AIAA JOURNAL
Vol. 44, No. 7, July 2006

Use of Finite Element Methods in Frequency Domain Aeroacoustics

Preetham P. Rao* and Philip J. Morris†
Pennsylvania State University, University Park, Pennsylvania 16802

The linearized Euler equations are used widely in many aerodynamic noise prediction schemes. The linearized Euler equations also support instability waves, which can obscure the acoustic solution. An approach to suppress the instability waves is to solve the linearized Euler equations in the frequency domain. Also, the frequency domain approach is far less expensive compared to the time domain approach in terms of the computational time though only a single frequency can be computed at one time. The application and comparison of two unstructured grid methods for the solution of aeroacoustic problems in the frequency domain are presented. Algorithms are developed using the discontinuous Galerkin and the streamline upwind Petrov–Galerkin finite element methods in two spatial dimensions on conforming triangular meshes using nodal polynomial basis sets. Unlike other existing frequency domain solvers, no assumption is made regarding the nature of the mean flow. The application of the two methods to a two-dimensional benchmark problem involving the suppression of instability waves is presented. The two methods are compared in terms of the computational cost. It is concluded that the discontinuous Galerkin method is prohibitively more expensive compared to the streamline upwind Petrov–Galerkin method for aeroacoustic applications in the frequency domain using conforming meshes and nodal polynomials.

Nomenclature

$\boldsymbol{A}$	=	coefficient matrix
$\boldsymbol{B}$	=	elemental edge flux matrix
$\boldsymbol{C}$	=	reaction matrix
h	=	size of Ω^e
$\boldsymbol{I}$	=	identity matrix
j, k, r	=	indices
$\boldsymbol{K}$	=	stiffness matrix
M	=	Mach number of mean flow
$\boldsymbol{M}$	=	mass matrix
N	=	size of basis set
N_{el}	=	number of nonoverlapping elements in Ω
N_{nz}	=	number of nonzeros in global matrix
p	=	acoustic pressure
p_d	=	degree of basis set polynomials
q	=	dependent variable(s)
$\mathcal{R}$	=	universal gas constant
s	=	acoustic source term
t	=	time
u, v	=	x and y acoustic velocity components
x, y	=	coordinates
x_b	=	coordinate in buffer layer
γ	=	ratio of specific heat coefficients
κ, ε	=	positive constants in damping function σ
λ	=	spectral radius of $\boldsymbol{A}$
ρ	=	acoustic density
σ	=	damping function in buffer layer
τ	=	stability parameter in streamline upwind Petrov–Galerkin formulation
ψ	=	basis set polynomial
Ω	=	bounded computational domain
ω	=	frequency of acoustic solution
$\otimes$	=	Kronecker product, defined in Appendix B
$\odot$	=	row multiple, defined in Appendix B

Superscripts

e	=	value or approximation of a quantity in Ω^e
NE	=	value or approximation of a quantity in the neighboring element
$\sim$	=	spatial component of acoustic solution

Presented as Paper 2004-2962 at the AIAA/CEAS 10th Aeroacoustics Conference, Manchester, England, United Kingdom, 10–12 May 2004; received 19 August 2004; revision received 28 November 2005; accepted for publication 19 January 2006.

*Graduate Research Assistant, Department of Aerospace Engineering. Member AIAA.

†Boeing/A. D. Welliver Professor, Department of Aerospace Engineering. Fellow AIAA.

I. Introduction

AERODYNAMIC noise in a turbulent flow originates from the unsteady motion of turbulent structures in the flow, usually in a shear layer or near a solid surface. The sound generated then propagates to the far field through the nonuniformly moving air. Computational fluid dynamics codes based on some form of turbulence modeling can be used to characterize the sources of noise in the flowfield. Because acoustic perturbations are small, linear methods can be used to simulate the propagation of sound to far-field observer locations. One such method involves the solution of the linearized Euler equations (LEE). The LEE describe the propagation of sound waves in a nonuniform mean flow.

The LEE also support spatially growing solutions for certain source frequencies. However, these instabilities, which are known as the Kelvin–Helmholtz instabilities, are limited in a real flow by the nonlinear effects present in the full governing equations, namely, the Navier–Stokes equations. In other words, the instability wave solutions are a consequence of the simplification of the full equations of motion. Often, the acoustic solution is obscured by the nonphysical instability solution. Various techniques have been attempted by several researchers to suppress the instability wave solution in the time domain solution. These techniques range from assuming zero gradient in the mean flow to numerical filtering of the instability waves. A detailed discussion of the techniques is given by Agarwal et al.[1] In the same study, an approach to suppress the instability waves was presented, that of solving the LEE in the frequency domain. In the frequency domain, a harmonic response is assumed for the acoustic variables. The frequency of response is chosen as the frequency of oscillation of the source. The transformation to the frequency domain modifies the originally hyperbolic LEE to an elliptic system of equations. Because the boundary conditions affect

the solution at every point inside the computational domain for an elliptic system, the algebraic system of equations obtained after numerical discretization needs to be assembled and solved globally. The acoustic solution is reconstructed easily in the time domain by an inverse Fourier transform. Because a global solution is obtained in a single-matrix inversion process, the frequency domain approach is far less time consuming compared to a time domain solution. However, the frequency domain solution is limited to a single frequency of excitation. The comparison of the frequency and time domain solution approaches for practical aeroacoustic applications is a relatively new area of research.

The use of iterative techniques to solve the resulting global matrix in the frequency domain approach has been discussed by Agarwal.[2] It has been shown that the use of an iterative technique to solve the global matrix is usually equivalent to a pseudo-time-marching method and, hence, produces an instability wave solution. Hence, the solution of the global matrix needs to be sought using direct methods such as Gaussian elimination or lower–upper (LU) decomposition techniques to suppress the instability waves. It is possible that other iterative techniques could be used as long as they do not mimic a time-marching procedure.

Several researchers have attempted to solve the linearized Euler equations in the frequency domain.[1] However, in all such implementations, some assumption has been made regarding the mean flow gradients, or the equations themselves have been altered to obtain a Green's function for the resulting set of equations. For example, the assumption of no gradients in the direction of the mean flow allows the reduction of the LEE to Lilley's equation, for which an analytical solution can be found. Agarwal et al.[1] solved the LEE in the frequency domain using a second-order finite difference method. The LEE were, however, reduced to a single, Lilley's equation. A frequency domain solution of the LEE for the acoustic propagation and scattering of jet noise around an aircraft was sought by Xu et al.[3] They did not reduce the LEE to an equation with an analytical solution. However, the mean flow around the aircraft was assumed to be irrotational, so that the momentum equation reduced to an algebraic relation between the density and the velocity potential. Hall[4] developed a finite element method (FEM) for computing the aeroacoustic response of turbomachinery blades in the frequency domain. The descritization was developed for the full potential equations of flow in two spatial dimensions, with an assumption of a uniform mean flow. The resulting numerical equations were solved using LU decomposition. The development of a frequency domain solver for the case of a general mean flow configuration is still an issue that needs much attention because, in realistic applications, simplifying assumptions cannot be made. A three-dimensional block structured LEE solver, LINFLUX, was developed by Montgomery and Verdon.[5] The solver used the finite difference method (FDM). Several aeroacoustic and aeroelastic problems for a turbomachinery cascade were solved. LINFLUX allowed for a spatially nonuniform mean flow within the LEE.

Numerous practical applications involving complex geometry, such as aerodynamic noise prediction from internal mixer configurations in a jet engine or fan exhaust noise, demand the use of unstructured grids. It is very difficult and expensive to apply the FDM on unstructured grids, and FEM is a natural alternative. The primary goal of the present work is the initial development and demonstration of a finite element frequency domain solver of the full LEE for a general mean flow configuration. The application of two versions of the FEM to the solution of the LEE in the frequency domain is studied in two spatial dimensions: the discontinuous Galerkin (DG) and the continuous streamline upwind Petrov–Galerkin (SUPG) methods. The problem considered is a benchmark problem involving the shear-layer instabilities in a subsonic jet flow in two dimensions from the workshop on benchmark problems.[6]

The DG method was originally developed for hyperbolic conservation laws.[7] In recent years, it has been applied to elliptic equations.[8] Later, a comprehensive discussion of the literature on the DG methods applied to elliptic equations was provided by Arnold et al.[9] The basic principle of the DG method lies in allowing the solution to be discontinuous across each element in the computational domain. The discretized Galerkin equation is solved locally in each element. The communication between the elements is achieved by a calculation of the Riemann flux across the element boundaries. A detailed discussion of various aspects of the implementation of the DG method for problems in aeroacoustics is given by Rao.[10] Although initially regarded as computationally expensive, later developments by Atkins and Shu[11] and Rao and Morris[12] have made the DG method more efficient in terms of computational cost. The solution process using the DG method for a hyperbolic problem involves marching in time, and at each time step the domain is swept element after element. Separate linear systems are solved for each element. However, for an elliptic problem, the solution is globally dependent, and, hence, an assembly of elements is necessary. In the DG method, because each element has a separate set of variables, the size of the resulting global matrix is large compared to the size of an equivalent matrix obtained using a continuous Galerkin FEM or FDM. In this paper, it is shown in Sec. V that the DG method requires a very large usage of memory compared to a continuous Galerkin method when an assembly of elements is performed.

