JOURNAL OF COMPUTATIONAL PHYSICS 98, 145–162 (1992)

A Computational Model of the Cochlea Using the Immersed Boundary Method

RICHARD P. BEYER, JR.

Department of Applied Mathematics, University of Washington, Seattle, Washington 98125

Received April 20, 1990; accepted December 27, 1990

In this work we describe a two-dimensional computational model of the cochlea (inner ear). The cochlea model is solved by modifying and extending Peskin's immersed boundary method, originally applied to solving a model of the heart. This method solves the time-dependent incompressible Navier–Stokes equations in the presence of immersed boundaries. The fluid equations are specified on a fixed Eulerian grid while the immersed boundaries are specified on a moving Lagrangian grid. The immersed boundaries exert forces locally on the fluid. These local forces are seen by the fluid as external forces that are added to the other forces, pressure and viscous, acting on the fluid. The modifications and extension of Peskin's method involve both the fluid solver and the calculation and transfer of immersed-boundary forces to the fluid. For the fluid, the Navier–Stokes equations are solved on a doubly periodic rectangular grid in a second-order accurate manner using a projection method developed by J. Bell, P. Colella, and H. Glaz (Lawrence Livermore National Laboratory Report UCRL-98225, 1988). The extension of the immersed-boundary forces from the moving grid to the fixed fluid grid and the restriction of the fluid velocities from the fixed fluid grid to the moving grid have been modified to be second-order accurate. The calculation of the immersed-boundary forces can be done either explicitly or implicitly or a combination of both. The cochlea is modelled as two fluid chambers separated by a flexible partition whose stiffness varies exponentially along its length. The stapes is represented by a moving piston and the bony outer walls are allowed to be either straight or tapered. A travelling wave propagates along the flexible partition under the influence of the moving piston, and the dependence of this wave on partition stiffness is studied. Also included are studies of transient signal analysis and comparisons of model results to experimental data and asymptotic results. © 1992 Academic Press, Inc.

1. INTRODUCTION

The beautiful complexity of the ear has attracted a large number of researchers over the years. Because of this complexity, an understanding of how the ear functions is incomplete, and theories of auditory system operation are still evolving. This report is concerned with modelling one part of the auditory system operation—the biomechanical behavior of the cochlea, or inner ear. In order to describe this behavior, we use a computational model based on the equations of motion of a coupled fluid/immersed-boundary system. Our method is based on one originally developed by Peskin [11] for modelling fluid dynamics in the heart and has some features in common with particle-in-cell methods [5]. The immersed boundaries are used to describe both the flexible and rigid parts of the cochlea, such as the basilar membrane and the surrounding bone. These various parts of the cochlea are represented by material points. Each point exerts a force on the surrounding fluid in a specified manner depending on its position. In addition, each point moves in accordance with the local fluid velocity. Thus the fluid and the points interact to describe the response of the cochlea to vibrational stimulus. The advantage of this type of method is that the fluid only feels the presence of the immersed boundaries as an external force field, and hence we can model the fluid dynamics on a regular rectangular grid. Since the immersed boundaries are not boundaries of the fluid domain, we eliminate the need to deal with complicated discretization of derivatives at fluid grid points near the immersed boundary. We can, therefore, easily model immersed boundaries with complicated geometries.

In order to model the cochlea with this immersed-boundary approach, we have had to modify Peskin's original semi-implicit method. His original method was designed to model the heart and, as such, needed to allow for large movements of the immersed boundaries. In addition, the material points were interconnected in order to model elastic boundaries. This resulted in a nonlinear system of equations describing the movement of the boundary points that was solved as a minimization problem for an appropriate energy functional. However, the material points in the cochlea are either completely stationary as in the bone or move only slightly as in the basilar membrane. Furthermore, there is little mechanical coupling of the points on the membrane. We have taken advantage of this and uncoupled these boundary points with the result that the equations of motion describing the boundary point movement are a linear system.

We have used this modified semi-implicit method to model a simple version of the cochlea. The model is successful at capturing the essential behavior of the travelling wave and at describing the dependence of peak basilar membrane displacement on frequency. However, while we can model

0021-9991/92 $3.00
Copyright © 1992 by Academic Press, Inc.
All rights of reproduction in any form reserved.

moving boundaries adequately, the nature of the cochlea problem leads to new difficulties. Stationary or nearly rigid points are modelled as very stiff springs, and when the spring stiffness is too large, the semi-implicit method suffers from stability problems. This is due to the physical stiffness of the boundaries resulting in a numerical "stiffness" that causes the calculated forces to be too large and the immersed-boundary points to overshoot their correct positions. The result is that stationary boundary points oscillate about their equilibrium positions in an undesirable manner. The usual method for solving "stiff" differential equations is to use an implicit method.

With the above problems in mind, we further modified Peskin's method to give a fully implicit version that deals successfully with the stiffness problem. Stationary points no longer exhibit unwanted oscillations. Going to a fully implicit method also has the advantage of giving us more choice in the mechanical properties of the points we can model. Now we are allowed to specify that certain points have specified velocities while other boundary points obey specified force laws.

In addition to the stability issue discussed above, we have also improved the accuracy of the immersed-boundary method. Peskin's original method is first-order accurate in time. In order to capture the behavior of the cochlea with a minimum of computational effort, we have made the method second-order accurate. These improvements fall into two categories: those involving the fluid equations and those involving the boundary forces. The accuracy improvement in the fluid equations is achieved by using an incompressible Navier–Stokes solver developed by Bell, Colella, and Glaz [3] in place of the projection method of Chorin [6]. The accuracy of the boundary force calculations is improved by making changes to the way forces and velocities are communicated between the fluid grid and the immersed-boundary grid.

2. COCHLEA MODEL

2.1. *Assumptions*

The human cochlea is a liquid-filled tube, 35 mm long, that has an average cross-sectional area of 2 mm^2. This tube is curled up and coils about 2.5 times. The coiled tube has three liquid-filled chambers: the scala vestibuli, the scala media, and the scala tympani. The scala vestibuli is separated from the scala media by Reissner's membrane which is very flexible. The scala media and the scala tympani are separated by the basilar membrane which varies in flexibility along its length and is responsible for the unusual behavior of the cochlea. On top of the basilar membrane sit other structures, mainly the tectoral membrane and the Organ of Corti which contains the hair cells. Other mammalian cochlea are similar.

