AIAA JOURNAL
Vol. 40, No. 2, February 2002

Three-Dimensional Rectangular Duct Code with Application to Impedance Eduction

Willie R. Watson[*]
NASA Langley Research Center, Hampton, Virginia 23681-2199

A zero flow, fully three-dimensional, variable impedance, rectangular duct aeroacoustics code that spans the frequency spectrum of interest in duct liner research is developed. The governing equations and boundary conditions in the duct are solved numerically using the finite element methodology. The methodology makes use of a state-of-the-art, sparse equation solver to obtain the capability to study high-frequency sound waves that may require millions of grid points for resolution. Noise suppression levels predicted from the code are in excellent agreement with those obtained from mode theory. The single-processor performance of the solver, relative to that of the more commonly used band solver, increases with frequency. At a frequency of 17 kHz, the band solver is 4.25 times slower and consumes 2.5 times more memory than the fully sparse equation solver. The duct aeroacoustics code is combined with an optimization algorithm and used successfully to educe the impedance spectrum of a ceramic liner. The primary problem with using the methodology to perform optimization studies at frequencies above 14 kHz is excessive central processor unit time. The results support the recommendation that research be directed toward exploitation of the multiprocessor capability of the solver to further reduce central processor unit time.

Nomenclature

$\vec{A}$, $\overleftarrow{A}$	=	amplitude of right- and left-moving acoustic pressure mode
$[A]$, $[\bar{A}]$, $[B]$	=	system matrices; order is *NMQ*
$[A_q]$, $[B_q]$	=	major blocks of $[A]$; order is *NM*
$[A^e]$, $[A^S]$, $[A^V]$	=	local element matrices; order is 8
$[\mathcal{A}]$, $[\mathcal{B}]$, $[\mathcal{C}]$, $[\mathcal{F}]$, $[\underline{\mathcal{G}}]$	=	local wall impedance matrices; order is 8
$[\bar{a}_r]$, $[\bar{b}_r]$, $[\bar{c}_r]$	=	minor blocks of $[A]$; order is M
$[B^V]$, $[C^V]$, $[D^V]$, $[G^V]$	=	local element volume matrices; order is 8
$[c_q]$, $[f_q]$	=	major blocks of $[\mathcal{L}]$; order is *NM*
c_0, ρ_0	=	ambient sound speed and density
$[\mathcal{D}]$, $[\mathcal{L}]$	=	diagonal and unit lower triangular matrices; order is *NMQ*
$\{DA\}$, $\{OA\}$	=	vector of diagonal and off-diagonal coefficients in a sparse matrix
$[d_q]$, $[\mathcal{I}]$	=	diagonal matrix and identity matrix; order is *NM*
E, N_m	=	error function and three-dimensional basis functions
$\{F\}$, $\{F_R\}$	=	vectors of length *NMQ* containing source effects
$\{F_1\}$, $\{F_2\}$	=	vectors of length *NM* containing source pressures
f, p_s	=	source frequency and source plane acoustic pressure
H, W, L	=	height, width, and length of duct
h, w, l	=	height, width, and length of a finite element
i	=	$\sqrt{-1}$
k, k_x, k_y, k_z	=	free-space, transverse, spanwise, and axial wave numbers
k_{x0}	=	initial estimate for k_x
N, M, Q	=	number of transverse, spanwise and axial grid lines in the duct
NA, *NL*	=	number of nonzeroes in upper triangular part of a sparse matrix
$\{NC\}$, $\{CN\}$	=	pointer vectors for a sparse matrix
NF	=	number of fill-ins
P, p	=	acoustic pressure eigenmode and acoustic pressure disturbance
R, χ	=	normalized resistance and normalized reactance
t	=	dimensional time
V, S	=	computational volume and computational surface
x, y, z	=	transverse, spanwise, and axial coordinates
$\underline{\beta}^L$, $\underline{\beta}^R$, β^B, β^U	=	normalized admittances of wall lining
$\bar{\beta}$, $\bar{\zeta}$	=	normalized admittance and normalized impedance of the duct exit
ΔdB	=	noise suppression level, dB
ζ^L, ζ^R, ζ^B, ζ^U	=	normalized impedances of wall lining
$\{\underline{\Phi}\}$, $\{\underline{\Phi}^e\}$	=	global and local vector of node pressures
$\{\bar{\Phi}\}$, $\{\bar{\Phi}_R\}$	=	intermediate vectors of length *NMQ* for forward substitution
ϕ	=	axial acoustic power
∇^2,$\vec{\nabla}$	=	Laplace and gradient operator
$\cdot$	=	vector dot product

Subscripts

KI	=	matrix coefficient in the *K*th row and *I*th column of a matrix
m	=	local node counter, 1, 2, . . . , 8
q	=	major block counter, 1, 2, . . . , Q
R	=	reordered system
r	=	minor block counter, 1, 2, . . . , N
s	=	source plane index

Superscripts

e	=	element number
R, L, B, U	=	right sidewall, left sidewall, lower wall, and upper wall of duct
S,V	=	element surface and element volume
T	=	matrix or vector transpose
$*$	=	complex conjugate

Received 22 November 2000; revision received 3 August 2001; accepted for publication 8 August 2001.

[*] Senior Research Scientist, Computational Modeling and Simulation Branch, Aerodynamics, Aerothermodynamics, and Acoustics Competency, Mail Stop 128; W.R.Watson@Larc.NASA.Gov. Member AIAA.

I. Introduction

FAN noise accounts for a significant portion of community noise radiated from commercial aircraft engines. Noise reduction research today focuses on reducing the perceived noise levels of future aircraft by half relative to current levels. Installation of acoustic treatment, that is, liners, into the nacelles of aircraft engines remains one of the most effective means for achieving these noise reduction goals. However, future aircraft engines are expected to have engine ducts with much shorter length to diameter ratios than traditional aircraft. The shorter length to diameter ratios, that is, higher bypass ratios, will severely tax the ability of conventional liners to absorb engine noise effectively. To achieve the required noise reduction for the higher bypass ratios, more advanced liners are needed.[1] These include double- and triple-layer liners as well as those liners with variable surface impedances. The acoustic treatment must be optimized to provide sufficient noise reduction for the higher bypass ratios of the engine ducts.

To optimize the treatment for maximum sound suppression, fully three-dimensional aeroacoustic codes, which account for realistic geometries and increased liner complexity, are needed. The aeroacoustic codes must be numerically based because exact solutions are not possible for realistic geometries nor for wall impedance variations. Recent research in impedance eduction techniques has also highlighted the need for three-dimensional numerically based codes to perform accurate impedance measurements.[2] Currently, industry and government design codes treat only two-dimensional designs. Although several approximate three-dimensional models are available,[3] these models make simplifying assumptions that are not generally valid for acoustic disturbances propagating within the walls of an aircraft nacelle. Within an aircraft nacelle, the engine noise is often dominated by a few harmonics of a fundamental frequency. It is, therefore, convenient to use a frequency-domain analysis to take full advantage of the presence of only a few harmonics in the acoustic field.

From a cursory examination of the physical size of engine nacelles on current aircraft, and from knowledge of the predominant fan tones, a three-dimensional acoustic model will require upward of 100,000 degrees of freedom at even moderate frequencies. A second complication is that the matrix equations resulting from frequency-domain analysis are complex and indefinite. Complex, indefinite systems of matrix equations require special techniques to obtain their solution, and these techniques become difficult to implement efficiently for large systems of matrix equations. Currently, band solvers are used to obtain the solution to large sparse systems of equations in nacelle aeroacoustics.[2] When applied in three-dimensional computational methods, however, the band solvers require an excessive amount of CPU time and RAM. This requirement has limited nacelle aeroacoustic codes to the study of low-frequency sound sources in either two-dimensional or axisymmetric nacelles.

The purpose of this work is to develop a zero flow, fully three-dimensional, variable impedance, rectangular duct code for duct liner aeroacoustics. To the author's knowledge, this represents the first effort to develop a fully three-dimensional aeroacoustic code that is capable of spanning the full range of frequencies of current interest in duct liner research. First, the frequency-domain differential equation and boundary conditions are presented. The solution for the acoustic field is then approximated by a conventional finite element method. The finite element method leads to a large, sparse linear system of equations. A fully sparse equation solver is applied to reduce the CPU time and RAM required for three-dimensional solutions. RAM and CPU time statistics of the fully sparse solver are compared to that of the more commonly used band solver. Noise suppression levels computed from the fully sparse solver are compared to those predicted from a mode theory for the full range of frequencies of current interest in nacelle liner research. As an application, the three-dimensional model is combined with an optimization algorithm and used to educe the impedance spectrum of a ceramic liner.

