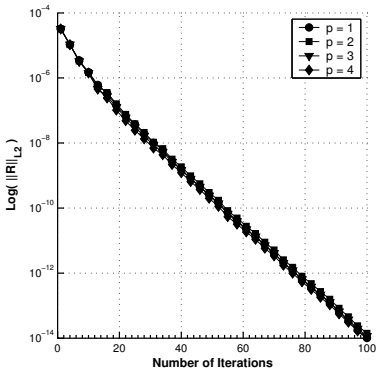
where α represents an under-relaxation factor which may be required for robustness purposes. Although the Newton scheme provides the most rapid convergence based on number of non-linear iterations, each iteration requires the inversion of the large Jacobian matrix $\left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}_p}\right]$ which is prohibitively expensive. Therefore, we proceed by forming approximations to the Jacobian, notably by replacing the full Jacobian matrix by the block diagonal entries corresponding to the coupling between all modes within each element. The resulting scheme can then be written as:

$$\Delta \mathbf{U}_p^{n+1} = [D^n]^{-1}(-\mathbf{R}(\mathbf{U}_p^n)) \tag{13}$$

where $[D^n]$ represents the block diagonal approximation to the full Jacobian, which can be inverted by inverting each individual element sub-block using LU decomposition. This approach is referred to as the non-linear element Jacobi scheme. An alternate approach consists of splitting the original Jacobian into a block diagonal component $[D^n]$ and an off-diagonal component $[O^n]$, and writing the solution scheme as:

$$\Delta \mathbf{U}_p^{k+1} = [D^n]^{-1} \left(-\mathbf{R}(\mathbf{U}_p^n) - [O^n] \Delta \mathbf{U}_p^k \right) \tag{14}$$

We refer to this approach as the linearized element Jacobi scheme. In this approach, a number of subiterations k is performed for each non-linear update n, whereas in the non-linear element Jacobi approach, each cycle corresponds to a non-linear update. For linear problems, these two schemes are identical, and for non-linear problems they should yield similar asymptotic convergence rates.[48] However, the linearized scheme requires less frequent non-linear residual updates, which provides a substantial time savings due to the expensive nature of evaluating the non-linear residual. On the other hand, the linearized approach requires the storage of the additional off-diagonal Jacobian matrix entries, which is often impractical for three-dimensional simulations. A desirable feature of these element-Jacobi solvers is that they deliver convergence rates which are independent of the order of accuracy p of the discretization. This is demonstrated in Figure 7, where the inviscid flow over a the three-dimensional bump previously depicted in Figure 4 is computed using discretizations ranging from p=1 to p=4, and all cases display similar convergence rates using the non-linear Jacobi solver. On the other hand, the cost per iteration grows quickly with increasing p order, mainly due to the increase in the size of the element blocks to be inverted at each iteration, and the viability of this approach is limited to relatively moderate p orders (i.e $p < 6$). Various techniques can be used to improve the effectiveness of these schemes, including freezing the factored Jacobians for a number of successive non-linear iterations, and using Gauss-Seidel variants which make use of the latest information at neighboring cells at each iteration. On the other hand, although all these schemes produce p-independent convergence rates, the number of iterations required increases with mesh resolution, and more elaborate techniques such as multigrid methods are required to obtain a solver which is both p- and h-independent.



(a)

Figure 7. Demonstration of p-independent convergence rate for non-linear element Jacobi scheme for three-dimensional flow over bump case

Multigrid methods are known as efficient techniques for accelerating convergence to steady state for both linear and non-linear problems,[48,49] and can be applied with a suitable existing relaxation technique. The rapid convergence property relies on an efficient reduction of the high frequency error modes on each grid level of a fine-to-coarse sequence of grids.

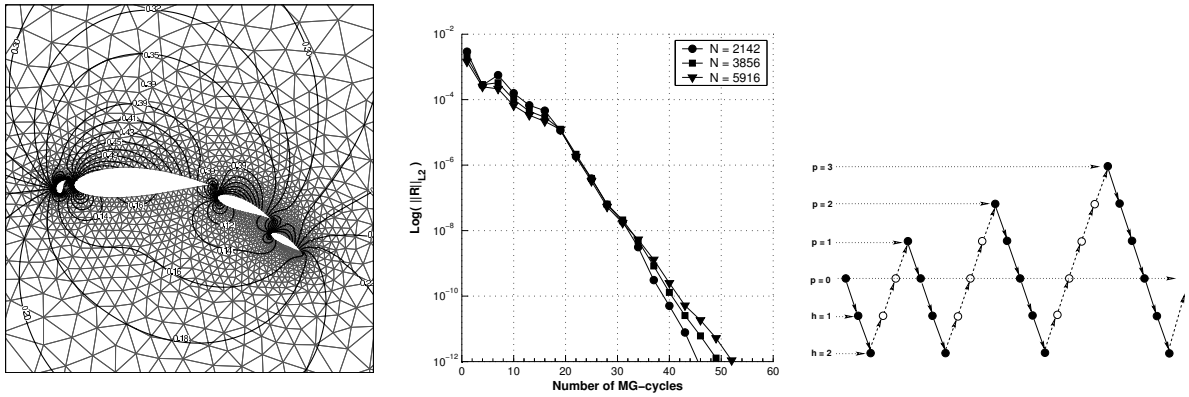
The spectral or p-multigrid approach[15–17,36,50] is based on the same concepts as a traditional h-multigrid method, but makes use of "coarser" levels which are constructed by reducing the order of accuracy of the

discretization, rather than using physically coarser grids with fewer elements. Thus, all grid levels contain the same number of elements, which alleviates the need to perform complex interpolation between grid levels and/or to implement agglomeration-type procedures. Furthermore, the formulation of the interpolation operators between fine and coarse grid levels is greatly simplified when a hierarchical basis set is employed for the solution approximation. The main advantage is due to the fact that the lower order basis functions are a subset of the higher order basis (*i.e.* hierarchical) and the *restriction* and *prolongation* operators become simple projection operators into a lower and higher order space, respectively.[15,16,36] Therefore their formulation is obtained by a simple deletion or augmentation of the basis set. The *restriction* from fine to coarse level is obtained by disregarding the higher order modal coefficients and transferring the values of the low order modal coefficients exactly. Similarly, the *prolongation* from coarse to fine levels is obtained by setting the high order modes to zero and injecting the values of the low order coefficients exactly.