As an alternative to the memory-expensive DG method for frequency domain problems, the streamline upwind Petrov–Galerkin (SUPG) method is considered. It is well known that the conventional Galerkin methods are unstable for convection-dominated problems (see Ref. 13). The SUPG method was developed by stabilizing the conventional Galerkin method by the addition of an artificial dissipation function to the trial space functions of the Galerkin formulation (see Refs. 14 and 15). The artificial dissipation is controlled with a stabilizing parameter, usually denoted as τ. The stabilizing parameter is determined based on several factors, including the element size and the convection velocity.[13] The choice of the stabilizing parameter has been the subject of many research studies.[16–18] A consistent methodology for the design of the stabilizing parameter is yet to be established. The methods used by many of the researchers are either ad hoc or involve very complex calculations for specific problems. Several researchers[19–21] have already applied the SUPG method to the compressible Navier–Stokes and Euler equations. In this paper, the application of the SUPG method to the linearized Euler equations in two spatial dimensions is considered.

The remainder of this paper is organized as follows. In Sec. II, the Galerkin method is applied to the LEE in the frequency domain in two dimensions. This is followed by a note on the application of the DG method in Sec. III. An outline of the methodology for the assembly of equations is presented. The solution using the DG method for the plane jet benchmark problem[6] is presented in Sec. III.C. A comparison of the DG method with the continuous Galerkin method in terms of the computational expense is carried out in Sec. IV, particularly for applications in frequency domain aeroacoustics. Then, in Sec. V, a stabilized continuous Galerkin method is used to seek the suppression of the instability wave solution. Finally, the results are summarized in Sec. VI.

II. Frequency Domain Formulation

Consider the LEE in two dimensions, written in nondimensional, conservation form, in Cartesian coordinates:

$$\frac{\partial \boldsymbol{q}}{\partial t} + \frac{\partial \boldsymbol{A}_r \boldsymbol{q}}{\partial x_r} + \boldsymbol{C}\boldsymbol{q} = \boldsymbol{s}, \qquad \text{for} \qquad r = 1, 2 \tag{1}$$

with $x_1 \equiv x$ and $x_2 \equiv y$,

$$\boldsymbol{q} = \begin{Bmatrix} \rho \\ u \\ v \\ p \end{Bmatrix}, \qquad \boldsymbol{A}_1 = \begin{bmatrix} M_x & \rho_0 & 0 & 0 \\ 0 & M_x & 0 & 1/\rho_0 \\ 0 & 0 & M_x & 0 \\ 0 & \gamma P_0 & 0 & M_x \end{bmatrix}$$

$$\boldsymbol{A}_2 = \begin{bmatrix} M_y & 0 & \rho_0 & 0 \\ 0 & M_y & 0 & 0 \\ 0 & 0 & M_y & 1/\rho_0 \\ 0 & 0 & \gamma P_0 & M_y \end{bmatrix}$$

$$\boldsymbol{C} = \begin{bmatrix} 0 & 0 \\ \frac{1}{\rho_0}\left(M_x\frac{\partial M_x}{\partial x} + M_y\frac{\partial M_x}{\partial y}\right) & -\frac{\partial M_y}{\partial y} \\ \frac{1}{\rho_0}\left(M_x\frac{\partial M_y}{\partial x} + M_y\frac{\partial M_y}{\partial y}\right) & \frac{\partial M_y}{\partial x} \\ 0 & (1-\gamma)\frac{\partial P_0}{\partial x} \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 \\ \frac{\partial M_x}{\partial y} & -\frac{\partial (1/\rho_0)}{\partial x} \\ -\frac{\partial M_x}{\partial x} & -\frac{\partial (1/\rho_0)}{\partial y} \\ (1-\gamma)\frac{\partial P_0}{\partial y} & (\gamma-1)\left(\frac{\partial M_x}{\partial x} + \frac{\partial M_y}{\partial y}\right) \end{bmatrix} \tag{2}$$

and $\boldsymbol{s}$, a vector of time-varying acoustic source terms. Here, ρ_0, M_x, M_y, and P_0 are the values of the mean density, velocity components, and pressure, respectively, nondimensionalized using appropriate reference values. The mean flow quantities can be spatially varying. $\boldsymbol{A}_r$ denotes the coefficient matrices. The matrix $\boldsymbol{C}$, made up of the derivatives of the mean flow quantities, is called the reaction matrix.

In the frequency domain approach, a time harmonic response is assumed for the acoustic variables of the form

$$\boldsymbol{q} = \tilde{\boldsymbol{q}}\exp(-i\omega t)$$

with the frequency of response ω equal to the frequency of oscillation of the source. A tilde represents the spatial variation of the variables. Under this assumption, the linearized Euler equations reduce to

$$-i\omega\tilde{\boldsymbol{q}} + \frac{\partial \boldsymbol{A}_r\tilde{\boldsymbol{q}}}{\partial x_r} + \boldsymbol{C}\tilde{\boldsymbol{q}} = \tilde{\boldsymbol{s}} \qquad \text{for} \qquad r = 1, 2 \tag{3}$$

Let the computational domain Ω be divided into N_{el} number of nonoverlapping elements Ω^e that span the entire domain. In this study, the two-dimensional element Ω^e is taken to be an unstructured triangle with straight edges. Let $\tilde{\boldsymbol{q}}^e$ represent an approximation to the variable $\tilde{\boldsymbol{q}}$ in Ω^e using functions of a basis set ψ_j as

$$\tilde{\boldsymbol{q}} \approx \sum_{j=1}^{N} \tilde{\boldsymbol{q}}_j^e \psi_j \tag{4}$$

The size of the basis set N is a function of the degree of the basis polynomials p_d and the number of spatial dimensions. Here, $\tilde{\boldsymbol{q}}^e$ is a $4N$ vector of the nodal values for all variables in $\tilde{\boldsymbol{q}}$,

$$\tilde{\boldsymbol{q}}^e = \left\{\tilde{\rho}_j^e, \tilde{u}_j^e, \tilde{v}_j^e, \tilde{p}_j^e\right\}^T, \qquad \text{for} \qquad j = 1, \ldots, N \tag{5}$$

Similarly, let $\tilde{\boldsymbol{s}}^e$ be the vector of the nodal acoustic source values for each equation in Eqs. (3). In the present study, the basis set polynomials are chosen as Lagrange interpolation polynomials. The use of Lagrange polynomials reduces the computational cost as well as the complexity in the algorithm, as noted by Rao and Morris.[12] When Lagrange polynomials are used, the flux expressions are also easily interpolated. For example, the approximation for the term $M_x\tilde{u}$ can be written as

$$M_x\tilde{u} \approx \sum_{j=1}^{N} \left(M_{xj}^e \tilde{u}_j^e\right)\psi_j \tag{6}$$

with M_{xj}^e being the values of M_x at the nodes of the element. The application of the traditional Galerkin method to Eq. (3) using Eq. (4) yields, for $k = 1, \ldots, N$,

$$\int_{\Omega^e} \psi_k \left(-i\omega\tilde{\boldsymbol{q}}^e + \frac{\partial \boldsymbol{A}_r\tilde{\boldsymbol{q}}^e}{\partial x_r} + \boldsymbol{C}\tilde{\boldsymbol{q}}^e = \tilde{\boldsymbol{s}}^e\right) \mathrm{d}\Omega^e = 0 \tag{7}$$