The main function of the cochlea is to convert mechanical vibrations induced by sound waves into electrical nerve impulses. When the fluid is excited, a pressure wave travels through the fluid of the cochlea, deflecting the basilar membrane. When the basilar membrane is deflected, the hair cells undergo a shearing motion due to the motion of the tectoral membrane and the Organ of Corti. The movement of these hairs cells results in the firing of nerves, which send signals up the auditory nerve to the brain. The cochlea also serves as a filtering mechanism that results in the separation of differing sound frequencies.

The cochlea's frequency discrimination is due mainly to the varying stiffness of the basilar membrane. As a wave moves down the basilar membrane, the decreasing stiffness of the membrane causes the amplitude of the wave to grow. The wave continues to grow until frictional damping of the fluid dominates and causes the wave to rapidly decay. The location of the maximum displacement depends on the frequency of excitation. As the location changes, different groups of hair cells are activated which the brain senses as sounds of different pitch. This is known as "cochlear tuning."

There has been extensive work over the years in cochlea modelling and Viergever [18, 16], Zwislocki [21], Rhode [13], and Allen [1] have given reviews of these models. In most of these models, the fluid is assumed to be incompressible, inviscid, and linear. This assumption results in having to solve Laplace's equation for the pressure inside the cochlea. The boundary conditions are specified such that at the basilar membrane, the dynamic equilibrium between the membrane and the fluid is governed by

$$M(x)\frac{\partial^2 h(x,t)}{\partial t^2} + R(x)\frac{\partial h(x,t)}{\partial t} + K(x)\,h(x,t) = -2[p(x,0,t)],$$

where M, R, and K are the mass, damping, and stiffness of the membrane, $h(x, t)$ is the instantaneous displacement of the membrane, $p(x, y, t)$ is the pressure, and $[p(x, 0, t)]$ is the hydrodynamic pressure difference between the top and the bottom sides of the membrane.

In our model we include both the viscosity of the fluid and the nonlinear convection and solve the full incompressible Navier–Stokes equations. With fluid viscosity present as the damping mechanism, we let the mass M and the damping R of the membrane be zero. The force the membrane exerts on the fluid is then the force of a massless spring.

Our model of the cochlea reflects the fact that certain assumptions and simplifications have been made about the geometry and mechanical properties of the cochlea. These simplifications are made in order to give a computationally tractable model while still describing the essential physics of the cochlea. Figure 1 shows our two-dimensional depiction

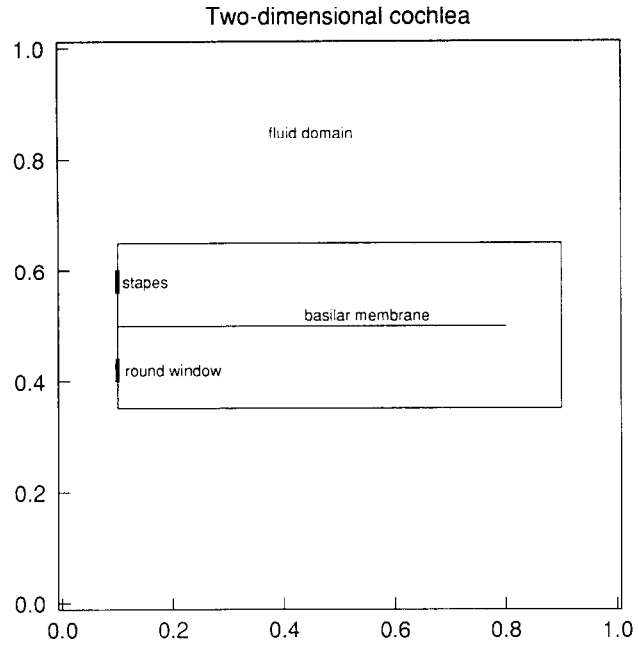

FIG. 1. Two-dimensional version of the cochlea.

of the cochlea. The assumptions and simplifications are as follows:

Spiral coiling. Because we are using a two-dimensional model of the cochlea, we have had to make certain geometrical simplifications in characterizing its three-dimensional structure. We replace the spiral coil with a straight tube. The effect of this simplification has been discussed in Von Békésy [19], Viergever [17], Loh [10], and Steele and Zias [15], and they conclude that the coiling should only slightly affect the basilar membrane motion. The outer walls of the cochlea in Fig. 1, excluding the stapes and round window, are rigid.

Interior structures. Structures inside the cochlea such as the limbus, tectorial membrane, and the Organ of Corti are neglected and only the basilar membrane is included. While the presence of these structures affects the behavior of the basilar membrane to some extent, we lump them together with the basilar membrane in order to reduce the complexity of the moving interior structures. The three channels of the cochlea, the scala vestibuli, the scala media, and the scala tympani, are reduced to two channels by neglecting the very flexible Riessner's membrane. The stapes is assumed to behave like a piston as discussed by Viergever [16]. The stapes and round window are also assumed to have equal cross sections.

Incompressible fluid. As discussed by Viergever [18], the fluid in the cochlea can be regarded as incompressible for most frequencies except, perhaps, the highest in the hearing range. Even at the highest audiofrequencies, the effects of compressibility will only have a minor effect on basilar membrane motion. Consequently we assume that the fluid is incompressible with a density of 1 g cm^{-3} and viscosity of 0.01 g s^{-1} cm^{-1}.

Basilar membrane. We summarize here the discussion given in Le Veque, Peskin, and Lax [9] concerning the simplifications required for a two-dimensional version of the basilar membrane. The real basilar membrane has some mass. It is useful in terms of the immersed-boundary method to neglect this mass. We note Von Békésy's observation [19] that cochlea tuning was absent when the fluid was drained from the ear and the basilar membrane was excited by air pressure alone. This observation eliminates basilar membrane resonance as the mechanism for cochlea tuning and provides some justification for neglecting inertial effects.

