JOURNAL OF COMPUTATIONAL PHYSICS **127,** 52–65 (1996)
ARTICLE NO. 0157

Nonreflecting Boundary Conditions for Time-Dependent Scattering

MARCUS J. GROTE*,[1] AND JOSEPH B. KELLER†,[2]

*Scientific Computing and Computational Mathematics, Stanford University, Stanford, California 94305-2140; †Departments of Mathematics and Mechanical Engineering, Stanford University, Stanford, California 94305-2125

Received May 19, 1995; revised November 6, 1995

An exact nonreflecting boundary condition was derived previously for use with the time dependent wave equation in three space dimensions. Here it is shown how to combine that boundary condition with finite difference methods and finite element methods. Uniqueness of the solution is proved, stability issues are discussed, and improvements are proposed for numerical computation. Numerical examples are presented which demonstrate the improvement in accuracy over standard methods. © 1996 Academic Press, Inc.

1. INTRODUCTION

We wish to calculate numerically the time dependent field $u(\boldsymbol{x}, t)$ scattered from a bounded scattering region in three-dimensional space. In this region, there may be one or more scatterers, and the equation for u may have variable coefficients and nonlinear terms. As usual, we surround the scattering region by an artificial boundary $\mathcal{B}$ and confine the computation to the region Ω bounded by $\mathcal{B}$. Then, to formulate a problem for u in Ω, we make u satisfy a boundary condition on $\mathcal{B}$. The boundary conditions commonly imposed produce spurious reflections from $\mathcal{B}$. To avoid this spurious reflection, we have devised an exact nonreflecting boundary condition [1]. In doing so, we chose $\mathcal{B}$ to be a sphere of radius a, and we assumed that u satisfied the wave equation outside $\mathcal{B}$. The boundary condition is local in time and nonlocal on $\mathcal{B}$, and it does not involve high-order derivatives on $\mathcal{B}$.

Now we shall show how to combine this boundary condition with the finite difference method, or with the finite element method, to obtain a computational problem in Ω. In formulating the computational problem, it is necessary to truncate the exact boundary condition and to retain only a finite number of spherical harmonics. We shall show how to modify this truncated boundary condition to improve its accuracy. We shall also examine the stability of the ordinary differential equation which occurs in the boundary condition.

Finally, we shall solve a sequence of scattering problems by using an explicit finite difference method and our boundary condition. We shall also solve the same problems by using two of the standard artificial boundary conditions. Comparison of these solutions with the "exact" solution, obtained by computing in a very large domain so that spurious reflections are postponed, shows that our boundary condition is much more accurate than the standard ones. Our boundary condition also has the advantage that it remains accurate when the radius of the artificial boundary is made smaller, so that the computational domain is reduced.

Previously, we proved that the problem in Ω with our exact boundary condition is satisfied by the restriction to Ω of the solution of the initial-boundary value problem in the infinite region [1]. Now we shall prove that the problem in Ω has a unique solution.

2. FORMULATION

We consider time-dependent scattering from a bounded scattering region in three-dimensional space. We surround this region by a sphere $\mathcal{B}$ of radius a. Outside $\mathcal{B}$, we assume that the scattered field $u(\mathbf{x}, t)$ satisfies

$$\frac{\partial^2 u}{\partial t^2} - \Delta u = 0, \quad u(\mathbf{x}, 0) = 0, \quad \partial_t u(\mathbf{x}, 0) = 0. \tag{2.1}$$

We suppose that the domain Ω within $\mathcal{B}$ is bounded internally by the smooth surface Γ. In Ω, we consider the following simple model problem:

$$\frac{\partial^2 u}{\partial t^2} - \nabla \cdot (c\nabla u) = f, \quad \text{in } \Omega \times (0, T) \tag{2.2}$$

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega \tag{2.3}$$

$$\frac{\partial}{\partial t} u(\mathbf{x}, 0) = \dot{u}_0(\mathbf{x}), \quad \mathbf{x} \in \Omega \tag{2.4}$$

$$c\frac{\partial u}{\partial \nu} = h \quad \text{on } \Gamma \times (0, T) \tag{2.5}$$

[1] Supported by an IBM graduate fellowship. Email: grote@cims.nyu.edu.

[2] Supported in part by AFOSR, NSF, and ONR. Email: keller@math.stanford.edu.

0021-9991/96 $18.00

$$\left(\frac{\partial}{\partial r}+\frac{\partial}{\partial t}\right)[ru] = -\sum_{n=1}^{\infty}\sum_{m=-n}^{n} \mathbf{c}_n \cdot \mathbf{z}_{nm} Y_{nm} \quad \text{on } \mathscr{B}\times(0,T) \tag{2.6}$$

$$\frac{d}{dt}\mathbf{z}_{nm}(t) = \mathbf{A}_n \mathbf{z}_{nm}(t) + \mathbf{u}_{nm}(t), \quad \mathbf{z}_{nm}(0) = 0. \tag{2.7}$$

Here ν is the outward normal on Γ, and $Y_{nm}(\theta, \varphi)$ is the nmth spherical harmonic normalized over the *unit* sphere,

$$\begin{aligned} Y_{nm}(\theta,\varphi) = {} & [(2n+1)(n-|m|)!/4\pi(n+|m|)!]^{1/2} \\ & \times e^{im\varphi} P_n^{|m|}(\cos\theta). \end{aligned} \tag{2.8}$$

If the problem considered is real, one can use real instead of complex spherical harmonics. In that case everything remains the same, but the normalization constant in (2.8) will change. In (2.2) we require that $c = c(\mathbf{x}) > 0$. The source term $f(\mathbf{x}, t, u, \nabla u)$ may be nonlinear.

Equation (2.6) is the exact nonreflecting boundary condition which was derived in [1]. It involves the vector functions $\mathbf{z}_{nm}(t)$, which are the solutions of the *linear first-order ordinary* differential equations (2.7). In (2.6) the vectors $\mathbf{c}_n = \{c_n^j\}$ are defined by

$$c_n^j = \frac{n(n+1)j}{2a^j}, \tag{2.9}$$

and in (2.7) the $n \times m$ matrices $\mathbf{A}_n = [A_n^{ij}]$ are defined by

$$A_n^{ij} = \begin{cases} -n(n+1)/(2a^j) & \text{if } i = 1, \\ (n+i)(n+1-i)/(2i), & \text{if } i = j+1, \\ 0, & \text{otherwise.} \end{cases} \tag{2.10}$$

The vector function $\mathbf{u}_{nm}(t)$ in (2.7) has the nmth Fourier coefficient of u on $\mathscr{B}$ as its only nonzero component:

$$\mathbf{u}_{nm} = [(Y_{nm}, u|_{r=a}), 0, ..., 0]^{\mathrm{T}}. \tag{2.11}$$

Here

$$(Y_{nm}, u|_{r=a}) = \int_0^{2\pi}\int_0^{\pi} \overline{Y_{nm}(\theta,\varphi)}\, u(a,\theta,\varphi,t)\sin\theta\, d\theta\, d\varphi \tag{2.12}$$

Both $\mathbf{u}_{nm}$ and $\mathbf{z}_{nm}$ vary in time, whereas $\mathbf{c}_n$ and $\mathbf{A}_n$ are constants.

It was shown in [1] that the restriction to Ω of the solution to the initial-boundary value problem (2.2)–(2.5), with Ω replaced by the infinite region outside Γ, solves the initial-boundary value problem (2.2)–(2.7). Now we shall show that the solution of this problem is unique.

THEOREM 2.1. *Suppose that the initial-boundary value problem* (2.2)–(2.5) *in the infinite region outside* Γ *has a unique smooth solution. Then so does* (2.2)–(2.7). *The two solutions coincide in* Ω.

The proof of Theorem 2.1 is given in Appendix A.

The Cauchy problem (2.2)–(2.5) with $f \equiv 0$ and $c \equiv 1$ in the infinite region outside Γ is well-posed with respect to the initial data $u_0(\mathbf{x})$ and $\dot u_0(\mathbf{x})$. Since its solution coincides with that of the initial-boundary value problem (2.2)–(2.7), we immediately obtain the following result:

COROLLARY 2.1. *The initial-boundary value problem* (2.2)–(2.7) *with* $f \equiv 0$ *and* $c \equiv 1$ *is well-posed with respect to the initial data* $u_0(\mathbf{x})$ *and* $\dot u_0(\mathbf{x})$.

We remark that Corollary 2.1 does not imply that (2.2)–(2.7) is well-posed with respect to arbitrary perturbations in the boundary condition (2.6).

3. FINITE ELEMENT FORMULATION

We shall now derive the finite element formulation for the problem (2.2)–(2.7) in the computational domain Ω. To derive the weak form of the problem, we denote by $\mathscr{V}$ the Sobolev space $H^1(\Omega)$, which contains square-integrable functions with square-integrable first derivatives. Next we define the sesquilinear forms

$$\mathscr{A}[w,u] = \int_\Omega \nabla\overline{w}\cdot(c\nabla u)\, d\Omega, \tag{3.1}$$

$$(w,u) = \int_\Omega \overline{w} u\, d\Omega, \tag{3.2}$$

$$(w,u)_{\mathscr{B}} = \int_{\mathscr{B}} \overline{w} u\, d\mathscr{B}, \tag{3.3}$$

$$(w,u)_\Gamma = \int_\Gamma \overline{w} u\, d\Gamma. \tag{3.4}$$

We multiply (2.2)–(2.4) by a weighting function $\overline{w} \in \mathscr{V}$ and integrate over Ω. Then we use integration by parts in the integrated (2.2) to get

$$(w,\ddot u) + \mathscr{A}[w,u] - (w, \partial_r u)_{\mathscr{B}} = (w,f) + (w,h)_\Gamma. \tag{3.5}$$

Next, we use (2.6) to eliminate $\partial_r u$ on $\mathscr{B}$ from (3.5).

