AIAA JOURNAL
Vol. 36, No. 5, May 1998

Quadrature-Free Implementation of Discontinuous Galerkin Method for Hyperbolic Equations

Harold L. Atkins*
NASA Langley Research Center, Hampton, Virginia 23681-2199
and
Chi-Wang Shu†
Brown University, Providence, Rhode Island 02912

A discontinuous Galerkin formulation that avoids the use of discrete quadrature formulas is described and applied to linear and nonlinear test problems in one and two space dimensions. For many practical problems, this approach requires fewer operations and less storage than conventional implementations but preserves the compactness and robustness that is inherent in the discontinuous Galerkin method. Test problems include the linear and nonlinear one-dimensional scalar advection of both smooth and discontinuous initial value problems, the two-dimensional scalar advection of smooth initial value problems that are discretized by using unstructured grids with varying degrees of smoothness and regularity, and two-dimensional linear Euler solutions on unstructured grids.

Introduction

COMPUTATIONAL methods for aeroacoustics must possess accuracy properties that exceed those of conventional second-order computational fluid dynamics (CFD) methods. At the same time, many problems of interest involve complex geometries that cannot be easily treated by common high-order methods that normally require a smooth, structured grid. In addition to the geometrically complex problem, we are particularly interested in strongly nonlinear flows that contain shock waves as a major source of sound generation, such as in the case of jet noise. In an effort to satisfy these requirements, the relatively untried discontinuous Galerkin method is being tested for hyperbolic problems. Some advantages of this approach include the ease with which the method can be applied to both structured and unstructured grids and its suitability for parallel computer architectures.

The discontinuous Galerkin method also has several useful mathematical properties. Johnson and Pitkäräta[1] proved stability and error estimates for linear scalar advection. In a series of papers, Cockburn and Shu[2] and Cockburn et al.[3,4] discussed the discontinuous Galerkin method with the use of approximate Riemann solvers, limiters, and total-variation-diminishing (TVD) Runge–Kutta time discretizations for nonlinear hyperbolic problems. In Ref. 2, the general formulation in one space dimension is provided; this work includes an analysis of the accuracy and stability (in terms of total variation) and a detailed description of the implementation. Numerical examples using the scalar equations are also provided. In Ref. 3, the method is applied to one-dimensional systems. Stability of the initial boundary value problem for a linear system is proved, and numerical experiments are presented for the one-dimensional Euler equations. In Ref. 4, the method is generalized to multispace dimensions. A primary result in Ref. 4 is the design of a limiter that applies to general triangulations, maintains a high order of accuracy in smooth regions, and guarantees maximum norm stability. Jiang and Shu[5] have also proven that the discontinuous Galerkin method satisfies a local cell entropy inequality for the square entropy $\eta(U) = U^2$ for arbitrary triangulations in any space dimension and for any order of accuracy. This proof trivially implies L_2 stability of the method for nonlinear shocked problems in the scalar case.

Although the discontinuous Galerkin method has not been widely used in the CFD arena, several instances exist in which this method has been applied to the Euler or Navier–Stokes equations.[3,4,6–9] Halt and Agarwal[6] applied the method of moments (similar to the discontinuous Galerkin method) to the steady, two-dimensional Euler equations for subsonic flows. Bassi and Rebay[7] applied the discontinuous Galerkin method to two-dimensional Euler equations for transonic flows and demonstrated the importance of the proper treatment of curved boundaries. In Ref. 8, Bassi and Rebay extended their method to the Navier–Stokes equations by introducing the gradient of the solution as an auxiliary variable. Most recently, Lowrie et al.[9] presented a fully discrete discontinuous Galerkin method for the unsteady Euler equations. Biswas et al.[10] applied the discontinuous Galerkin method in an h–p version in an adaptive-grid environment and considered the issues in using limiters for moments, as well as for parallel implementations.

In the works just described, the integrals that appear in the formulation are evaluated with quadrature formulas. In the present work, the discontinuous Galerkin method is implemented in a form that avoids the use of quadrature formulas. This approach allows the discontinuous Galerkin method to be efficiently implemented for multidimensional elements for which optimal quadrature formulas of the required accuracy are not readily available. In addition, the quadrature-free approach provides a significant reduction in the computational work, even in two dimensions, for which near-optimal quadrature formulas have been developed. This work focuses on the new quadrature-free formulation of the discontinuous Galerkin method and employs only periodic test problems to demonstrate the method. We note that the new formulation introduces no difficulties with regard to the imposition of boundary conditions, and the quadrature-free approach presented can easily handle boundary-condition procedures commonly used in finite volume and finite element methods. A thorough treatment of a variety of boundary conditions applied in the context of the quadrature-free formulation can be found in Ref. 11

In the first section, the discontinuous Galerkin method is described in general. Then, additional motivation for using the quadrature-free approach is given, along with specific details of the formulation. Storage requirements and operation count are also discussed, and the results of a stability analysis are given. The quadrature-free form is shown to significantly reduce the two primary arguments against the method: the high storage requirements and a high operation count. The last section presents numerical results for one- and two-dimensional test problems. A linear scalar equation is used to verify the general properties of the discontinuous Galerkin method for solution expansions up to a degree of 12.

Presented as Paper 96-1683 at the AIAA/CEAS 2nd Aeroacoustics Conference, State College, PA, May 6–8, 1996; received July 12, 1996; revision received Aug. 25, 1997; accepted for publication Sept. 22, 1997.

*Research Scientist, Aerodynamic and Acoustics Methods Branch. Senior Member AIAA.

†Professor, Division of Applied Mathematics.

The nonlinear Burgers' equation is used to demonstrate the shock-capturing capabilities of the method in one space dimension. The method is applied to both scalar advection and the linear Euler equations in two space dimensions to demonstrate the unstructured grid capability.

General Discontinuous Galerkin Method

Consider an arbitrary domain Γ in which the solution is governed by a conservation equation of the form

$$U_t + \nabla \cdot \vec{F} = 0 \tag{1}$$

The discontinuous Galerkin method can be obtained by partitioning the domain onto smaller, nonoverlapping elements Γ_i that cover the domain and then applying a traditional Galerkin method[12] to each element. The Galerkin approach within an element is defined by selecting a finite-dimensional basis set, approximating the solution as an expansion in that basis, and then projecting the governing equation onto each member of that basis set:

$$B \equiv \{b_k, 1 \le k \le N(p,d)\} \tag{2}$$

$$U_{\Gamma_i} \approx V_i \equiv \sum_{j=1}^{N} v_{i,j} b_j \tag{3}$$

$$\int_{\Gamma_i} b_k (V_t + \nabla \cdot \vec{F})_i \, d\Gamma = 0, \qquad 1 \le k \le N(p,d), \quad 1 \le i \le I \tag{4}$$

where I is the total number of elements. The basis set must be constructed from the lower-order terms of a complete and linearly independent set. In the fully discrete approach,[9] the basis set contains both temporal and spatial functions. In the semidiscrete approach, which is used here, the basis set contains only spatial functions, and the solution expansion coefficients $v_{i,j}$ are functions of time. The number of terms in the expansion, $N(p,d)$, depends on the degree of the expansion p and the number of time–space dimensions d that are represented. When the basis functions are polynomials of degree p, the order of accuracy of the method has been proven[1] to be at least $p + \frac{1}{2}$. In practice, however, the order of accuracy of the method is observed in most cases to be $p+1$. Except where the full form is needed for clarity, $N(p,d)$ will be denoted simply as N.