Another simplification in going from three dimensions to two involves the mechanical coupling of the basilar membrane. Since the basilar membrane is a narrow plate clamped along its edges, longitudinal coupling is negligible compared with the transverse coupling. The basilar membrane is very stiff in the transverse direction and acts as a very flexible membrane in the longitudinal direction. This coupling of the plate to its edges is represented in the two-dimensional basilar membrane as a restoring force that is independent at each point along the basilar membrane, with magnitude proportional to the vertical displacement of the membrane at each point. This gives a Hooke's Law force of the type we will describe in more detail below. Experimentally (e.g., [19]), the stiffness of the membrane is known to vary exponentially along its length. This gives a stiffness, or spring constant, of the form

$$S_0 e^{\lambda x}, \tag{1}$$

where S_0 is a constant, $\lambda \approx -1.4\ \text{cm}^{-1}$ in humans and x varies from zero at the basal end of the basilar membrane to 3.5 cm at the apical end in humans. In this section we assume the restoring force acts at each point along the basilar membrane separately. The restoring force used here is purely elastic. In reality, the membrane may be viscoelastic, with a velocity-dependent friction term.

2.2. *Equations*

With the above restrictions and assumptions in mind, we use the following equations to describe the mathematical model of the cochlea. For the fluid we use the incompressible Navier–Stokes equations:

$$\rho(\mathbf{u}_t + \mathbf{u}\cdot\nabla\mathbf{u}) = -\nabla p + \mu\nabla^2\mathbf{u} \tag{2}$$

$$\nabla\cdot\mathbf{u} = 0. \tag{3}$$

In addition, we specify the conditions

$$\mathbf{u} = 0 \tag{4}$$

on the rigid boundary. Along the boundary that coincides with the stapes, we specify the sinusoidal motion

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{a}_0 + \mathbf{a}_1 \sin(\omega t) \tag{5}$$

where $\mathbf{a}_0$ and $\mathbf{a}_1$ are constants. Along the boundary that coincides with the round window, the velocity would be the negative of (5) in order to satisfy conservation of mass. Along the basilar membrane, we have

$$u(x, h(x, t), t) = 0, \qquad v(x, h(x, t), t) = h_t(x, t), \tag{6}$$

where $y = h(x, t)$ is the instantaneous position of the basilar membrane, and u and v are the horizontal and vertical components of $\mathbf{u}$. The basilar membrane exerts a vertical stress on the fluid that takes the form

$$[p] = -S_0 \mathbf{h}(\mathbf{x}, t) \exp(-\lambda \mathbf{x}), \tag{7}$$

where $[p]$ is the jump in pressure across the membrane. Since the fluid is incompressible, the viscous stress is zero and this jump in pressure is the jump in the total vertical stress.

To solve the above mathematical model, we use a modified version of Peskin's immersed boundary method.

3. NUMERICAL METHOD

3.1. *Overview*

Peskin's original method [11] was aimed at solving a model of the blood flow pattern in the heart. His method has also been extended by Fauci [7] to study aquatic animal locomotion and by Fogelson [8] to study blood flow during clotting. With this method the whole structure is imbedded in a large rectangular fluid domain with periodic boundary conditions and a uniform computational grid. This allows the Navier–Stokes equations to be solved via finite difference methods on a regular grid and fast Fourier transforms, which has the advantage of great computational efficiency and speed. The boundaries of the mathematical model become immersed boundaries within the enlarged fluid domain and are represented numerically by a separate set of discrete grid points that move relative to the fluid grid. The boundary points affect the fluid grid via the forces they exert. The fluid in turn acts on the immersed boundaries by moving them at the local fluid velocity. In this section we review Peskin's method with the aim of providing the foundations for the immersed-boundary method on which our computational model of the cochlea is built.

The fluid region is described by the time-dependent incompressible Navier–Stokes equations

$$\rho(\mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u}) = -\nabla p + \mu \nabla^2 \mathbf{u} + \mathbf{f} \tag{8}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{9}$$

in a two-dimensional rectangular periodic domain, Ω. Immersed-boundaries are represented by a singular force density, $\mathbf{f}(\mathbf{x}, t)$, which is nonzero only in the regions containing the immersed boundaries. Associated with the immersed-boundary $\mathbf{X}(x, t)$ is a boundary force with intensity $\mathbf{F}$. The force density $\mathbf{f}$ used in the Navier–Stokes equations is then given by

$$\mathbf{f}(\mathbf{x}, t) = \int_B \mathbf{F}(s, t)\, \delta(\mathbf{x} - \mathbf{X}(s, t))\, ds, \tag{10}$$

where B is the domain of the immersed boundary. The force $\mathbf{f}$ has a delta function singularity since we are integrating a two-dimensional delta function only in one dimension. Even though the interaction of the fluid and the boundary is local, the entire fluid feels the effects of a force at a boundary point since $\nabla \cdot \mathbf{f}$ acts as a source term for the pressure field.

The immersed-boundary forces $\mathbf{F}_k$, acting at each point $\mathbf{X}_k$ of the immersed boundary, will depend on the nature of the immersed boundary. For a purely elastic boundary, the forces depend on the positions of the boundary points and have the general functional relationship of

$$\mathbf{F} = \mathbf{F}(\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_m). \tag{11}$$

As an example, for Fauci's swimming organism [7], the forces are given by

$$\mathbf{F}_k = -\partial_k E = -\left(\frac{\partial E}{\partial x_k}, \frac{\partial E}{\partial y_k}\right), \tag{12}$$

where E is an energy function that depends on both the length of the segments of the boundary and on the angle between the segments. In our model of the cochlea, we model the forces by Hooke's law. That is, we have assumed that the restoring force acting to return the basilar membrane to its resting position is purely elastic and acts at each point of the membrane independently of the other points as

$$\mathbf{F}_k = S(\mathbf{X}_0 - \mathbf{X}_k), \tag{13}$$

where S is the stiffness, and $\mathbf{X}_0$ is the equilibrium position of the point. In Fauci's case [7] the boundary points are not independent but coupled, making the determination of the forces in (12) a minimization problem. By treating the boundary points along the basilar membrane as independent, we have reduced the problem to solving for the forces directly from the displacements.

In addition to the equations of motion for the fluid, the equation of motion for the immersed-boundary points is

$$\frac{\partial \mathbf{X}(s, t)}{\partial t} = \mathbf{u}(\mathbf{X}(s, t), t) = \int_\Omega \mathbf{u}(\mathbf{x}, t)\, \delta(\mathbf{x} - \mathbf{X}(s, t))\, d\mathbf{x}. \tag{14}$$

In other words, the immersed boundary moves at the local fluid velocity.