II. Governing Equations and Boundary Conditions

Figure 1 shows the three-dimensional, rectangular-duct geometry and coordinate system used in this study. The volume enclosed by the duct is W units in width, H units in height, and L units in length. At the source and exit plane of the domain, respectively, the source plane pressure p_s and the exit plane normalized (dimensionless) impedance $\bar{\zeta}$ are known. The source plane pressure and normalized exit impedance are assumed functions of position along their respective boundaries. Throughout this paper, all impedances are normalized with the characteristics impedance, that is, $\rho_0 c_0$, of the air in the duct. Each wall of the duct is lined with a locally reacting sound absorbing material, that is, liner, with normalized impedance denoted by ζ. The normalized impedance is a function of position along the duct axis and the duct perimeter so that

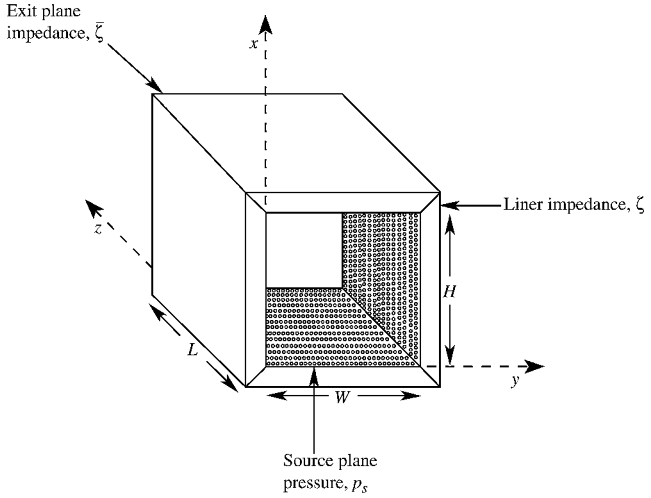

Fig. 1 Three-dimensional rectangular duct and coordinate system.

$$\zeta = \left\{ \begin{array}{l} \zeta^L \ [\text{if } y = 0, \text{ and } (0 \le x \le H,\ 0 \le z \le L)] \\ \zeta^R \ [\text{if } y = W, \text{ and } (0 \le x \le H,\ 0 \le z \le L)] \\ \zeta^B \ [\text{if } x = 0, \text{ and } (0 \le y \le W,\ 0 \le z \le L)] \\ \zeta^U \ [\text{if } x = H, \text{ and } (0 \le y \le W,\ 0 \le z \le L)] \end{array} \right\} \tag{1}$$

The equation that describes the propagation of linear, steady-state, acoustic pressure disturbances within the duct depicted in Fig. 1 is the Helmholtz equation (see Ref. 4):

$$\nabla^2 p + k^2 p = 0 \tag{2}$$

where a time convention of the form $e^{i2\pi ft}$ has been assumed and the free space wave number k is defined in the usual manner, that is, $k = (2\pi f)/c_0$. Although not considered here, Eq. (2) may be generalized to include uniform mean flows or be suitably transformed to cylindrical coordinates and used to study cylindrical duct geometries. Along the source plane of the duct ($z = 0$) the sound source pressure is known:

$$p = p_s \tag{3}$$

At the exit plane ($z = L$) the ratio of acoustic pressure to the normal component of acoustic particle velocity must equal the exit impedance. When expressed in terms of the acoustic pressure disturbance, the exit boundary condition is[4]

$$\frac{\partial p}{\partial n} = -ik\left(\frac{p}{\bar{\zeta}}\right) \tag{4}$$

The duct walls are assumed locally reacting, so that the wall boundary condition is[4]

$$\frac{\partial p}{\partial n} = -ik\left(\frac{p}{\zeta}\right) \tag{5}$$

When the normalized wall impedance function ζ is known, Eqs. (2–5) form a well-posed boundary value problem that can be solved to obtain the acoustic pressure disturbance in the duct. Once this pressure disturbance is obtained, it can be used to obtain the noise suppression in decibels ΔdB that is produced by the wall lining[4]:

$$\Delta \mathrm{dB} = 10\left[\log_{10}\frac{\phi(0)}{\phi(L)}\right]$$

$$\phi(z) = \frac{1}{2}\mathrm{Re}\left\{\frac{i}{2\pi\rho_0 f}\int_0^W\int_0^H\left[p(x,y,z)\frac{\partial p^*(x,y,z)}{\partial z}\right]\mathrm{d}x\,\mathrm{d}y\right\} \tag{6}$$

Generally, it is desirable to design an optimal liner. This is achieved by repeatedly cycling through the solution to the boundary value problem [Eqs. (2–5)] for given values of the wall impedance function ζ, until the impedance function that maximizes the sound suppression ΔdB is obtained. Conversely, if the upper wall acoustic pressure profile is known, the impedance ζ can be determined using an iterative procedure.[2] This procedure consists of repeated cycling through the solution to the boundary value problem described by Eqs. (2–5) and obtaining the upper wall acoustic pressure profile for each impedance function ζ. As each new wall pressure profile is computed, it is compared to the known profile until convergence is achieved within an accepted error tolerance.

Whether the objective is to obtain the wall lining that maximizes the noise suppression for a given wall impedance function (optimal liner design) or to obtain the unknown impedance for a known upper wall pressure (impedance eduction), an exact solution for the acoustic pressure field is not obtainable except for severely restrictive assumptions on the impedance functions. The question is, thus, not whether to use a numerical technique to obtain the acoustic pressure disturbance, but rather which numerical technique to use. Because of the promising results obtained by the finite element methodology in other fields of continuum mechanics and from initially promising applications to two-dimensional aeroacoustics applications,[2] the finite element methodology was chosen for this study.

III. Finite Element Methodology

The general description of the finite element methodology that is presented in this paper is detailed in the following step-by-step procedure[5,6]: 1) discretization of the computational domain, 2) approximations for the unknown variables, 3) derivation of the element stiffness matrix, 4) assembly of the system matrix, and 5) implementation of boundary conditions.

This sequence of steps describes the actual processes followed in setting up the system matrix equation, which must be solved by an equation solver.

A. Discretization of the Computational Domain

When applied to the current three-dimensional, duct aeroacoustics problem, the finite element method may be interpreted as an approximation to the continuous acoustic field as an assemblage of rectangular prism elements as shown in Fig. 2. Here, evenly spaced points N, M, and Q are assumed in the transverse, spanwise, and axial directions, respectively. A typical rectangular prism element with transverse, spanwise, and axial dimensions h, w, and l is shown in Fig. 3. The rectangular prism element consists of eight local node numbers labeled 1, 2, ... 8, respectively, as shown in Fig. 3. The objective is to obtain the unknown acoustic pressure disturbances at each of the NMQ nodes.

B. Approximations of the Unknown Variables

Within each element, p is approximated as a combination of eight linearly independent basis functions:

$$p = \sum_{m=1}^{m=8} N_m p_m \tag{7}$$

$$N_1 = \left(\frac{1-x}{h}\right)\left(\frac{1-y}{w}\right)\left(\frac{1-z}{l}\right)$$

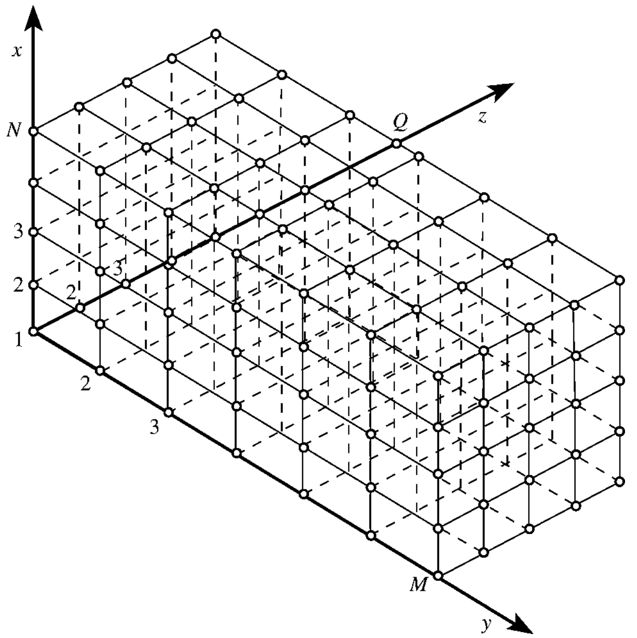

Fig. 2 Finite element discretization of three-dimensional computational domain.