These calculations lead to the weak form of the problem, which can be stated as follows:

Find $u(t) \in \mathscr{V}$ such that for all $w \in \mathscr{V}$,

$$\begin{aligned} (w,\ddot u) + \mathscr{A}[w,u] + \left(w, \dot u + \frac{u}{a}\right)_{\mathscr{B}} = {} & (w,f) \\ & + (w,h)_\Gamma - \frac{1}{a}\sum_{n=1}^{\infty}\sum_{m=-n}^{n} \mathbf{c}_n\cdot\mathbf{z}_{nm}(w,Y_{nm})_{\mathscr{B}} \end{aligned} \tag{3.6}$$

$$(w, u(0, \cdot)) = (w, u_0) \tag{3.7}$$

$$(w, \dot{u}(0, \cdot)) = (w, \dot{u}_0) \tag{3.8}$$

$$\dot{\mathbf{z}}_{nm}(t) = \mathbf{A}_n \mathbf{z}_{nm}(t) + \mathbf{u}_{nm}(t), \quad \mathbf{z}_{nm}(0) = 0. \tag{3.9}$$

The finite element method [2] is obtained by approximating the weak form (3.6)–(3.9). The domain Ω is discretized into a finite number of elements, and each element is associated with a finite number of nodes. Then u and w are approximated by

$$u^h(\mathbf{x}, t) = \sum_{I \in \eta} d_I(t) N_I(\mathbf{x}), \tag{3.10}$$

$$w^h(\mathbf{x}) = \sum_{I \in \eta} w_I N_I(\mathbf{x}). \tag{3.11}$$

Here η is the set of nodes, N_I is the shape function associated with node I, and $d_I(t)$ and w_I are coefficients. We now substitute (3.10) and (3.11) into (3.6)–(3.9) with the sum over n truncated at some finite value N and require the resulting equation to hold for all values of w_I. This yields the finite element matrix form of the problem for the vector of unknowns $d = \mathbf{d}(t)$:

$$\mathbf{M}\ddot{\mathbf{d}} + \mathbf{C}\dot{\mathbf{d}} + \mathbf{K}\mathbf{d} = \mathbf{f}, \quad t > 0 \tag{3.12}$$

$$\dot{\mathbf{z}}_{nm}(t) = \mathbf{A}_n \mathbf{z}_{nm}(t) + \mathbf{u}_{nm}(t), \quad \mathbf{z}_{nm}(0) = 0 \tag{3.13}$$

$$\mathbf{d}(0) = \mathbf{d}_0 \tag{3.14}$$

$$\dot{\mathbf{d}}(0) = \mathbf{v}_0. \tag{3.15}$$

The matrices **M**, **K**, and **C**, are defined by

$$\mathbf{M} = [M_{IJ}], \quad \mathbf{K} = [K_{IJ}], \quad \mathbf{C} = [C_{IJ}], \tag{3.16}$$

$$M_{IJ} = (N_I, N_J), \quad K_{IJ} = \mathscr{A}\,[N_I, N_J], \quad C_{IJ} = \frac{1}{a}(N_I, N_J)_{\mathscr{B}}. \tag{3.17}$$

The vectors **f**, **d**, $\mathbf{d}_0$, and $\mathbf{v}_0$ are defined by

$$\mathbf{f} = \{f_I\}, \quad \mathbf{d}_0 = \{d_{J0}\}, \quad \mathbf{v}_0 = \{v_{J0}\}, \tag{3.18}$$

$$f_I = (N_I, f) + (N_I, h)_\Gamma - \frac{1}{a} \sum_{n=1}^{\infty} \sum_{m=-n}^{n} \mathbf{c}_n \cdot \mathbf{z}_{nm} (N_I, Y_{nm})_{\mathscr{B}}, \tag{3.19}$$

$$d_{J0} = (u_0, N_J), \quad v_{J0} = (\dot{u}_0, N_J). \tag{3.20}$$

The quantities **d**, $\dot{\mathbf{d}}$, and $\ddot{\mathbf{d}}$ are the displacement, the velocity, and the acceleration vectors, respectively. **M** is the mass matrix, **K** is the stiffness matrix, and **C** is a damping term due to the absorbing boundary condition. We note that **M** is symmetric and positive definite, and that both **K** and **C** are symmetric and positive semi-definite. Thus, we can use a standard time-marching scheme from the Newmark family [3], such as the central difference method, to integrate (3.12). The solution of (3.13) is computed concurrently, for instance, using the explicit second-order Adams method, or the implicit trapezoidal rule [4]. The matrix **C** is almost empty, since only terms along the boundary contribute to its nonzero entries.

4. FINITE DIFFERENCE FORMULATIONS

Instead of using the finite element method, we can use a finite difference method to solve (2.2)–(2.7). We shall now describe how to do this, choosing $c(\mathbf{x}) = 1$ in Ω for simplicity.

We opt for the leap-frog method, which is a standard *explicit* time-marching method for the wave equation. The wave equation (2.2) is discretized both in time and in space at $(\mathbf{x}, t)$ using second-order centered finite differences. We denote by U^h the numerical grid function. Let U^k be the numerical solution and f^k the source f at some grid point $\mathbf{x}$ at time $t_k = k\Delta t$. Then the basic step to advance the numerical solution in time is

$$U^{k+1} = 2U^k - U^{k-1} + (\Delta t)^2(\Delta^h U^k + f^k), \quad k = 1, 2, ... \tag{4.1}$$

Here Δ^h denotes a finite difference approximation to the Laplacian.

The boundary condition (2.6) is necessary when we wish to advance the numerical solution on $\mathscr{B}$ using (4.1). Indeed, the radial part of Δ^h requires values of U^h outside Ω. Let $\Delta_r u = r^{-2}\partial_r(r^2\delta_r u)$ denote the radial part of the Laplacian. Next, let r_l denote the lth grid point in the radial direction. Hence, $r_{l+1} = r_l + \Delta r$ and $r_{l+1/2} = r_l + \Delta r/2$. Then a second-order finite difference approximation to Δ_r is

$$\Delta_r^h U_l = \frac{r_{l+1/2}^2 U_{l+1} - (r_{l+1/2}^2 + r_{l-1/2}^2) U_l + r_{l-1/2}^2 U_{l-1}}{(r_l \Delta r)^2}. \tag{4.2}$$

To calculate $\Delta_r^h U^h$ at $r_l = a$, we need U_{l+1} which lies outside Ω. We eliminate it by using the finite difference approximation to (2.6) at $r_l = a$, θ_i, φ_j, with the sum over n truncated at some finite value N. This yields

$$\frac{r_{l+1} U_{l+1}^k - r_{l-1} U_{l-1}^k}{2\Delta r} + \frac{r_l (U_l^{k+1} - U_l^{k-1})}{2\Delta t} = -\sum_{n=1}^{N} \sum_{m=-n}^{n} \mathbf{c}_n \cdot \mathbf{z}_{nm}^k \, Y_{nm}(\theta_i, \varphi_j). \tag{4.3}$$

Thus we have the two equations (4.1) and (4.3) for the two unknowns U_l^{k+1} and U_{l+1}^k, which enables us to solve for U_l^{k+1} on $\mathscr{B}$. We underline that to compute U^{k+1} on $\mathscr{B}$, we only need the values of $\mathbf{z}_{nm}(t)$ at $t = t_k$, because both the differential equation and the boundary condition are discretized in time about $t = t_k$.

The numerical solution $\mathbf{z}_{nm}^h$ to the ordinary differential equation (2.7) is computed concurrently with U^h. Because the eigenvalues of $\mathbf{A}_n$ lie in the left half of the complex plane (see Section 6), we opt for the implicit second-order trapezoidal rule [4, Section II.7]:

$$\left(I - \frac{\Delta t}{2}\mathbf{A}_n\right)\mathbf{z}_{nm}^{k+1} = \left(I + \frac{\Delta t}{2}\mathbf{A}_n\right)\mathbf{z}_{nm}^{k} + \frac{\Delta t}{2}(\mathbf{u}_{nm}^{k+1} + \mathbf{u}_{nm}^{k}). \tag{4.4}$$

The inner products $(Y_{nm}, U^h|_{r=a})$, needed for $\mathbf{u}_{nm}^k$ and $\mathbf{u}_{nm}^{k+1}$, are computed using Simpson's fourth-order quadrature rule. Since the trapezoidal rule is unconditionally stable, there is no restriction on Δt in the integration of (2.7). The work required in solving the linear systems (4.4) is negligible, because the matrices $\mathbf{A}_n$ are very small and remain constant in time. We have also implemented the explicit second-order Adams method [4, Section III.1] to integrate (2.7), which lead to similar results. The additional stability constraint on Δt, imposed by the explicit Adams method, is usually less restrictive than that due to the leapfrog method. Of course, this depends on N and on the mesh size used in the computation.

The complete algorithm proceeds as follows:

0. Initialize U^h at t_0 and t_1, and set $\mathbf{z}_{nm}^0$ and $\mathbf{z}_{nm}^1$ to zero.
1. Compute U^{k+1} at all inner points in Ω using (4.1).
2. Compute U^{k+1} at $\mathscr{B}$ using (4.1) and (4.3).
3. Compute $\mathbf{z}_{nm}^{k+1}$ using (4.4), and go to 1.