In order to solve the above equations of motion, the fluid domain and the immersed boundary have to be discretized. The two-dimensional fluid domain is divided into an $N \times N$ square grid (or a $M \times N$ uniform rectangular grid) with a grid cell dimension of $h = 1/N$. Defined on the fluid grid are the fluid quantities: velocity $\mathbf{u}_{ij}^n = \mathbf{u}(ih, jh, n\,\Delta t)$, pressure $\mathbf{p}_{ij}^n$, and force $\mathbf{f}_{ij}^n$, where $i, j = 1, 2, ..., N$ and $\mathbf{u}, \mathbf{f} \in \mathbb{R}^{2N^2}$, and $\mathbf{p} \in \mathbb{R}^{N^2}$. Defined on the immersed-boundary are the location, velocity, and force: $\mathbf{X}_k^n = \mathbf{X}(k\,\Delta s, n\,\Delta t)$, $\mathbf{U}_k^n$, and $\mathbf{F}_k^n$, where $k = 1, 2, ..., m$ and $\mathbf{X}, \mathbf{U}, \mathbf{F} \in \mathbb{R}^{2m}$.

With the above discretization we can numerically solve the equations of motion. The solution of the Navier–Stokes equations (8)–(9) is done using either Chorin's method (as implemented by Peskin [11]) or the method developed by Bell, Colella, and Glaz [3].

The integral equations (10) and (14) are discretized as

$$\mathbf{f}_{ij} = \sum_{k=1}^{2m} \mathbf{F}_k D_{ij}(\mathbf{X}_k)\,\Delta s, \tag{15}$$

$$\mathbf{U}_k = \sum_{i,j=1}^{N} \mathbf{u}_{ij} D_{ij}(\mathbf{X}_k)\,h^2, \tag{16}$$

where $\mathbf{U}_k$ is the velocity of the kth immersed-boundary point, Δs is the length of each immersed-boundary segment, and D_{ij} is a discrete form of the delta function. The discrete delta function in (15)–(16) is defined in the same manner as Peskin's [11] by

$$D_{ij}(\mathbf{x}) = d(x - ih)\,d(y - jh), \tag{17}$$

where h is the fluid grid spacing and

$$d(r) = \begin{cases} \dfrac{1}{4h}\left(1 + \cos\left(\dfrac{\pi r}{2h}\right)\right), & |r| < 2h, \\ 0, & |r| \geqslant 2h. \end{cases} \tag{18}$$

Since (15) describes an extension of immersed boundary forces to the fluid grid, we denote this operation by

$$\mathbf{EF} = \mathbf{f}, \tag{19}$$

where $\mathbf{E} \in \mathbb{R}^{2N^2 \times 2m}$ and is called the extension operator. Likewise, (16) is a restriction of fluid velocities to the immersed boundary, so we denote this operation by

$$\mathbf{Ru} = \mathbf{U}, \tag{20}$$

where $\mathbf{R} \in \mathbb{R}^{2m \times 2N^2}$ and is called the restriction operator. The discrete delta function D_{ij} forms the heart of the extension and restriction operations, and we will present improvements to (19) and (20) in Section 3.4 by using other versions of D_{ij}.

3.2. *The Semi-Implicit Method*

In this section we discuss how we actually solve equations (8)–(11), (14) using a modified form of Peskin's original semi-implicit method. In order to advance the state of our coupled fluid-elastic boundary system by one time step, we start with the fluid velocities, $\mathbf{u}_{ij}^n$, and the boundary position, $\mathbf{X}_k^n$, at time level t_n. With this information, we can solve for the new fluid velocity, $\mathbf{u}_{ij}^{n+1}$, and the new boundary position, $\mathbf{X}_k^{n+1}$ at time level t_{n+1}. We solve the fluid equations with Chorin's method as in Fauci [7] or as in Bell *et al.* [3]. To update the boundary position, we discretize (14) in a second-order accurate manner using the trapezoidal rule

$$\frac{\mathbf{X}^{n+1} - \mathbf{X}^n}{\Delta t} = \frac{1}{2}(\mathbf{R}^n\mathbf{u}^n + \mathbf{R}^{n+1}\mathbf{u}^{n+1}), \tag{21}$$

rather than the first-order accurate manner in [11]. Since the immersed boundaries are either stationary or only move slightly, we approximate the restriction operator at time level $n+1$ with its value at the previous time. Then (21) becomes

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \frac{\Delta t}{2}(\mathbf{R}^{n-1}\mathbf{u}^n + \mathbf{R}^n\mathbf{u}^{n+1}), \tag{22}$$

where $\mathbf{R}^{n-1}\mathbf{u}^n$ is used in place of $\mathbf{R}^n\mathbf{u}^n$, since we do not compute and save $\mathbf{R}$, but, rather, we save the matrix-vector product $\mathbf{U}^n = \mathbf{R}^{n-1}\mathbf{u}^n$ at each time step.

Since we do not know the fluid velocity $\mathbf{u}^{n+1}$ at time level $n+1$, we use an approximation $\mathbf{u}^*$ to obtain

$$\mathbf{X}^* = \mathbf{X}^n + \frac{\Delta t}{2}(\mathbf{R}^{n-1}\mathbf{u}^n + \mathbf{R}^n\mathbf{u}^*), \tag{23}$$

where $\mathbf{X}^*$ is our estimated boundary location at time $n+1$. The estimate of the fluid velocity at time level $n+1$, $\mathbf{u}^*$, is given by

$$\mathbf{u}^* = \mathbf{u}^n + \Delta t\,\mathbf{E}^n\mathbf{F}^*, \tag{24}$$

where, following Peskin [11], we neglect all terms in the fluid dynamics equations except the forces exerted by the boundary. This estimated fluid velocity is used only to estimate the boundary position. Once the estimated boundary position, $\mathbf{X}^*$, is calculated, we no longer use $\mathbf{u}^*$. Substituting (24) into (23) gives

$$\mathbf{X}^* = \mathbf{X}^n + \frac{\Delta t}{2}(\mathbf{R}^{n-1} + \mathbf{R}^n)\,\mathbf{u}^n + \frac{(\Delta t)^2}{2}\mathbf{R}^n\mathbf{E}^n\mathbf{F}^* \tag{25}$$

for the estimate of the immersed-boundary location.