Multigrid strategies are based on a recursive application of a two-level solution mechanism, where the second (coarser) grid is solved exactly, and used to accelerate the solution on the finer grid.[49] Because the exact solution of the coarse grid problem at each multigrid cycle is most often prohibitively expensive, the recursive application of multigrid to solve the coarse grid problem offers the preferred approach for minimizing the computational cost of the multigrid cycle, thus resulting in a complete sequence of coarser grids. For spectral (p)-multigrid methods, the recursive application of lower order discretizations ends with the $p = 0$ discretization on the same grid as the fine level problem. For fine meshes, the (exact) solution of this $p = 0$ problem at each multigrid cycle remains relatively expensive. Alternatively, an approximate solution of the $p = 0$ level using a fixed number of iterations results in overall degraded convergence and compromises the h-independent property of the solution strategy. In order to achieve a truly h-independent approach, the $p = 0$ level can be solved using an agglomeration multigrid strategy. In this scenario, the p-multigrid scheme reverts to an agglomeration multigrid scheme once the $p = 0$ level has been reached, making use of a complete sequence of physically coarser agglomerated grids, thus the designation *hp*-multigrid. This procedure has the potential of resulting in a truly h- and p-independent solution strategy for high-order accurate discontinuous Galerkin discretizations. Figure 8(a) illustrates the computed Mach contours for inviscid two-dimensional flow over an idealized four-element airfoil geometry using a fifth-order accurate DG discretization, which was used to study the h-independence of the h-p multigrid solver. In Figure 8(b), the convergence rate of the multigrid solver is plotted as a function of the number of cycles, for mesh sizes ranging from 2142 to 5916 cells. The similar convergence rates obtained in all cases provides a good demonstration of the relative insensitivity of the multigrid scheme to the mesh size. A full multigrid cycling strategy is employed in these calculations, where the $p = 0$ problem is first solved using h-multigrid, and then used recursively as the initial condition for the solution of the higher-order problems as depicted in Figure 8(c). When combined with the p-independence property of the underlying element Jacobi smoother, the h-p multigrid solver results in an optimal solver which is insensitive to both h and p.

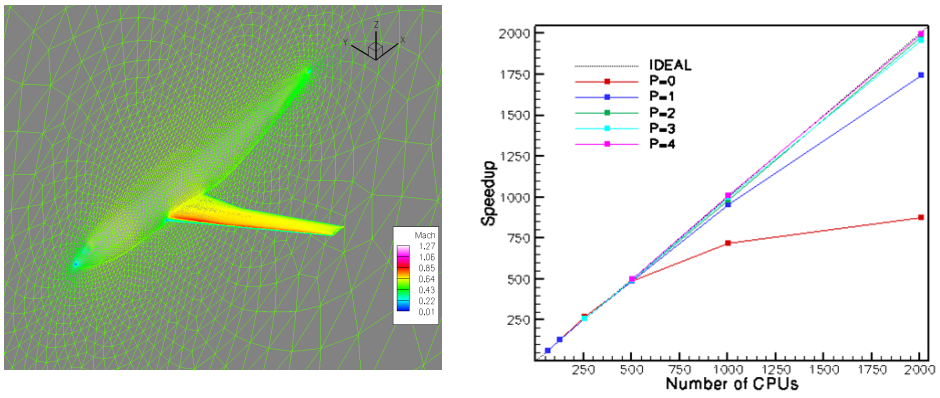
Since one of the principal advantages of DG discretizations is their scalability on massively parallel computer architectures, it is important that this property not be compromised by the chosen solution scheme. Because the h-p multigrid approach relies on the element Jacobi smoother on each level of the multigrid sequence, the locality and compact stencil of this smoother ensures good scalability of the overall multigrid scheme. This is illustrated in the scalability results for a three-dimensional test case consisting of the inviscid flow over a wing-body configuration on a mesh of 2.5 million cells solved using the h-p multigrid scheme. Figure 9(a) depicts the computed surface pressure distribution computed for this case, while Figure 9(b) documents the scalability of this case using the h-p multigrid algorithm, running on up to 2008 cpus of the NASA Columbia Supercomputer. As seen in this plot, although poor scalability is observed for the $p = 0$ case, which corresponds to a first-order accurate finite-volume discretization, the scalability improves with higher p values with near perfect scalability for fourth and fifth-order accurate discretizations on this relatively coarse mesh, largely due to the increased work over the same stencil inherent in these higher-order discretizations.

The introduction of additional viscous terms for the simulation of the full Navier-Stokes equations entails additional complications for the hp-multigrid solver applied to DG discretizations. On the one hand, the



(a) (b) (c)

Figure 8. (a) Mach contours computed on mesh of 2142 elements for 4-element airfoil case using fifth-order accurate discretization ($p = 4$); (b) Demonstration of h-independent convergence rate on meshes of various sizes for flow over two-dimensional airfoil case; (c) Illustration of full multigrid cycling strategy for h-p multigid algorithm.



(a) (b)

Figure 9. (a) Three dimensional Discontinuous Galerkin computation of inviscid flow over wing-body configuration on mesh of 2.5 million cells; (b) Scalability of h-p multigrid algorithm on NASA Columbia Supercomputer for 2.5 million cell mesh as a function of discretization order of accuracy.

cost of the element Jacobi or Gauss-Seidel smoother is not affected dramatically by the addition of the viscous terms, since the dominant cost is related to the factorization and multiplication of the diagonal and off-diagonal blocks which retain the same structure (although the off-diagonal blocks are more dense in the Navier-Stokes case). However, the approximations necessary to generate a $p = 0$ discretization of the viscous terms leads to inconsistencies between the $p = 0$ and higher $p \geq 1$ levels, resulting in non-optimal convergence of the h-p multigrid algorithm.[51, 52] One approach to this problem consists of solving the $p = 1$ level more exactly, with additional ($p \leq 1$ multigrid) subiterations within each fine level multigrid cycle. A more effective approach consists of using the entire h-p multigrid scheme as a preconditioner for a Krylov scheme such as GMRES.[52] Figure 10 depicts the convergence obtained for the viscous laminar airfoil case previously discussed in Figure 6 using a fifth-order accurate ($p = 4$) discretization on a mesh of 7701 cells, illustrating the benefit of additional sub-cycling on the $p = 1$ level, and the superior performance of the Krylov scheme.