5. MODIFIED BOUNDARY CONDITIONS

In computation, the sum over n in (2.6) must be truncated at a finite value N. Then the boundary condition becomes inexact for the modes beyond the point of truncation. It reduces to $(\partial_r + \partial_t)[ru_{nm}] = 0$ at $r = a$ for the modes $u_{nm} = (U, Y_{nm})$, $n > N$. This raises the question whether we can replace this inexact condition by a more accurate one, as we did in [5] for the reduced wave equation. We shall show how to modify the truncated boundary condition so that it remains exact for the low modes $n \leq N$, but becomes more accurate for the high modes $n > N$.

We now briefly recall the derivation of (2.6)–(2.7). First, we introduce the polar coordinates r, θ, φ. Next, we define G_n, the integral operator:

Definition 5.1.

$$G_n[u](r,t) \equiv \begin{cases} ru(r,t), & \text{if } n = 0 \\ -r\displaystyle\int_r^\infty \frac{(r^2 - s^2)^{n-1}u(s,t)}{(2s)^{n-1}(n-1)!}\,ds, & \text{if } n \geq 1. \end{cases} \tag{5.1}$$

We apply G_n to $u_{nm}(r, t) = (U, Y_{nm})$ and find (Lemma 3.2 in [1]) that $G_n[u_{nm}]$ satisfies the one-dimensional wave equation outside $\mathscr{B}$:

$$\frac{\partial^2}{\partial t^2}G_n[u](r,t) - \frac{\partial^2}{\partial r^2}G_n[u](r,t) = 0, \quad r \geq a. \tag{5.2}$$

We note that $G_n[u_{nm}]$ is finite for any t, because $u(\mathbf{x}, t)$ vanishes for $\|\mathbf{x}\| \geq a + t$. Since the solution u is purely outgoing, $G_n[u_{nm}](r, t)$ is outgoing too. Then from (5.2) it has the form $g(t - r)$, and thus

$$\left(\frac{\partial}{\partial t} + \frac{\partial}{\partial r}\right)G_n[u_{nm}](r,t) = 0, \quad r \geq a. \tag{5.3}$$

An exact boundary condition is (5.3) applied to $u_{nm}(r, t)$ at $r = a$.

To use this boundary condition in computation, we must reformulate (5.3) and derive an equivalent but more tractable form, which does not involve integrals or high-order derivatives. In (Lemma 7.1 in [1]) it was shown that

$$ru_{nm}(r,t) = (-1)^n \sum_{j=0}^{n} \frac{\gamma_{nj}}{r^j}\left(-\frac{\partial}{\partial r}\right)^{n-j} G_n[u_{nm}](r,t), \tag{5.4}$$

where the constants γ_{nj} are defined by

$$\gamma_{nj} = \frac{(n+j)!}{(n-j)!j!2^j}. \tag{5.5}$$

We now apply $\partial_r + \partial_t$ to (5.4), and use (5.3) to obtain

$$\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right)[ru_{nm}(r,t)] = (-1)^n \sum_{j=1}^{n} \frac{-j\gamma_{nj}}{r^{j+1}}\frac{\partial^{n-j}}{\partial t^{n-j}}G_n[u_{nm}](r,t), \quad r \geq a. \tag{5.6}$$

To simplify the notation, we define

$$w_{nm}(t) = (-1)^n G_n[(u, Y_{nm})](a,t)/a. \tag{5.7}$$

Next, we set $r = a$ in (5.6), multiply by Y_{nm}, and sum over n and m. This yields

$$\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right)[r\, u] = -\sum_{n=1}^{\infty} \sum_{m=-n}^{n} Y_{nm} \sum_{j=1}^{n} \frac{j\gamma_{nj}}{a^j} \frac{d^{n-j}}{dt^{n-j}} w_{nm}, \quad r = a. \tag{5.8}$$

By using (5.3) in (5.4) and definition (5.7) for $w_{nm}(t)$, we find that w_{nm} is the solution of the *ordinary* differential equation,

$$\frac{d^n}{dt^n} w_{nm}(t) = (u, Y_{nm})(a, t) - \sum_{j=1}^{n} \frac{\gamma_{nj}}{a^j} \frac{d^{n-j}}{dt^{n-j}} w_{nm}(t), \tag{5.9}$$

with initial conditions

$$w_{nm}(0) = \frac{d}{dt} w_{nm}(0) = \cdots = \frac{d^{n-1}}{dt^{n-1}} w_{nm}(0) = 0. \tag{5.10}$$

We note that (5.8) and (5.9) appear in a simpler form than in (Theorem 7.1 of [1]), because we have multiplied the original $w_{nm}(t)$ by $(-1)^n/a$. To obtain the final formulation (2.6), we rescale (5.9) to eliminate the large γ_{nj} that appear in (5.8) and (5.9) by setting $\mathbf{z}_{nm}(t) = [z^1_{nm}(t), ..., z^n_{nm}(t)]^{\mathrm{T}}$, with

$$z^j_{nm}(t) = \frac{\gamma_{nj}}{\gamma_{n1}} \frac{d^{n-j}}{dt^{n-j}} w_{nm}(t), \quad j = 1, ..., n. \tag{5.11}$$

In deriving (5.6), we applied $B_1 = \partial_r + \partial_t$ to (5.4). This operator annihilates any spherically symmetric outgoing wave, such as the leading term of the large r expansion of $ru(\mathbf{x}, t)$, where u is any solution of (2.1):

$$ru(r, \theta, \phi, t) = \sum_{j=0}^{\infty} \frac{g_j(t - r, \theta, \phi)}{r^j}, \quad r \geq a. \tag{5.12}$$

Bayliss and Turkel [6] derived a sequence of local operators, which annihilate increasingly many leading terms in (5.12). We shall now derive an equivalent sequence of operators B_l, which look slightly different because we work with ru instead of u. Then we shall show how to use them to modify the boundary condition (5.6).

To derive these operators, we note that for any $m \geq 0$

$$\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t} + \frac{m}{r}\right) \frac{g(t - r)}{r^m} = 0. \tag{5.13}$$

We use (5.12) and (5.13) with $m = 2$ to obtain

$$\begin{aligned} B_2[ru] &= \left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t} + \frac{2}{r}\right)\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right)[ru] \\ &= \sum_{j=2}^{\infty} \frac{j(j-1)g_j}{r^{j+2}}. \end{aligned} \tag{5.14}$$

Next we use (5.12) and (5.13) with $m = 3$ to get

$$\begin{aligned} B_3[ru] &= \left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t} + \frac{4}{r}\right)\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t} + \frac{2}{r}\right)\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right)[ru] \\ &= \sum_{j=3}^{\infty} \frac{-j(j-1)(j-2)g_j}{r^{j+3}}. \end{aligned} \tag{5.15}$$

For $l \geq 1$, we define the operator B_l

$$B_l = \prod_{j=0}^{l-1} \left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t} + \frac{2j}{r}\right), \tag{5.16}$$

where the rightmost term is $j = 0$. Upon applying B_l to (5.12), we find that

$$B_l[ru] = \mathcal{O}(r^{-2l}), \quad r \geq a. \tag{5.17}$$

Instead of applying $B_1 = \partial_r + \partial_t$ to (5.4), we can apply a higher order differential operator B_l with $l > 1$. Thus, when the sum over n is truncated at a finite value N, the boundary condition on the modes $n > N$ becomes

$$B_l[ru] = 0, \quad r = a. \tag{5.18}$$

In view of (5.17), we expect that using (5.18) with $l > 1$ instead of with $l = 1$ will yield a smaller error.

When we apply B_2 to (5.4) and use (5.3), we get

$$\begin{aligned} &B_2[ru_{nm}(r, t)] = \\ &\quad (-1)^n \sum_{j=2}^{n} \frac{j(j-1)\gamma_{nj}}{r^{j+2}} \frac{\partial^{n-j}}{\partial t^{n-j}} G_n[u_{nm}](r, t), r \geq a. \end{aligned} \tag{5.19}$$

Similarly, when we apply B_3 to (5.4) we get

$$\begin{aligned} B_3[ru_{nm}(r, t)] &= (-1)^n \sum_{j=3}^{n} \frac{-j(j-1)(j-2)\gamma_{nj}}{r^{j+3}} \\ &\quad \times \frac{\partial^{n-j}}{\partial t^{n-j}} G_n[u_{nm}](r, t), r \geq a. \end{aligned} \tag{5.20}$$

We note that the term $j = 1$ in (5.6) vanishes in (5.19) and that the terms $j = 1$ and $j = 2$ vanish in (5.20). As a consequence, $B_2[ru] = 0$ is an exact boundary condition for the harmonic modes $n = 0, 1$, and $B_3[ru] = 0$ is an exact boundary condition for the modes $n = 0, 1, 2$. Hence, annihilating the leading l terms of the large distance expansion (5.12) naturally imposes the exact boundary condition on the harmonic modes $0 \leq n \leq l - 1$.

This suggests that the local boundary conditions, derived in (Theorem 5.1 of [1]) and also derived by Thompson and Pinsky [7], coincide with those derived in [6]. Both sequences of local operators require increasingly high-order derivatives to reduce the amount of spurious reflection at $\mathscr{B}$. In contrast, the formulation (2.6) corresponding to B_1 does not involve any derivatives beyond first-order and can be made arbitrarily accurate by increasing the value of N.