III. DG Method

In the DG approach, Eq. (7) is integrated by parts, discretized, and solved within each element. The basis polynomials in Eq. (4) are defined locally in the element. A brief discussion of the application of the DG method following a Galerkin formulation similar to Eq. (7) is provided in Appendix A for a scalar elliptic equation. The discretized equation following the DG approach within an element for Eq. (3) is written as

$$\boldsymbol{P}^e\tilde{\boldsymbol{q}}^e + \sum_{j=1}^{3} \boldsymbol{P}^{\text{NE}j}\tilde{\boldsymbol{q}}^{\text{NE}j} = \boldsymbol{f}^e \tag{8}$$

with

$$\boldsymbol{P}^e = \left[-i\omega\boldsymbol{I}\otimes\boldsymbol{M}^e + \boldsymbol{A}_r^e \odot \boldsymbol{K}_r^e + \boldsymbol{C}^e \odot \boldsymbol{M}^e\right] + \frac{1}{2}\sum_{j=1}^{3}\left[\left[\boldsymbol{A}_r^e\boldsymbol{n}_r^j\right]\odot\boldsymbol{B}^{ej} + \alpha\boldsymbol{I}\otimes\boldsymbol{B}^{ej}\right] \tag{9}$$

$$\boldsymbol{P}^{\text{NE}j} = \frac{1}{2}\sum_{j=1}^{3}\left[\left[\boldsymbol{A}_r^{\text{NE}}\boldsymbol{n}_r^j\right]\odot\boldsymbol{B}^{ej} - \alpha\boldsymbol{I}\otimes\boldsymbol{B}^{ej}\right] \tag{10}$$

$$\boldsymbol{f}^e = \left[\boldsymbol{I}\otimes\boldsymbol{M}^e\right]\tilde{\boldsymbol{s}}^e \tag{11}$$

The superscript NEj denotes the values from the neighbor element on side j. The expression for the mass matrix $\boldsymbol{M}^e$ and the stiffness matrices $\boldsymbol{K}_r^e$ are given in in Appendix A. The elemental edge matrix $\boldsymbol{B}^{ej}$ for a given side j of the triangular element is also given in Appendix A. The symbols $\otimes$ and $\odot$ are defined in Appendix B.

The elemental equations need to be assembled to form a global matrix. In the global assembly, the only connection between the elements is through the elemental boundary matrices. Because the solution in each element is discontinuous, each element has its own set of $4N$ nodal variables, for the nodal values of the four variables in $\tilde{\boldsymbol{q}}^e$. There are a total of $4N \times N_{\text{el}}$ variables in the computational domain. The structure of the global matrix is shown in Fig. 1. The global matrix is made of smaller block matrices. The numerical equation from each element occupies a set of $4N$ rows. The diagonal block matrix in a set of rows for an element corresponds to the contribution from the element. There are three more blocks in the same set of rows, from the three neighbors surrounding the element.

A. Boundary Conditions

In this study, a buffer-zone-type boundary condition is implemented. The physical domain of a problem is surrounded by a buffer zone of sufficient width. The acoustic amplitudes are damped by adding an imaginary component to the frequency only in the buffer zone, similar to the method used by Agarwal et al.[1] In the buffer zone,

$$\omega \rightarrow \omega[1 + i\sigma(x, y)] \tag{12}$$

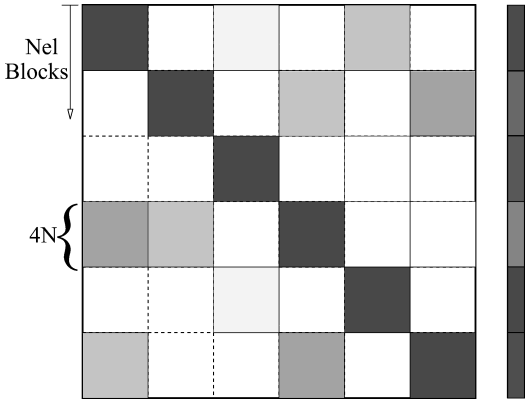


Fig. 1 Structure of global matrix and vector of variables in frequency domain DG method.

The damping function σ is varied smoothly from zero at the beginning to a maximum value at the end of the buffer zone. The values of the acoustic variables are assumed to be zero on the boundary of the computational domain. For a one-dimensional problem, the damping function is taken as

$$\sigma(x) = \varepsilon\left[1 - \exp\left(\kappa x_b^2\right)\right] / [1 - \exp(\kappa)] \tag{13}$$

with x_b being the coordinate in the buffer region normalized by the width of the region and where ε and κ are taken as positive constants.

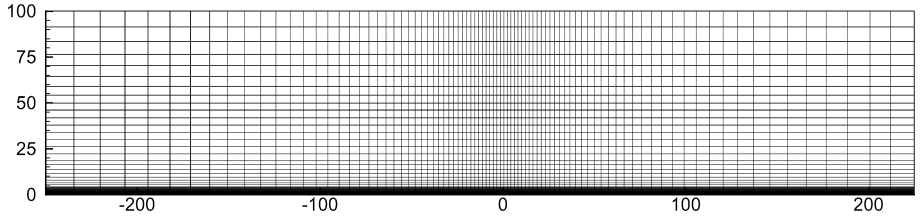
B. Direct Solver

In the present work, a sparse LU decomposition solver routine from MATLAB®[22] is used to solve algebraic systems of reasonably small sizes, of the order of a few tens of thousand rows. For algebraic systems of a larger size, a parallel solver called SuperLU[23] is used. The size of the global matrix is stated from this point in terms the number of rows. SuperLU is an open source sparse matrix LU decomposition solver with message-passing-interface-based parallel implementation written in C. It has been extensively tested on various parallel platforms. The only constraint with the SuperLU solver is that the sparse matrix must be written in the Harwell–Boeing format. The alternative is to modify the solver's source code itself, but this is a tedious process that, in turn, would consume more time and effort. In the present work, the elements of the global matrix are written out in the coordinate-storage format, and then a modified MATLAB routine from Li[24] is used to write out the global matrix in the Harwell–Boeing format.

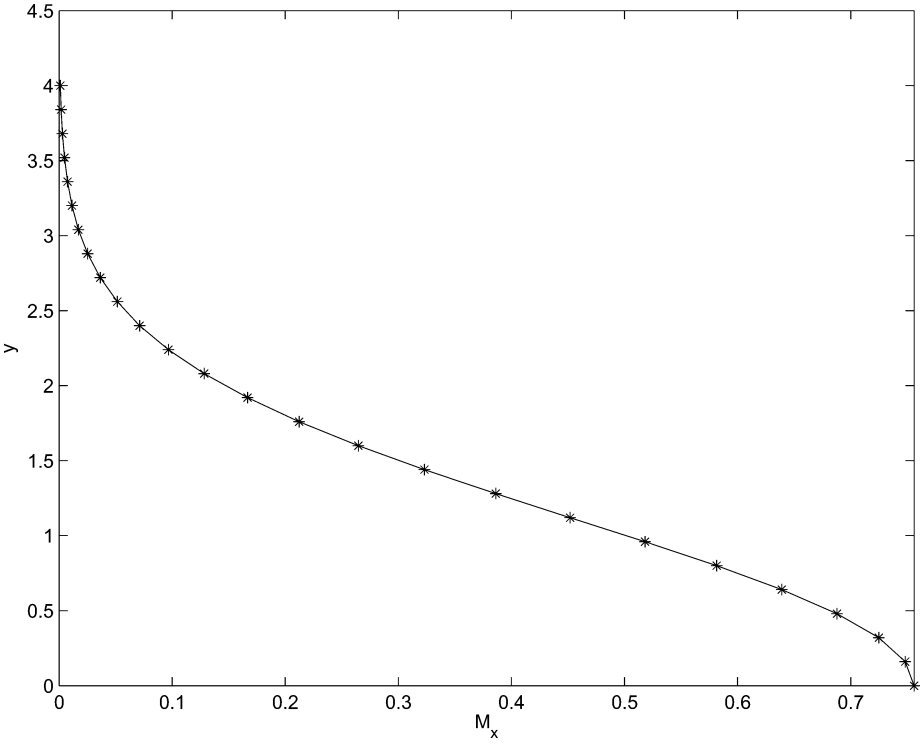
C. Application to Benchmark Problem

The benchmark problem from the fourth Computational Aeroacoustics workshop[6] is chosen as the test problem. The problem concerns the radiation and refraction of acoustic waves generated by a time harmonic source immersed in the shear layer of a jet. The geometry of the computational domain, and the mean flow profile, are shown in Fig. 2. The problem domain is $-50 < x < 150$ and $0 < y < 50$. The source is located at the origin. The frequency of excitation of the source is 76 rad/s. Along the centerline $y = 0$, the Mach number of the mean flow, which is only in the x direction, is 0.756 based on the speed of sound in the jet. The gradient of the mean flow in the y direction is very high, between $y = 0$ and $y \approx 4$, compared to the region $4 < y < 50$. An instability wave is excited by the source at the specified frequency of the source, but only the acoustic solution is obtained by solving the LEE in the frequency domain. The instability wave and the acoustic solution from Agarwal et al.[1] are shown in Fig. 3, where the instantaneous pressure values at $y = 15$ at the end of a cycle of oscillation of the source are plotted.

The physical domain is surrounded by a buffer domain in which the acoustic amplitudes are damped by adding an imaginary component to the frequency. The computational domain is $-250 < x < 225$ and $0 < y < 100$. The length of the buffer zone and the number of points in the buffer zone were selected to be similar to those in the study by Agarwal et al.[1] The reasons for the selections can be found therein. The mesh consists of similar right-angle triangles obtained by halving the rectangles of a 89×49 grid. There are 8500 elements in the mesh. The mesh is clustered near the origin in both the x and y directions to resolve the mean flow gradient and the source. The



a) 8500-element, 89 × 49 grid



b) Profile of mean flow in *x* direction

Fig. 2 Benchmark problem.