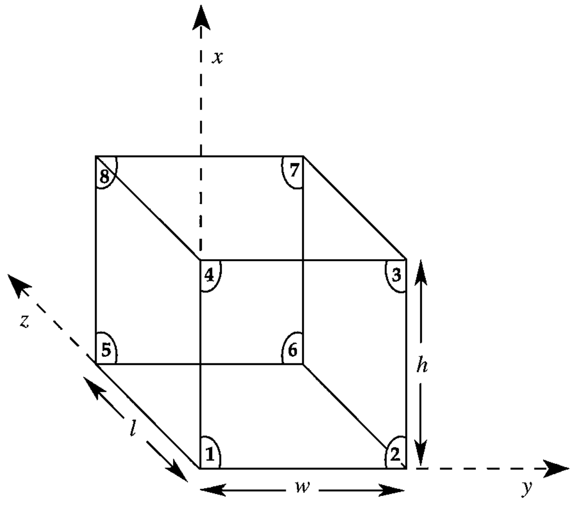

Fig. 3 Typical three-dimensional element.

$$N_2 = \left(\frac{1-x}{h}\right)\left(\frac{y}{w}\right)\left(\frac{1-z}{l}\right)$$

$$N_3 = \frac{(xy)}{(wh)}\left(\frac{1-z}{l}\right), \qquad N_4 = \left(\frac{x}{h}\right)\left(\frac{1-y}{w}\right)\left(\frac{1-z}{l}\right)$$

$$N_5 = \left(\frac{1-x}{h}\right)\left(\frac{1-y}{w}\right)\left(\frac{z}{l}\right), \qquad N_6 = \left(\frac{1-x}{h}\right)\left(\frac{y}{w}\right)\left(\frac{z}{l}\right)$$

$$N_7 = \frac{(xyz)}{(whl)}, \qquad N_8 = \left(\frac{x}{h}\right)\left(\frac{1-y}{w}\right)\left(\frac{z}{l}\right) \tag{8}$$

The basis functions $N_1, N_2, \ldots, N_8$ comprise a complete set of basis functions. The normalized admittance functions, that is, the reciprocal of the normalized impedance functions, are expanded in similar series:

$$1/\bar{\zeta} = \bar{\beta} = \left(\frac{1-x}{h}\right)\left[\left(\frac{1-y}{w}\right)\bar{\beta}_5 + \left(\frac{y}{w}\right)\bar{\beta}_6\right] + \left(\frac{x}{h}\right)\left[\left(\frac{y}{w}\right)\bar{\beta}_7 + \left(\frac{1-y}{w}\right)\bar{\beta}_8\right]$$

$$\frac{1}{\zeta^L} = \beta^L = \left(\frac{1-z}{l}\right)\left[\left(\frac{1-x}{h}\right)\beta_1^L + \left(\frac{x}{h}\right)\beta_4^L\right] + \left(\frac{z}{l}\right)\left[\left(\frac{1-x}{h}\right)\beta_5^L + \left(\frac{x}{h}\right)\beta_8^L\right]$$

$$\frac{1}{\zeta^R} = \beta^R = \left(\frac{1-z}{l}\right)\left[\left(\frac{1-x}{h}\right)\beta_2^R + \left(\frac{x}{h}\right)\beta_3^R\right] + \left(\frac{z}{l}\right)\left[\left(\frac{1-x}{h}\right)\beta_6^R + \left(\frac{x}{h}\right)\beta_7^R\right]$$

$$\frac{1}{\zeta^B} = \beta^B = \left(\frac{1-z}{l}\right)\left[\left(\frac{1-y}{w}\right)\beta_1^B + \left(\frac{y}{w}\right)\beta_2^B\right] + \left(\frac{z}{l}\right)\left[\left(\frac{1-y}{w}\right)\beta_5^B + \left(\frac{y}{w}\right)\beta_6^B\right]$$

$$\frac{1}{\zeta^U} = \beta^U = \left(\frac{1-z}{l}\right)\left[\left(\frac{y}{w}\right)\beta_3^U + \left(\frac{1-y}{w}\right)\beta_4^U\right] + \left(\frac{z}{l}\right)\left[\frac{(x)}{(w)}\beta_7^U + \left(\frac{1-y}{w}\right)\beta_8^U\right] \tag{9}$$

C. Derivation of the Element Stiffness Matrix

Galerkin's finite element method is used to minimize the field error and develop the element stiffness matrix. The field error function is defined as

$$E = \nabla^2 p + k^2 p \tag{10}$$

Contributions to the minimization of the field error function for a typical element due to local node m are

$$\int_V E N_m \,\mathrm{d}V = \int_V [\nabla^2 p + k^2 p] N_m \,\mathrm{d}V \tag{11}$$

The second derivative terms in Eq. (11) are reduced to first derivatives using Green's second identity:

$$\int_V E N_m \,\mathrm{d}V = \int_V \left[-\vec{\nabla} p \cdot \vec{\nabla} N_m + k^2 p N_m\right] \mathrm{d}V + \int_S \frac{\partial p}{\partial n} N_m \,\mathrm{d}S \tag{12}$$

Elimination of the second derivative terms from the surface integral in Eq. (12) is required so that the linear basis functions N_m can be used. Elimination of the second derivative terms from the integral also has the advantage that all impedance boundary conditions can be incorporated into this surface integral. The incorporation of the boundary condition into the surface integral also allows a choice of basis functions that do not have to satisfy explicitly any impedance boundary conditions.

The contribution to the surface integral

$$\int_S \frac{\partial p}{\partial n} N_m \,\mathrm{d}S \tag{13}$$

is identically zero for all elements except those that lie along an impedance boundary. Substituting the exit plane boundary condition into the surface integral in Eq. (13) gives, along the exit boundary,

$$\int_S \frac{\partial p}{\partial n} N_m \,\mathrm{d}S = -ik \int_S \left(\frac{p}{\bar{\zeta}}\right) N_m \,\mathrm{d}S \tag{14}$$

whereas for elements that lie along the upper, lower, and sidewalls of the duct

$$\int_S \frac{\partial p}{\partial n} N_m \,\mathrm{d}S = -ik \int_S \left(\frac{p}{\zeta}\right) N_m \,\mathrm{d}S \tag{15}$$

The contribution to the minimization of the field error for each element e when collected for each of the eight local nodes m is expressed in matrix form as

$$\begin{Bmatrix} \int_V E N_1 \,\mathrm{d}V \\ \int_V E N_2 \,\mathrm{d}V \\ \vdots \\ \int_V E N_8 \,\mathrm{d}V \end{Bmatrix} = [A^e]\{\Phi^e\} \tag{16}$$

$$[A^e] = [A^V] + [A^S], \qquad \{\Phi^e\} = \{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8\}^T \tag{17}$$

In Eq. (17), $[A^V]$ and $[A^S]$ are the contributions to $[A^e]$ due to the element volume and the impedance boundaries, respectively. Each matrix in Eq. (17) is an 8 × 8, complex, symmetric matrix that is given explicitly as

$$[A^V] = \frac{k^2 whl}{216}[G^V] - \frac{wl}{36h}[B^V] - \frac{hl}{36w}[C^V] - \frac{wh}{36l}[D^V] \tag{18}$$

$$[G^V] = \begin{bmatrix} 8 & 4 & 2 & 4 & 4 & 2 & 1 & 2 \\ 4 & 8 & 4 & 2 & 2 & 4 & 2 & 1 \\ 2 & 4 & 8 & 4 & 1 & 2 & 4 & 2 \\ 4 & 2 & 4 & 8 & 2 & 1 & 2 & 4 \\ 4 & 2 & 1 & 2 & 8 & 4 & 2 & 4 \\ 2 & 4 & 2 & 1 & 4 & 8 & 2 & 4 \\ 1 & 2 & 4 & 2 & 2 & 2 & 8 & 4 \\ 2 & 1 & 2 & 4 & 4 & 4 & 4 & 8 \end{bmatrix} \tag{19}$$

$$[B^V] = \begin{bmatrix} 4 & 2 & -2 & -4 & 2 & 1 & -1 & -2 \\ 2 & 4 & -4 & -2 & 1 & 2 & -2 & -1 \\ -2 & -4 & 4 & 2 & -1 & -2 & 2 & 1 \\ -4 & -2 & 2 & 4 & -2 & -1 & 1 & 2 \\ 2 & 1 & -1 & -2 & 4 & 2 & -2 & -4 \\ 1 & 2 & -2 & -1 & 2 & 4 & -4 & -2 \\ -1 & -2 & 2 & 1 & -2 & -4 & 4 & 2 \\ -2 & -1 & 1 & 2 & -4 & -2 & 2 & 4 \end{bmatrix} \tag{20}$$

$$[C^V] = \begin{bmatrix} 4 & -4 & -2 & 2 & 2 & -2 & -1 & 1 \\ -4 & 4 & 2 & -2 & -2 & 2 & 1 & -1 \\ -2 & 2 & 4 & -4 & -1 & 1 & 2 & -2 \\ 2 & -2 & -1 & 4 & 1 & -1 & -2 & 2 \\ -2 & 2 & 1 & -1 & 4 & -4 & -2 & 2 \\ -2 & 2 & 1 & -1 & -4 & 4 & 2 & -2 \\ -1 & 1 & 2 & -2 & -2 & 2 & 4 & -4 \\ 1 & -1 & -2 & 2 & 2 & -2 & -4 & 4 \end{bmatrix} \tag{21}$$