For later reference, we derive the full boundary condition with $l = 2$. We multiply (5.19) by Y_{nm} and sum over n and m. Next, we set $r = a$ and use (5.11) and (5.7) to obtain

$$B_2[ru] = \sum_{n=2}^{\infty} \sum_{m=-n}^{n} \tilde{\mathbf{c}}_n \cdot \mathbf{z}_{nm}(t) Y_{nm}, \quad r = a. \qquad (5.21)$$

Here $\mathbf{z}_{nm}$ is the solution of (2.7), and $\tilde{\mathbf{c}}_n$ replaces $\mathbf{c}_n$ in (2.9):

$$\tilde{\mathbf{c}}_n(j) = \frac{j(j-1)\gamma_{n1}}{a^{j+1}}, \quad j = 1, ..., n. \qquad (5.22)$$

6. STABILITY

When used in computation, the exact boundary condition (2.6) is truncated at some finite value N. We now show that the energy of the solution of the initial-boundary value problem in Ω remains bounded, despite the incorrect boundary condition imposed upon the higher modes $n > N$. Because the higher and lower modes are treated differently in the boundary condition, we shall assume that they remain independent of one another for all time. We assume that Γ is either a sphere or absent and that $c = 1$ in Ω. For simplicity, we let h and f equal zero. Then the energy of the system in the absence of external forces is

$$E_\Omega[u](t) = \int_\Omega \frac{1}{2} (|u_t(\mathbf{x}, t)|^2 + \|\nabla u(\mathbf{x}, t)\|^2) \, d\mathbf{x}. \qquad (6.1)$$

Similarly, we denote by $E_\infty[u](t)$ the energy (6.1), with Ω replaced by the infinite region outside Γ.

By assumption, the lower and the higher modes fully decouple throughout Ω. We denote by $u_{\leq N}$ the first $N + 1$ modes and by $u_{>N}$ the remaining higher modes:

$$u = \underbrace{\sum_{n=0}^{N} \sum_{|m|\leq n} u_{nm} Y_{nm}}_{u_{\leq N}} + \underbrace{\sum_{n=N+1}^{\infty} \sum_{|m|\leq n} u_{nm} Y_{nm}}_{u_{>N}}. \qquad (6.2)$$

Next, we multiply (2.2) across by $\overline{u_t}$ and integrate over Ω. Using integration by parts and the boundary condition $(\partial_r + \partial_t)[ru] = 0$ at $r = a$, we get

$$\begin{aligned} \frac{d}{dt} E_\Omega[u_{>N}](t) = &-\int_{\mathscr{B}} |\partial_t u_{>N}|^2 \, d\mathscr{B} \\ &- \frac{d}{dt} \frac{1}{a} \int_{\mathscr{B}} |u_{>N}|^2 \, d\mathscr{B}. \end{aligned} \qquad (6.3)$$

Then we integrate (6.3) from 0 to T and use that u vanishes on $\mathscr{B}$ at $t = 0$ to obtain the inequality

$$E_\Omega[u_{>N}](T) \leq E_\Omega[u_{>N}](0). \qquad (6.4)$$

The boundary condition imposed at $r = a$ on $u_{\leq N}$ is exact. Therefore, as a consequence of Theorem 2.1, $u_{\leq N}$ coincides with the lower modes of the restriction to Ω of the solution to the initial-boundary problem (2.2)–(2.5) in the infinite region outside Γ. The energy of a solution to the homogeneous initial-boundary value problem is conserved [8], Section 6.8). Since the initial data is confined to Ω, we have

$$\begin{aligned} E_\Omega[u_{\leq N}](t) \leq E_\infty[u_{\leq N}](t) &= E_\infty[u_{\leq N}](0) \\ &= E_\Omega[u_{\leq N}](0). \end{aligned} \qquad (6.5)$$

By adding (6.4) to (6.5), we conclude that the total energy $E_\Omega[u](t)$ remains bounded by the initial energy $E_\Omega[u](0)$ for all time.

When used in computation, the truncated boundary condition is approximated numerically. This introduces both discretization and rounding errors, which could lead to numerical instability. We shall now discuss the stability of the first-order system of ordinary differential equations (2.7), which is used to update the quantities $\mathbf{z}_{nm}(t)$.

The stability of the ordinary differential equation (2.7) is determined by the eigenvalues of $\mathbf{A}_n$. Let $\mathbf{D}_n = [D_n^{ij}]$ be the $n \times n$ diagonal matrix with $D_n^{jj} = \gamma_{nj}$. Then $\mathbf{D}_n^{-1} \mathbf{A}_n \mathbf{D}_n$ is the companion (or Frobenius) matrix associated with the polynomial

$$P_n(x) = x^n + \frac{\gamma_{n1}}{a} x^{n-1} + \cdots + \frac{\gamma_{nn}}{a^n}. \qquad (6.6)$$

Hence, the eigenvalues of $\mathbf{A}_n$ coincide with the zeroes of P_n. We multiply (6.6) by $2^n a^n$, set $z = 2ax$, and denote the resulting polynomial by $Q_n(z)$. It is

$$Q_n(z) = \sum_{j=0}^{n} \frac{(n+j)!}{(n-j)!j!} z^{n-j}. \qquad (6.7)$$

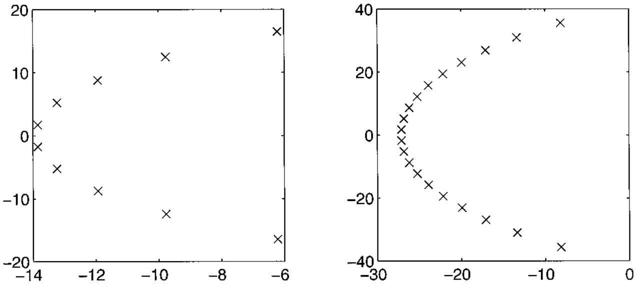

FIG. 1. The roots of Q_{10} (left) and Q_{20} (right).

The roots of P_n are simply the roots of Q_n multiplied by $1/(2a)$. For $n = 1$, $Q_1(z) = z + 2$, and the root is -2. For $n = 2$, $Q_2(z) = z^2 + 6z + 12$, and the roots are $-3 \pm i\sqrt{3}$. In Fig. 1 we see that for small values of n, the roots of Q_n lie on a curve of parabolic shape in the left-half plane. Therefore for small values of n, the real parts of the eigenvalues of $\mathbf{A}_n$ are strictly negative, and the ordinary differential equation (2.7) is asymptotically stable. As we increase n from 10 to 20, we observe that the roots tend to move farther away from the imaginary axis. This results in a stronger obliteration of the past for the higher Fourier modes of the solution on $\mathscr{B}$.

For larger values of n, the problem of computing the eigenvalues is extremely ill-conditioned, so that meaningful results cannot be obtained even in double precision. To show that the roots lie strictly in the left half of the complex plane *without* calculating them, we shall use the Routh–Hurwitz criterion [4, Section I.13]. This criterion provides a finite algorithm based solely on the coefficients of Q_n, which are integers, to decide whether the roots lie strictly in the left half of the complex plane. We have implemented this algorithm in Mathematica using *exact arithmetic*, and we have verified that the roots of Q_n lie strictly in the left half of the complex plane for all n up to $n = 150$. Therefore the ordinary differential equation (2.7) is asymptotically stable for $n \leq 150$, and probably for all n.

7. NUMERICAL EXAMPLES

We shall now compare the accuracy of the various boundary conditions via numerical experiments. First, we shall present a detailed study of the accuracy and the convergence properties of the different boundary conditions. To do so, we shall consider a model problem, where a locally supported time-harmonic source excites the medium inside Ω. Second, we shall present computations for two standard test problems: scattering of an incident plane wave from a sphere, and radiation from a piston on a sphere.

7.1. Model Problem

7.1.1. *Description*

We consider the problem (2.2)–(2.7) with $c(\mathbf{x}) = 1$, $h = 0$, and both u_0 and $\dot{u}_0$ equal zero. The obstacle Γ is the sphere of radius 0.5, and the artificial boundary $\mathscr{B}$ is located at $r = 1$. Hence the computational domain Ω is the region $0.5 \leq r \leq 1$. Initially, the medium is at rest; it is then excited locally by a time-harmonic source distributed within a sphere of radius $r_f = 0.15$ centered at $\mathbf{x}_f$ ($r = 0.75$ and $\theta = 0$). The source strength is

$$f(\mathbf{x}, t) = \begin{cases} \alpha \sin(\omega t) \sin^2((1 - \|\mathbf{x} - \mathbf{x}_f\|/r_f)\pi/2), & \text{if } \|\mathbf{x} - \mathbf{x}_f\| < r_f, \\ 0, & \text{otherwise.} \end{cases} \tag{7.1}$$

We choose the scaling constant $\alpha = 5 \times 10^4$ to make the solution $\mathscr{O}(1)$. The source f has its maximal value at $\mathbf{x}_f$ and decays with distance from $\mathbf{x}_f$.

We shall compare the numerical solution U^h of (2.2)–(2.7) with the numerical solution U^h_∞ of (2.2)–(2.5) in the infinite domain. To compute U^h_∞ we consider the larger domain $\|\mathbf{x}\| \leq R$, with $R = 5$. Since the propagation speed c is one and since the support of f lies inside Ω, the influence of the boundary at $r = R$ will not be sensed inside Ω until $t = 8$. Therefore inside Ω, U^h_∞ coincides with the numerical solution of the initial-boundary value problem (2.2)–(2.5) in the infinite region outside Γ for $0 \leq t \leq 8$. The setup for this model problem is drawn to scale in Fig. 2.