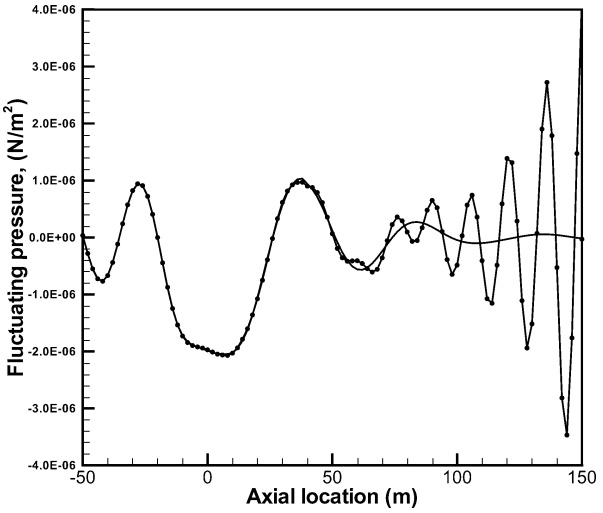


Fig. 3 Instability wave and acoustic solution from Agarwal et al.[1]; instantaneous pressure at $y = 15$: ——, acoustic solution and –·–·–·–, instability wave solution.

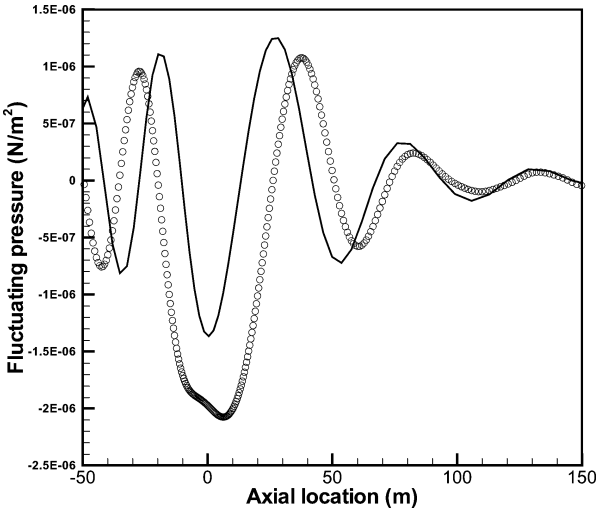


Fig. 4 Comparison of DG solution with exact solution from Agarwal et al.[1] for two-dimensional frequency domain problem; instantaneous pressure at $y = 15$: ○, solution from Agarwal et al.[1] and ——, DG solution.

result obtained using a first-order polynomial basis set is shown in Fig. 4. The acoustic solution from the DG method is compared with a solution from a second-order finite difference method given by Agarwal et al.[1] The size of the global matrix is over 10^5. (The size of a matrix has been defined in Sec. III.B as just the number of rows.) The solution time using the SuperLU solver on four processors is approximately 3.5 min. By comparison, a time domain solver used on the same number of elements takes a few hours to obtain a solution at the far field. However, the DG solution needs improvement. Although the solution was sought on the same grid using both the DG and SUPG methods, to be discussed later, the solution from the SUPG method is more accurate. The large error in the DG method is attributed to the implementation of the boundary conditions. The solution from the DG method depends heavily on the methods to implement the boundary conditions, which could be changed by extending the buffer zone and changing the damping coefficients, or by using a perfectly matched layer[25] (PML). The order of approximation and the number of grid points could be increased. However, an analysis of the computational needs and a comparison with other finite element methods, as discussed later, will show that such improvements are unnecessary because the DG method itself proves to be far more expensive for aeroacoustic problems in the frequency domain.

IV. Evaluation of DG Method

As mentioned earlier, the size of the global matrix in the DG method is given by $4NN_{\text{el}}$. The number of nonzeros N_{nz} in the matrix can be calculated as follows. In the discretized equation (8) for the two-dimensional LEE, the global matrix is made up of element volume matrices and element edge matrices. The total number of these matrices in $\boldsymbol{P}^e$ is equal to the total number of terms in the matrix $\boldsymbol{A}_r^e \boldsymbol{n}_r^{(j)}$, that is, 10. Here, $n_r^{(j)}$ is the normal to the side j of a triangle. Each volume matrix is of size N. For simplicity, the volume matrices will be considered to be full. The number of nonzeros in the edge matrices is equal to the number of entries in the mass matrix in one dimension and is given by $(p_d + 1)^2$. Given the three edges, the total number of nonzeros in an elemental matrix is given by $10[3(p_d + 1)^2 + N^2]$. Hence, the total number of nonzeros in the global matrix is given by

$$N_{\text{nz}} = 10\left[3(p_d + 1)^2 + N^2\right]N_{\text{el}} \tag{14}$$

with

$$N = (p_d + 1)(p_d + 2)/2$$

Consider a modest 8500-element grid. The size of the global matrix is 10^5, and the number of nonzeros is over 1.75×10^6. The size of the Harwell–Boeing format file containing the sparse global matrix is 92 MB. The sizes of this matrix are 220 and 660 MB, using uniformly second- and third-degree basis sets. Thus, the files size grows rapidly, and it becomes increasingly difficult to deal with the computer memory. Evidently, the number of nonzeros increases as the square of the degree of the basis set and is directly proportional to the number of elements.

In an unstructured grid, the number of elements is generally twice the number of nodes. With a conventional continuous Galerkin method, the size of the global matrix is equal to the number of nodes. Hence, the global matrix from the DG method is twice the size of that from a continuous FEM. Moreover, the edge flux terms are absent in a continuous Galerkin formulation. Therefore, the number of nonzeros is also considerably smaller compared to the DG method. Given the same 8500-element grid, using a continuous Galerkin method described in the following section, the number of nonzeros is 4.5 times less than that of the DG method. Whereas the solution of the linear system using the DG method requires a parallel solver, the solution using the SUPG method is obtained in MATLAB on a single processor in relatively less time.

Some useful conclusions can be drawn from this comparison: The DG method is suited for seeking a high-order-accurate solution on unstructured grids. However, for problems requiring a global assembly, the DG method is expensive in memory requirements compared to conventional continuous Galerkin methods. The memory requirement is critical, especially for the case of the LEE in the frequency domain where the problem involves multiple variables. Even then, the use of the DG method is recommended in the following cases:

1) The physics of the problem can be expressed using a single governing equation.

2) An adaptive refinement needs to be performed on unstructured meshes. The DG method can incorporate the h and p_d refinements very easily, for example, by the use of hanging nodes in the mesh. However, for an elliptic system of equations, as in the case of LEE, the use of the DG method poses prohibitive computational memory constraints compared to the continuous FEM. The memory requirements will be much greater for three-dimensional calculations. This is due to the increased number of nodal variables per element in three dimensions, especially for higher orders. For example, the number of variables in a second-order triangular element in two dimensions is $N = 6$, and, for a three-dimensional tetrahedron, $N = 10$. In addition, because the number of surfaces is higher in three dimensions compared to the number of sides in two dimensions, the global matrix is expected to be denser due to the additional surface flux terms.

Although FDM are as good as continuous FEM in terms of the memory requirements, they are not applicable on unstructured grids. In the following section, the application of continuous Galerkin

methods to frequency domain aeroacoustic problems is considered. Particularly, the emphasis is on the use of the SUPG method.

V. SUPG Method

A brief discussion on the application of the SUPG method in the discretization of a scalar advection equation is given in Appendix A. The weak form obtained from the SUPG formulation for the system of equations (3) over an element can be written as

$$\int_{\Omega^e} \psi\left(-i\omega\tilde{\boldsymbol{q}} + \frac{\partial \boldsymbol{A}_r\tilde{\boldsymbol{q}}}{\partial x_r} + \boldsymbol{C}\tilde{\boldsymbol{q}}\right)\mathrm{d}\Omega^e + \int_{\Omega^e} \tau^e \boldsymbol{A}_k^T \frac{\partial \psi}{\partial x_k}\left(-i\omega\tilde{\boldsymbol{q}} + \frac{\partial \boldsymbol{A}_r\tilde{\boldsymbol{q}}}{\partial x_r} + \boldsymbol{C}\tilde{\boldsymbol{q}}\right)\mathrm{d}\Omega^e = \int_{\Omega^e} \psi\tilde{\boldsymbol{s}}\,d\Omega^e + \int_{\Omega^e} \tau^e \boldsymbol{A}_k^T \frac{\partial \psi}{\partial x_k}\tilde{\boldsymbol{s}}\;\mathrm{d}\Omega^e \tag{15}$$

In the conventional SUPG method, the test function is given by $\psi + \boldsymbol{A}_r(\partial\psi/\partial x_r)$. However, the use of $\boldsymbol{A}_r^T$ in the place of $\boldsymbol{A}_r$ in the stabilizing term has been shown to produce superior results for nonlinear problems.[17] Also, Beau et al.[19] have used A_r^T in the stabilizing term for the Euler equations applied to compressible flows with shocks. In this work, because the linearized version of the compressible Euler equations is used, $\boldsymbol{A}_r^T$ is used in the formulation.