$$[D^V] = \begin{bmatrix} 4 & 2 & 1 & 2 & -4 & -2 & -1 & -2 \\ -4 & 4 & 2 & 1 & -2 & -4 & -2 & -1 \\ -2 & 2 & 4 & 2 & -1 & -2 & -4 & -2 \\ 2 & -2 & -4 & 4 & -2 & -1 & -2 & -4 \\ -2 & 2 & 1 & -1 & 4 & 2 & 1 & 2 \\ -2 & 2 & 1 & -1 & -4 & 4 & 2 & 1 \\ -1 & -2 & -4 & -2 & 1 & 2 & 4 & 2 \\ -2 & -1 & -2 & -4 & 2 & 1 & 2 & 4 \end{bmatrix} \tag{22}$$

$$[A^S] = -\frac{ikhw}{144}[\mathcal{A}] - \frac{ikwl}{144}[\mathcal{B}] - \frac{ikwl}{144}[\mathcal{C}] - \frac{ikhl}{144}[\mathcal{F}] - \frac{ikhl}{144}[\mathcal{G}] \tag{23}$$

$$[\mathcal{A}] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathcal{A}_{55} & \mathcal{A}_{56} & \mathcal{A}_{57} & \mathcal{A}_{58} \\ 0 & 0 & 0 & 0 & \mathcal{A}_{56} & \mathcal{A}_{66} & \mathcal{A}_{67} & \mathcal{A}_{57} \\ 0 & 0 & 0 & 0 & \mathcal{A}_{57} & \mathcal{A}_{67} & \mathcal{A}_{77} & \mathcal{A}_{78} \\ 0 & 0 & 0 & 0 & \mathcal{A}_{58} & \mathcal{A}_{57} & \mathcal{A}_{78} & \mathcal{A}_{88} \end{bmatrix}, \qquad \begin{aligned} \mathcal{A}_{55} &= 9\bar\beta_5 + 3\bar\beta_6 + \bar\beta_7 + 3\bar\beta_8 \\ \mathcal{A}_{56} &= 3\bar\beta_5 + 3\bar\beta_6 + \bar\beta_7 + \bar\beta_8 \\ \mathcal{A}_{57} &= \bar\beta_5 + \bar\beta_6 + \bar\beta_7 + \bar\beta_8 \\ \mathcal{A}_{58} &= 3\bar\beta_5 + \bar\beta_6 + \bar\beta_7 + 3\bar\beta_8 \\ \mathcal{A}_{66} &= 3\bar\beta_5 + 9\bar\beta_6 + 3\bar\beta_7 + \bar\beta_8 \\ \mathcal{A}_{67} &= \bar\beta_5 + 3\bar\beta_6 + 3\bar\beta_7 + \bar\beta_8 \\ \mathcal{A}_{77} &= \bar\beta_5^E + 3\bar\beta_6 + 9\bar\beta_7 + 3\bar\beta_8 \\ \mathcal{A}_{78} &= \bar\beta_5 + \bar\beta_6 + 3\bar\beta_7 + 3\bar\beta_8 \\ \mathcal{A}_{88} &= 3\bar\beta_5 + \bar\beta_6 + 3\bar\beta_7 + 9\bar\beta_8 \end{aligned} \tag{24}$$

$$[\mathcal{B}] = \begin{bmatrix} \mathcal{B}_{11} & 0 & 0 & \mathcal{B}_{14} & \mathcal{B}_{15} & 0 & 0 & \mathcal{B}_{18} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathcal{B}_{14} & 0 & 0 & \mathcal{B}_{44} & \mathcal{B}_{48} & 0 & 0 & \mathcal{B}_{48} \\ \mathcal{B}_{15} & 0 & 0 & \mathcal{B}_{48} & \mathcal{B}_{55} & 0 & 0 & \mathcal{B}_{58} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathcal{B}_{18} & 0 & 0 & \mathcal{B}_{48} & \mathcal{B}_{58} & 0 & 0 & \mathcal{B}_{88} \end{bmatrix}, \qquad \begin{aligned} \mathcal{B}_{11} &= 9\beta_1^L + 3\beta_4^L + 3\beta_5^L + \beta_8^L \\ \mathcal{B}_{14} &= 3\beta_1^L + 3\beta_4^L + \beta_5^L + \beta_8^L \\ \mathcal{B}_{15} &= 3\beta_1^L + \beta_4^L + 3\beta_5^L + \beta_8^L \\ \mathcal{B}_{18} &= \beta_1^L + \beta_4^L + \beta_5^L + \beta_8^L \\ \mathcal{B}_{44} &= 3\beta_1^L + 9\beta_4^L + \beta_5^L + 3\beta_8^L \\ \mathcal{B}_{48} &= \beta_1^L + 3\beta_4^L + \beta_5^L + 3\beta_8^L \\ \mathcal{B}_{55} &= 3\beta_1^L + \beta_4^L + 9\beta_5^L + 3\beta_8^L \\ \mathcal{B}_{58} &= \beta_1^L + \beta_4^L + 3\beta_5^L + 3\beta_8^L \\ \mathcal{B}_{88} &= \beta_1^L + 3\beta_4^L + 3\beta_5^L + 9\beta_8^L \end{aligned} \tag{25}$$

$$[\mathcal{C}] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathcal{C}_{22} & \mathcal{C}_{23} & 0 & 0 & \mathcal{C}_{26} & \mathcal{C}_{27} & 0 \\ 0 & \mathcal{C}_{23} & \mathcal{C}_{33} & 0 & 0 & \mathcal{C}_{27} & \mathcal{C}_{37} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathcal{C}_{26} & \mathcal{C}_{27} & 0 & 0 & \mathcal{C}_{66} & \mathcal{C}_{67} & 0 \\ 0 & \mathcal{C}_{27} & \mathcal{C}_{37} & 0 & 0 & \mathcal{C}_{67} & \mathcal{C}_{77} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad \begin{aligned} \mathcal{C}_{22} &= 9\beta_2^R + 3\beta_3^R + 3\beta_6^R + \beta_7^R \\ \mathcal{C}_{23} &= 3\beta_2^R + 3\beta_3^R + \beta_6^R + \beta_7^R \\ \mathcal{C}_{26} &= 3\beta_2^R + \beta_3^R + 3\beta_6^R + \beta_7^R \\ \mathcal{C}_{33} &= 3\beta_2^R + 9\beta_3^R + \beta_6^R + 3\beta_7^R \\ \mathcal{C}_{36} &= \beta_2^R + \beta_3^R + \beta_6^R + \beta_7^R \\ \mathcal{C}_{37} &= \beta_2^R + 3\beta_3^R + \beta_6^R + 3\beta_7^R \\ \mathcal{C}_{66} &= 3\beta_2^R + \beta_3^R + 9\beta_6^R + 3\beta_7^R \\ \mathcal{C}_{67} &= \beta_2^R + \beta_3^R + 3\beta_6^R + 3\beta_7^R \\ \mathcal{C}_{77} &= \beta_2^R + 3\beta_3^R + 3\beta_6^R + 9\beta_7^R \end{aligned} \tag{26}$$

$$[\mathcal{F}] = \begin{bmatrix} \mathcal{F}_{11} & \mathcal{F}_{12} & 0 & 0 & \mathcal{F}_{15} & \mathcal{F}_{16} & 0 & 0 \\ \mathcal{F}_{12} & \mathcal{F}_{22} & 0 & 0 & \mathcal{F}_{16} & \mathcal{F}_{26} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathcal{F}_{15} & \mathcal{F}_{16} & 0 & 0 & \mathcal{F}_{55} & \mathcal{F}_{56} & 0 & 0 \\ \mathcal{F}_{16} & \mathcal{F}_{26} & 0 & 0 & \mathcal{F}_{56} & \mathcal{F}_{66} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad \begin{aligned} \mathcal{F}_{11} &= 9\beta_1^B + 3\beta_2^B + 3\beta_5^B + \beta_6^B \\ \mathcal{F}_{12} &= 3\beta_1^B + 3\beta_2^B + \beta_5^B + \beta_6^B \\ \mathcal{F}_{15} &= 3\beta_1^B + \beta_2^B + 3\beta_5^B + \beta_6^B \\ \mathcal{F}_{16} &= \beta_1^B + \beta_2^B + \beta_5^B + \beta_6^B \\ \mathcal{F}_{22} &= 3\beta_1^B + 9\beta_2^B + \beta_5^B + 3\beta_6^B \\ \mathcal{F}_{26} &= \beta_1^B + 3\beta_2^B + \beta_5^B + 3\beta_6^B \\ \mathcal{F}_{55} &= 3\beta_1^B + \beta_2^B + 9\beta_5^B + 3\beta_6^B \\ \mathcal{F}_{56} &= \beta_1^B + \beta_2^B + 3\beta_5^B + 3\beta_6^B \\ \mathcal{F}_{66} &= \beta_1^B + 3\beta_2^B + 3\beta_5^B + 9\beta_6^B \end{aligned} \tag{27}$$