The divergence term is cast in a weak form by applying integration by parts as

$$\int_{\Gamma_i} b_k \sum_{j=1}^{N} (v_{i,j})_t b_j \, d\Gamma - \int_{\Gamma_i} \nabla b_k \cdot \vec{F}_i \, d\Gamma + \int_{\partial\Gamma_i} b_k \vec{F}_i^R \cdot d\vec{s} = 0 \qquad 1 \le k \le N, \quad 1 \le i \le I \tag{5}$$

where $d\vec{s}$ is an outward-pointing, surface-element normal. Because each element has its own approximate solution, the global solution is discontinuous at the edge between elements. This discontinuity is resolved through the use of an approximate Riemann flux vector $\vec{F}^R$. The approximate Riemann flux provides the crucial coupling between elements, as well as the correct upwind bias that is required to ensure stability.

Before discretization, Eq. (5) is represented in terms of coordinates that are local to the commutational element Ω_i:

$$\int_{\Omega_i} b_k \sum_{j=1}^{N} (v_{i,j})_t b_j J_i \, d\Omega - \int_{\Omega_i} \nabla b_k \cdot \boldsymbol{J}_i^{-1} \vec{F}_i J_i \, d\Omega + \int_{\partial\Omega_i} b_k \boldsymbol{J}_i^{-1} \vec{F}_i^R \cdot J_i \, d\vec{s} = 0, \qquad 1 \le k \le N, \quad 1 \le i \le I \tag{6}$$

where $\boldsymbol{J}_i \equiv \partial(x,y,z)/\partial(\xi,\eta,\zeta)$, $J_i = |\boldsymbol{J}_i|$, and (ξ,η,ζ) are the local coordinates within the element.

The first integral is usually written as a mass matrix times a vector that contains the time derivative of the dependent variables: $\boldsymbol{M}_i(\partial \boldsymbol{V}_i/\partial t)$, where

$$\boldsymbol{M}_i = [(m_{k,j})_i], \qquad (m_{k,j})_i \equiv \int_{\Omega_i} b_k b_j J_i \, d\Omega$$

$$\boldsymbol{V}_i \equiv [v_{i,1}, v_{i,2}, \ldots, v_{i,N}]^T$$

In most cases, the mass matrix can be computed and inverted in advance of the main calculation; however, unless further assumptions or constraints are made, a distinct copy of $\boldsymbol{M}_i$ must be stored for each element. Note that an important distinction between the discontinuous Galerkin method and the usual finite element method is that in the discontinuous Galerkin method the mass matrix is local to the generating element. Thus, the resulting N equations can be used to temporally evolve the unknown coefficients of the solution expansion by using simple explicit methods.

Some other advantages of the discontinuous Galerkin method become apparent from a quick inspection of Eq. (6). The solution within an element is not reconstructed by looking to neighboring elements; thus, the accuracy of the method within a given element depends only on the size and shape of that element and on the degree to which the solution is approximated. The accuracy does not depend on the smoothness of the mesh, and the algorithm does not depend on identifying a suitable cluster of neighboring elements. As a consequence, for elements adjacent to boundaries the evaluation of the two volume integral terms is identical to the treatment of elements in the interior of the domain. Furthermore, adjacent elements communicate with one another only through the approximate Riemann flux vector in the element boundary integral term. Thus, each element can be thought of as a separate entity that obtains some boundary data from its neighbors. In light of this observation, clearly the physical boundary conditions are easily imposed either by supplying the approximate Riemann flux with the appropriate external data or by reformulating the flux to impose the desired conditions. A thorough discussion of boundary conditions can be found in Ref. 11.

Throughout the remainder of this discussion, the subscript that identifies the element is omitted unless its use is necessary for clarity. In addition, although the preceding discussion applies to a general basis set, the remaining discussion will assume that the basis functions are polynomials. In particular, the results presented later use a basis set that is constructed from simple monomials, e.g., $B \equiv \{1, \xi, \eta, \xi^2, \xi\eta, \eta^2, \ldots\}$.

Quadrature-Free Approach

In all previous implementations, the spatial integrals are evaluated with quadrature formulas that are appropriate to the element shape and to the required degree of accuracy. Quadrature formulas, especially Gaussian quadrature formulas, are usually the most accurate and efficient means for evaluating integrals. However, this feature is based on the assumption that the data are readily available at the quadrature points, i.e., the data are stored there, which is typical for most finite element and spectral element methods. In the discontinuous Galerkin method, however, the formulation requires the evaluation of both volume and surface integrals, and no single set of N quadrature points exists that can be used to evaluate all integrals to the required accuracy. Thus, the usual practice in the implementation of the discontinuous Galerkin method is to choose a basis set and treat the expansion coefficients as the dependent variables. As a consequence, the evaluation of the volume integral using quadrature formulas requires N operations at each quadrature point simply to obtain the data needed to evaluate the quadrature formula. Because all integrals must be exact for polynomials of degree $2p$, the operation count for the complete evaluation of the volume integral is greater than $2N$ operations for each of the N equations. In contrast, if the basis functions are judiciously chosen such that integrals of products of the basis functions can be evaluated exactly, then the complete volume integral can be evaluated in fewer than $N+1$ operations per equation without the use of quadrature formulas.

An equally important benefit of the quadrature-free approach is that it provides a systematic means for implementing the discontinuous Galerkin method in any number of dimensions using polynomials of any degree. Aside from the use of tensor products of one-dimensional Gaussian formulas, there is no general procedure for obtaining multidimensional quadrature formulas. Optimal quadrature formulas, those that minimize the number of quadrature points and possess desirable properties such as symmetry and low condition number, are not readily available, and the development of such formulas is an active area of research. Thus, most implementations

of the discontinuous Galerkin methodology have been limited to two dimensions using polynomials of degree $p = 2$ or 3.