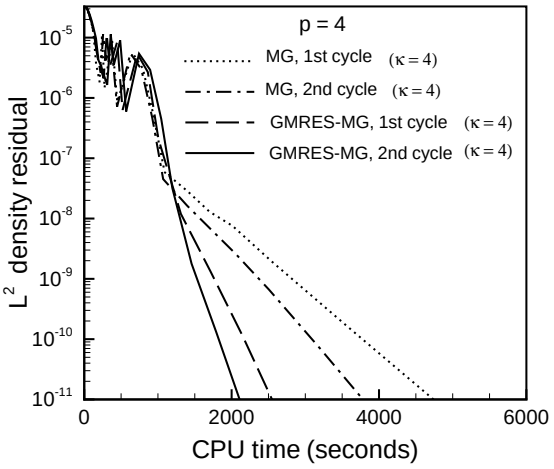


Figure 10. Convergence for viscous airfoil flow problem using the standard h-p multigrid cycling strategy with four additional subcyles at $p = 0$ (1st cycle), and using four additional subcycles at p=1 (2nd cycle), and corresponding convergence curve using the same h-p multigrid algorithm as a preconditioner for GMRES.

IV. Discretely Conservative Arbitrary Lagrangian-Eulerian (ALE) Formulation

While much work has been performed with DG methods for steady-state or time-dependent problems on static meshes, relatively few investigations have looked at the formulation of DG methods on dynamically deforming meshes.[20–22] The use of dynamically deforming meshes is important for many engineering applications with relative body motion, such as aeroelasticity. In order to use DG methods on moving meshes, the discretization must first be cast in Arbitrary Lagrangian Eulerian (ALE) form. For this, the time derivative must be taken out of the integral in equation (9) which, after simplification[22] yields the ALE form of the DG discretization:

$$\frac{\partial}{\partial t}\int_{\widehat{\Omega}_k} \phi_i \mathbf{U}_p \, |J_k| d\widehat{\Omega}_k - \int_{\widehat{\Omega}_k} \nabla\phi_i \cdot (\mathbf{F}(\mathbf{U}_p) - \mathbf{U}_g\mathbf{U}_p) \, J_k^{-1}|J_k| d\widehat{\Omega}_k + \int_{\partial\widehat{\Omega}_k} \phi_i (\mathbf{F}^*(\mathbf{U}_p) - \mathbf{U}_g\mathbf{U}_p)\cdot\mathbf{n}\,|J_k| d(\partial\widehat{\Omega}_k) = 0 \tag{15}$$

where $\mathbf{U}_g$ denotes the grid velocity field. An important consideration is whether the particular ALE form of the discretization preserves discrete conservation exactly. The statement of discrete conservation is embodied

in the Geometric Conservation Law (GCL),[53] which simply states that a uniform flow, which is an exact solution of any consistent discretization, should be preserved exactly in the presence of arbitrary mesh motion. Unfortunately, prescribing a constant flow field $\mathbf{U}_p$ in equation (15) and simplifying the resulting equations, leads to a set of constraints on the grid velocity field $\mathbf{U}_g$ which is most often indeterminate (different number of degrees of freedom versus constraints). This is compounded by the fact that the spatial and temporal orders of accuracy of the grid velocity field may be arbitrary and different than those of the flow field $\mathbf{U}_p$, particularly for prescribed grid motion cases.

In reference[22] we have derived an approach for formulating fully conservative ALE discretizations which preserve the design order of spatial and temporal accuracy of the underlying static mesh schemes, while permitting the use of arbitrary mesh motion descriptions. The approach is most easily explained in the context of a first-order accurate backwards difference temporal discretization (BDF1). In this case, the discrete equations obtained by the method of lines can be recovered through a space-time discretization, using a tensor product basis set in space-time given by:

$$\Phi(x_i, t) = \phi(x_i)\psi(t) \tag{16}$$

where $\phi(x_i)$ represents the basis functions of the original spatial DG discretization, and the choice $\psi(t) = 1$ corresponds to a BDF1 temporal discretization. The resulting equations take on a form that corresponds to a time-integrated version of equation (15), namely:

$$\begin{aligned} \int_{\widehat{\Omega}} \phi_i^{n+1} \mathbf{U}^{n+1} |J_k|^{n+1} d\widehat{\Omega}_k - \int_{\widehat{\Omega}(t)} \phi_i^n \mathbf{U}^n |J_k|^n d\widehat{\Omega}_k \quad &= \\ \int_t^{t+\Delta t} \int_{\widehat{\Omega}_k} \nabla\phi_i \cdot \left[\mathbf{F}(\mathbf{U}_p) - \mathbf{U}_p\mathbf{U}_g\right] J_k^{-1} |J_k| d\widehat{\Omega}_k dt \quad &- \quad \int_t^{t+\Delta t} \int_{\partial\widehat{\Omega}_k} \phi_i \left[\mathbf{F}^*(\mathbf{U}_p) - \mathbf{U}_p\mathbf{U}_g\right] \cdot \mathbf{n} \, |J_k| d(\partial\widehat{\Omega}_k) dt \end{aligned} \tag{17}$$