$$[\mathcal{G}] = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathcal{G}_{33} & \mathcal{G}_{34} & 0 & 0 & \mathcal{G}_{37} & \mathcal{G}_{38} \\ 0 & 0 & \mathcal{G}_{34} & \mathcal{G}_{44} & 0 & 0 & \mathcal{G}_{38} & \mathcal{G}_{48} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathcal{G}_{37} & \mathcal{G}_{38} & 0 & 0 & \mathcal{G}_{77} & \mathcal{G}_{78} \\ 0 & 0 & \mathcal{G}_{38} & \mathcal{G}_{48} & 0 & 0 & \mathcal{G}_{78} & \mathcal{G}_{88} \end{bmatrix}, \qquad \begin{aligned} \mathcal{G}_{33} &= 9\beta_3^U + 3\beta_4^U + 3\beta_7^U + \beta_8^U \\ \mathcal{G}_{34} &= 3\beta_3^U + 3\beta_4^U + \beta_7^U + \beta_8^U \\ \mathcal{G}_{37} &= 3\beta_3^U + \beta_4^U + 3\beta_7^U + \beta_8^U \\ \mathcal{G}_{38} &= \beta_3^U + \beta_4^U + \beta_7^U + \beta_8^U \\ \mathcal{G}_{44} &= 3\beta_3^U + 9\beta_4^U + \beta_7^U + 3\beta_8^U \\ \mathcal{G}_{48} &= \beta_3^U + 3\beta_4^U + \beta_7^U + 3\beta_8^U \\ \mathcal{G}_{77} &= 3\beta_3^U + \beta_4^U + 9\beta_7^U + 3\beta_8^U \\ \mathcal{G}_{78} &= \beta_3^U + \beta_4^U + 3\beta_7^U + 3\beta_8^U \\ \mathcal{G}_{88} &= \beta_3^U + 3\beta_4^U + 3\beta_7^U + 9\beta_8^U \end{aligned} \tag{28}$$

D. Assembly of the System Matrix

Assembly of the system matrix, that is, a matrix representing the entire system, from the element stiffness matrices is a fundamental feature of all finite element computations. Appropriate shifting of rows and columns is all that is required to add the local element matrix $[A^e]$ directly into the system matrix.[5,6] The system of equations resulting from the assembly of the elements is of the form

$$[\bar{A}]\{\Phi\} = \{0\} \tag{29}$$

$$[\bar{A}] = \begin{bmatrix} [A_1] & [B_2] & & & \\ [B_2]^T & [A_2] & [B_3] & & \\ & [B_3]^T & [A_3] & [B_4] & \\ & & \ddots & \ddots & \ddots \\ & & & [B_Q]^T & [A_Q] \end{bmatrix}$$

$$\{\Phi\} = \{\{\Phi_1\}, \{\Phi_2\}, \{\Phi_3\}, \ldots, \{\Phi_Q\}\}^T \tag{30}$$

Each major block ($[A_q]$ and $[B_q]$) is an $NM \times NM$ complex matrix, and the diagonal major blocks ($[A_q]$) are symmetric.

E. Implementation of the Sound-Source Boundary Condition

The system of equations given by Eq. (29) can be solved to obtain the solution vector once the sound-source boundary condition, that is, Eq. (3), has been substituted. To satisfy the sound-source boundary condition, all nodal values of the acoustic pressure at the source plane are simply set to the known value of source pressure p_s. This is achieved by modifying Eq. (30) to

$$[A]\{\Phi\} = \{F\} \tag{31}$$

$$[A] = \begin{bmatrix} [\mathcal{I}] & [0] & & & \\ [0] & [A_2] & [B_3] & & \\ & [B_3]^T & [A_3] & [B_4] & \\ & & \ddots & \ddots & \ddots \\ & & & [B_Q]^T & [A_Q] \end{bmatrix}$$

$$\{F\} = \begin{Bmatrix} \{F_1\} \\ \{F_2\} \\ \{0\} \\ \vdots \\ \{0\} \end{Bmatrix}, \qquad \{F_2\} = -[B_2]^T\{F_1\} \tag{32}$$

where $\{F_1\}$ is a vector of length NM that contains the value of the source pressure p_s at the NM nodes in the source plane. Note that only the first NM rows and columns of $[\bar{A}]$ [see Eq. (29)] are modified by application of the sound-source boundary condition. The substitution of a sound-source boundary condition such as Eq. (3) into a matrix equation such as Eq. (29) is described in detail elsewhere.[5]

IV. Equation Solvers

The system matrix equation [Eq. (31)] may be solved on a digital computer to obtain the solution vector. The choice of an equation solver that can function in an efficient manner is of paramount concern. In the simplest formulation of an equation solver, both the zero and nonzero coefficient within the system matrix $[A]$ are stored and operated upon. This approach requires approximately $(NMQ)^2$ storage locations, and the work, that is, the total number of arithmetic operations, required to obtain the solution vector is approximately $(NMQ)^3$. This method of solution is impractical for three-dimensional aeroacoustics applications because there is not enough RAM to store the $(NMQ)^2$ coefficients on modern computers.

Fortunately, the system matrix equation (31) is amenable to solution techniques that take advantage of the special character of the system matrix. In particular, the system matrix is symmetric and thinly populated or sparse. The two most commonly used methods for solving sparse equation systems such as Eq. (31) are the band solver and the fully sparse solver.

A. Band Solver

The banded form of Eq. (31) is well suited for repeated solutions of large systems of equations. The band solver (BS) used in this paper is based on a Gauss–Doolittle factorization of the coefficient matrix, followed by the sequential operations of forward and backward substitution to obtain the solution vector. The Gauss–Doolittle factorization of $[A]$ is readily obtainable (see Ref. 7):

$$[A] = [\mathcal{L}][\mathcal{D}][\mathcal{L}]^{\mathcal{T}} \tag{33}$$

$$[\mathcal{L}] = \begin{bmatrix} [c_1] & & & & \\ [f_2] & [c_2] & & & \\ & [f_3] & [c_3] & & \\ & & \ddots & \ddots & \\ & & & [f_Q] & [c_Q] \end{bmatrix}$$

$$[\mathcal{D}] = \begin{bmatrix} [d_1] & & & & \\ & [d_2] & & & \\ & & [d_3] & & \\ & & & \ddots & \\ & & & & [d_Q] \end{bmatrix} \tag{34}$$

The matrix coefficients in $[\mathcal{L}]$ and $[\mathcal{D}]$ are determined by applying the rules of block matrix multiplication to Eq. (33).[7]

Equation (31) decomposes into two triangular systems,

$$[\mathcal{L}]\{\bar{\Phi}\} = \{F\} \tag{35}$$

$$[\mathcal{D}][\mathcal{L}]^T\{\Phi\} = \{\bar{\Phi}\} \tag{36}$$

The solutions to the lower triangular system [Eq. (35)] and the upper triangular system [Eq. (36)] are obtained using forward and backward substitution, respectively. We achieve economy of RAM and CPU time by taking into account that only the Q diagonal major blocks $[A_q]$, and the $(Q-1)$ off-diagonal major blocks $[B_q]$ need to be stored or operated on during the solution phase. During the forward and backward substitution phases of the solution, we need to retain only $[c_q]$, $[f_q]$, and $[d_q]$. Thus, to achieve economy of RAM, we overwrite $[c_q]$ into $[A_q]$ and $[f_q]$ into $[B_q]$ during the solution phase.

B. Fully Sparse Solver

BS achieves economy of RAM and CPU time by allowing for storage and arithmetic operations only on coefficients within the major blocks of $[A]$. However, a significant number of coefficients within the major blocks of $[A]$ are zero. Each major block $[A_q]$ and $[B_q]$ [see Eq. (32)] is an $NM \times NM$ block-tridiagonal matrix:

$$\begin{bmatrix} [\bar{a}_1] & [\bar{c}_1] & & & \\ [\bar{b}_2] & [\bar{a}_2] & [\bar{c}_2] & & \\ & [\bar{b}_3] & [\bar{a}_3] & [\bar{c}_3] & \\ & & \ddots & \ddots & \ddots \\ & & & [\bar{b}_N] & [\bar{a}_N] \end{bmatrix} \tag{37}$$

Each minor block of $[A]$ ($[\bar{a}_r]$, $[\bar{b}_r]$ and $[\bar{c}_r]$) is a complex $M \times M$ tridiagonal matrix. Thus, the major blocks of $[A]$ are also highly sparse.