The choice of the stability parameter matrix has a significant influence on the result obtained using the SUPG method. The simplest method of selecting the stability parameter involves replacing the convection speed by functions of the eigenvalues of the flux coefficient matrices. In this work, an expression similar to that provided by Beau et al.[19] is used. The stability parameter matrix is expressed as

$$\tau^e = \max\left(\alpha h_r^e / \lambda_r\right)\boldsymbol{I} \tag{16}$$

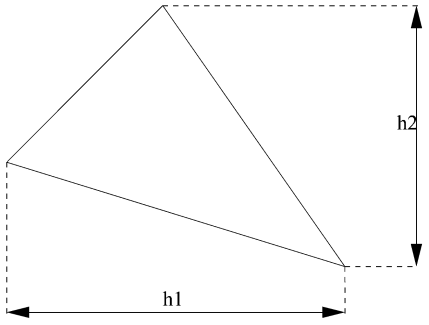


Fig. 5 Element lengths in triangle.

with $\boldsymbol{I}$ being a 4×4 identity matrix. The element size h_r^e is the width of the element in the rth direction, as shown in Fig. 5. Here, λ_r is the spectral radius of the coefficient matrix $\boldsymbol{A}_r$.

A. Discretized Equation

The discretized form of the weak formulation in an element can be written as

$$\boldsymbol{P}^e\tilde{\boldsymbol{q}}^e = \boldsymbol{f}^e \tag{17}$$

with

$$\boldsymbol{P}^e = \left[-i\omega\boldsymbol{I}\otimes\boldsymbol{M}^e + \boldsymbol{A}_r \odot \boldsymbol{K}_r^e + \boldsymbol{C}^e \odot \boldsymbol{M}^e\right] + \left[-i\omega\left[\tau^e\boldsymbol{A}_k^T\right]\odot\boldsymbol{K}_k^{eT} + \left[\tau^e\boldsymbol{A}_k^T\boldsymbol{A}_r\right]\odot\boldsymbol{V}_{kr} + \left[\tau^e\boldsymbol{A}_k^T\boldsymbol{C}\right]\odot\boldsymbol{M}^e\right] \tag{18}$$

$$\boldsymbol{f}^e = \left[\boldsymbol{I}\otimes\boldsymbol{M}^e + \left[\tau^e\boldsymbol{A}_k^T\right]\odot\boldsymbol{K}_k^{eT}\right]\tilde{\boldsymbol{s}}^e, \qquad \text{for} \qquad k, r = 1, 2 \tag{19}$$

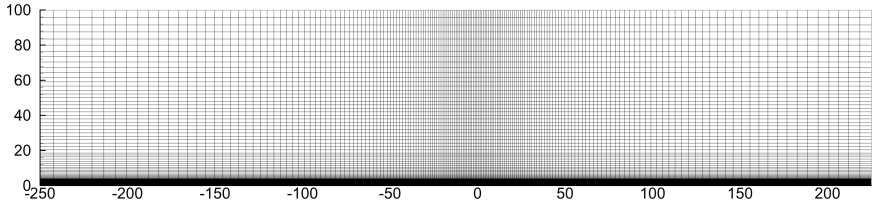
The element matrices and the symbols are as defined before for the DG algorithm. The diffusion matrix in an element is defined as

$$\boldsymbol{V}_{kr} = \int_{\Omega^e} \frac{\partial\psi}{\partial x_k}\frac{\partial\psi}{\partial x_r}\,\mathrm{d}\Omega^e \tag{20}$$

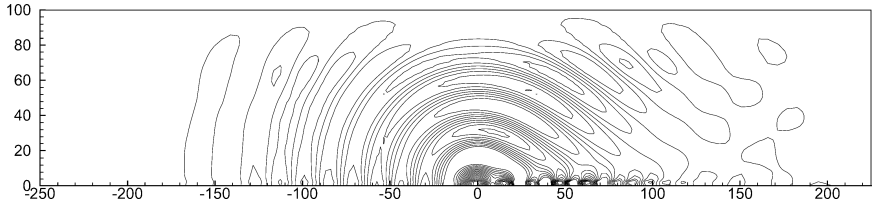
A global assembly is carried out with the element matrices and vectors of Eq. (17). After imposing the boundary conditions, the resulting linear system is solved using direct solvers.

B. Application to Benchmark Problem

The SUPG method is applied here to the benchmark problem described in Sec. III.C. The computational domain is taken to be $-250 < x < 225$ and $0 < y < 100$. In the buffer region, a damping function of the form given by Eq. (13) is used, with $\varepsilon = 5$ and $\kappa = 1.0$. The computation is performed on two grids made of similar right-angle triangles. The first grid has 89×49 nodes and the second has 177×97 nodes in the x and y directions, respectively. The second grid is shown in Fig. 6a, without the triangulation. The triangles are created by diagonally halving the rectangular elements. The nodes are clustered near the origin in both directions to resolve the source and the high mean flow gradient. The resulting numerical linear algebraic system is solved using a direct solver in MATLAB,[22] using LU decomposition. The size of the global matrix to be inverted is $(4 \times N_{\text{nodes}})^2$. The actual LU decomposition and the solution took only 4 min for the finer grid case on a 1-GHz Pentium IV processor with 512-MB RAM. A contour plot of the acoustic pressure obtained using a first-order SUPG method is shown in Fig. 6b. The values of the instantaneous pressure along $y = 15$ at the end of a cycle of oscillation of the source for the 89×49 grid is shown in Fig. 7a. The improved solution, which is much closer to the exact solution, is shown in Fig. 7b. The values of the pressure along $y = 50$ using the finer grid is shown in Fig. 8. The agreement between the fine grid solution and the exact solution is very good.