In the following section, we derive a set of matrices used in the quadrature-free implementation to evaluate the volume and boundary integrals. Furthermore, by placing the restriction that $\boldsymbol{J}$ is constant within each element on the geometry of all but a few isolated elements, these matrices are the same for all elements of a given type. This approach results in a method with low storage requirements and low computational cost that retains the ability to treat unstructured grids in an accurate and robust manner.

To derive the quadrature-free form, the flux vector $\vec{F}$ must first be approximated in terms of the basis set:

$$\vec{F} = \sum_{j=1}^{M} \vec{f}_j b_j, \qquad M \geq N(p, d) \tag{7}$$

If $\vec{F}(U)$ is linear, then $M = N$; however, when $\vec{F}(U)$ is nonlinear, M must be at least $N(p+1, d)$ to obtain the design accuracy of $p+1$.

The flux expansion is trivially obtained when $\vec{F}(U)$ is linear. Similarly, for common test problems such as the nonlinear Burgers' equation, $\vec{F}(U) = \frac{1}{2}U(x)^2$ can be obtained by multiplication so long as triple products of the basis functions are also easily integrated. Such is the case when the basis functions are polynomials in the local coordinates of the computational element. The more complex flux functions, such as those of the Euler equations, can be treated in any of several ways described in Ref. 13 and generally do not introduce any complications for smooth flows.

Flows with nonlinear discontinuities, such as shocks, require careful attention, and research is ongoing in this area.[14] We note, however, that when quadrature formulas are used to evaluate the integral of a nonlinear flux, the higher-order terms inherent in the flux contribute to the result in an undesirable manner. Although quadrature formulas are designed to be exact for polynomials of a specified degree, the contribution of polynomials that are of higher degree is not usually considered in the design of the formulas. For sufficiently smooth solutions, the contribution of the higher-order terms is also of higher order and is negligible; however, the higher-order terms can be quite large in the neighborhood of a captured discontinuity and will contribute significantly to the quadrature result. In approximating the functional form of the flux, we are given direct control over how the higher-order terms are truncated, filtered, or otherwise dealiased.

Volume Integrals

With the flux given by Eq. (7), the volume integral of the flux in Eq. (6) can easily be rewritten as a matrix times a vector in a manner that is similar to the treatment of the mass matrix:

$$\left[\int_{\Omega} \nabla b_k b_j \boldsymbol{J}^{-1} J \, \mathrm{d}\Omega\right] \cdot \begin{bmatrix} \vec{f}_1 \\ \vec{f}_2 \\ \vdots \\ \vec{f}_M \end{bmatrix} \tag{8}$$

Furthermore, under the constraint that $\boldsymbol{J}$ is constant within each element, the terms in the preceding integral can be regrouped as

$$\left[\int_{\Omega} \nabla b_k b_j \, \mathrm{d}\Omega\right] \cdot \begin{bmatrix} \boldsymbol{J}^{-1} J \vec{f}_1 \\ \boldsymbol{J}^{-1} J \vec{f}_2 \\ \vdots \\ \boldsymbol{J}^{-1} J \vec{f}_M \end{bmatrix} \equiv \vec{\boldsymbol{A}} \cdot \vec{\boldsymbol{F}} \tag{9}$$

Similarly, the mass matrix can be redefined as

$$\left[\int_{\Omega} b_k b_j \, \mathrm{d}\Omega\right] JV \equiv JMV \tag{10}$$

Because the integrals now depend only on the shape of the computational element, the matrices $\boldsymbol{M}$ and $\vec{\boldsymbol{A}}$ are the same for all elements of a given type, e.g., a single copy of $\boldsymbol{M}^{-1}\vec{\boldsymbol{A}}$ applies to all triangles. Also, because the evaluation of the matrix components requires only the integration of polynomials over simple linear elements, the matrices can be evaluated exactly. Unique elements, such as those that lie on a curved domain boundary, will require their own distinct copy of $\boldsymbol{M}^{-1}\vec{\boldsymbol{A}}$; however, only a few such elements are likely to be needed. Furthermore, by approximating the boundary curvature as a polynomial (or other easily integrable function), the matrices can still be evaluated exactly.[11]

Boundary Integrals

The boundary integral is partitioned into segments that are associated with the sides of the element $\partial\Omega_j$, such that $\partial\Omega = \cup\partial\Omega_j$. The process of rewriting the integral on any segment as a constant matrix times a vector function of the solution is complicated only by the fact that the solutions on either side of the element are given in terms of different local coordinate systems. This minor difficulty is eliminated by expressing both solutions in terms of a common coordinate $\bar{\xi}$ that is associated with the edge. In two dimensions, for example, let $\xi_j(\bar{\xi})$ and $\eta_j(\bar{\xi})$ for $\bar{\xi}_0 < \bar{\xi} < \bar{\xi}_1$ define the relation between the element and edge coordinates on edge j of an arbitrary element. Then, the solution on the edge in terms of the edge coordinate is simply

$$\bar{V}_j(\bar{\xi}) = \sum_{k=1}^{N(p,d)} v_k b_k[\xi_j(\bar{\xi}), \eta_j(\bar{\xi})] \equiv \sum_{k=1}^{N(p,d-1)} \bar{v}_{j,k} \bar{\xi}^{(k-1)} \tag{11}$$

The coefficients of the edge solution are given explicitly by $\bar{\boldsymbol{V}}_j = \boldsymbol{T}_j \boldsymbol{V}$, where $\boldsymbol{T}_j = [(t_{l,k})_j]$ and $(t_{l,k})_j$ is the coefficient of $\bar{\xi}^l$ in the term $b_k[\xi_j(\bar{\xi}), \eta_j(\bar{\xi})]$.

After $\bar{\boldsymbol{V}}_j$ has been computed on every edge of every element, the flux through an edge can be computed without regard for the type of elements or the orientation of the coordinate system of the elements that border the edge. At each edge, we arbitrarily designate one element to be on the left and the other to be on the right and denote element quantities with the subscripts l and r, respectively. In the present work, the Riemann flux is approximated by a simple Lax–Friedrichs flux of the form

$$J\boldsymbol{J}^{-1}\vec{F}_l^R \cdot \vec{n} \equiv \frac{1}{2}\Big\{\big[J\boldsymbol{J}^{-1}(\vec{F}(\bar{V}_l(\bar{\xi})) + \vec{F}(\bar{V}_r(-\bar{\xi})))\big] \cdot \vec{n} - \alpha(\bar{V}_l(\bar{\xi}) - \bar{V}_r(-\bar{\xi}))\Big\} = \sum_{k=1}^{\bar{N}} \bar{f}_k^R \bar{\xi}^{(k-1)} \tag{12}$$

where $\mathrm{d}\vec{s} = \vec{n}\,\mathrm{d}\bar{\xi}$, $\vec{n}$, points from left to right, α is some smooth positive function that is greater in magnitude than the eigenvalues of the Jacobian of $\frac{1}{2}[J\boldsymbol{J}^{-1}(\vec{F}_l + \vec{F}_r)] \cdot \vec{n}$, and $\bar{N} = N(p, d-1)$; $\vec{F}_l(\bar{V}(\bar{\xi}))$ and $\vec{F}_r(\bar{V}(\bar{\xi}))$ can either be computed directly from the solutions $\bar{V}_l(\bar{\xi})$ and $\bar{V}_r(\bar{\xi})$, respectively, or the volume flux can be mapped to the edge coordinate by the same procedure used to obtain the edge solution.