The sparseness of the major blocks of $[A]$ can be exploited for improved efficiency by using a more sophisticated refinement of the sparse solver than a BS provides. In the refined version of the sparse solver [fully sparse solver (FSS)], all computation and storage are performed only with the nonzero coefficients within the major blocks of $[A]$. In this section the major tasks involved in the application of FSS to the solution of Eq. (31) are briefly explained. The success of FSS is due to improved technologies, for example, sparse data storage formats, sparse reordering algorithms, sparse symbolic factorization, sparse numerical factorization, and forward/backward solution phases, and bookkeeping strategies that are ideal for implementation on a digital computer. More detailed information may be found in Refs. 8 and 9. The FSS equation-solving strategy and storage formats are demonstrated on the following symmetric system of equations:

$$[A]\{\Phi\} = \{F\} \tag{38}$$

$$[A] = \begin{bmatrix} A_{11} & A_{12} & A_{13} & 0 & A_{15} & A_{16} \\ A_{12} & A_{22} & 0 & A_{24} & 0 & 0 \\ A_{13} & 0 & A_{33} & 0 & 0 & 0 \\ 0 & A_{24} & 0 & A_{44} & A_{45} & 0 \\ A_{15} & 0 & 0 & A_{45} & A_{55} & 0 \\ A_{16} & 0 & 0 & 0 & 0 & A_{66} \end{bmatrix}$$

$$\{F\} = \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \\ F_6 \end{Bmatrix}, \qquad \{\Phi\} = \begin{Bmatrix} \Phi_1 \\ \Phi_2 \\ \Phi_3 \\ \Phi_4 \\ \Phi_5 \\ \Phi_6 \end{Bmatrix} \tag{39}$$

1. Sparse Data Storage Format

The sparse descriptions of the symmetric system matrix are fully described by the four one-dimensional vectors $\{OA\}$, $\{CN\}$, $\{DA\}$ and $\{NC\}$:

$$\{OA\} = \{A_{12}, A_{13}, A_{15}, A_{16}, A_{24}, A_{45}\}^T$$

$$\{CN\} = \{2, 3, 5, 6, 4, 5\}^T \tag{40}$$

$$\{DA\} = \{A_{11}, A_{22}, A_{33}, A_{44}, A_{55}, A_{66}\}^T$$

$$\{NC\} = \{4, 1, 0, 1, 0, 0\}^T \tag{41}$$

The pointer vectors $\{CN\}$ and $\{NC\}$ fully describe the nonzero pattern of $[A]$, whereas the numerical values of $[A]$ are contained in $\{OA\}$ and $\{DA\}$. Note that $\{DA\}$ contains the numerical values of the diagonal coefficients of $[A]$, and $\{OA\}$ contains the numerical values associated with the off-diagonal nonzero coefficients. The pointer vector $\{NC\}$ contains the number of off-diagonal nonzero coefficients in the upper triangular part of each row of $[A]$, and $\{CN\}$ contains the column numbers associated with the off-diagonal nonzero coefficients in $\{OA\}$. The columnwise storage order presented here was chosen because it is easier to implement than a rowwise order.

2. Sparse Reordering Algorithms

For maximum efficiency, one should never factorize and solve a sparse system such as Eq. (31) directly. The system of equations should be reordered using a reordering vector {REORDER} to obtain

$$[B]\{\Phi_R\} = \{F_R\} \tag{42}$$

The numerical values in $[B]$, $\{\Phi_R\}$ and $\{F_R\}$ are identical to those in $[A]$, $\{\Phi\}$, and $\{F\}$, respectively; however, these numerical values have been moved to different locations in the reordered system. The reordered system [Eq. (42)] may be factorized and solved much more efficiently than the original system [Eq. (31)]. The reordering vector, that is, {REORDER}, is then used to reorder vector $\{\Phi_R\}$ and recover the solution for $\{\Phi\}$. The purpose of reordering algorithms[10] [such as multiple minimum degrees (MMD) or nested dissection] is to provide {REORDER}.

Without reordering, the Gauss–Doolittle factorization of $[A]$ is

$$[A] = [\mathcal{L}][\mathcal{D}][\mathcal{L}]^{\mathcal{T}} \tag{43}$$

$$[\mathcal{L}]^T = \begin{bmatrix} 1 & \mathcal{L}_{12} & \mathcal{L}_{13} & 0 & \mathcal{L}_{15} & \mathcal{L}_{16} \\ 0 & 1 & \bar{\mathcal{L}}_{23} & \mathcal{L}_{24} & \bar{\mathcal{L}}_{25} & \bar{\mathcal{L}}_{26} \\ 0 & 0 & 1 & \bar{\mathcal{L}}_{34} & \bar{\mathcal{L}}_{35} & \bar{\mathcal{L}}_{36} \\ 0 & 0 & 0 & 1 & \mathcal{L}_{45} & \bar{\mathcal{L}}_{46} \\ 0 & 0 & 0 & 0 & 1 & \bar{\mathcal{L}}_{56} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{44}$$

During the factorization, many of the zero-value terms appearing in the upper triangular part of $[A]$ are nonzero in the upper triangular part of $[\mathcal{L}]^{\mathcal{T}}$. These extra nonzero terms created during the factorization of $[A]$ are referred to as fill-ins and are denoted by $\bar{\mathcal{L}}_{KI}$. In the upper triangular part of $[\mathcal{L}]^{\mathcal{T}}$ one has eight extra (or new) nonzero fill-ins. As a result, one defines

$$NF = 8, \qquad NL = NA + NF = 6 + 8 = 14 \tag{45}$$

In general, the number of nonzero coefficients in the upper triangular part of $[\mathcal{L}]^{\mathcal{T}}$, that is, NL, is much larger than those in $[A]$, that is, NA.

Applying the MMD reordering algorithm to $[A]$ will result in the following reordering vector:

$$\{\text{REORDER}\} = \{4, 6, 3, 5, 1, 2\}^T \tag{46}$$

The methodology used by MMD to determine {REORDER} involves dynamic programming techniques. (A full discussion of this methodology is beyond the scope of this paper.) Reordering the rows and columns in Eq. (31) according to {REORDER} gives

$$[B]\{\Phi_R\} = \{F_R\} \tag{47}$$

$$\{B\} = \begin{bmatrix} A_{44} & 0 & 0 & A_{45} & 0 & A_{24} \\ 0 & A_{66} & 0 & 0 & A_{16} & 0 \\ 0 & 0 & A_{33} & 0 & A_{13} & 0 \\ A_{45} & 0 & 0 & A_{55} & A_{15} & 0 \\ 0 & A_{16} & A_{13} & A_{45} & A_{11} & A_{12} \\ A_{24} & 0 & 0 & 0 & A_{12} & A_{22} \end{bmatrix}$$

$$\{F_R\} = \begin{Bmatrix} F_4 \\ F_6 \\ F_3 \\ F_5 \\ F_1 \\ F_2 \end{Bmatrix}, \qquad \{\Phi_R\} = \begin{Bmatrix} \Phi_4 \\ \Phi_6 \\ \Phi_3 \\ \Phi_5 \\ \Phi_1 \\ \Phi_2 \end{Bmatrix} \tag{48}$$

Now one factorizes $[B]$ to obtain

$$[B] = [\mathcal{L}][\mathcal{D}][\mathcal{L}]^T, \qquad [\mathcal{L}]^T = \begin{bmatrix} 1 & 0 & 0 & \mathcal{L}_{14} & 0 & \mathcal{L}_{16} \\ 0 & 1 & 0 & 0 & \mathcal{L}_{25} & 0 \\ 0 & 0 & 1 & 0 & \mathcal{L}_{35} & 0 \\ 0 & 0 & 0 & 1 & \mathcal{L}_{45} & \bar{\mathcal{L}}_{46} \\ 0 & 0 & 0 & 0 & 1 & \mathcal{L}_{56} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{49}$$

Note that the reordered matrix $[B]$ can be factorized more efficiently than $[A]$ because there is only one fill-in, that is, $\bar{\mathcal{L}}_{46}$, in $[\mathcal{L}]^T$.

3. Sparse Symbolic Factorization

The reordered matrix $[B]$ is fully described by the following four vectors:

$$\{NC\} = \{2, 1, 1, 1, 1, 0\}^T$$

$$\{DA\} = \{A_{44}, A_{66}, A_{33}, A_{55}, A_{11}, A_{22}\}^T \tag{50}$$

$$\{CN\} = \{4, 6, 5, 5, 5, 6\}^T$$

$$\{OA\} = \{A_{45}, A_{24}, A_{16}, A_{13}, A_{15}, A_{12}\}^T \tag{51}$$

Before performing the numerical factorization of the factored matrix, it is necessary to go through the sparse symbolic factorization so that 1) the nonzero pattern of $[\mathcal{L}]^{\mathcal{T}}$ can be determined (including the locations of fill-ins) and 2) the value of NL can be determined to allocate adequate RAM for $[\mathcal{L}]^{\mathcal{T}}$.