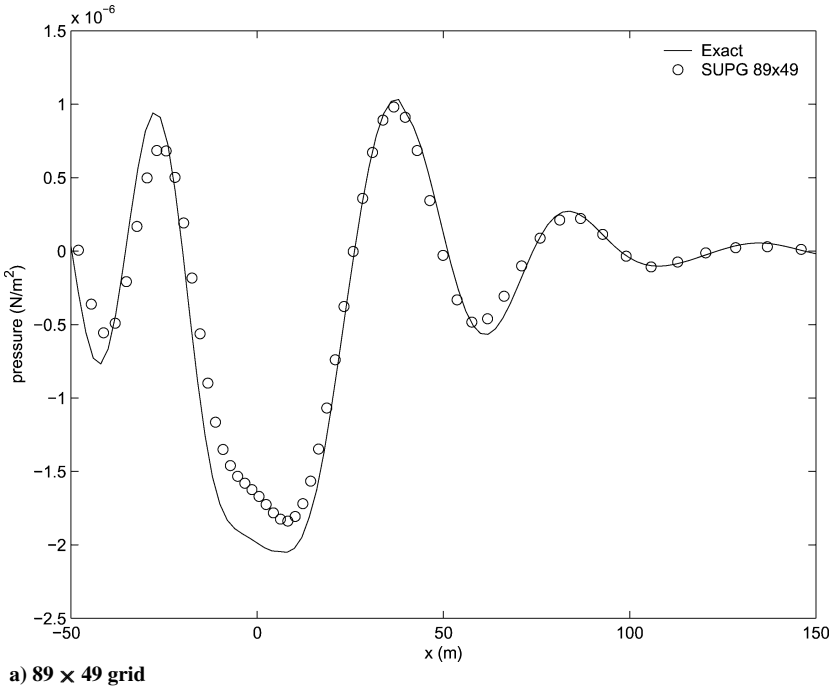


a) 177 × 97 grid used

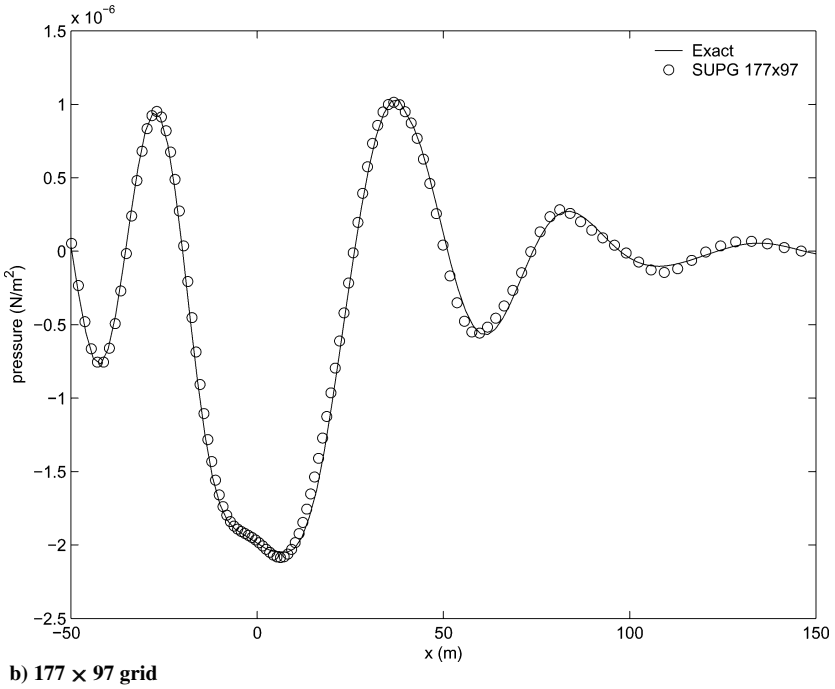


b) Contour plot of acoustic pressure

Fig. 6 SUPG method in 2*d* LEE.



a) 89 × 49 grid



b) 177 × 97 grid

Fig. 7 Comparison of exact and SUPG solutions for instantaneous pressure at $y = 15$.

C. Observations and Future Directions

The SUPG method has an inherent damping that increases with the size of the element. This can be seen from expression (16). This means that the artificial dissipation is decreased as the grid resolution is increased. The grid in the buffer region is coarser than that in the physical domain, and, hence, there is an inherent damping of the acoustic waves in the SUPG method. This needs to be taken into account while choosing the parameters in the damping function in Eq. (13). There is also a need to reduce the length of the buffer region surrounding the problem domain to cut down on the computation time. This requires a study of the damping parameters, as well as an implementation of the PML in the frequency domain.

The results presented here have been obtained using a value of $\alpha = 0.125$ in Eq. (16). It has been observed that the solution is very sensitive to the value of α and, subsequently, the stabilizing parameter matrix. The design of the stability parameter matrix requires significant research and analysis. In this implementation of the SUPG method, a fairly simple expression (16) was chosen for the stability parameter matrix. Methods using the inverse estimates of the coefficient matrices and the source matrices, as mentioned by Codina,[17] could be more satisfactory. Also, for higher-order solutions, estimates based on the element matrices $\boldsymbol{M}^e$ and $\boldsymbol{K}^e$ have to be evaluated because they have been documented[18] to produce a better solution in such cases.

VI. Discussion

In this paper, an application of two unstructured grid methods has been studied for the solution of the LEE in the frequency domain. This results in a suppression of the instability waves inherent to the solution. The solvers have been developed using the DG and SUPG

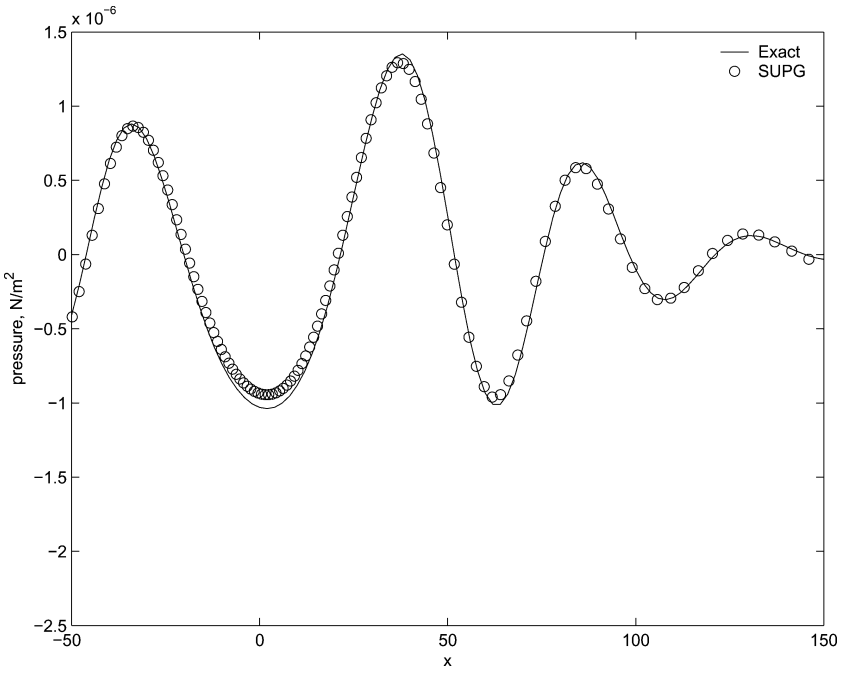


Fig. 8 Comparison of exact and SUPG solutions for instantaneous pressure at $y = 50$.

FEM in two spatial dimensions. The test problem considered is a recent benchmark problem of acoustic refraction and propagation in the shear layer of a parallel two-dimensional jet, involving the suppression of instability waves. The transformation of the LEE to the frequency domain produces an elliptic system of equations. This demands an assembly of elements in the computational domain to produce a global matrix. The resulting algebraic system needs to be solved directly to suppress the instability waves. The DG method produces a prohibitively large algebraic system compared to that produced by the continuous Galerkin method. The number of nonzeros in the global matrix increases as the square of the order of approximation. The size of the global matrix is of the order of the number of elements in the domain. The continuous Galerkin method, on the other hand, produces much smaller matrices. The size of the matrix to be inverted is critical for practical applications where the full LEE have to be used and, hence, multiple variables are involved.

Recently, Marburg and Schneider[26] studied the influence of element types on numerical errors in acoustic applications. They considered the application of the boundary element method (BEM) to the three-dimensional Helmholtz equation using continuous and discontinuous elements. The study concluded that the discontinuous elements are more efficient compared to continuous elements. Although the dimensionality of the surface mesh was the same as the problems considered in this paper for the two-dimensional LEE, the discontinuous elements proved to be more efficient compared to the continuous elements. The BEM, although similar to an FEM, has a few restrictions such as its inapplicability to nonlinear problems due to the difference in the basic formulation from the FEM. When FEM is used with discontinuous elements, there is an additional need for the calculation of fluxes on element boundaries. The basis set in the study consisted of Legendre polynomials. The present study uses nodal Lagrange polynomials. The choice of the basis set polynomial affects the boundary flux calculations considerably, as noted by Rao.[10] Hence, within the limits of the usage of nodal polynomials, it is observed in the present study that the use of discontinuous elements is computationally expensive for frequency domain applications if adaptive mesh and nonconforming elements are not employed.

For the purposes of comparison, let us assume that the mesh consisted of rectangular elements. It may be argued that the DG method can be easily defined on the mesh with linear elements, whereas the SUPG method, being a continuous method, cannot be defined with linear elements on a rectangular mesh. Bilinear elements would have to be used for the SUPG method, which would increase the number of degrees of freedom. However, in this paper we have interpolated the mean flow, along with the variables in the LEE on triangular elements, to evaluate the terms with their products appearing in the flux formulations as shown in Appendix A. Note that a linear interpolation of rectangular elements of the mean flow would introduce more errors in the mean flow solution used. Also, a nonnodal interpolation that is required on rectangular grids would result in a larger number of variables in the system, corresponding to the flux products of the LEE. The situation where rectangular meshes are used needs further investigation. Under circumstances where triangular meshes and nodal interpolation are used, the SUPG method has better efficiency compared to the DG method.

VII. Conclusions

The SUPG method has been applied to the LEE in the frequency domain using a simple expression for the stabilizing parameter. The suppression of instability waves in a parallel jet has been demonstrated with the use of the SUPG method. Unlike other frequency domain solutions in the present literature, no assumptions are made regarding the nature of the mean flow. The solver based on the SUPG method has been shown to be applicable for a rotational mean flow, which makes it more suitable for practical problems in aeroacoustics. However, the efficiency of the SUPG method has a few limitations. Continuous elements and, hence, the SUPG method are harder to use with adaptive grids and nonconforming elements. The resources required for the SUPG method could indeed become prohibitive due to the restriction of a conforming mesh, especially for high-frequency applications in three-dimensional problems.