However, since this results from a conservative space-time formulation, if all terms are integrated exactly, then discrete conservation is naturally obtained. In order to proceed, a functional form of the grid speed field $\mathbf{U}_g$ is required, and this is obtained by assuming the grid motion to be completely defined by the location of the points defining each mesh cell (or alternatively the quadrature points) in time, and expanding the $\mathbf{U}_g$ field in terms of a set of spatial basis functions. Thus, the product of the grid velocity terms with the flow field $\mathbf{U}_p\mathbf{U}_g$ present in the integrands of equation (17) results in higher-order polynomial functions in space, while the metric terms $\mathbf{n}$ and $J^{-1}, |J_k|$ become non-linear functions in time, requiring the use of higher-order quadrature rules in both space and time. Although this approach is only directly applicable to a first-order accurate BDF1 temporal discretization, it may be extended to more accurate BDF2 and implicit Runge-Kutta discretizations through application to each individual time interval or stage which makes up a multi-step or implicit Runge-Kutta scheme, respectively. For example, after dividing the space-time formulation of equation (17) through by the time step to return to the form given in equation (15), and after some manipulation, the GCL preserving formulation for a BDF2 temporal scheme yields the following expressions for the composite grid speed terms to be evaluated at each spatial quadrature point in equation (15):

$$\left[\nabla\phi_i \mathbf{U}_g \, J_k^{-1} |J_k|\right]^{n+1} = \frac{3}{2\Delta t} \int_t^{t+\Delta t} \nabla\phi_i \cdot \mathbf{U}_g \, J_k^{-1} |J_k| dt - \frac{1}{2\Delta t} \int_{t-\Delta t}^{t} \nabla\phi_i \cdot \mathbf{U}_g \, J_k^{-1} |J_k| dt \tag{18}$$

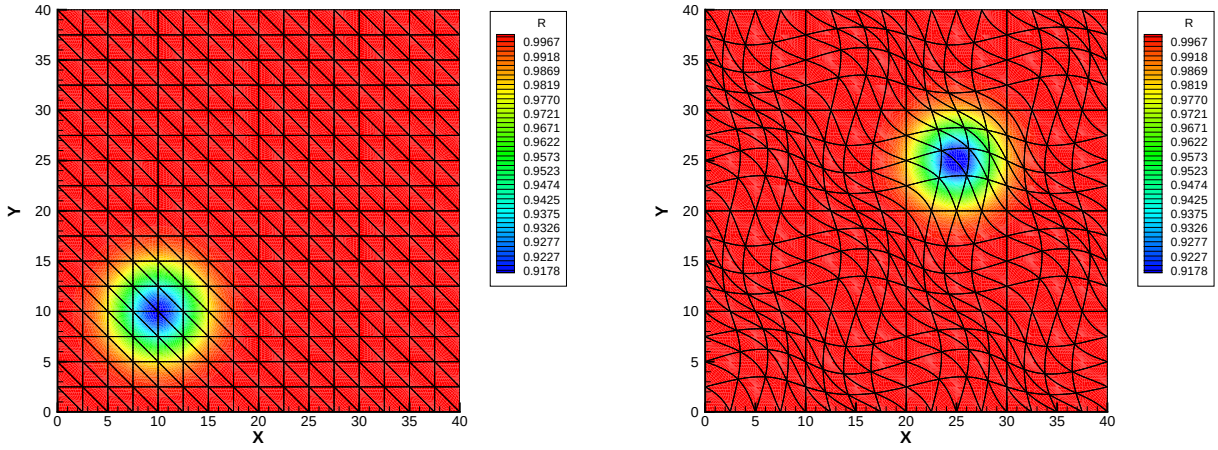
and

$$\left[\phi_i \mathbf{U}_g \cdot \mathbf{n} \, |J_k|\right]^{n+1} = \frac{3}{2\Delta t} \int_t^{t+\Delta t} \phi_i \mathbf{U}_g \cdot \mathbf{n} \, |J_k| dt - \frac{1}{2\Delta t} \int_{t-\Delta t}^{t} \phi_i \mathbf{U}_g \cdot \mathbf{n} \, |J_k| dt \tag{19}$$

where it is understood that exact quadrature rules must be used on the right-hand side time integrals, given the known functional forms of $\mathbf{U}_g$, $\mathbf{n}$, J_k^{-1} and $|J_k|$.

Figure 11 illustrates a test case used to validate this approach, which consists of a vortex being convected through a deforming mesh of curved elements. Prior to the vortex test case, it was verified that the simulation of uniform flow on this deforming mesh is indeed preserved exactly (machine zero residuals at each time step). Using BDF1 and BDF2 temporal discretizations, the convecting vortex case was then used to demonstrate the retention of first or second-order accuracy, respectively, in the presence of the moving mesh, as demonstrated in the temporal accuracy plot depicted in Figure 12 for this case.

Work is currently underway to validate this discretely conservative ALE formulation to higher-order temporal discretizations such as implicit Runge-Kutta schemes. For higher-order temporal discretizations, one may either attempt to apply the current formulation at each stage of a Runge-Kutta scheme, or to formulate a space-time discretization making use of the same temporal basis functions which define the implicit Runge-Kutta scheme, when this is feasible.



(a) BDF1

(b) BDF2

Figure 11. (a) Initial density contours and mesh for convecting vortex problem using $p = 4$ spatial discretization; (b) Density contours for BDF2 scheme at time $t = 3/4\, t_{max}$ using a time-step of $\delta t = 0.25$ and $p = 4$ spatial discretization.

V. Adjoint Sensitivity Methods

Adjoint methods have become popular in CFD simulations over the last decade. These methods were initially adopted for use in design optimization problems[54] and more recently have been used for error estimation and adaptive mesh refinement purposes.[24–26, 28]

Adjoint methods enable the calculation of the sensitivities for a single simulation output with respect to any number of simulation inputs at the cost of a single solution of the adjoint equations. Adjoint formulations can be obtained by transposing the linearization process, and are thus often thought of as reverse linearization techniques. These methods are particularly well suited for design optimization procedures, where one is most often interested in some global quantity such as lift, drag or heating. For error estimation purposes, adjoint methods provide a mechanism for determining the effect of local errors (in space and time) on the final global simulation output of interest, as well as sensitivities of the output with respect to local changes in the mesh configuration. This, in turn, can be used to drive an adaptive meshing strategy, which tailors the computational mesh to the particular objective under consideration.

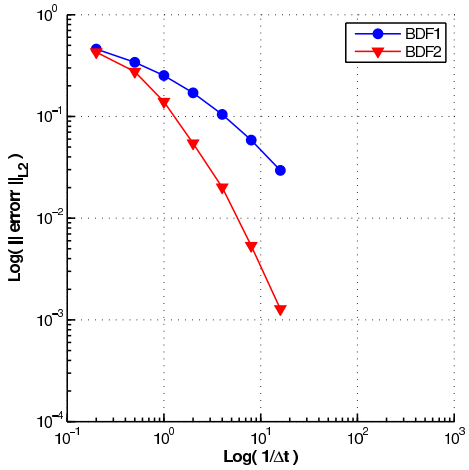


Figure 12. Temporal accuracy as a function of time-step size at $t = 10$ for the convecting vortex problem in the presence of dynamically deforming mesh.