The instantaneous error $E^h(t)$ is defined as

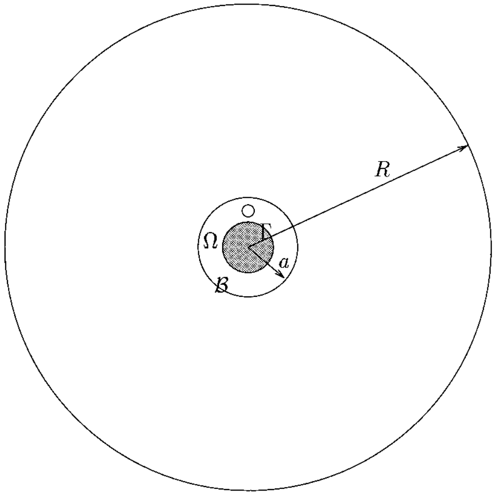

FIG. 2. The computational domains Ω for U^h, and $0.5 \leq r \leq 5$ for U^h_∞, are shown drawn to scale. The support of f lies in the small sphere inside Ω.

$$E^h(t) = \|U^h_\infty(\cdot, t) - U^h(\cdot, t)\|_{L_2(\Omega)}. \tag{7.2}$$

We denote by E^h_T the maximal error over the time interval [0, 8],

$$E^h_T = \max_{t\in[0,8]} E^h(t). \tag{7.3}$$

Since the solution can be scaled by any constant, the magnitude of the absolute error $E^h(t)$ is irrelevant. We shall only use it as a means to compare the performance of the various boundary conditions in relationship to one another.

7.1.2. *Implementation Details*

The test problem described above is axisymmetric and independent of φ. Therefore, it suffices to compute the solution in the r, θ plane for $0 \le \theta \le \pi$, with the symmetry condition $\partial_\theta u = 0$ at $\theta = 0$ and $\theta = \pi$. The grid in Ω is evenly spaced with 10, 20, or 40 intervals in $0.5 \le r \le 1$, and 60, 120, or 240 intervals in $0 \le \theta \le \pi$, respectively. For the computation of U^h_∞, we simply extend the mesh into the larger domain with the same grid spacing up to $r = R$. Both U^h and U^h_∞ are computed using the explicit second-order leap-frog method described in Section 4.

The stability condition for the leap-frog method on an equispaced grid with spacing h, in three space dimensions, and with $c = 1$, is $\Delta t \le h/\sqrt{3}$. Since we shall use a polar grid, we simply set Δt equal to the shortest edge in the mesh divided by $\sqrt{3}$. We keep Δt fixed throughout the computation.

In Fig. 3, the solution U^h_∞ is shown at $t = 3.5$ for a frequency $\omega = 2\pi$, just before the wave front reaches the external boundary $r = R$.

We shall compute U^h by using various boundary conditions at $\mathscr{B}$. The boundary condition $B_l[ru] = 0$ with $l = 1$ and $l = 2$ is denoted by BT1 and BT2, respectively, to acknowledge [6]. We denote (2.6) by NR1(N) and (5.21) by NR2(N). We recall that NR1(0) coincides with BT1, and that NR2(0) and NR2(1) coincide with BT2.

To implement (5.21), we first expand B_2 as

$$B_2 = \frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial t^2} + 2\frac{\partial^2}{\partial r \partial t} + \frac{2}{r}\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right). \tag{7.4}$$

All the terms but the cross derivatives can be approximated with a second-order finite difference formula centered at $r = a$ and $t = t_k$. The radial derivative in the term $\partial_r\partial_t$ is approximated by second-order extrapolation into Ω, by passing a parabola through rU^h at the grid locations r_{l-2}, r_{l-1}, and $r_l = a$.

7.1.3. *Numerical Results*

We begin with a calculation at the low frequency $\omega = \pi/4$ on the 20×120 grid. In Fig. 4 we compare the solutions obtained using BT1, BT2, and NR1(20), with the "exact" solution U^h_∞. The solution is shown at the north pole of $\mathscr{B}$, $r = 1$ and $\theta = 0$, as a function of time. Both BT1 and BT2 yield rather large errors. The solution obtained using NR1(20) cannot be distinguished from U^h_∞ on the left graph. The error $E^h(t)$ is shown on the right. By using NR1(20), instead of BT1 or BT2, the error is reduced by almost three orders of magnitude in accuracy. The error in using NR1(20) is mainly due to the discretization error and not to the boundary condition.

In Fig. 5, we perform the same comparison with a higher frequency $\omega = \pi$. Here the solution is shown at the south pole, $r = 1$ and $\theta = \pi$. We see that imposing BT1 introduces a large error both in the amplitude and in the phase. The error due to the second-order condition BT2 is pointwise about 10% at the south pole and is smaller than in the previous case $\omega = \pi/4$. This is to be expected, since local boundary conditions tend to be more accurate for higher wave numbers. Again, the solution U^h obtained with NR1(20) coincides with U^h_∞ on the graph. The error $E^h(t)$ for NR1(20) is slightly larger than in the previous case $\omega = \pi/4$, because we have used the same grid in both calculations.

We shall now verify that the numerical solution U^h computed using NR1(N) indeed converges to U^h_∞ inside Ω if N is large enough. We set $N = 25$, and progressively refine the initial grid 10×60 by a factor 2 in r and in θ. At each refinement, the error drops by a factor 4, as is shown in Fig. 6. This shows the second-order convergence to the exact solution U^h_∞ as $\Delta r, \Delta t \to 0$. In that sense the boundary condition is exact, even if truncated and used in a numerical scheme: the error introduced at the boundary is negligible in comparison to the discretization error of the numerical method used in the interior of Ω. In contrast, the solutions obtained using BT1 and BT2 do not improve as we refine

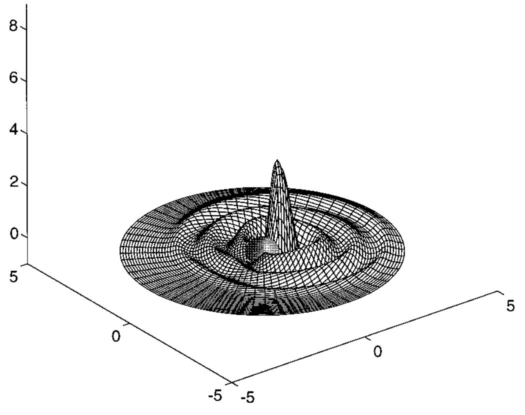

FIG. 3. The solution U^h_∞ at $t = 3.5$ is shown for $\omega = 2\pi$.

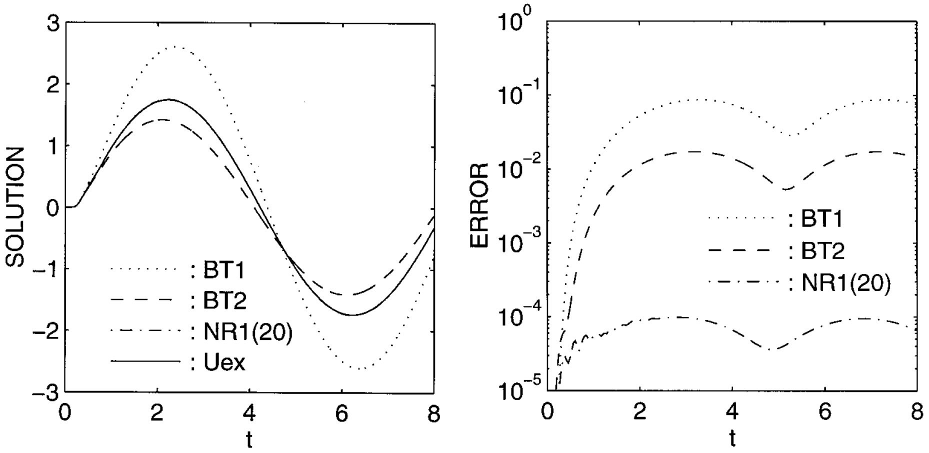

FIG. 4. Results for $\omega = \pi/4$. Left: The solutions U^h, computed using the boundary conditions BT1, BT2, and NR1(20), are compared with the exact solution U^h_∞ on $\mathscr{B}$ at $\theta = 0$. Right: The L_2 error $E^h(t)$ is shown for the same three boundary conditions.

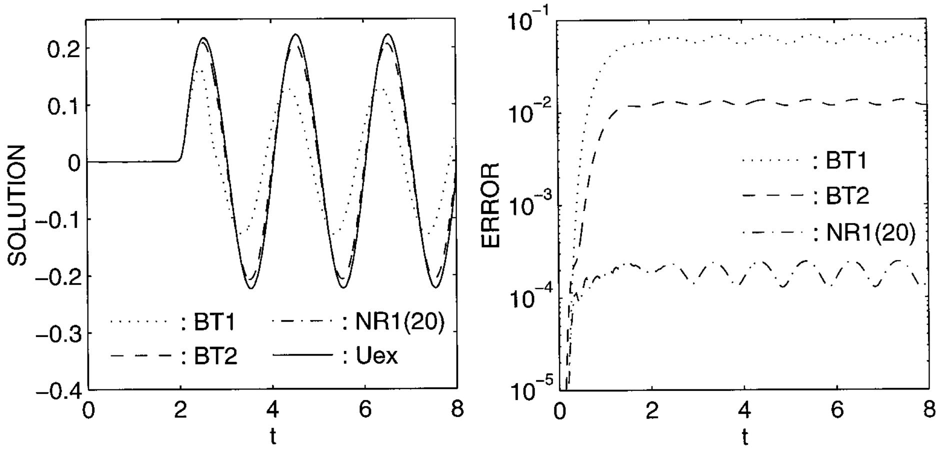

FIG. 5. Results for $\omega = \pi$. Left: The solutions U^h, computed using the boundary conditions BT1, BT2, and NR1(20), are compared with the exact solution U^h_∞ on $\mathscr{B}$ at $\theta = \pi$. Right: The L_2 error $E^h(t)$ is shown for the same three boundary conditions.