Finally, the boundary integral is evaluated in the edge frame of reference by expressing b_k in terms of the edge coordinates. The integral on a boundary segment becomes

$$\left[\int_{\partial\Omega_j} (b_k)(\boldsymbol{J}^{-1}\vec{\boldsymbol{F}}^R) \cdot \mathrm{d}\vec{s}\right] = \left[\int_{\bar{\xi}_0}^{\bar{\xi}_1} b_k(\xi_j(\bar{\xi}), \eta_j(\bar{\xi}))\bar{\xi}^l \, \mathrm{d}\bar{\xi}\right] \times \begin{bmatrix} \bar{f}_1^R \\ \bar{f}_2^R \\ \vdots \\ \bar{f}_{\bar{N}}^R \end{bmatrix} \equiv \boldsymbol{B}_j \bar{\boldsymbol{F}}^R \tag{13}$$

The final form of the semidiscrete equation is

$$\boldsymbol{V}_t = \boldsymbol{J}^{-1}\left[\boldsymbol{M}^{-1}\vec{\boldsymbol{A}} \cdot \vec{\boldsymbol{F}} - \sum_{k=1}^{n_e} (\boldsymbol{M}^{-1}\boldsymbol{B}_k \bar{\boldsymbol{F}}_k)\right] \tag{14}$$

where n_e is the number of edges and the matrices $\boldsymbol{M}^{-1}\vec{\boldsymbol{A}}$ and $\boldsymbol{M}^{-1}\boldsymbol{B}_k$ are constant matrices that apply to all elements of a given type. Furthermore, these matrices can be evaluated easily and exactly for any type of element shape.

Computational Effort

The quadrature-free form substantially reduces the computational resources required to implement the discontinuous Galerkin method. To illustrate this point, the work required to evaluate the volume integral of the flux both with and without the use of quadrature formulas is examined. Note that, because the edge integral term is of a lower dimensionality, evaluation of the volume integral dominates the work requirements. Also, because the work required to evaluate a nonlinear flux can vary greatly depending on the specific form of the flux, this analysis is limited to the linear case in which the evaluation of $\vec{F}(U)$ is assumed to require m operations independent of the order of the method. Under these assumptions, the work required to compute the flux vector $\vec{F}$ from the solution vector V is mN, and the work to evaluate the matrix product $M^{-1}\vec{A}\cdot\vec{F}$ is, at most, N^2. All elements employed possess symmetry properties with respect to one or more of the element coordinates that cause many components of $M^{-1}\vec{A}$ to be zero in a predictable pattern. In the case of a triangular element, for instance, over half of the elements are zero. In the case of a square element, which is symmetric with respect to both ξ and η, the ratio is higher. Thus, a conservative estimate of the work to evaluate the volume integral is $N(N+m)/2$.

Under geometric constraints similar to those imposed in the quadrature-free form, the work required to evaluate the volume integral using a quadrature formula with N_q quadrature points is given by the sum of three components: NN_q operations to evaluate the solution at each quadrature point, mN_q operations to compute the flux from the solution at each quadrature point, and another NN_q to evaluate the quadrature formula for each equation. Thus, the total work using quadrature formulas is $N_q(2N+m)$.

If $N_q = N$, the work using quadrature formulas would be four times that of the quadrature-free approach; however, N_q is generally greater then N for relevant values of p and d (the degree of the solution polynomial and the number of time–pace dimensions, respectively). The value of N as a function of degree p and d is given by

$$N(p,d) = \left[\prod_{k=1}^{d}(p+k)\right] \Big/ d!$$

There is no general formula for the number of quadrature points, except for the case in which a tensor product of one-dimensional formulas is used, in which case $N_q = p^d$.

Time Integration and Stability

The solution is advanced in time with a three-stage TVD Runge–Kutta method[15]:

$$W^0 = V^{n-1}$$
$$W^k = \beta_k W^0 + (1-\beta_k)[W^{k-1} + \Delta t R(W^{k-1})], \qquad k = 1,2,3 \quad (15)$$
$$V^n = W^3$$

where $\beta_k = 0, \frac{3}{4}$, and $\frac{1}{3}$ for $k = 1, 2$, and 3, respectively.

Fourier stability analysis has been applied to a generalized K-stage form of Eq. (15), i.e., $k = 1, 2, \ldots, K$, for the one-dimensional linear case of $U_t + aU_x = 0$ to determine the stability limit $\lambda_K \equiv a\Delta t/\Delta x$. Results are given in Table 1 for $K = 1, 2$ and 3 and for $0 \le p \le 11$. The rapid drop in the stability limit as the order of the method is increased would normally be alarming in comparison with stability constraints of explicit finite difference methods. However, if we require that comparisons be made among methods that have the same total number of variables, then the size of the element in the discontinuous Galerkin method would be larger (by a factor of $p+1$ in one dimension) than the mesh size of a comparable finite difference calculation. Thus, most of the drop in the stability limit can be attributed to the definition of Δx. The last column of Table 1 gives $\lambda_K(p+1)$, which gives the discontinuous Galerkin stability limit in a form that facilitates comparison with the stability limit of a finite difference method.

Results

One-Dimensional Test

The one-dimensional version of this method has been tested on the linear and nonlinear problems

$$U_t + aU_x = 0 \quad (16)$$

Table 1 K-stage Runge–Kutta methods $\lambda_K \equiv a\Delta t/\Delta x$ applied to discontinuous Galerkin method of degree p (order $p+1$)

p	λ_1	λ_2	λ_3	$\lambda_3(p+1)$
0	1.0	1.00	1.256	1.256
1	0.001	0.333	0.409	0.818
2	—[a]	0.06	0.209	0.627
3	—[a]	0.02	0.13	0.52
4	—[a]	0.01	0.089	0.445
5	—[a]	0.006	0.066	0.396
6	—[a]	0.004	0.051	0.306
7	—[a]	0.003	0.04	0.32
8	—[a]	0.002	0.033	0.297
9	—[a]	0.002	0.027	0.27
10	—[a]	0.001	0.023	0.253
11	—[a]	0.001	0.02	0.24

[a]Unstable method.