On completion of the sparse symbolic factorization phase, the pointer vectors $\{NC\}$ and $\{CN\}$ that fully describe the nonzero pattern in $[\mathcal{L}]^{\mathcal{T}}$ are completely known, and NL is determined:

$$\{NC\} = \{2, 1, 1, 2, 1, 0\}^T, \qquad \{CN\} = \{4, 6, 5, 5, 5, 6, 6\}^T$$

$$NL = NA + NF = 6 + 1 = 7 \tag{52}$$

4. Sparse Numerical Factorization Phase

In this phase, the numerical values in $[\mathcal{L}]^{\mathcal{T}}$ and $[\mathcal{D}]$ for the reordered matrix $[B]$ are computed:

$$\{OA\} = \{\mathcal{L}_{14}, \mathcal{L}_{16}, \mathcal{L}_{25}, \mathcal{L}_{35}, \mathcal{L}_{45}, \mathcal{L}_{46}, \mathcal{L}_{56}\}^T \tag{53}$$

$$\{DA\} = \{1, 1, 1, 1, 1, 1\}^T \tag{54}$$

$$\mathcal{D}_{II} = \left\{ \begin{array}{ll} B_{11} & (I = 1) \\ B_{II} - \displaystyle\sum_{K=1}^{I-1} \mathcal{D}_{KK}\mathcal{L}_{KI} & (I = 2, 3, \ldots, 6) \end{array} \right\} \tag{55}$$

$$\mathcal{L}_{IJ} = \left\{ \begin{array}{ll} \dfrac{B_{IJ}}{\mathcal{D}_{II}} & (I = 1, J = 2, \ldots, 6) \\ B_{IJ} - \displaystyle\sum_{K=1}^{I-1} \frac{\mathcal{D}_{KK}\mathcal{L}_{KI}\mathcal{L}_{KJ}}{\mathcal{D}_{II}} & (I \neq 1, J = I + 1, \ldots, 6) \end{array} \right\} \tag{56}$$

Note that the first column of $[\mathcal{D}]$ and $[\mathcal{L}]$ is computed first. Subsequent columns of $[\mathcal{L}]$ and $[\mathcal{D}]$ are computed using Eqs. (55) and (56). Because the diagonals of $[\mathcal{L}]$ are known a priori, the diagonal values of $[\mathcal{D}]$ are overwritten into the unit diagonal values of $[\mathcal{L}]$ for computational efficiency.

5. Forward/Backward Solution Phase

The solution to the reordered system [Eq. (47)] is obtained in two phases.

1) In the first phase (forward solution phase), an intermediate solution vector $\{\Phi_R\}$ is computed from the solution of the matrix equation

$$[\mathcal{L}]\{\bar{\Phi}_R\} = \{F_R\} \tag{57}$$

2) In the second phase (backward solution phase), the solution vector for the reordered system $\{\Phi_R\}$ is computed from the solution to the matrix equation

$$[\mathcal{D}][\mathcal{L}]^T\{\Phi_R\} = \{\bar{\Phi}_R\} \tag{58}$$

The vector $\{\Phi_R\}$ is reordered using {REORDER} to obtain the solution vector $\{\Phi\}$, which satisfies the system matrix equation (38).

V. Mode Solution

Exact solutions for the acoustic pressure field in the duct do not exist for arbitrary forms of the normalized impedance functions. It is possible, however, to develop a modal solution when these impedance functions are constant. This modal solution will be used to validate the accuracy of the equation solvers that were presented in the preceeding discussions. The single mode solution (in the duct of Fig. 1) for constant impedance is obtained by applying separation of variables to the boundary value problem described in Eqs. (2–5):

$$p(x, y, z) = [\vec{A}e^{-ik_z z} + \overleftarrow{A}e^{ik_z z}]P \tag{59}$$

$$\vec{A} = \frac{\int_0^W \int_0^H p_s P \,\mathrm{d}x\,\mathrm{d}y}{\int_0^W \int_0^H P^2 \,\mathrm{d}x\,\mathrm{d}y}, \qquad \overleftarrow{A} = \frac{(k_z\bar{\beta} - k)}{(k_z\bar{\beta} + k)}\vec{A}e^{-2ik_z L} \tag{60}$$

$$(k_z)^2 = (k)^2 - (k_x)^2 - (k_y)^2 \tag{61}$$

$$(k_y W)\tan(k_y W) - \frac{(ikW)[\beta^R + \beta^L]}{\left[1 + \beta^L\beta^R(kW)^2\big/(k_y W)^2\right]} = 0 \tag{62}$$

$$(k_x H)\tan(k_x H) - \frac{i(kH)[\beta^U + \beta^B]}{\left[1 + \beta^B\beta^U(kH)^2\big/(k_x H)^2\right]} = 0 \tag{63}$$

$$P = \left[\cos(k_x x) + \frac{ik\beta^B}{k_x}\sin(k_x x)\right]\left[\cos(k_y y) + \frac{ik\beta^L}{k_y}\sin(k_y y)\right] \tag{64}$$

Note that k_y and k_x are obtained by solving the transcendentalequations (62) and (63), respectively.

VI. Results

An in-house computer code that assembles the system matrix in either BS or FSS format was written and combined with the appropriate solver to provide the capability to solve three-dimensional aeroacoustics problems. RAM and CPU time statistics were compared for both solvers, and the noise suppression predicted from the solver solutions was compared to that predicted from the mode theory presented in the preceding section. Results were computed on an in-house SGI ORIGIN 2000 computer platform with 13 GB of RAM and eight processors. Results were run on a single processor with double-precision (64-bit) arithmetic. The duct geometry was identical to that of the NASA Langley Research Center flow impedance tube.[2] This three-dimensional duct has a square cross section 0.0508 m in width ($W = H = 0.0508$ m) and 0.812 m in length ($L = 0.812$ m). The upper and two side walls of the duct are rigid ($\zeta^L = \zeta^R = \zeta^U = \infty$), and the lower wall has a constant impedance. A much more complete description of the duct is given in Ref. 2. All calculations were performed at standard atmospheric conditions, and the source frequency was chosen to span the full range of frequencies currently of interest in duct liner research.

A. Rigid-Wall Duct Studies

In the rigid-wall duct, that is, $\zeta^L = \zeta^R = \zeta^B = \zeta^U = \infty$, the transcendental equations (62) and (63) have exact analytical solutions for the spanwise and transverse wave numbers:

$$k_y W = 0, \pi, 2\pi, \ldots \tag{65}$$

$$k_x H = 0, \pi, 2\pi, \ldots \tag{66}$$

These exact wave numbers [Eqs. (65) and (66)] have been used to determine the finite element grid required to resolve all cut-on (propagating) modes in the duct. Table 1 shows the minimum values of N, M, and Q required to resolve all cut-on modes for frequencies up to 21 kHz. Note that the current industry practice is to use scale models as small as one-fifth the size of full-scale engines. The full-scale frequency range of interest is typically 0.5–4.0 kHz. Thus, the frequency range of interest for the smallest scale, that is, a one-fifth scale model, is 2.5–20 kHz. Therefore, computational results for frequencies beyond 21 kHz are not presented in this paper. Further, in determining N, M, and Q in Table 1, the author has used the generally accepted rule that approximately 12 points per wavelength are required (in each of the three coordinate directions) to resolve accurately a cut-on mode. Note that at the highest frequency (21 kHz)

Table 1 Grid points required to resolve cut-on modes in the rigid-wall duct

f, kHz	N	M	Q	NMQ
4	6	6	114	4,104
7	12	12	200	28,800
11	18	18	313	101,412
14	24	24	399	229,824
17	30	30	484	435,600
21	36	36	599	776,304

Table 2 CPU time requirements in the rigid-wall duct

f, kHz	BS, min	FSS, min
4	0.01	0.02
7	0.51	0.44
11	7.64	3.57
14	54.00	16.77
17	254.70	60.17
21	——[a]	197.83

[a]Insufficient RAM for BS.

Table 3 RAM requirements in the rigid-wall duct

f, kHz	BS, GB	FSS, GB
4	0.006	0.006
7	0.110	0.110
11	0.800	0.510
14	3.170	1.600
17	9.290	3.780
21	——[a]	8.460

[a]Insufficient RAM for BS.

Table 4 Noise suppression in the rigid-wall duct

f, kHz	BS, dB	FSS, dB
4	0.000	0.000
7	0.000	0.000
11	0.001	0.001
14	0.001	0.001
17	0.002	0.002
21	——[a]	0.002

[a]Insufficient RAM for BS.

Table 5 Normalized resistance and reactance for the ceramic material

f, kHz	R	χ
4	0.36	0.76
7	2.61	−3.66
11	0.67	1.14
14	1.60	−2.26
17	0.59	0.54
21	1.23	−1.59

Table 6 Transverse wave numbers in the soft-wall duct

f, kHz	k_{x0}	k_x
4	π/H	$25.57+1.94i$
7	$2\pi/H$	$13.48+25.95i$
11	$3\pi/H$	$27.72+1.61i$
14	$4\pi/H$	$36.97+5.09i$
17	$5\pi/H$	$29.85+1.09i$
21	$6\pi/H$	$33.54+2.28i$

Table 7 Noise suppression in the soft-wall duct

f, kHz	Mode theory, dB	BS, dB	FSS, dB
4	14.68	14.62	14.62
7	76.19	75.29	75.29
11	5.42	5.38	5.38
14	31.09	30.98	30.98
17	137.36	137.14	137.14
21	7.37	7.34	7.34

a matrix order (NMQ) of 776,304 is required to resolve accurately all cut-on modes.