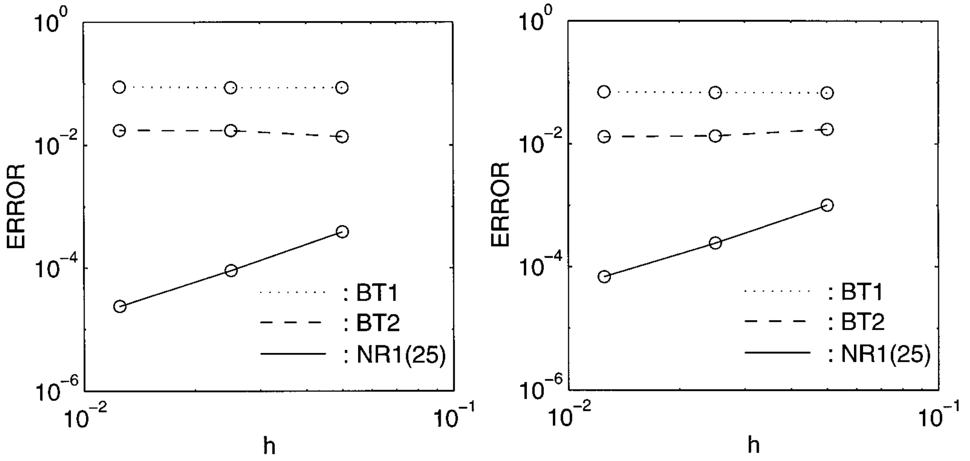

FIG. 6. The maximal errors E^h_T in the numerical solutions U^h, computed with the boundary condition NR1(25) on the three grids 10 × 60, 20 × 120, and 40 × 240, are shown versus $h = \Delta r$. Left: for $\omega = \pi/4$. Right: for $\omega = \pi$.

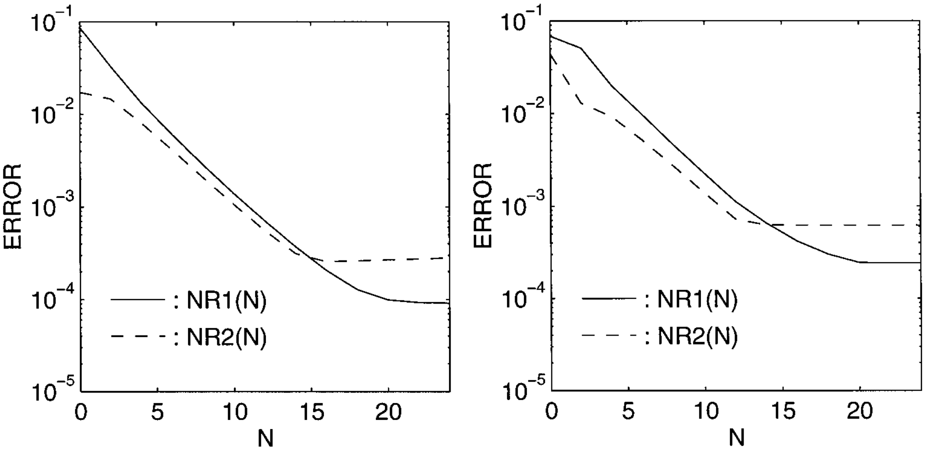

FIG. 7. The maximal errors E^h_T in the numerical solutions, computed on the grid 20 × 120 using the boundary conditions NR1(N) and NR2(N), are shown versus N. Left: for $\omega = \pi/4$. Right: for $\omega = \pi$.

the mesh. This clearly indicates that the error introduced by imposing them at $\mathscr{B}$ dominates in the computation. If one refines the mesh further, it may be necessary to increase the value of N to ensure that the error due to the boundary condition remains negligible.

Next, we compare the solutions U^h computed using the boundary conditions NR1(N) and NR2(N). The grid 20 × 120 is kept fixed while we increase the value of N. We recall that NR1(0) is identical to BT1 and that NR2(0) and NR2(1) are identical to BT2. Figure 7 shows that we obtain an improvement of two or three orders in magnitude over BT1 and BT2 if we use the exact boundary condition. We attribute the small discrepancy between NR1 and NR2 for large values of N to the less accurate finite difference approximation used in (7.4).

Figure 7 also displays the subtle interplay between the error due to discretization and the error due to truncating the boundary condition. For a fixed grid there is a maximal value for N, below which the error does not decrease anymore. From that point on, the boundary condition becomes more accurate than the numerical method, and it is pointless to increase N without further refining the underlying mesh.

7.1.4. *Long Time Stability*

The issue of long time stability has arisen in the numerical implementation of an exact nonreflecting boundary condition due to Ting and Miksis [9]. By using Kirchhoff's formula for the time-dependent wave equation in three space dimensions, they derived an exact nonreflecting boundary condition. It is nonlocal in both space and time, but it involves only a fixed amount of past information. When Givoli and Cohen [10] combined this boundary condition with a standard nondissipative finite-difference scheme in the interior domain, the numerical solution exhibited a long time instability. They were able to eliminate this instability by using the dissipative Lax–Wendroff scheme in the interior domain. In contrast, the boundary conditions, which were derived in this paper and combined with a nondissipative second-order centered finite difference stencil inside Ω, never exhibited any unstable behavior. Whether the explicit Adams method or the trapezoidal rule was used to integrate the system of ordinary differential equations on the artificial boundary, the overall numerical scheme was stable in all our test runs.

To demonstrate the long time stability of our numerical method, we proceed as in [10]. We consider the model problem in 7.1.1 with the source f a short-time pulse, that is, with the temporal part $\sin(\omega t)$ of $f(\mathbf{x}, t)$ set to zero for $t \geq \pi/\omega$. In Fig. 8, the solutions U^h for $\omega = \pi/4$, using the boundary conditions BT1, BT2, NR1(20), and NR2(20), are shown on $\mathscr{B}$ at $\theta = 0$ for a long time interval. The results indicate that the numerical method using any one of these boundary conditions is stable.

7.1.5. *Computational Effort and Storage Requirement*

To improve the accuracy of local boundary conditions, one can either use increasingly higher order differential operators B_l, or simply move the artificial boundary farther away from the region of interest. Typically the latter is done, because it does not require any major modifications of the computer program. The price, of course, is the increased amount of memory and execution time required to store and compute the solution in the superfluous external layer appended to Ω.

We now compare the effect of increasing the radius a of the artificial boundary using a local boundary condition, with that of increasing N in (2.6) or (5.21). The results displayed in Fig. 9 are computed for $\omega = \pi/4$ on the 20 ×

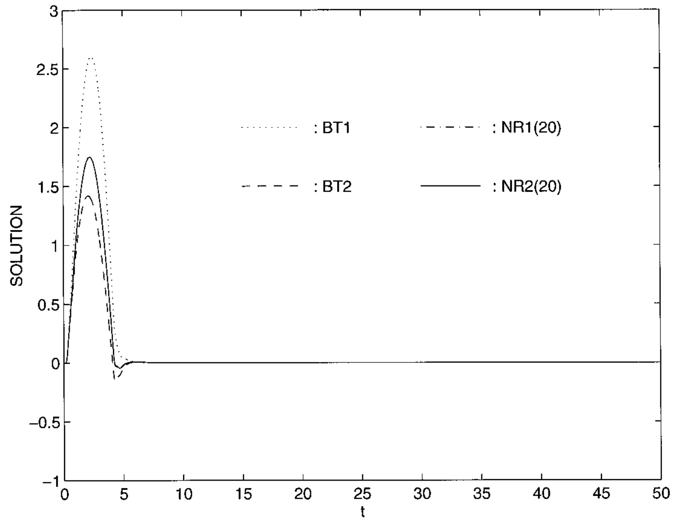

FIG. 8. Demonstration of long time stability: the solutions U^h, computed with $f(\mathbf{x}, t)$ a short-time pulse using the boundary conditions BT1, BT2, NR1(20), and NR2(20), are shown on $\mathcal{B}$ at $\theta = 0$.

120 mesh. The maximal error E_T^h is always computed over $0.5 \leq r \leq 1$, regardless of a. On the left E_T^h is shown versus the radius a for BT1 and BT2. As expected, both local boundary conditions improve as we move the artificial boundary farther away from the obstacle. On the right, a is displayed versus N, to answer the question: "how far away must the artificial boundary be located, so that the error obtained using BT1 or BT2 be at least as small as that using NR1(N) or NR2(N), respectively?"

Here for BT1, the radius a must be larger than 3.5 to reduce by a factor $\frac{1}{3}$ the initial error obtained by using BT1 at $a = 1$. This increases memory requirements and execution time by a factor 7. In contrast, we obtain the same reduction in the error by applying NR1(8) at $a = 1$. This results in less than 10% extra computer time to compute the inner products (U, Y_{nm}) over $\mathcal{B}$ and advance the functions $\mathbf{z}_{n0}(t)$ for $1 \leq n \leq 8$. For BT2, the error decays much faster as we increase a. Still, to achieve the tenfold reduction in the error obtained with NR2(8), we must take a greater than 1.5. This doubles the size of the computational domain, which doubles both storage requirements and total execution time. Using NR2(8) with $a = 1$ instead of BT2 with $a = 1.5$ again increases the total execution time by less than 10%. We conclude this section with the following three key remarks.