Appendix A: Formulation of DG and SUPG Methods

The formulation of the DG method and the SUPG method are described for a two-dimensional elliptic conservation equation. For simplicity, the governing equation is chosen to be the conservation equation of a scalar variable in a bounded domain $\Omega \in \mathbb{R}^2$ given by

$$cq + \nabla \cdot \boldsymbol{F} = s \tag{A1}$$

where c is a constant and q is the conserved variable. $\boldsymbol{F}$ is the flux, which is a function of q, and s is a source term independent of q. Let the domain be divided into a number of triangular elements Ω^e, and in each element q is approximated in a fashion similar to that in Eq. (4) using Lagrange polynomials.

A. DG Algorithm

An application of the the Galerkin method to Eq. (A1), followed by an integration by parts, yields

$$\int_{\Omega^e} \psi_k \sum_{j=1}^{N} \psi_j c q_j^e \, \mathrm{d}\Omega^e - \int_{\Omega^e} \nabla\psi_k \cdot \boldsymbol{F}^e \, \mathrm{d}\Omega^e + \int_{\partial\Omega^e} \psi_k \bar{F}_R^e \, \mathrm{d}\Gamma = \int_{\Omega^e} \psi_k s^e \, \mathrm{d}\Omega^e \tag{A2}$$

for $k = 1, \ldots, N$. Here, $\mathrm{d}\Gamma$ is an edge of the triangle and $\bar{F}_R$ is the Riemann flux through the edge. The Riemann flux is approximated using the traces of the flux values of the element and the neighboring element on the common surface or edge. In the present work, the Riemann flux is approximated using the Lax–Friedrich's form (see Ref. 7)

$$\bar{F}_R^e = \tfrac{1}{2}\left[\left(\bar{\boldsymbol{F}}_+^e + \bar{\boldsymbol{F}}_-^e\right) \cdot \boldsymbol{n}^{\partial\Omega^e} - \alpha\left(\bar{q}_+^e - \bar{q}_-^e\right)\right] \tag{A3}$$

where $\bar{\boldsymbol{F}}_-^e$ is the trace of the flux of the element to the left of the edge and $\bar{\boldsymbol{F}}_+^e$ is that of the element to the right of the edge. Here, $\boldsymbol{n}^{\partial\Omega^e}$ is the unit normal to the edge from left to right and α is a smooth positive parameter chosen for the upwind bias. The value of α is chosen to be close to the convection speed, given by $\mathrm{d}F/\mathrm{d}u$. Then, for $k = 1, \ldots, N$,

$$\int_{\Omega^e} \psi_k \sum_{j=1}^{N} \psi_j c q_j^e \, \mathrm{d}\Omega^e - \int_{\Omega^e} \nabla\psi_k \cdot \boldsymbol{F}^e \, \mathrm{d}\Omega^e + \int_{\partial\Omega^e} \psi_k \bar{F}_R^e \, \mathrm{d}\Gamma = \int_{\Omega^e} \psi_k s^e \, \mathrm{d}\Omega^e \tag{A4}$$

In general, the flux can be a nonlinear function of q. The flux term is approximated using the same basis set, similar to Eq. (6),

$$F \approx \sum_{j=1}^{N} f_j \psi_j \tag{A5}$$

Here, the nodal values are denoted by $\boldsymbol{F}^e = \{f_j^e\}$. Let the source term also be approximated in fashion similar to q, with the nodal values denoted by $\boldsymbol{s}^e$. Then Eq. (24) is written as

$$c\boldsymbol{M}^e\boldsymbol{q}^e - \boldsymbol{K}^e\boldsymbol{f}^e + \boldsymbol{b}^e = \boldsymbol{M}^e\boldsymbol{s}^e \tag{A6}$$

The notation $\boldsymbol{q}^e = \{q_j^e\}$, $j = 1, \ldots, N$ has been used, and the expression for the terms in $\bar{F}_R^e$, to be explained later, are omitted. Here the elements of the mass matrix are

$$\boldsymbol{M}_{kj}^e = \int_{\Omega^e} \psi_k \psi_j \, \mathrm{d}\Omega^e \qquad \forall \qquad k, j = 1, \ldots, N \tag{A7}$$

and the elements of the stiffness matrix are

$$\boldsymbol{K}_{kj}^e = \int_{\Omega^e} \nabla\psi_k \psi_j \, \mathrm{d}\Omega^e \qquad \forall \qquad k, j = 1, \ldots, N \tag{A8}$$

It is convenient to carry out the integrations in the formulation (A2) in transformed coordinates. The integrals are calculated on a mapped element. In this study, a right-angle triangle is used. The final equation is evaluated and assembled in the real coordinates of Ω.

The edge flux term is formed by the sum of the contributions from the three sides of the triangular element. Let the contribution from side j be denoted as $\boldsymbol{b}^{ej}$. By the use of the Lax–Friedrich's expression in Eq. (A3), the edge flux on the side j is

$$\boldsymbol{b}^{ej} = \frac{1}{2}\int_{\Omega^e} \psi(\bar{F}^{ej} + \bar{F}^{\mathrm{NE}j})\boldsymbol{n}_r^j \, \mathrm{d}\Gamma^{ej} - \frac{\alpha}{2}\int_{\Omega^e} \psi(\bar{\boldsymbol{q}}^{\mathrm{NE}j} - \bar{\boldsymbol{q}}^{ej}) \, \mathrm{d}\Gamma^{ej} \tag{A9}$$

When the flux expansion is used,

$$\boldsymbol{b}^{ej} = \frac{1}{2}\left[\int_{\Omega^e} \psi\psi^T \, \mathrm{d}\Gamma^{ej}\right]\left\{\overline{\boldsymbol{F}_r^{ej}} + \overline{\boldsymbol{F}_r^{\mathrm{NE}j}}\right\}\boldsymbol{n}_r^j - \frac{\alpha}{2}\left[\int_{\Omega^e} \psi\psi^T \, \mathrm{d}\Gamma^{ej}\right] \times \{\overline{\boldsymbol{q}^{\mathrm{NE}j}} - \overline{\boldsymbol{q}^{ej}}\} \tag{A10}$$

Now, Lagrange polynomials reduce to their one-dimensional equivalents along the edges. Thus, the matrix

$$\left[\int_{\Omega^e} \psi\psi^T \, \mathrm{d}\Gamma^{ej}\right]$$

is a rearrangement of the one-dimensional mass matrix. The exact structure of this matrix is given by Rao.[10] This matrix of size $N \times N$ is denoted as the edge flux matrix

$$\boldsymbol{B}^{ej} = \left[\int_{\Omega^e} \psi\psi^T \, \mathrm{d}\Gamma^{ej}\right] \tag{A11}$$

B. SUPG Algorithm

For simplicity, it is assumed that the flux varies linearly with the variable in Eq. (A1), with a constant speed of advection denoted by $\boldsymbol{a}$:

$$\boldsymbol{F} = \boldsymbol{a}q \tag{A12}$$

The discretized weak form of the traditional Galerkin approach using the element approximation (4) is

$$\int_{\Omega^e} \psi_k(\boldsymbol{a} \cdot \nabla\psi_j) q_j^e \, \mathrm{d}\Omega^e + \int_{\Omega^e} c(\psi_k\psi_j) q_j^e \, \mathrm{d}\Omega^e = \int_{\Omega^e} \psi_k s \, \mathrm{d}\Omega^e \qquad k, j = 1, \ldots, N \tag{A13}$$

The weak form has an unsymmetrical term associated with the convection operator. In the SUPG method, an upwind biased diffusion term is added[27] to the weak form by modifying the test function

$$\psi \Rightarrow \psi + \tau^e \boldsymbol{a} \cdot \nabla\psi \tag{A14}$$

Here, τ^e is a stabilizing parameter controlling the artificial dissipation and is determined based on the element size, the reaction coefficient c, and the convection velocity. A possible candidate for τ^e when $c = 0$ is

$$\tau^e = \alpha h^e / 2|\boldsymbol{a}| \tag{A15}$$

with α as a positive constant. The discretized weak form in Eq. (33) now becomes

$$\int_{\Omega^e} \psi_k\left(\boldsymbol{a} \cdot \nabla\psi_j q_j^e \, \mathrm{d}\Omega^e + c\psi_j q_j^e - s\right) \mathrm{d}\Omega^e + \int_{\Omega^e} \tau^e \boldsymbol{a} \cdot \nabla\psi_k\left(\boldsymbol{a} \cdot \nabla\psi_j q_j^e + \psi_j q_j^e - s\right) \mathrm{d}\Omega^e = 0 \tag{A16}$$

The second term corresponds to the SUPG formulation. The product of the gradients of the test and trial functions in the first term of the SUPG expression is essentially the artificial diffusion term.