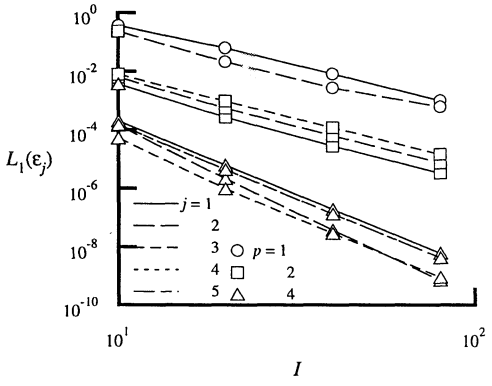


Fig. 1 Convergence of L_1 error for $p = 1$, 2, and 4.

and

$$U_t + \tfrac{1}{2}(U^2)_x = 0 \quad (17)$$

on the domain $0 < x < 1$ with periodic boundary conditions.

A linear problem is solved first with smooth initial conditions $U(0,x) = \frac{1}{2} + \sin(2\pi x)$ to demonstrate the general accuracy properties of the method. The numerical solution is initialized by expanding the initial condition in a Taylor's series about the center of each element. All components of the numerical solution are compared with the Taylor's series of the exact solution after the solution has advected for several periods. The L_n norm of the error in the jth component of the solution is defined as

$$L_n(\varepsilon_j) \equiv \left[\left(\sum_{i=1}^{I} |v_{i,j} - u_{i,j}|^n\right) \Big/ I\right]^{\frac{1}{n}}$$

where $u_{i,j}$ denotes the Taylor coefficient of the exact solution in cell i and I denotes the number of elements. A mesh-refinement study has been performed for $p = 1$–5. The time step was chosen to be sufficiently small such that the error would be dominated by the spatial operator; however, for $p > 2$ the time step varied as $\Delta t \propto (\Delta x)^{(p+1)/3}$ so that the temporal accuracy would be of the same order as the spatial accuracy. Figure 1 shows the L_1 error for each component of the solution for $p = 1$, 2, and 4. The convergence rate of the solution between any two grids a and b is defined by

$$\sigma_n^j \equiv \ell n\left\{\frac{[L_n(\varepsilon_j)]_a}{[L_n(\varepsilon_j)]_b}\right\} \Big/ \ell n\left(\frac{h_a}{h_b}\right) \quad (18)$$

Table 2 gives the convergence rate between the two finest grids in the refinement sequence. Although most cases converge at the design rate of $p+1$, the v_1 term of the $p = 1$ case converges at a rate of approximately 3, which is 1 higher than expected. This faster convergence is fortuitous and occurs only because the basis functions are incidentally orthogonal. Because v_2 is only second order and v_3 is undefined, a solution of degree greater than 1 cannot be recovered at any point other than the element center without departing from the Galerkin framework. Furthermore, although v_1 converges faster than the design order, its error is still considerably larger than that of the formally third-order case ($p = 2$).

Table 2 Convergence rates of $L_1(\varepsilon_j)$ between two finest grids

	j				
p	1	2	3	4	5
1	2.990	2.133			
2	3.064	3.049	3.022		
4	4.99	5.74	5.00	5.00	5.00

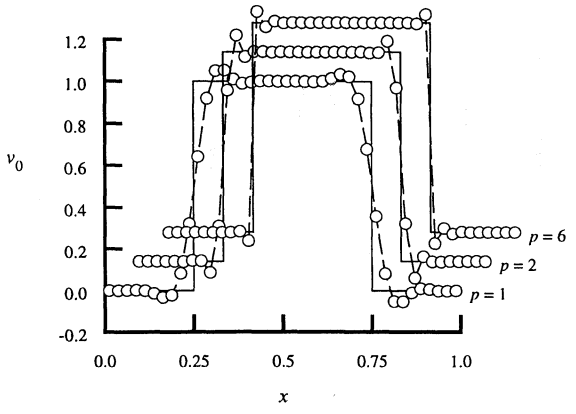


Fig. 2 Solution of linear problem after one time period with $p = 1$, 2, and 6: ——, exact solution, and o– – -, numerical solution.

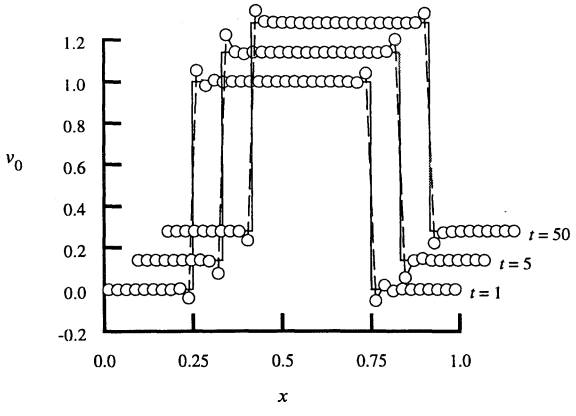


Fig. 3 Solution of linear problem with $p = 6$ after 1, 5, and 50 time periods: ——, exact solution, and o– – -, numerical solution.

In the second test case, the discontinuous Galerkin method is applied to the linear problem with a discontinuous initial solution:

$$U(0,x) = \begin{cases} 1 & \frac{1}{4} < x < \frac{3}{4} \\ 0 & x \le \frac{1}{4}, \quad x \ge \frac{3}{4} \end{cases} \tag{19}$$

Figures 2 and 3 show several results for this case on a grid with 40 elements. Figure 2 shows the solutions after one period for $p = 1$, 2, and 6. Each solution has small overshoots; however, they are confined to the neighborhood of the discontinuity, and the resolution of the discontinuity improves as p increases. Similar results were observed up to $p = 11$ (12th order), which was the highest order tested. Figure 3 shows three solutions for $p = 6$ (seventh order) that have advected for 1, 5, and 50 periods. The overshoots neither grow nor spread in time to any noticeable extent; this behavior is in sharp contrast to that of more traditional methods. A typical finite difference approach, for instance, would tend to smear a contact discontinuity over a region that grows linearly in time.