Because the formulation of adjoint methods for spatial error estimation was originally developed within the context of a variational framework, these were naturally applied to DG discretizations at an early stage.[23,55,56] In recent work, we have formulated adjoint spatial error estimation techniques based on a non-variational framework,[25,26,28,57–59] although the variational and non-variational approach can be shown to lead to the same final discrete equations.

We consider the case where a current solution $\bar{\mathbf{w}} = \mathbf{w} + \delta\mathbf{w}$ is given as a perturbation of the exact solution $\mathbf{w}$. As an example, $\mathbf{w}$ may represent the exact solution on a fine mesh, while $\bar{\mathbf{w}}$ denotes the solution on a coarser mesh, which does not satisfy the fine mesh discrete equations when interpolated to the fine mesh. Alternatively, $\bar{\mathbf{w}}$ may correspond to the solution on the same mesh but computed using a lower order (p) discretization. The corresponding change in the output functional of interest can be written as

$$\delta J = J(\bar{\mathbf{w}}) - J(\mathbf{w}) \tag{20}$$

and the corresponding change in the residual is given as:

$$\delta\mathbf{R} \quad = \quad \mathbf{R}(\bar{\mathbf{w}}) - \mathbf{R}(\mathbf{w}) = \mathbf{R}(\bar{\mathbf{w}}) \tag{21}$$

where the last equality follows due to the fact that the $\mathbf{w}$ values satisfy the residual equations exactly on the fine mesh. Linearizing about the perturbed solution yields:

$$\delta J \quad = \quad \frac{\partial J}{\partial \mathbf{w}}\delta\mathbf{w} \tag{22}$$

and

$$\delta\mathbf{R} \quad = \quad \frac{\partial \mathbf{R}}{\partial \mathbf{w}}\delta\mathbf{w}, \quad or \quad \delta\mathbf{w} = \left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^{-1}\delta\mathbf{R} \tag{23}$$

which, upon substitution into equation (22) yields

$$\delta J \quad = \quad \mathbf{\Lambda}^T\delta\mathbf{R} \tag{24}$$

using the definition of the adjoint solution variable $\mathbf{\Lambda}$:

$$\left[\frac{\partial \mathbf{R}}{\partial \mathbf{w}}\right]^T \mathbf{\Lambda} = \left[\frac{\partial J}{\partial \mathbf{w}}\right]^T \tag{25}$$

Thus, an estimate of the variation of the functional output which would be observed by recomputing the solution on a finer grid (i.e. an estimate of the error in this functional) can be obtained as the inner product between the adjoint solution and the residual obtained using the current solution projected or reconstructed onto a finer mesh. This variation corresponds to an estimate of the global error in the functional output, and a grid refinement criteria can be devised by flagging regions of the mesh where element contributions to this inner product are relatively large. In order to avoid the expense of solving the adjoint field $\mathbf{\Lambda}$ on the fine level, which would be equivalent to the cost of computing the flow solution on the fine level, the adjoint is solved only on the coarse level and then projected onto the fine level where it is used to compute the inner product defined in equation (24).

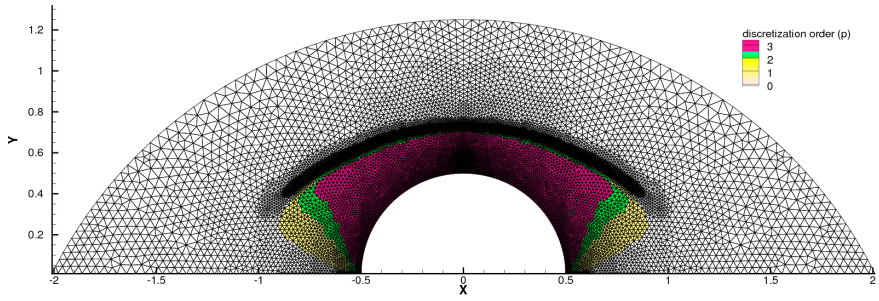
In the context of spatial error adaptivity, high-order methods offer the possibility of pursuing both mesh (h) refinement and order (p) refinement. The choice between h and p refinement is guided by the smoothness of the solution, with smooth solution regions targeted for p refinement, and non-smooth regions targeted for h refinement. In order to guide this process, a local smoothness indicator is required. Of the various indicators proposed in the literature, the smoothness indicator based on the ratio of the L2 norm of the highest modes to the remaining modes[60–62] has been found to provide reliable results, as well as an indicator based on the magnitude of the inter-element jumps.[61,63] A successful h-p adaptive refinement strategy should result in a low order discretization and a refined mesh in regions of discontinuities, and high-order discretizations in other regions of the domain critical to the accuracy of the output objectives of interest, while providing exponential (optimal) error reduction with increasing numbers of degrees of freedom.[5–7]

Figures 13 and 14 illustrate an h-p adaptive computation for an inviscid Mach 6 flow over a circular cylinder, using the adjoint-based error criterion to drive the adaptive process, for an output objective defined as the surface integrated temperature on the cylinder, and using the smoothness indicator based on inter-element jumps.[63] The adaptive procedure begins with a first-order accurate ($p = 0$) discretization throughout the domain, and refines the mesh while naturally maintaining the $p = 0$ discretization upstream of the shock (where the flow is uniform) and within the shock region, which is captured without the need for any limiting. At the same time, the accuracy order is raised in the region between the shock and the cylinder, while refinement in h or p does not occur in lateral regions near the cylinder, since the flow in these regions does not impact the objective of interest (surface temperature). Figure 15 denotes the convergence of the adaptive refinement process for the surface integrated temperature objective. At each level, the error in the functional computed by equation (24) can be added to the current functional value to predict the value of the functional computed on the next finer level. As seen in Figure 15, the agreement between the predicted functional value and the actual value computed on the next level of refinement improves with increasing refinement, with very good agreement occurring at the last level of refinement, where the linearization assumption inherent in equations (22) through (24) becomes most valid.