There are a variety of functional relationships to choose from for the immersed-boundary force, $\mathbf{F}^*$. The simplest would be to model each point on the immersed boundary as a Hooke's law spring,

$$\mathbf{F}^* = \mathbf{S}(\mathbf{X}_0^{n+1} - \mathbf{X}^*), \tag{26}$$

where $\mathbf{S} \in \mathbb{R}^{2m \times 2m}$ is the diagonal stiffness matrix and $\mathbf{X}_0^{n+1}$ is the known equilibrium position at time $n+1$. More sophisticated models might allow $\mathbf{S}$ to be a nonlinear function of displacement or incorporate a damping term, in which case (26) would be modified to include a velocity term. This would allow viscoelastic behavior to be treated.

In order to solve for $\mathbf{F}^*$, we substitute (25) into (26) and solve for $\mathbf{F}^*$ to obtain

$$\mathbf{F}^* = \left(\mathbf{S}^{-1} + \frac{(\Delta t)^2}{2}\mathbf{R}^n\mathbf{E}^n\right)^{-1} \times \left(\mathbf{X}_0^{n+1} - \mathbf{X}^n - \frac{\Delta t}{2}(\mathbf{R}^{n-1} + \mathbf{R}^n)\,\mathbf{u}^n\right). \tag{27}$$

Using (27) we calculate the force exerted at the estimated new boundary position, $\mathbf{X}^*$, at time level $n+1$ and use this to solve the full Navier–Stokes equations to get $\mathbf{u}^{n+1}$.

When we actually solve (27), we make the additional approximation $\mathbf{R}^n\mathbf{E}^n = \beta\mathbf{I}$, where $\beta = 3/8h$. The details of this derivation are given in the Appendix.

We then solve for the estimate of the force at time-level $n+1$,

$$\mathbf{F}^* = \left(\mathbf{S}^{-1} + \frac{(\Delta t)^2}{2}\beta\mathbf{I}\right)^{-1} \times \left(\mathbf{X}_0^{n+1} - \mathbf{X}^n - \frac{\Delta t}{2}(\mathbf{R}^{n-1} + \mathbf{R}^n)\,\mathbf{u}^n\right). \tag{28}$$

This force is then extended to the grid by the extension operation, $\mathbf{f}^* = \mathbf{E}^n\mathbf{F}^*$, and the velocity $\mathbf{u}^{n+1}$ at time t_{n+1} is calculated by solving the Navier–Stokes equations with source terms $\mathbf{f}^*$. Finally, the boundary point locations are updated.

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \frac{\Delta t}{2}(\mathbf{R}^n\mathbf{u}^{n+1} + \mathbf{R}^{n+1}). \tag{29}$$

3.3. *The Implicit Method*

The semi-implicit method suffers from stability problems due to the large spring constants used to model the rigid boundaries. This stiffness makes it necessary to restrict the time step to values much smaller than desirable in order to avoid unwanted oscillations in the nearly rigid boundary points. Using the following implicit method eliminates these unwanted oscillations.

In deriving the implicit method, we start with the discretized Navier–Stokes equations,

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + G\mathbf{p}^{n+1/2} = \frac{\mu}{2}DG\,(\mathbf{u}^{n+1} + \mathbf{u}^n) - [(uD_x^0 + vD_y^0)\mathbf{u}]^{n+1/2} + \frac{1}{2}(\mathbf{f}^{n+1} + \mathbf{f}^n) \tag{30}$$

$$D\mathbf{u}^{n+1} = 0, \tag{31}$$

where D and G are the discrete divergence and gradient operator, and D_x^0, D_y^0 are the centered difference operators in the x and y directions. The nonlinear convection term at time level $n+\frac{1}{2}$ is calculated by an unsplit second-order Godunov method. The solution to (30) and (31) is then $\mathbf{u}^{n+1}$ and $\mathbf{p}^{n+1/2}$. A full account of these finite difference operators is given in [3].

Collecting terms and simplifying gives

$$\mathbf{u}^{n+1} = \mathbf{V}\left(\mathbf{b}^{n+1/2} + \frac{\Delta t}{2}\mathbf{E}^{n+1}\mathbf{F}^{n+1}\right), \tag{32}$$

where $\mathbf{V} = (\mathbf{I} - (\mu\,\Delta t/2)\,DG)^{-1}$ is the "viscous operator" and

$$\mathbf{b}^{n+1/2} = \left(\mathbf{I} + \frac{\mu\,\Delta t}{2}DG\right)\mathbf{u}^n - \Delta t\,G\mathbf{p}^{n+1/2} - \Delta t\,[(uD_x^0 + vD_y^0)\mathbf{u}]^{n+1/2} + \frac{\Delta t}{2}\mathbf{f}^n,$$

which includes all the information at time level n and $n+\frac{1}{2}$.

Applying (31) involves projecting $\mathbf{u}$ onto the divergence-free subspace, and this is done with the projection operator $\mathbf{P}$. The result is then

$$\mathbf{u}^{n+1} = \mathbf{PV}\left(\mathbf{b}^{n+1/2} + \frac{\Delta t}{2}\mathbf{E}^{n+1}\mathbf{F}^{n+1}\right). \tag{33}$$

Briefly, to project a vector field onto the divergence-free subspace, the divergence-free part of the vector field is represented as a linear combination of basis elements. These basis elements are specified as linear combinations of $G^{\perp}$. A complete account of this projection operation is found in [3].

In the semi-implicit algorithm, we treated fixed immersed-boundary points as obeying Hooke's law with very large stiffnesses. Now we treat the immersed-boundary points as being one of two kinds: either flexible points that obey Hooke's law as before, or fixed points whose velocities are specified (either zero or non-zero). The latter points are used to model fixed or moving walls whose exact location is

known at each time. In the implicit algorithm, we calculate forces $\mathbf{F}^{n+1}$ so that after extending these forces, solving the Navier–Stokes equations, and restricting the velocities, we get exactly the desired velocities at the fixed points, while the imposed forces at the flexible points and the resulting velocities satisfy Hooke's law with the appropriate stiffness.