Next, the discontinuous Galerkin method is applied to a linear test case that was prescribed as part of the Institute for Computer Applications in Science and Engineering/NASA Langley Research Center (ICASE/LaRC) Workshop on Benchmark Problems in Computational Aeroacoustics[16]; these results are compared with the results of a finite difference method also described in Ref. 16. The test case consists of a Gaussian pulse that is advected across a uniform

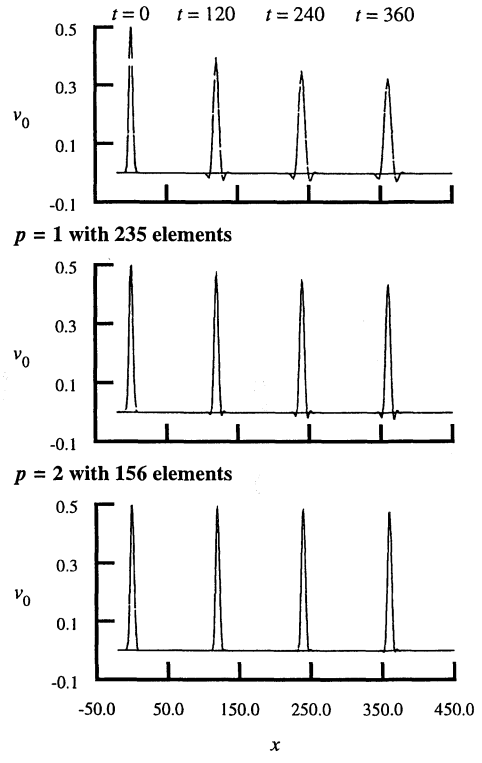


Fig. 4 Discontinuous Galerkin method applied to advection of Gaussian pulse.

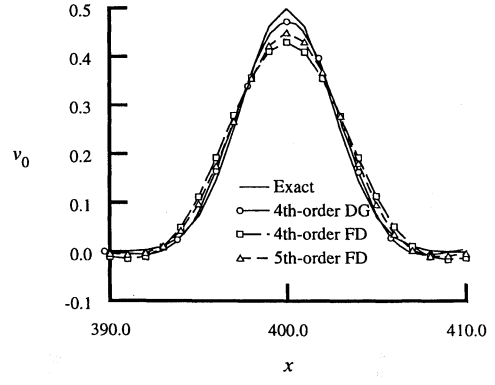


Fig. 5 Comparison of fourth-order discontinuous Galerkin (DG) method with fourth- and fifth-order finite difference (FD) method.

domain. The Gaussian pulse has a half-width of 6 and is initially centered on the origin of a domain that ranges from –20 to 450. Results are shown in Fig. 4 for $p = 1$, 2, and 3; however, as p is increased, the number of elements is decreased such that the total number of variables is approximately 470 (the number of points specified in the workshop). In Fig. 5, the results of the fourth-order discontinuous Galerkin method at $t = 400$ are compared in detail with the results of a fourth- and fifth-order finite difference method. The results obtained with the fourth-order discontinuous Galerkin method with only 117 elements are considerably better than that of either the fourth- or the fifth-order finite difference methods with 470 points. (Note that smooth curves are generated for the results obtained with the discontinuous Galerkin method by evaluating the solution at several points within each element.)

The last one-dimensional test case is a nonlinear problem [Eq. (17)] in which a shock forms from an initially smooth solution. This problem was used not only to demonstrate the robustness of the method but also to investigate the effect of truncating the nonlinear flux at various levels. We expect, based on the formulation, that the nonlinear flux must be expanded to $M = N(p+1, d)$ terms such that the degree of $\nabla b_k \cdot \vec{F}$ is the same as the degree of $b_k V$ to obtain the design rate of convergence of $p + 1$. This expected

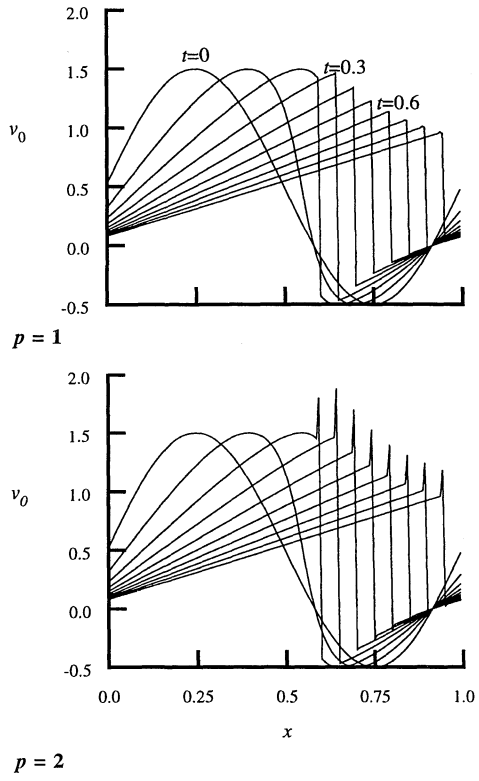


Fig. 6 Solution of nonlinear equation.

convergence property was verified by a mesh-refinement study in which the calculation was stopped just before shock formation.[13] The mesh refinement was performed for several values of p, and the finest grid contained 320 elements. The results indicate that the convergence rate of the L_∞ error is approximately $p + 1$ when $M > N(p, d)$ but drops to approximately p when $M = N(p, d)$.

Figure 6 shows solutions for $p = 1$ and 2 (second and third order) in which the shock has formed and has begun to propagate. In both cases, the solutions are obtained without the use of limiters, added dissipation, or entropy correction terms. However, for the case in which $p = 2$, the nonlinear flux needs to be fully expanded [$M = N(2p, d)$]; otherwise, the solution diverges shortly after shock formation. Although all higher-order cases ($p > 2$) are unstable, the authors know of no other third-order linear method in which the exact same scheme is both stable for shocks and accurate for smooth problems. Total-variation-bounded limiters have been shown to be effective in stabilizing higher-order cases,[2,4] and research is continuing in this area.[14]

Two-Dimensional Test

The discontinuous Galerkin method is applied to the scalar advection equation in two dimensions to demonstrate its robust treatment of unstructured grids. The test problem is given by

$$U_t + aU_x + bU_y = 0$$
$$U(0, x, y) = [\sin(\pi x)\sin(\pi y)]^4 \tag{20}$$

defined on the periodic domain $0 < x, y < 1$. The approximate solution V_i is initialized from the Taylor's expansion of the exact initial condition. The baseline case is chosen to be a uniform Cartesian grid that is triangulated in a regular manner, as shown in Fig. 7. Figure 8 shows the L_1 error in the v_1 component of the solution. As in the one-dimensional case, the time step was small so that the spatial error was dominant, and for $p > 2$ the time step was proportional to $(\Delta x)^{(p+1)/3}$. In mesh-refinement studies, the first grid in the sequence is coarsened as the order of the method is increased, so that the total number of variables is roughly the same. The abscissa in Fig. 8 is the square root of the total number of variables, which facilitates comparison with a fourth-order finite difference method that has a five-point centered stencil. The higher-order convergence of the $p = 1$ case that was observed in one dimension is

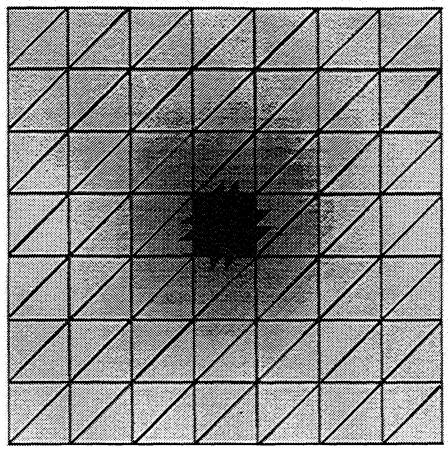


Fig. 7 Triangulated grid and solution of scalar advection problem.