Appendix B: Kronecker Product and Row Multiple

A. Kronecker Product

The result of the Kronecker product $\otimes$ of a matrix $\boldsymbol{X}$ of size $M \times M$ with a matrix $\boldsymbol{Y}$ of size $N \times N$ is defined to be a matrix of size $MN \times MN$, made up of $M \times M$ blocks of $\boldsymbol{Y}$ multiplied with the

elements of $\boldsymbol{X}$. The $(k, j)^{th}$ block, which is of size $N \times N$, is given by

$$[\boldsymbol{X} \otimes \boldsymbol{Y}]^{(kj)} = [\boldsymbol{X}^{(kj)}\boldsymbol{Y}], \qquad \forall \qquad k, j = 1, \ldots, M \tag{B1}$$

An example illustrating the Kronecker product is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \otimes [\boldsymbol{D}] = \begin{bmatrix} a[\boldsymbol{D}] & b[\boldsymbol{D}] \\ c[\boldsymbol{D}] & d[\boldsymbol{D}] \end{bmatrix} \tag{B2}$$

B. Row Multiple

The symbol $\odot$ is defined to be a row multiple of two matrices, $\boldsymbol{X}$ and $\boldsymbol{Y}$, of sizes $M \times M$ and $N \times N$, respectively. This is a product defined in the case when the elements of $\boldsymbol{X}$ are nodal variables, expanded in a basis set of size N. For example, the flux coefficient matrices are made up of mean flow values that are nodal variables. The row multiple of $\boldsymbol{X}$ with $\boldsymbol{Y}$ is then defined to be made of $M \times M$ blocks of $\boldsymbol{Y}$, with the columns in every row of $\boldsymbol{Y}$ multiplied by the corresponding expansion coefficients of the elements of $\boldsymbol{X}$. The $(kj)^{th}$ block of the row multiple, which is of size $N \times N$, is defined to be

$$[\boldsymbol{X} \odot \boldsymbol{Y}]^{(kj)} = \left[\boldsymbol{X}^{e(kj)}{}_m \boldsymbol{Y}_{nm}\right] \qquad \forall \qquad k, j = 1, \ldots, M$$
$$\forall \qquad m, n = 1, \ldots, N \tag{B3}$$

with

$$\boldsymbol{X}^{(kj)} \approx \sum_{n=1}^{N} \boldsymbol{X}^{e(kj)}{}_n \psi_n$$

In Eq. (B3), the Einstein summation convention is not applied to the repeated index m.

References

[1]Agarwal, A., Morris, P. J., and Mani, R., "The Calculation of Sound Propagation in Nonuniform Flows: Suppression of Instability Waves," *AIAA Journal*, Vol. 42, No. 1, 2004, pp. 80–88.

[2]Agarwal, A., "The Prediction of Tonal and Broadband Slat Noise," Ph.D. Dissertation, Dept. of Aerospace Engineering, Pennsylvania State Univ., University Park, PA, May 2004.

[3]Xu, J., Stanescu, D., Hussaini, M. Y., and Farassat, F., "Computation of Engine Noise Propagation and Scattering Off an Aircraft," *International Journal of Aeroacoustics*, Vol. 1, No. 4, 2002, pp. 403–420.

[4]Hall, K. C., "A Variational Finite Element Method for Computational Aeroacoustic Calculations of Turbomachinery Noise," *Proceedings of the 2nd CAA Workshop on Benchmark Problems*, NASA CP 3352, 1997, pp. 269–278.

[5]Montgomery, D., and Verdon, J. M., "A Three-Dimensional Linearized Unsteady Analysis for Turbomachinery Blade Rows," NASA CR 4770, March 1997.

[6]"Radiation and Refraction of Sound Waves Through a 2-D Shear Layer," *Proceedings of the 4th CAA Workshop on Benchmark Problems*, NASA CP 2004-212954, 2004, pp. 23, 24.

[7]Cockburn, B., Karniadakis, G. E., and Shu, C. W., "The Development of Discontinuous Galerkin Methods," *Discontinuous Galerkin Methods: Theory, Computation and Applications*, edited by B. Cockburn, G. Karniadakis, and C. W. Shu, Lecture Notes in Computational Science and Engineering 11, Springer, New York, 1999, pp. 5–50.

[8]Cockburn, B., and Shu, C. W., "The Local Discontinuous Galerkin Method for Time Dependent Convection–Diffusion Systems," *SIAM Journal of Numerical Analysis*, Vol. 35, No. 6, 1998, pp. 2440–2463.

[9]Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, D., "Discontinuous Galerkin Methods for Elliptic Problems," *Discontinuous Galerkin Methods: Theory, Computation and Applications*, edited by B. Cockburn, G. Karniadakis, and C. W. Shu, Lecture Notes in Computational Science and Engineering 11, Springer, New York, 1999, pp. 89–101.

[10]Rao, P. P., "High Order Unstructured Grid Methods for Computational Aeroacoustics," Ph.D. Dissertation, Dept. of Aerospace Engineering, Pennsylvania State Univ., University Park, PA, May 2004.

[11]Atkins, H. L., and Shu, C. W., "Quadrature Free Implementation of Discontinuous Galerkin Method for Hyperbolic Equations," *AIAA Journal*, Vol. 36, No. 5, 1998, pp. 775–782.

[12]Rao, P. P., and Morris, P. J., "Application of a Generalized Quadrature Free Discontinuous Galerkin Method in Aeroacoustics," AIAA Paper 2003-3120, 2003.

[13]Donea, J., and Huerta, A., *Finite Element Methods for Flow Problems*, Wiley, Hoboken, NJ, 2003.

[14]Hughes, T. J. R., and Brooks, A., "A Multidimensional Upwind Scheme with No Crosswind Diffusion," *Finite Elements for Convection Dominated Flows*, edited by T. Hughes, Vol. AMD 34, American Society of Mechanical Engineers, New York, 1979.

[15]Kelly, D. W., Nakazawa, S., and Zienkiewicz, O. C., "A Note on Anisotropic Balancing Dissipation in the Finite Element Method Approximation to Convection Diffusion Problems," *International Journal for Numerical Methods in Engineering*, Vol. 15, No. 11, 1980, pp. 1705–1711.

[16]Hughes, T. J. R., and Mallet, M., "A New Finite Element Formulation for Computational Fluid Dynamics: III. The Generalized Streamline Operator for Multidimensional Advective Diffusive Systems," *Computer Methods in Applied Mechanics and Engineering*, Vol. 58, No. 3, 1986, pp. 305–328.

[17]Codina, R., "On Stabilized Finite Element Methods for Linear Systems of Convection–Diffusion-Reaction Equations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 188, No. 1, 2000, pp. 61–82.

[18]Tezduyar, T. E., and Osawa, Y., "Finite Element Stabilization Parameters Computed from Element Matrices and Vectors," *Computer Methods in Applied Mechanics and Engineering*, Vol. 190, No. 3, 2000, pp. 411–430.

[19]Beau, G. J. L., Ray, S. E., Aliabadi, S. K., and Tezduyar, T. E., "SUPG Finite Element Computation of Compressible Flows with the Entropy and Conservation Variables Formulations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 104, No. 3, 1993, pp. 397–422.

[20]Hughes, T. J. R., "Recent Progress in the Development and Understanding of the SUPG Methods with Special Reference to the Compressible Euler and Navier–Stokes Equations," *International Journal for Numerical Methods in Fluids*, Vol. 7, No. 11, 1987, pp. 1261–1275.

[21]Tezduyar, T., and Sathe, S., "Stabilization Parameters in SUPG and PSPG Formulations," *Journal of Computational and Applied Mechanics*, Vol. 4, No. 1, 2003, pp. 71–88.

[22]*MATLAB—The Language of Scientific Computing*, 13th ed., Mathworks, Natick, MA, 2003, URL: http://www.mathworks.com.

[23]Demmel, J. W., Gilbert, J. R., and Li, X. S., *SuperLU*, Lawrence Berkeley National Lab., Univ. of California, Berkeley, CA, 2nd ed., URL: http://acts.nersc.gov/superlu/ [cited 21 April 2006].

[24]Li, X., "zm2hb.m"—*A Program to Convert Sparse Matrices from Coordinate Format to Harwell–Boeing Format*, Univ. of California, Berkeley, CA, URL: http://math.nist.gov/MatrixMarket/src/zm2hb.m [cited 21 April 2006].

[25]Tam, C. K. W., Auriault, L., and Cambuli, F., "Perfectly Matched Layer as an Absorbing Boundary Condition for the Linearized Euler Equations in Open and Ducted Domains," *Journal of Computational Physics*, Vol. 144, No. 1, 1998, pp. 213–244.

[26]Marburg, S., and Schneider, S., "Influence of Element Types on Numeric Error for Acoustic Boundary Elements," *Journal of Computational Acoustics*, Vol. 11, No. 3, 2003, pp. 363–386.

[27]Zienkiewicz, O. C., and Taylor, R. L., *The Finite Element Method*, Vol. 3, 5th ed., Butterworth–Heinemann, Oxford, 2000.

D. Gaitonde
Associate Editor