Before we present the implicit method, we introduce new notation for the two groups of immersed-boundary points. If there are m total immersed-boundary points, we divide the points into two groups, q fixed points and $m-q$ flexible points. This division of points leads to the following partitioning for the restriction operator, the boundary point locations, and the boundary forces:

$$\mathbf{R}=\begin{bmatrix}\mathbf{R}_{\mathrm{I}}\\ \mathbf{R}_{\mathrm{II}}\end{bmatrix}, \qquad \mathbf{X}=\begin{bmatrix}\mathbf{X}_{\mathrm{I}}\\ \mathbf{X}_{\mathrm{II}}\end{bmatrix},$$
$$\mathbf{F}=\begin{bmatrix}\mathbf{F}_{\mathrm{I}}\\ \mathbf{F}_{\mathrm{II}}\end{bmatrix}\begin{matrix}2q\\ 2(m-q)\end{matrix}$$

For the q fixed points whose velocity is specified, we have

$$\mathbf{R}_{\mathrm{I}}^{n}\mathbf{u}^{n+1}=\bar{\mathbf{u}}_{\mathrm{I}}^{n+1}, \tag{34}$$

where

$\mathbf{R}_{\mathrm{I}}\in\mathbb{R}^{2q\times 2N^2}$, the restriction operator for the fixed points,

$\bar{\mathbf{u}}_{\mathrm{I}}^{n+1}\in\mathbb{R}^{2q}$, the specified velocity of the fixed points.

For the $m-q$ flexible points, we have a coupled set of requirements: the new position is determined by the velocities according to

$$\mathbf{X}_{\mathrm{II}}^{n+1}=\mathbf{X}_{\mathrm{II}}^{n}+\frac{\Delta t}{2}\left(\mathbf{R}_{\mathrm{II}}^{n-1}\mathbf{u}^{n}+\mathbf{R}_{\mathrm{II}}^{n}\mathbf{u}^{n+1}\right), \tag{35}$$

and the resulting positions are related to the new forces by Hooke's law,

$$\mathbf{F}_{\mathrm{II}}^{n+1}=\mathbf{S}_{\mathrm{II}}(\mathbf{X}_{\mathrm{II}}^{0}-\mathbf{X}_{\mathrm{II}}^{n+1}). \tag{36}$$

Here

$\mathbf{X}_{\mathrm{II}}^{0}\in\mathbb{R}^{(2m-q)}$, the equilibrium position of the flexible points,

$\mathbf{X}_{\mathrm{II}}\in\mathbb{R}^{2(m-q)}$, the location of the flexible points,

$\mathbf{R}_{\mathrm{II}}\in\mathbb{R}^{2(m-q)\times 2N^2}$, the restriction operator for the flexible points,

$\mathbf{F}_{\mathrm{II}}\in\mathbb{R}^{2(m-q)}$, the force exerted by the flexible points,

$\mathbf{S}_{\mathrm{II}}\in\mathbb{R}^{2(m-q)\times 2(m-q)}$, the diagonal stiffness matrix for the flexible points.

Combining (35), (36), and (33) and introducing a full $2m\times 2m$ inverse stiffness matrix,

$$\mathbf{S}^{-1}=\begin{bmatrix}0 & 0\\ 0 & \mathbf{S}_{\mathrm{II}}^{-1}\end{bmatrix}\begin{matrix}2q\\ 2(m-q)\end{matrix}$$
$$\begin{matrix}2q & 2(m-q)\end{matrix} \tag{37}$$

we obtain a system of equations for $\mathbf{F}^{n+1}$,

$$\left[\frac{\Delta t}{2}\mathbf{R}^{n}\mathbf{P}\mathbf{V}\mathbf{E}^{n}+\frac{2}{\Delta t}\mathbf{S}^{-1}\right]\mathbf{F}^{n+1}$$
$$=\begin{bmatrix}\bar{\mathbf{u}}_{\mathrm{I}}^{n+1}\\ 0\end{bmatrix}-\begin{bmatrix}0\\ \mathbf{R}_{\mathrm{II}}^{n-1}\mathbf{u}^{n}\end{bmatrix}$$
$$+\frac{2}{\Delta t}\begin{bmatrix}0\\ \mathbf{X}_{\mathrm{II}}^{0}-\mathbf{X}_{\mathrm{II}}^{n}\end{bmatrix}-\mathbf{R}^{n}\mathbf{P}\mathbf{V}\mathbf{b}^{n+1/2}. \tag{38}$$

Equation (38) allows us to solve for the boundary forces at the new time level such that the fixed points have specified velocities and the flexible points obey Hooke's law. Note that as any $\mathbf{S}_{ii}\to\infty$, we expect a smooth transition from flexible to fixed behavior for this point since the corresponding element of $\mathbf{R}_{\mathrm{II}}^{n-1}\mathbf{u}^{n}$ approaches zero. This element would then be classified as being in the type I group with $\bar{\mathbf{u}}_{\mathrm{I}}^{n+1}=0$.

In our implicit algorithm, we cannot immediately solve (38) for $\mathbf{F}^{n+1}$ and then solve (33) for $\mathbf{u}^{n+1}$. The reason is that $\mathbf{b}^{n+1/2}$ is dependent on the forces sent to the Navier–Stokes solver. Instead, we solve (33) and (38) iteratively as follows. First we introduce

$$\mathbf{F}^{n+1,\nu}=\mathbf{F}^{n+1,\nu-1}+\delta\mathbf{F}^{n+1,\nu-1} \tag{39}$$

and require $\delta\mathbf{F}^{n+1,\nu-1}\to 0$ as $\nu\to\infty$. Now (33) becomes

$$\mathbf{u}^{n+1,\nu}=\mathbf{P}\mathbf{V}\left(\mathbf{c}^{n+1/2,\nu-1}+\frac{\Delta t}{2}\mathbf{E}\delta\mathbf{F}^{n+1,\nu-1}\right), \tag{40}$$

where $\mathbf{c}^{n+1/2,\nu-1}=\mathbf{b}^{n+1/2}+(\Delta t/2)\,\mathbf{E}\mathbf{F}^{n+1,\nu-1}$.