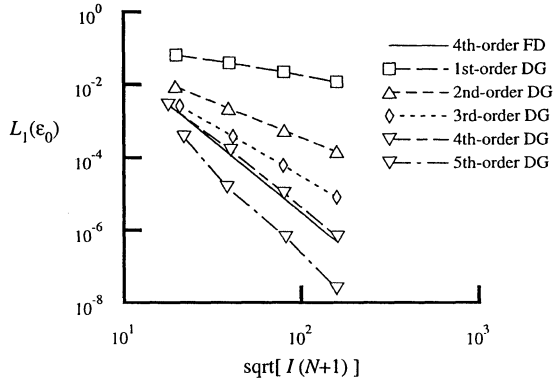
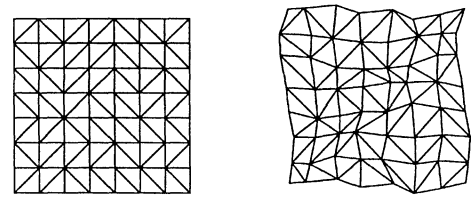
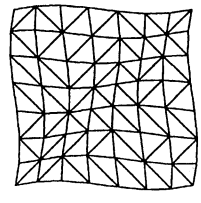


Fig. 8 Convergence of solution for scalar advection on unstructured grid for various orders of accuracy.

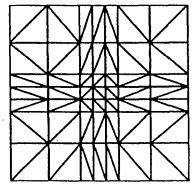


a) Random triangulation

c) Random perturbation of 20% of the mean cell spacing



b) Smoothly clustered toward diagonal



d) Discontinuous mesh variation

Fig. 9 Variation on baseline unstructured grid.

not observed in the two-dimensional case. The accuracies of the fourth-order discontinuous Galerkin and finite difference methods are quite similar.

One of the major motivations for pursuing a discontinuous Galerkin method is its ability to maintain accuracy for complex geometries. Here, the baseline grid is altered in several ways to test and demonstrate this capability. Figure 9 shows four of the variations that were tested. In the first case (Fig. 9a), grid A is uniform as in the baseline case, but the triangulation has been performed randomly. In the second case (Fig. 9b), grid B is generated from the grid of Fig. 9a (grid A) by smoothly clustering the grid toward a diagonal. The grid of Fig. 9c (grid C) is generated from the grid of Fig. 9b (grid B) by randomly perturbing each grid point by an amount that is less than

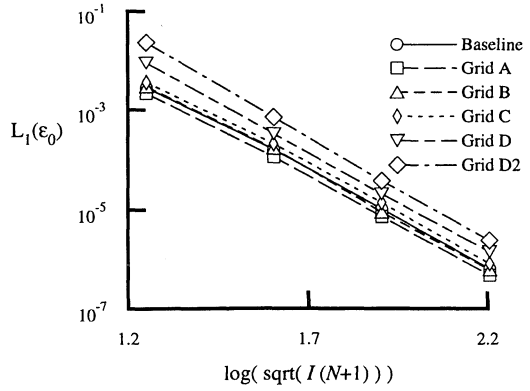


Fig. 10 Convergence of solution for scalar advection for fourth-order method on variations of baseline unstructured grid.

20% of the average mesh size. In the last case (Fig. 9d), the grid (D) is generated from the baseline by imposing a piecewise-constant mesh spacing that places half of the points in a narrow band around the axis.

In most cases, the measured error was insensitive to the grid modification or the direction of propagation, i.e., the value of a and b. Figure 10 gives the L_1 error for $p = 3$ (fourth order) on each of the grids shown in Fig. 9. Results for other values of p were similar. The results in Fig. 10 are for $a = 1$ and $b = 0$, except for case D2, where a and $b = 1$. The slight increase in error in case D2 is attributed more to the increase in mesh size along the propagation path than to the discontinuous manner in which the mesh size changes.

In the last test case, the discontinuous Galerkin method is applied to another problem prescribed as part of the ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics.[16] The linear Euler equations are solved on a square domain of dimensions $-100 < x, y < 100$ with initial conditions that place a compact acoustic source at $x = y = 0$ and a convecting disturbance at $x = 67$, $y = 0$. Here, the equations have been recast in a form that emphasizes the decoupling of the convection terms from the acoustic terms that occurs in this linear system:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{E}}{\partial x} + \frac{\partial \boldsymbol{F}}{\partial y} = 0 \tag{21}$$

where

$$\boldsymbol{U} = \begin{bmatrix} \rho - P \\ p \\ u \\ v \end{bmatrix}, \qquad \boldsymbol{E} = \begin{bmatrix} M_x(\rho - P) \\ M_x P + u \\ M_x u + P \\ M_x v \end{bmatrix}$$

$$\boldsymbol{F} = \begin{bmatrix} M_y(\rho - P) \\ M_y P + v \\ M_y u \\ M_y v + P \end{bmatrix}$$

$M_x = 0.5$, $M_y = 0$, and

$$(\rho - P)(0, x, y) = 0.1 \exp\left[(-\ell_n(2))\frac{(x - 67)^2 + y^2}{25}\right]$$

$$P(0, x, y) = \exp\left[(-\ell_n(2))\left(\frac{x^2 + y^2}{9}\right)\right]$$

$$u(0, x, y) = 0.04xP(0, x, y)$$

$$v(0, x, y) = 0.04yP(0, x, y)$$

The Workshop, which targeted finite difference methods, prescribed a grid of 200 × 200. In the present calculation, a uniform $n \times n$ Cartesian grid is used that has been randomly triangulated (as shown in Fig. 9a) and in which the number of elements is chosen as a function of the degree p such that the total number of unknowns equals approximately 200^2. Computations were performed for $p = 0$, 1, 2, and 3 (first, second, third, and fourth order) with $n = 141$, 81, 57, and 44, respectively. Figure 11 shows P at $t = 40$ for the $p = 3$ case. The wave fronts remain smooth and cylindrical in spite of the fact that the initial disturbance was smaller than the element size. A more quantitative comparison is shown in Fig. 12. The pressure P is plotted on the $x = 0$, $t = 40$ line for $p = 0$–3 at the resolution given earlier. Also shown is a fine-grid solution with $p = 3$ and $n = 132$ and the solution from a fifth-order finite difference method[16] on a 200 × 200 grid. The enlargement of the right peak shows that all solutions of third-order accuracy or better yield similar results.