First, the boundary conditions NR1 and NR2 achieve the desired accuracy with a minimal amount of storage. Indeed, the additional memory needed to store the functions $\mathbf{z}_{nm}(t)$, $|m| \leq n$, $n = 1, ..., N$ is only of the order of $\sim 2N^3/3$ scalar values, independently of the grid size used. Moreover, to avoid recomputing the spherical harmonics at each step, one may store $\exp(im\varphi)$ and $P_n^{|m|}(\cos\theta)$ separately and form their product when needed for calculating the inner products (Y_{nm}, U^h) over $\mathcal{B}$. Both extra storage requirements are negligible when compared to the storage required by the numerical solution on the three-dimensional grid.

Second, to compute the $(N+1)^2 - 1$ Fourier coefficients (Y_{nm}, U^h), $|m| \leq n$, $1 \leq n \leq N$, over $\mathcal{B}$, it is not necessary to compute $\sim N^2$ inner products over the entire sphere. Instead, one computes only $2N + 1$ inner products with $\exp(im\phi)$, $|m| \leq N$, over $\mathcal{B}$, and then computes $\sim N^2$ *one-dimensional* inner products with $P_n^{|m|}(\cos\theta)$ over $[0, \pi]$. Let p denote the number of grid points in $[0, \pi]$ and let $q \simeq 2p^2$ be the number of grid points on $\mathcal{B}$. Then, since N is much smaller than p—typically $N \simeq p/10$ for the numerical solution to be resolved, the cost of computing the Fourier coefficients is reduced from $\sim N^2 q$ to $\sim 2Nq$. Similarly, the

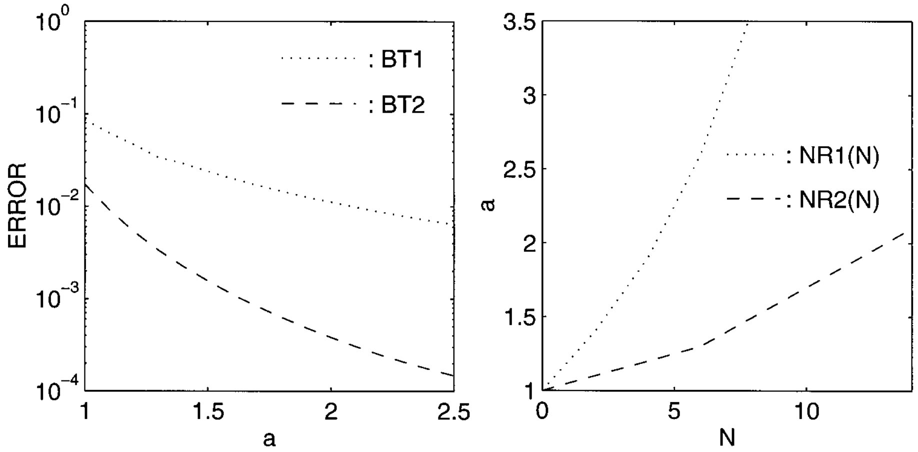

FIG. 9. Left: the maximal errors E_T^h, computed over $0.5 \leq r \leq 1$ using the boundary conditions BT1 and BT2, are shown versus the radius a for $\omega = \pi/4$. Right: the smallest radius a, for which E_T^h obtained using BT1 (or BT2) is less than E_T^h obtained using NR1(N) (or NR2(N), respectively), is shown versus N.

cost of computing the right of (2.6) is $\sim 2\,Nq$. As a point of comparison, advancing the solution on $\mathscr{B}$ using the second-order finite difference stencil requires about $9q$ operations, since updating the value at one grid point requires a weighted average of nine neighboring grid points. Thus, the computational effort per grid point on $\mathscr{B}$ is $\sim 4N/9$ times the cost per grid point of updating the solution in the interior domain.

Third, for fixed values of N and a, the accuracy of the boundary conditions NR1 and NR2 improves as the frequency ω is reduced, because the solution becomes smoother. In contrast, the accuracy of local boundary conditions such as BT1 and BT2, which are based on the large distance expansion of the solution, tends to deteriorate as the low frequency content of the solution increases.

7.2. Plane Wave Scattering

We shall now calculate the scattered field of a plane wave impinging upon the sphere of radius r_0. At the surface of the sphere, we impose the acoustically "hard" boundary condition $\partial_r u = 0$. The incident field u^i propagates along the z-axis. If we let $t = 0$ at the time of impact, the incident field is

$$u^i(\mathbf{x}, t) = \sin \omega(z - r_0 + t) H[z - r_0 + t], \qquad (7.5)$$

where $H[x]$ denotes the Heaviside function. By differentiating u^i with respect to r, we obtain a boundary condition for the scattered field u at $r = r_0$,

$$\begin{aligned}\frac{\partial}{\partial r} u(r_0, \theta) &= \omega \cos\theta \cos(\omega(r_0 \cos\theta - r_0 + t)) \\ &\quad \times H[r_0 \cos\theta - r_0 + t].\end{aligned} \qquad (7.6)$$

We choose $\omega = 4\pi$, $r_0 = 0.5$, and as previously set the artificial boundary $\mathscr{B}$ at $r = 1$. The numerical solution in the infinite domain U^h_∞ is computed until $t = 4$ inside the region $r_0 \le r \le 3$. Thus, the truncation at $r = 3$ will not be sensed inside Ω until $t = 4.5$.

Both U^h_∞ and U^h, computed using BT1, BT2, and NR1(20), are shown in Fig. 10. The computational domain Ω is discretized using the finest mesh 40 × 240. All three boundary conditions perform extremely well for this test problem. On the left, the solutions are shown at $\mathscr{B}$ for $0° \le \theta \le 180°$ and $t = 4$, and are hard to distinguish from each other. On the right, the L_2-error $E^h(t)$ is shown: NR1(20) yields but a small improvement in accuracy over BT2. The high accuracy of the local boundary conditions was observed throughout the frequency range $\pi/4 \le \omega \le 4\pi$. For small wave numbers, the scattered field from a sphere is itself nearly spherical. Since BT1 and BT2 are both derived to annihilate such waves, they perform well for this simple test problem. In Fig. 11, the scattered field U^h_∞ is shown inside Ω at $t = 4$. The scattered field separates into two parts, the reflected and the shadow-forming wave, which appears behind the sphere.

7.3. Piston on a Sphere

As the final example, we consider the radiation from a circular piston on a sphere of radius 0.5—see [7; 11, Sect. 11.3]. The portion of the sphere from $\theta = 0°$ to $\theta = 15°$ is a piston, moving with radial velocity $\sin \omega t$, $\omega = 2\pi$. Elsewhere, the sphere is rigid and the solution vanishes. To avoid the extra numerical complications due to a discontinuous boundary condition, we let the solution decay smoothly to zero for $15° \le \theta \le 30°$. In addition, we use a third-order polynomial up to $t = 0.25$ to enforce C^1 continuity in t of the boundary data at $t = 0$. This problem is challenging because the waves generated at the piston pole $\theta = 0°$ are attenuated by a geometric spreading loss, as they travel along longitudes down to the south pole. In the region opposite the piston, the amplitude of the waves is significantly lower than it is near the piston.

We take the finest mesh 40 × 240 inside Ω, with $\mathscr{B}$ at $r = 1$. For the exact solution U^h_∞, we take the outer boundary at $R = 6$. Thus, the truncation at $r = R$ will not be sensed inside Ω until $t = 10.5$. In Fig. 12, the contour lines for U^h_∞ and U^h are shown at time $t = 10$. In the top part, U^h is computed using BT1, and we see that the contour lines differ severely, especially in the southern hemisphere $\theta \ge 90°$. In the bottom part, U^h is computed using BT2. Here U^h captures the physics of the solution much better, especially in the vicinity of the piston. Yet behind the sphere, a spurious reflection from $\mathscr{B}$ travels toward the obstacle. The contour lines of U^h, calculated using NR1(20), coincide with those of U^h_∞ perfectly and cannot be discerned on this figure.

Since the spurious reflections introduced by the local boundary operators appear to be most severe at the south pole, we take a closer look at that region. On the left of Fig. 13, the solutions U^h and U^h_∞ are shown at the south pole of $\mathscr{B}$, $r = 1$ and $\theta = 180°$, as functions of time. The solutions U^h are computed using BT1, BT2, and NR1(20). On the right of Fig. 13, the same solutions are displayed at time $t = 10$ along the z-axis, $0.5 \le r \le 1$ and $\theta = 180°$. The solutions U^h_∞ and U^h, calculated using NR1(20) coincide almost perfectly. The solutions U^h, calculated using BT1 and BT2, differ significantly from the exact solution. The spurious reflections they introduce at the outer boundary travel back into Ω and spoil the solution right up to the obstacle.

8. CONCLUSION

The original exact boundary conditions (2.6) derived in [1] and the modified boundary conditions (5.21) have both been found to be very accurate in numerical computations.