Going through the same derivations we used to obtain (38), we now have

$$\delta\mathbf{F}^{n+1,\nu-1}=\left[\frac{\Delta t}{2}\mathbf{R}^{n}\mathbf{P}\mathbf{V}\mathbf{E}^{n}+\frac{2}{\Delta t}\mathbf{S}^{-1}\right]^{-1}$$
$$\times\left(\begin{bmatrix}\bar{\mathbf{u}}_{\mathrm{I}}^{n+1}\\ 0\end{bmatrix}-\begin{bmatrix}0\\ \mathbf{R}_{\mathrm{II}}^{n-1}\mathbf{u}^{n}\end{bmatrix}\right.$$
$$\left.+\frac{2}{\Delta t}\begin{bmatrix}0\\ \mathbf{X}_{\mathrm{II}}^{0}-\mathbf{X}_{\mathrm{II}}^{n}\end{bmatrix}-\mathbf{R}^{n}\mathbf{P}\mathbf{V}\mathbf{c}^{n+1/2,\nu-1}\right). \tag{41}$$

We start the loop with $\mathbf{F}^{n+1,0}=0$ and solve (40) to obtain $\mathbf{R}^{n}\mathbf{P}\mathbf{V}\mathbf{c}^{n+1/2,\nu-1}$. Now we solve (41) to obtain $\delta\mathbf{F}^{n+1,\nu-1}$

which we then use in (39) to update $\mathbf{F}^{n+1,\nu}$. At this point we can restart the loop. In practice, however, $\mathbf{b}^{n+1/2}$ is nearly constant over a time step, and we only need to go through the loop once. Once this loop is done, we update the boundary location by

$$\mathbf{X}_{\mathrm{II}}^{n+1} = \mathbf{X}_{\mathrm{II}}^{n} + \frac{\Delta t}{2}(\mathbf{R}_{\mathrm{II}}^{n-1}\mathbf{u}^{n} + \mathbf{R}_{\mathrm{II}}^{n}\mathbf{u}^{n+1}). \tag{42}$$

Solving for $\delta\mathbf{F}^{n+1,\nu-1}$ in (41), we need

$$\left[\frac{\Delta t}{2}\mathbf{R}^{n}\mathbf{P}\mathbf{V}\mathbf{E}^{n} + \frac{2}{\Delta t}\mathbf{S}^{-1}\right] \tag{43}$$

each iteration. In order to avoid forming and factoring (43) each iteration and also each time step, we only form and factor it once in the beginning of the program in the initialization section using approximations $\mathbf{R}$ and $\mathbf{E}$ to $\mathbf{R}^{n}$ and $\mathbf{E}^{n}$ based on the zero displacement location of each boundary. This is justified by the fact that the displacements of the flexible boundaries are quite small.

3.4. *Discrete Delta Function Accuracy*

In order to discretize the integral equations (10) and (14), we have to introduce a discrete form of the delta function. The discrete delta function forms the heart of the extension and restriction operations, and any improvements that could be made would contribute to the overall improvement of the immersed boundary method. With this aim in mind, we present in the following an analysis of discrete delta function accuracy in the one-dimensional case.

The accurate representation of a delta function on a finite mesh raises two important issues which are the mesh accuracy and the translation invariance of such a representation. By mesh accuracy we mean the error due to the size of the mesh as measured in units of the mesh size, h. By translation invariance, we mean the effect of the location of the delta function in relation to the grid points. First we will discuss the accuracy issue and follow this with a discussion of the translation invariance issue.

Consider the following problem: Given a function defined on a finite grid, approximate the value of $f(\alpha)$ by interpolation,

$$f(\alpha) = \int_{-\infty}^{\infty} f(x)\,\delta(x-\alpha)\,dx \simeq \sum_{j} hd(x_j-\alpha)f(x_j). \tag{44}$$

If α coincides with one of the mesh points x_i, then the problem is easy since the solution is just $f(x_i)$. If α does not coincide with a grid point, then we have to approximate $\delta(x-\alpha)$ somehow and use this approximation $d(x-\alpha)$ in (44). Once a choice is made for $d(x-\alpha)$, we can evaluate the accuracy by looking at the resulting error

$$E = f(\alpha) - \sum_{j} hd(x_j-\alpha)f(x_j). \tag{45}$$

We investigate several different discrete delta functions and analyze the resulting error for the case where $f(x) = \exp(-|x-\alpha|)$. This function was chosen because it is continuous at α but all its derivatives are discontinuous. This is the same type of behavior we see in the fluid velocity across an immersed boundary. Shown in Fig. 2 are the various discrete delta functions that we study. These fall into three classes: the hat functions, Peskin's cosine functions, and the linear extrapolation functions.

The hat functions are defined as

$$d(r) = \begin{cases} (1-|r/w|)^{l}/w, & |r| < w \\ 0, & \text{otherwise,} \end{cases}$$

where w is the half width and the value of l is a "shape" parameter. When l is large the hat function becomes more rectangular. The top two plots in Fig. 2 are the hat functions. The left plot is the $4h$ and $2h$ hat with $l=1$. The right plot is the $2h$ hat with $l=1, 2, 3, 100$. The larger l, the more square the shape.

The cosine functions we studied are defined as

$$d(r) = \begin{cases} (1+\cos(\pi r/w))/2w, & |r| < w \\ 0, & |r| > w, \end{cases}$$

where w is again the half width. The lower left plot in Fig. 2 shows cosines of width $2h$, $4h$, and $6h$.

The linear extrapolation functions are defined as

$$d(r) = \begin{cases} \{2-4\varepsilon+(3-8\varepsilon)(r/h)+(1-3\varepsilon)(r/h)^2\}/h, & -2h < r < -h \\ \{2-2\varepsilon+(3\varepsilon-1)(r/h)^2\}/h, & -h < r < h \\ \{2-4\varepsilon+(8\varepsilon-3)(r/h)+(1-3\varepsilon)(r/h)^2\}/h, & h < r < 2h \\ 0, & \text{otherwise,} \end{cases}$$

where ε is used to perturb the linear extrapolation function to make it more translation invariant. This will be discussed further below. This discrete delta function is derived by a linear extrapolation using two points on each side of $r=0$. A straight line is fit to each set of two points and then extrapolated to the discontinuity. A weighted average of each extrapolation determines the value of the function at the discontinuity. These weights are determined by the relative distance the discontinuity is from each grid point.