VI. Conclusions and Future work

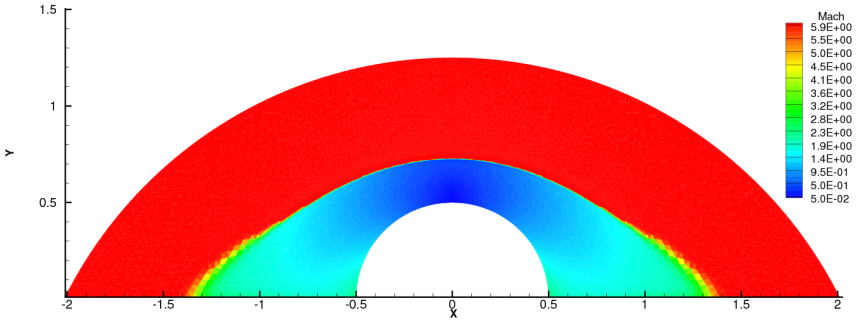
We have outlined various advances that we have pursued over the last five years in the interest of advancing discontinuous Galerkin methods to the point where they can be used in industrial aerodynamic problems. In general, the scientific community has achieved substantial progress in higher-order methods in general, and discontinuous Galerkin methods in particular over the last decade. Discretizations for convective and diffusive terms are now relatively well understood, and a renewed emphasis on efficient steady-state and time-implicit solvers is beginning to pay dividends. Sensitivity analysis methods have been demonstrated in the context of high-order methods, and advances are being pursued in optimization and error estimation.

However, substantial obstacles remain to be overcome. High-order DG methods are still relatively expensive compared to their low-order finite-volume counterparts. On the one hand, solvers such as hp multigrid



(a) 3^{rd} adapted mesh, 42234 elements, discretization orders ($p_{min} = 0, p_{max} = 3$)

Figure 13. *hp*-adapted meshes with the distributions of order of discretizations for hypersonic flow (Mach 6) over a half-circular cylinder.



(a) Mach number contours

Figure 14. Mach number contours on the final *hp*-adapted mesh for hypersonic flow (Mach 6) over a half-circular cylinder.

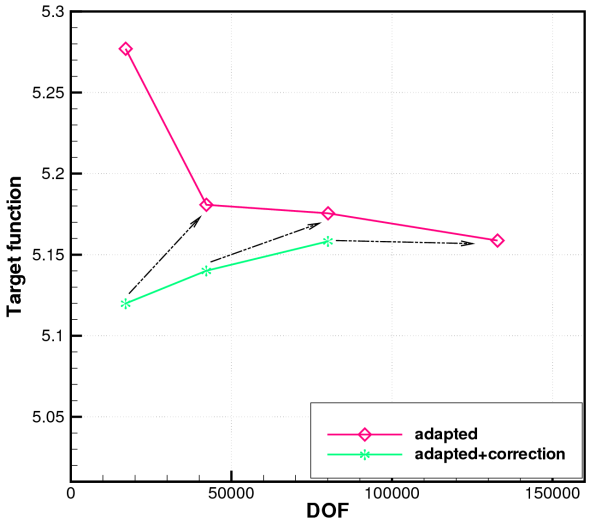


Figure 15. Comparison of functional convergence in terms of DOF for hypersonic flow over half-circular cylinder (Mach 6).

have proven to be successful at delivering optimal convergence rates in terms of numbers of cycles, but the cost per cycle remains relatively high. In our examples, the element Jacobi smoother used within the multigrid algorithm scales poorly with increasing order p, principally because it relies on factorizing the entire dense matrix in each element, the size of which grows superlinearly with the number of degrees of freedom within an element. This hinders use of the method for high p orders (i.e. $p \geq 6$) and more suitable smoothers for very high p order would be desirable.

However, this raises the question of what order of accuracy (p) is best suited for aerodynamic problems. In other communities, spectral and spectral-element simulations with very high order (up to $p = 10$ or higher) are not uncommon. The answer to this question lies in the characteristics of the problem at hand, and in the level of accuracy required of the simulation. Since high-order solutions are most effective in the presence of smooth solution behavior, it is likely that very high order discretizations will not be suitable for industrial aerodynamic problems, which naturally involve discontinuous and non-smooth behavior both from the flow physics and geometry definition points of view. Additionally, the level of accuracy required from an engineering standpoint may not be sufficient to justify the use of very high-order methods, although this assumption may not hold true for specific disciplines such as aeroacoustics and electromagnetics.

Finally, another important aspect for industrial simulations relates to the robustness of the methodology. Attempting to approximate a non-smooth solution with a high-order discretization can lead to well known robustness issues, which in turn may require the use of limiting or other local accuracy reducing techniques. Insufficient resolution, a pervasive issue in most industrial problems, can also result in lack of robustness for high-order discretizations, since the solution becomes non-smooth at the level of the mesh resolution in such cases. Thus, it seems that a variable order discretization capability will be required for ensuring robust simulation capability, and that ultimately, an h-p refinement capability will be key to achieving not only efficiency and accuracy objectives, but also robustness objectives.

In spite of these drawbacks, higher-order methods currently offer the best prospects for advancing the accuracy and reliability of aerodynamic simulation capabilities in the long run, due to their inherent mathematical properties, while at the same time offering an appealing approach for making effective use of rapidly

advancing parallel computer hardware.

VII. Acknowledgments

This work was originally supported by ONR grant N00014-04-1-0602 which ended in 2007, and is currently supported under NASA Grant NX07AC31A. We are also grateful for computer time provided by the NASA Advanced Supercomputing Division.

References

[1]Levy, D. W., Zickuhr, T., Vassberg, J., Agrawal, S., Wahls, R. A., Pirzadeh, S., and Hemsch, M. J., "Data Summary from the First AIAA Computational Fluid Dynamics Drag Prediction Workshop," *Journal of Aircraft*, Vol. 40, No. 5, 2003, pp. 875–882.

[2]Laflin, K., Klausmeyer, S. M., Zickuhr, T., Vassberg, J. C., Wahls, R. A., Morrison, J. H., Brodersen, O. P., Rakowitz, M. E., Tinoco, E. N., and Godard, J., "Data Summary from Second AIAA Computational Fluid Dynamics Drag Prediction Workshop," *Journal of Aircraft*, Vol. 42, No. 5, 2005, pp. 1165–1178.