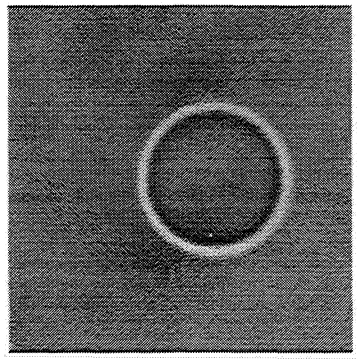


Fig. 11 Acoustic wave modeled by linear Euler equations: P at $t = 40$ for $p = 3$.

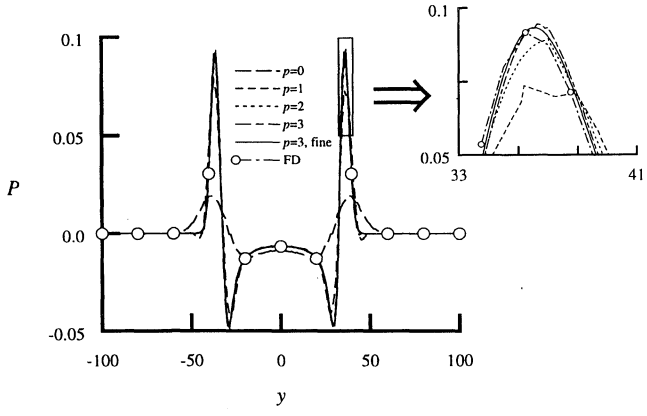


Fig. 12 Solution of linear Euler equations: pressure on $x = 0$ at $t = 40$.

Conclusions

A quadrature-free form of the discontinuous Galerkin method has been formulated for the hyperbolic conservation laws. This approach provides a systematic means to implement the discontinuous Galerkin method to any desired order of accuracy while reducing the operation count to about one-fourth of that of a quadrature-based method. The method is well suited to both unstructured and structured grids and has been tested on several one- and two-dimensional problems to demonstrate its accuracy and robustness. On smooth meshes, the accuracy of the discontinuous Galerkin method is comparable to or better than traditional high-order finite difference methods. Contact discontinuities are advected without the usual diffusion effect, and nonlinear discontinuities (shocks) are propagated by second- and third-order methods without the use of limiters. On two-dimensional unstructured grids, random and discontinuous mesh variations had little effect on the error and no effect on the convergence of the error.

Acknowledgments

The research of the second author was supported by NASA Langley Research Center Grant NAG-1-1145 and Contract NAS1-19480 while the author was in residence at the Institute for Computer Applications in Science and Engineering and by Army Research Office Grant DAAHO4-94-G-0205 and National Science Foundation Grant DMS-9500814.

References

[1]Johnson, C., and Pitkärata, J., "An Analysis of the Discontinuous Galerkin Method for a Scalar Hyperbolic Equation," *Mathematics of Computation*, Vol. 46, No. 176, 1986, pp. 1–26.

[2]Cockburn, B., and Shu, C.-W., "TVB Runge–Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework," *Mathematics of Computation*, Vol. 52, No. 186, 1989, pp. 411–435.

[3]Cockburn, B., Lin, S. Y., and Shu, C.-W., "TVB Runge–Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws III: One Dimensional Systems," *Journal of Computational Physics*, Vol. 84, No. 1, 1989, pp. 90–113.

[4]Cockburn, B., Hou, S., and Shu, C.-W., "The Runge–Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws IV: The Multidimensional Case," *Mathematics of Computation*, Vol. 54, No. 190, 1990, pp. 545–581.

[5]Jiang, G., and Shu, C.-W., "On Cell Entropy Inequality for Discontinuous Galerkin Methods," *Mathematics of Computation*, Vol. 62, No. 206, 1994, pp. 531–538.

[6]Halt, D. W., and Agarwal, R. K., "Compact Higher-Order Characteristic-Based Euler Solver for Unstructured Grids," *AIAA Journal*, Vol. 30, No. 8, 1992, pp. 1993–1999.

[7]Bassi, F., and Rebay, S., "Accurate 2D Euler Computations by Means of a High-Order Discontinuous Finite Element Method," *Lecture Notes in Physics*, Springer, New York, 1995, pp. 234–240.

[8]Bassi, F., and Rebay, S., "Discontinuous Finite Element High Order Accurate Numerical Solution of the Compressible Navier–Stokes Equations," *Numerical Methods for Fluid Dynamics*, Clarendon, Oxford, England, UK, 1995, pp. 295–302.

[9]Lowrie, R. B., Roe, P. L., and van Leer, B., "A Space-Time Discontinuous Galerkin Method for the Time-Accurate Numerical Solution of Hyperbolic Conservation Laws," AIAA Paper 95-1658, June 1995.

[10]Biswas, R., Devine, K. D., and Flaherty, J., "Parallel, Adaptive Finite Element Methods For Conservation Laws," *Applied Numerical Mathematics*, Vol. 14, No. 1–3, 1994, pp. 255–283.

[11]Atkins, H. L., "Continued Development of the Discontinuous Galerkin Method for Computational Aeroacoustic Applications," AIAA Paper 97-1581, May 1997.

[12]Fletcher, C. A. J., *Computational Galerkin Methods*, Springer–Verlag, New York, 1984.

[13]Atkins, H. L., and Shu, C.-W., "Quadrature-Free Implementation of the Discontinuous Galerkin Method for Hyperbolic Equations," AIAA Paper 96-1683, May 1996.

[14]Atkins, H. L., "Local Analysis of Shock Capturing Using Discontinuous Galerkin Methodology," AIAA Paper 97-2032, June 1997.

[15]Shu, C.-W., and Osher, S., "Efficient Implementation of Essentially Non-Oscillatory Shock-Capturing Schemes," *Journal of Computational Physics*, Vol. 77, No. 77, 1988, pp. 439–471.

[16]Atkins, H. L., "Application of Essentially Nonoscillatory Methods to Aeroacoustic Flow Problems," *Proceedings of ICASE/LaRC Workshop on Benchmark Problems in Computational Aeroacoustics*, edited by J. C. Hardin, J. R. Ristorcelli, and C. K. W. Tam, NASA CP-3300 1995, pp. 15–26.

S. Glegg
Associate Editor