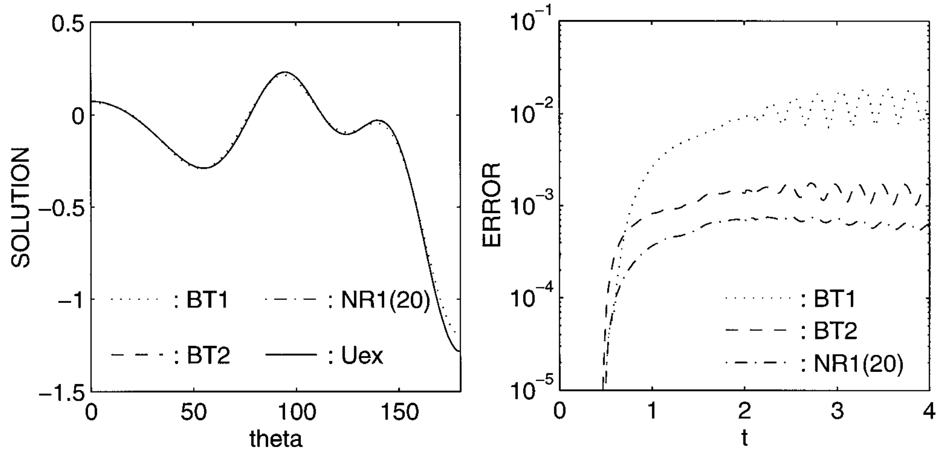

FIG. 10. Plane wave scattering. Left: The solutions U^h, computed using the boundary conditions BT1, BT2, and NR1(20), are compared with the exact solution U^h_∞ at $t = 4$ on $\mathscr{B}$ for at 0° ≤ θ ≤ 180°. Right: The error $E^h(t)$ is shown versus time for the same three boundary conditions.

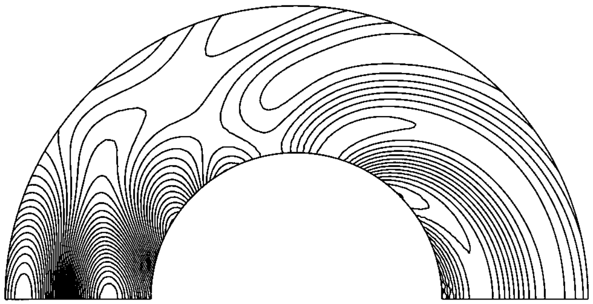

FIG. 11. Contour lines of the scattered field in Ω for an incident plane wave $e^{i\omega(t-z)}$ impinging upon a hard sphere for $\omega = 4\pi$, are shown at $t = 4$.

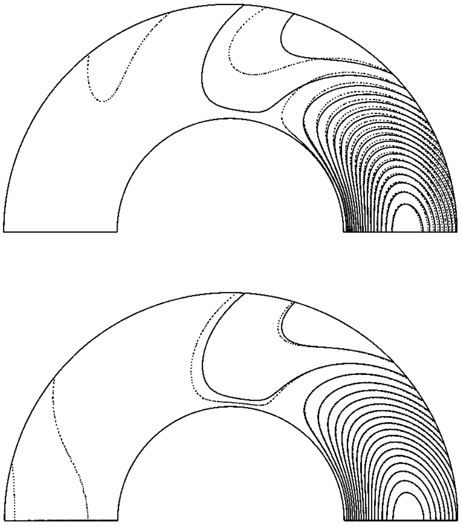

FIG. 12. Piston on sphere. Contour lines of the exact solution U^h_∞ (solid) and of the numerical solution U^h (dotted) are shown at $t = 10$. Top: U^h is computed using BT1. Bottom: U^h is computed using BT2.

Condition (2.6) requires only first derivatives of the solution, which makes it robust and easy to use. Condition (5.21) reduces the error on the truncated harmonic modes, but it is slightly more complicated to implement. They both can be viewed as improvements on the local boundary conditions BT1 and BT2 proposed in [6].

Exact boundary conditions allow the artificial boundary to be brought in as close as desired to the scatterer. They are easy to implement and require little extra memory. Although the formulation is global over the artificial boundary, it is explicit and does not require the solution of any large linear system. It only requires inner products with spherical harmonics of the solution on the artificial boundary. For equations with variable coefficients which tend to a constant at infinity, we expect the new boundary conditions to be more accurate than local conditions.

APPENDIX A

We shall now prove Theorem 2.1. Let u_1 and u_2 be two solutions of (2.5)–(2.7). Next, let v_i, $i = 1, 2$ be the unique "outer" solutions of

$$\frac{\partial^2}{\partial t^2} v_i - \Delta v_i = 0, \quad \|\mathbf{x}\| > a, t > 0, \tag{A.1}$$

$$v_i(\mathbf{x}, 0) = 0, \quad \partial_t v_i(\mathbf{x}, 0) = 0, \quad \|\mathbf{x}\| \geq a, \tag{A.2}$$

with the boundary condition

$$v_i(\mathbf{x}, t) = u_i(\mathbf{x}, t) \quad \text{on} \quad \|\mathbf{x}\| = a, t > 0. \tag{A.3}$$

Let $p_i = u_i$ in Ω and $p_i = v_i$ outside Ω. Then by (A.3), p_i is continuous across $\mathscr{B}$, so its time derivatives and tangential derivatives are continuous on $\mathscr{B}$. Since both u_i and v_i satisfy a second-order equation, we only need to show that the

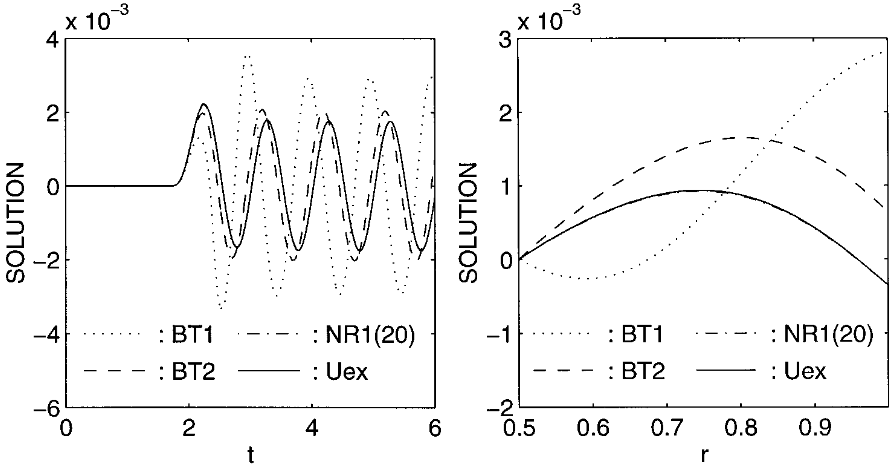

FIG. 13. Piston on a sphere. Left: The solutions U^h, computed using the boundary conditions BT1, BT2, and NR1(20), are compared with the exact solution U^h_∞ on $\mathscr{B}$ at $\theta = \pi$. Right: The same four solutions are shown at $t = 10$ along the z-axis in the region below the south pole, at $\theta = \pi$ and $0.5 \le r \le 1$.

normal derivative of p_i is continuous across $\mathscr{B}$ to show that p_i is a smooth solution of (2.2)–(2.5).

In the derivation of (Theorem 7.1 in [1]), only the fact that the solution satisfied the homogeneous wave equation (2.1) *outside* $\mathscr{B}$ with zero initial conditions was used. Thus by (A.1) and (A.2), v_i satisfies (5.8)–(5.10), with u_i replaced by v_i and w^i_{nm} replaced by some other quantity y^i_{nm}. Since $u_i = v_i$ at $r = a$, it follows that w^i_{nm} and y^i_{nm} satisfy the same ordinary differential equation (5.9) with identical initial conditions (5.10). Therefore w^i_{nm} and y^i_{nm} coincide for all time. Then (5.8) implies

$$\left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right)[ru_i]_{r=a^-} = \left(\frac{\partial}{\partial r} + \frac{\partial}{\partial t}\right)[rv_i]_{r=a^+}. \qquad \text{(A.4)}$$

By continuity of u_i and $\partial_t u_i$ on $r = a$, $\partial_r u_i$ at $r = a^-$ must equal $\partial_r v_i$ at $r = a^+$. In other words, the normal derivative of p_i is continuous across $r = a$. This implies that both p_1 and p_2 are smooth solutions of the initial-boundary value problem in the infinite region outside Γ. By the hypothesis of the theorem this problem has a unique solution. Therefore p_1 must equal p_2, which completes the proof.

REFERENCES

1. M. J. Grote and J. B. Keller, *SIAM J. Appl. Math.* 55(2), 280 (1995).
2. T. J. R. Hughes, *The Finite Element Method,* (Prentice–Hall, Englewood Cliffs, NJ, 1987).
3. N. M. Newmark, *J. Eng. Mech. Div. ASCE,* 67 1959.
4. E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations I* (Springer-Verlag, New York/Berlin, 1987).
5. M. J. Grote and J. B. Keller, *J. Comput. Phys.* **122,** 231 (1995).
6. A. Bayliss and E. Turkel, *Commun. Pure Appl. Math.* **33,** 707 (1980).
7. L. L. Thompson and P. M. Pinsky, “New Space-Time Finite Element Methods for Fluid–Structure Interactions in Exterior Domains”, in *Comput. Methods Fluid/Structure Interaction, AMD,* Vol. 178 (Am. Soc. Mech. Eng., New York, 1993), p. 101.
8. E. Zauderer, *Partial Differential Equations of Applied Mathematics,* 2nd ed. (Wiley-Interscience, New York, (1989).
9. L. Ting and M. J. Miksis, *J. Acoust. Soc. Am.* **80**(6), 1825 (1986).
10. D. Givoli and D. Cohen, *J. Comput. Phys.* **117,** 102 (1995).
11. P. M. Morse and H. Feshbach, *Methods of Theoretical Physics,* Vol. II (McGraw–Hill, New York, 1953).