The CPU time requirements in minutes for BS and FSS are given in Table 2. The CPU time shown in Table 2 is that required to obtain the solution vector $\{\Phi\}$ in the rigid-wall duct after the system matrix was assembled. The source was a plane wave source ($p_s = 1$), and the normalized exit impedance was chosen so that the duct was reflection free ($\bar{\zeta} = 1$). Although the CPU time for BS and FSS are comparable for frequencies at and below 7 kHz, FSS is considerably more efficient at the higher frequencies. Note that at a frequency of 17 kHz, BS requires 4.25 h of CPU time compared to 1 h for FSS. In addition, BS was unable to obtain a solution at a frequency of 21 kHz because the RAM requirements for BS exceeded the 13-GB limit of the computer platform.

RAM requirements (in gigabytes) for BS and FSS for the rigid-wall duct are tabulated in Table 3. RAM requirements for the two solvers are comparable for frequencies at and below 7 kHz; however, FSS consumes considerably less RAM at the higher frequencies. For example, BS requires 9.29 GB of RAM at 17 kHz, whereas FSS requires only 3.78 GB. Note that the RAM requirement of BS at a frequency of 21 kHz exceeded the 13-GB limit of the computer platform. On the other hand, the FSS obtained the solution vector using only 8.46 GB of RAM at this frequency.

To check the accuracy of the solution vectors obtained from the two solvers, the author has used the noise suppression in decibels as a metric. This metric is physically more meaningful than the error norm of the computed solution vector because it is the quantity perceived by the human ear as the noise source propagates down the duct. The solution vector $\{\Phi\}$ obtained from each equation solver was used to compute numerically the noise suppression levels ΔdB [see Eq. (6)], for the reflection-free plane wave source solution. It is easily shown that the reflection-free, plane wave source, mode solution in the rigid-wall duct is

$$p = e^{-ikz} \tag{67}$$

so there is zero noise suppression in the rigid-wall duct, that is, $\Delta\text{dB} = 0$. Equation (67) is easily derived from Eqs. (59–64) with $\beta^L = \beta^R = 0 + 0i$, $k_x = k_y = 0$, $p_s = 1$, and $\bar{\zeta} = 1$. The noise suppression levels ΔdB computed from the solver solutions for the rigid-wall duct are tabulated in Table 4 to three decimal digits of accuracy. The noise suppression levels are in excellent agreement with the mode solution value of zero decibels at all frequencies. As expected, BS and FSS give identical results for the noise suppression.

B. Soft-Wall Duct Studies

The acoustic pressure disturbance in the duct is more interesting when the lower wall of the duct is lined with a sound absorbing material. Results have been computed for a soft-wall duct containing a ceramic material, that is, liner. Table 5 shows the predicted resistance R and predicted reactance χ of the ceramic material, that is, $\zeta^B = R + i\chi$. It is easily shown that the reflection-free, nonplanar source, single-mode solution in the soft-wall duct is

$$p(x, y, z) = \left[\cos(k_x x) + \left(ik\beta^B\big/k_x\right)\sin(k_x x)\right]e^{-ik_z z} \tag{68}$$

Equation (68) is easily derived from Eqs. (59–64) with $k_y = 0$, $\bar{\zeta} = k/k_z$, and $p_s = [\cos(k_x x) + (ik\beta^B/k_x)\sin(k_x x)]$. In this case, the noise suppression is a finite value, but the transverse wave number k_x is difficult to determine because the transcendental equation (63) cannot be solved analytically for k_x. Consequently, the solution to this transcendental equation was obtained using a Newton–Raphson iterative method (see Ref. 7) with the rigid-wall solutions as the initial guesses. Table 6 shows the transverse wave number k_x obtained from the Newton–Raphson solution to the transcendental equation (63). The initial guess (k_{x0}) used to obtain the transverse wave number k_x is also specified in Table 6.

RAM and CPU time were computed for the soft-wall duct described in the preceding paragraph. As expected, the RAM and CPU time for the soft-wall duct were identical to those of the rigid-wall duct (see Tables 2 and 3) and are, therefore, not presented. Table 7 compares the FSS and BS noise suppression levels to those of mode theory in the soft-wall duct when the duct is 1 m in length ($L = 1$). Noise suppression levels obtained from the BS and FSS solution are identical, and they are in excellent agreement with the mode theory.

In a recent paper,[2] a numerical method for extracting the impedance of an acoustic material located in a two-dimensional duct

Table 8 Educed resistance and reactance spectrum of the ceramic liner

	Resistance R		Reactance χ	
f, kHz	Known	FSS educed	Known	FSS educed
4	0.36	0.33	0.76	0.71
7	2.61	2.67	−3.66	−3.62
11	0.67	0.63	1.14	1.09
14	1.60	1.54	−2.26	−2.21

was developed and validated. The impedance extraction method required input of the source pressure p_s, the exit impedance $\bar{\zeta}$, and upper wall pressure. The FSS computer code described here has been linked to this impedance extraction technique to provide the capability to extract impedances in three-dimensional sound fields. Because measured three-dimensional data were not available, the inputs required to educe the impedance were obtained from the mode solution given in Eq. (68) with $k_y = 0$ and k_x given in Table 6. Resistance and reactance educed using FSS are compared to the known values in Table 8. Excellent comparisons were obtained between the known and educed resistance and reactance values. Impedance eductions for frequencies above 14 kHz were not performed because the CPU times to educe the impedance became excessive.

VII. Conclusions

A zero flow, fully three-dimensional, rectangular duct code (FSS) that accounts for variable surface impedance liners and is capable of spanning the full range of frequencies currently of interest in nacelle liner research has been developed. FSS uses a state-of-the-art, fully sparse, equation solver (for solution of linear systems of equations) to acquire the capability to study high-frequency sound waves that may require nearly 800,000 grid points for resolution. Over the range of frequencies of current interest in nacelle liner research, noise suppression levels predicted from FSS are in excellent agreement with those predicted from mode theory. The single-processor performance (RAM and CPU time) of FSS has been compared to that of the more commonly used BS. BS and FSS have equal performance for frequencies equal to or below 7 kHz; however, the performance of FSS relative to BS increases monotonically with frequency. Results have shown that BS is 4.25 times slower and consumes 2.5 times more RAM than FSS at 17 kHz. As an application, FSS was combined with an optimization algorithm and used to educe the impedance spectrum of a ceramic liner, for frequencies up to 14 kHz. Good comparison between the FSS educed and known impedance spectrum was observed. The primary problem with using FSS to perform optimization studies at frequencies above 14 kHz is excessive CPU time. The results of this paper support the recommendation, therefore, that research be directed toward exploitation of the multiprocessor capability of FSS to further reduce CPU time so that optimization studies above 14 kHz are tractable.

Acknowledgment

The author gratefully acknowledges Duc T. Nguyen of Old Dominion University for his useful suggestions and assistance with the sparse assembly.

References

[1] Bielak, G., Kunze, R., Murray, P., Premo, J., Kosanchick, M., Hersh, A., Celano, B., Yu, J., Kwan, H., Chiou, S., Kelly, J., Betts, J., Follet, J., and Thomas, R., "Advanced Nacelle Acoustic Lining Concepts Development," NASA Contract Interim Rept. NAS1-97040, June 2000.

[2] Watson, W. R., Jones, M. G., and Parrott, T. L., "Validation of an Impedance Eduction Method in Flow," *AIAA Journal*, Vol. 37, No. 7, 1999, pp. 818–824.

[3] Dougherty, R. P., "A Wave-Splitting Technique for Nacelle Acoustic Propagation," *Proceedings of the 3rd AIAA/CEAS Aeroacoustic Conference*, AIAA, Reston, VA, 1997, pp. 550–556.

[4] Morse, P. M., and Ingard, K. U., *Theoretical Acoustics*, McGraw–Hill, New York, 1968, pp. 495, 496.

[5] Chandrakant, S., and Abel, J. F., *Introduction to the Finite Element Method*, Van Nostrand Reinhold, New York, 1972, pp. 176, 177, 192, 193.

[6] Zienkiewicz, O. C., *The Finite Element Method in Engineering Science*, McGraw–Hill, London, 1971, pp. 1–131.

[7] James, M. L., Smith, G. M., and Wolfold, J. C., *Applied Numerical Methods for Digital Computation*, 3rd ed., Harper and Row, New York, 1985, pp. 94–96, 178–185.

[8] Baddourah, M., "Solution of Matrix Equations Using Sparse Techniques," *The Role of Computers in Research and Development at Langley Research Center*, NASA CP-10159, 1994, pp. 147–155.

[9] Pissanetzsky, S., "Gauss Elimination with Supersparse Matrices," Brookhaven National Lab., Rept. BNL 26773, Upton, NY, Aug. 1979.

[10] George, A., and Liu, J. W., *Computer Solution of Large Sparse Positive Definite Systems*, Prentice–Hall, Englewood Cliffs, NJ, 1981, Chap. 5.

P. J. Morris
Associate Editor