[3]Vassberg, J. C., Tinoco, E. N., Mani, M., Brodersen, O. P., Eisfeld, B., Wahls, R. A., Morrison, J. H., Zickuhr, T., Laflin, K. R., and Mavriplis, D. J., "Summary of the Third AIAA CFD Drag Prediction Workshop," AIAA Paper 2007-0260.

[4]Mavriplis, D. J., "Grid Resolution Study of a Drag Prediction Workshop Configuration Using the NSU3D Unstructured Mesh Solver," AIAA-Paper 2005-4729.

[5]Bey, K. S. and Oden, J. T., "Hp-version discontinuous Galerkin method for hyperbolic conservation laws," *Comput. Methods Appl. Mech. Engr.*, Vol. 133, 1996, pp. 259–286.

[6]Solin, P., Segeth, P., and Zel, I., *High-Order Finite Element Methods*, Studies in Advanced Mathematics, Chapman and Hall, 2003.

[7]Karniadakis, G. E. and Sherwin, S. J., *Spectral/hp Element Methods for CFD*, Oxford University Press, 1999.

[8]Reed, W. H. and Hill, T. R., "Triangular mesh methods for the neutron transport equation," Los Alamos National Laboratory Technical Report LA-UR-73-479.

[9]Cockburn, B., Hou, S., and Shu, C., "TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case," *Math. Comp.*, Vol. 54, 1990, pp. 545–581.

[10]Cockburn, B. and Shu, C. W., "The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems," *Journal of Computational Physics*, Vol. 141, No. 2, 1998, pp. 199–2241.

[11]Cockburn, B., Karniadakis, G. E., and Shu, C. W., *Discontinuous Galerkin Methods: Theory, Computation and Applications*, Springer, 2000.

[12]Hesthaven, J. S. and Warburton, T., *Nodal Discontinuous Galerkin Methods*, Vol. 54, Springer, 2008, Texts in Applied Mathematics.

[13]Wang, Z. J., "Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids. Basic Formulation: Basic Formulation," *Journal of Computational Physics*, Vol. 178, No. 1, May 2002, pp. 210–251.

[14]Wang, Z. J., "On the connection between the spectral volume and the spectral difference method," *Journal of Computational Physics*, Vol. 227, No. 2, 2007, pp. 877–885.

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

[16]Nastase, C. and Mavriplis, D. J., "High-Order Discontinuous Galerkin Methods Using a Spectral Multigrid Approach," *Journal of Computational Physics*, Vol. 213, No. 1, March 2006, pp. 330–357.

[17]Mascarenhas, B., Helenbrook, B. T., and Atkins, H. L., "Application of the p-multigrid algorithm to discontinuous Galerkin formulations of the compressible Euler equations," AIAA Paper 2007-4331.

[18]Arabatzis, G., Vavilis, P., Toulopoulos, I., and Ekaterinaris, J., "Implicit High-Order Time Marching Schemes for the Linearized Euler Equations," *AIAA Journal*, Vol. 45, No. 8, 2007, pp. 1819–1826.

[19]Wang, L. and Mavriplis, D. J., "Implicit Solution of the Unsteady Euler Equations for High-Order Accurate Discontinuous Galerkin Discretizations," *Journal of Computational Physics*, Vol. 225, No. 2, Aug. 2007, pp. 1994–2015.

[20]Lomtev, I., Kirby, R., and Karniadakis, G., "A Discontinuous Galerkin method in moving domains," In Proc. of Discontinuous Galerkin Methods: Theory, Computation and Applications, eds. Cockburn et al., Springer-Verlag, NY.

[21]Persson, P. and Peraire, J., "Discontinuous Galerkin Solution of the Navier-Stokes Equations on Deformable Domains," AIAA Paper 2007–0513.

[22]Mavriplis, D. J. and Nastase, C., "On the Geometric Conservation Law for High-Order Discontinuous Galerkin Discretizations on Dynamically Deforming Meshes," AIAA Paper 2008-0778.

[23]Oden, J. T. and Prudhomme, S., "On goal-oriented error estimation for elliptic problems: Application to control of pointwise errors," *Comput. Methods Appl. Mech. Engr.*, Vol. 176, 1999, pp. 313–331.

[24]Hartmann, R. and Houston, P., "Adaptive Discontinuous Galerkin Finite Element Methods for the Compressible Euler Equations," *Journal of Computational Physics*, Vol. 183, No. 2, 2002, pp. 508–553.

[25]Giles, M. B. and Suli, E., "Adjoint Methods for PDEs: a posteriori error analysis and postprocessing by duality," *Acta Numerica*, 2002, pp. 145–236.

[26]Venditti, D. and Darmofal, D., "Grid adaptation for functional outputs: Application to 2-D inviscid compressible flow," *Journal of Computational Physics*, Vol. 176, 2002, pp. 40–69.

[27]Wintzer, M., Nemec, M., and Aftosmis, M., "Adjoint-Based Adaptive Mesh Refinement for Sonic Boom Prediction," AIAA Paper 2008-6593, 26th AIAA Applied Aerodynamics Conference, Honolulu, HI.

[28]Wang, L. and Mavriplis, D. J., "Adjoint-based h-p Adaptive Discontinuous Galerkin Methods for the Compressible Euler Equations," AIAA Paper 2009-0952.

[29]Davis, S. F., "Simplified Second-Order Godunov-Type Methods," *SIAM J. Sci. Statist. Comput.*, Vol. 9, No. 3, 1988, pp. 445–473.

[30]Roe, P. L., "Approximate Riemann solvers, parameter vectors and difference schemes," *J. Comp. Phys.*, Vol. 43, No. 2, 1981, pp. 357–372.

[31]Harten, A., Lax, P. D., and Van Leer, B., "On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws," *SIAM Review*, Vol. 25, No. 1, 1983, pp. 35–61.

[32]Toro, F. E., *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Applied Mechanics, Springer-Verlag, New York, NY, 1999.

[33]Batten, P., Clarke, N., Lambert, C., and Causon, D. M., "On the Choice of Wavespeeds for the HLLC Riemann Solver," *SIAM J. Sci. Comput.*, Vol. 18, No. 2, 1997, pp. 1553–1570.

[34]Batten, P., Leschiner, M. A., and Goldberg, U. C., "Average-State Jacobians and Implicit Methods for Compressible Viscous and Turbulent Flows," *J. Comput. Phys.*, Vol. 137, 1997, pp. 38–78.

[35]Szabo, B. and Babuska, I., *Finite Element Analysis*, John Wiley & Sons, Inc., New York, NY, 1991.

[36]Nastase, C. and Mavriplis, D. J., "A Parallel hp-Multigrid Solver for Three-Dimensional Discontinuous Galerkin Discretizations of the Euler Equations," AIAA Paper 2007-0512.

[37]Dunavant, D. A., "High Degree Efficient Symmetrical Gaussian Guadrature Rules for the Triangle," *Int. J. Numer. Meth. Engng.*, Vol. 21, 1985, pp. 1129–1148.

[38]Dunavant, D. A., "Economical Symmetrical Quadrature Rules for Complete Polynomials Over a Square Domain," *Int. J. Numer. Meth. Engng.*, Vol. 21, 1985, pp. 1777–1784.

[39]Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, D., "Unified analysis of discontinuous Galerkin methods for elliptic problems," *SIAM Journal of Numerical Analysis*, Vol. 39, 2002, pp. 1749–1779.

[40]Cockburn, B. and Shu, C. W., "The local discontinuous Galerkin for convection-diffusion systems," *SIAM Journal of Numerical Analysis*, Vol. 35, 1998, pp. 2440–2463.

[41]Peraire, J. and Persson, P. O., "The compact discontiuous Galerkin (CDG) method for elliptic problems," *SIAM Journal of Scientific Computing*, Vol. 30, No. 4, 2008, pp. 1806–1824.

[42]Bassi, F. and Rebay, S., "GMRES Discontinuous Galerkin solution of the compressible Navire-Stokes equations," in Dicontinuous Galerkin Methods: Theory, Computations and Applications, K. Cockburn, W. Shu (Eds.), Springer, Berlin.

[43]van Leer, B., "Discontinuous Galerkin for Diffusion," AIAA-Paper 2005-5108.

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

[45]Arnold, D. N., "An interior penalty finite element method with discontinuous elements," *SIAM Journal of Numerical Analysis*, Vol. 19, 1982, pp. 742–760.

[46]Shahbazi, K., "An explicit expression for the penalty parameter of the interior penalty method," *Journal of Computational Physics*, Vol. 205, No. 2, May 2005, pp. 401–407.

[47]Mavriplis, D. J., Jameson, A., and Martinelli, L., "Multigrid solution of the Navier-Stokes equations on triangular meshes," AIAA Paper 89-0120.

[48]Mavriplis, D. J., "An Assessment of Linear versus Non-Linear Multigrid Methods for Unstructured Mesh Solvers," *Journal of Computational Physics*, Vol. 175, Jan. 2002, pp. 302–325.

[49]Trottenberg, U., Schuller, A., and Oosterlee, C., *Multigrid*, Academic Press, London, UK, 2000.

[50]Luo, H., Baum, J. D., and Löhner, R., "A p-multigrid discontinuous Galerkin method for the Euler equations on unstructured grids," *Journal of Computational Physics*, Vol. 211, No. 2, 2006, pp. 768–783.

[51]Helenbrook, B. T. and Atkins, H. L., "Solving Discontinuous Galerkin Formulations of Poisson's Equation using Geometric and p Multigrid," *AIAA Journal*, Vol. 46, No. 4, April 2008, pp. 894–902.

[52]Shahbazi, K. and Mavriplis, D. J., "Multigrid Algorithms for High-Order Discontinuous Galerkin Discretizations of the Compressible Navier-Stokes Equations," submitted to Journal of Computational Physics.

[53]Guillard, H. and Farhat, C., "On the significance of the geometric conservation law for flow computations on moving meshes," *Computer Methods in Applied Mechanics and Engineering*, Vol. 190, 2000, pp. 1467–1482.

[54]Jameson, A., "Re-engineering the design process through computation," *J. of Aircraft*, Vol. 36, 1999, pp. 36–50.

[55]Estep, D., "A Posteriori Error Bounds and Global Error Control for Approximation of Ordinary Differential Equations," *SIAM Journal of Numerical Analysis*, Vol. 32, 1995, pp. 1–48.

[56]Houston, P., Rannacher, R., and Suli, E., "A posteriori error analysis for stabilized finite-element approximations of transport problems," *Comput. Methods Appl. Mech. Engr.*, Vol. 190, 2000, pp. 1483–1508.

[57]Mueller, J. D. and Giles, M. B., "Solution adaptive mesh refinement using adjoint error analysis," AIAA Paper 2001-2550.

[58]Mani, K. and Mavriplis, D. J., "Discrete Adjoint Based Time- Step Adaptation and Error Reduction in Unsteady Flow Problems," AIAA Paper 2007-3944, presented at the 18th AIAA Computational Fluid Dynamics Conference, Miami FL.

[59]Mani, K. and Mavriplis, D. J., "Error Estimation and Adaptation for Functional Outputs in Time- Dependent Flow Problems," AIAA Paper 2009-1495, presented at the AIAA Aerospace Sciences Meeting, Orlando FL.

[60]Mavriplis, C., "Adaptive Mesh Strategies for the Spectral Element Method," *Comput. Methods. Appl. Mech. Engrg.*, Vol. 116, 1994, pp. 77–86.

[61]Barter, G. E. and Darmofal, D. L., "Shock capturing with higher-order PDE-based artificial viscosity," AIAA Paper 2007-3823.

[62]Persson, P. and Peraire, J., "Sub-Cell Shock Capturing for Discontinuous Galerkin Methods," AIAA Paper 2006-0112.

[63]Krivodonova, L., Xin, J., Remacle, J. F., Chevaugeon, N., and Flaherty, J., "Shock detection and limiting with discontinuous Galerkin Methods for hyperbolic conservation laws," *Applied Numerical Mathematics*, Vol. 48, 2004, pp. 323–338.