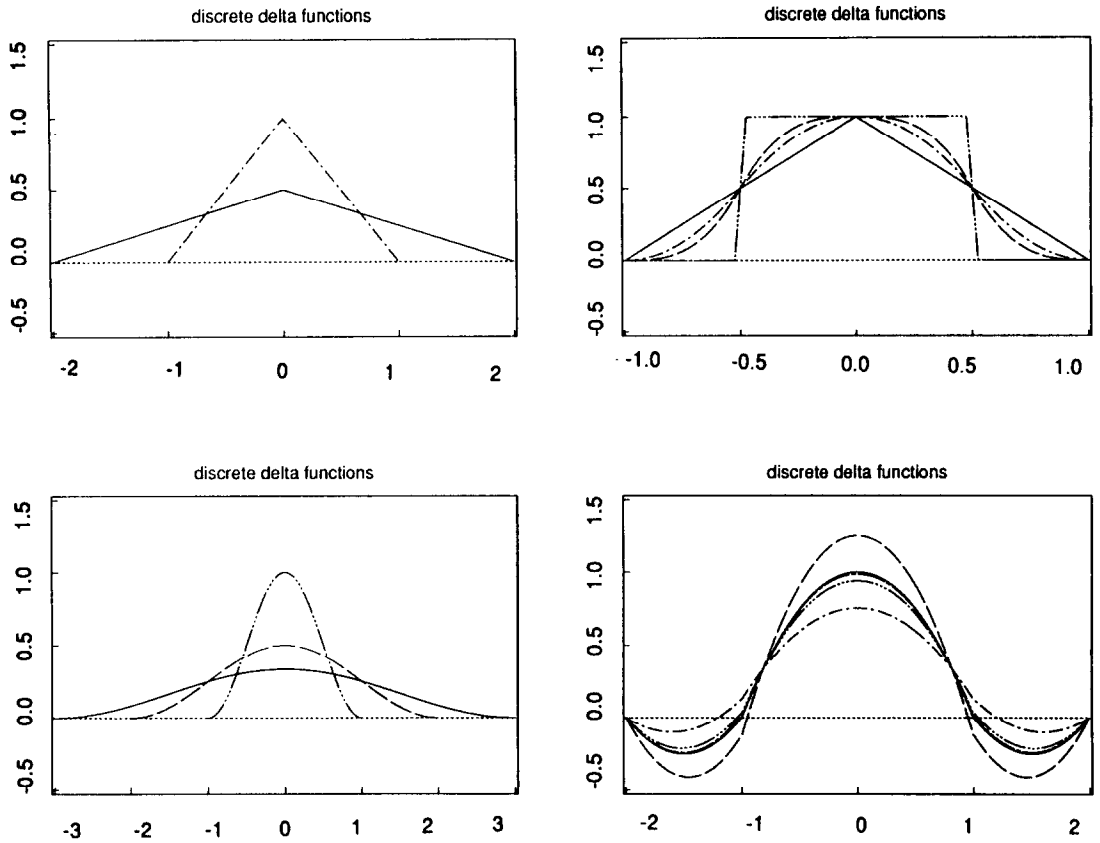

FIG. 2. Discrete delta functions.

The lower right plot in Fig. 2 shows the linear extrapolation functions for $\varepsilon = 0$, $h/8$, h, $2h$, $4h$. (There is not much difference between the case $\varepsilon = 0$ and $\varepsilon = h/8$, so they appear to be almost the same line). All of these functions also satisfy the requirement that $\int_{-\infty}^{\infty} d(x-\alpha)\,dx = 1$.

Each of these functions was used in (45) with four different mesh sizes $h = 1/N$, $N = 16$, 32, 64, 128. The resulting

and the cosines, are first-order accurate, while the extrapolation functions are second-order accurate. Shown in Fig. 3 are these results for a selection from each group. The slopes for the various hat and cosine functions are -1 which means the error as a function of h is first order. The slope for the linear extrapolation is -2 which means the error is second order.

To see why we get these results, we expand $f(z)$ in a Taylor series about the point α, substitute these expansions

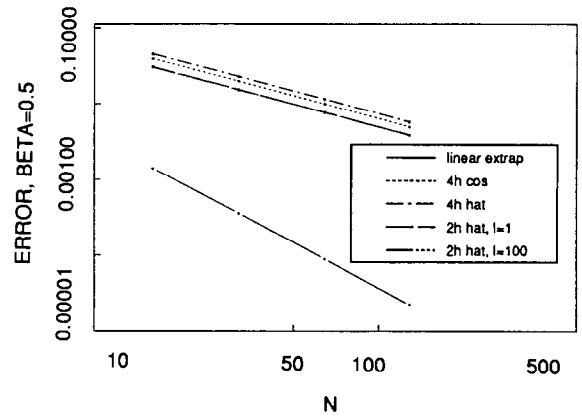

FIG. 3. Discrete delta function error as a function of grid size.

keep the analysis from becoming encumbered with excessive detail, we compare two cases, one from the interpolation-type discrete delta functions and one from the extrapolation type. We expand $f(x_{j-1})$, $f(x_j)$, $f(x_{j+1})$, and $f(x_{j+2})$ about $f(\alpha)$ to obtain

$$f(x_{j-1}) = f(\alpha) - (\alpha - (j-1)h) f'_{\alpha-} + \tfrac{1}{2}(\alpha - (j-1)h)^2 f''_{\alpha-} + \cdots \quad (46)$$

$$f(x_j) = f(\alpha) - (\alpha - jh) f'_{\alpha-} + \tfrac{1}{2}(\alpha - jh)^2 f''_{\alpha-} + \cdots \quad (47)$$

$$f(x_{j+1}) = f(\alpha) + ((j+1)h - \alpha) f'_{\alpha+} + \tfrac{1}{2}((j+1)h - \alpha)^2 f''_{\alpha+} + \cdots \quad (48)$$

$$f(x_{j+2}) = f(\alpha) + ((j+2)h - \alpha) f'_{\alpha+} + \tfrac{1}{2}((j+2)h - \alpha)^2 f''_{\alpha+} + \cdots, \quad (49)$$

where $f^{(j)}_{\alpha-}$ is the jth derivative of f just to the left of α and $f^{(j)}_{\alpha+}$ is the jth derivative of f just to the right of α. Next we substitute these into (45) and note that $\sum_i hd(x_i-\alpha)=1$. This gives for the error

$$\begin{aligned}E=&-hd(x_{j-1}-\alpha)\{-(\alpha-(j-1)h)f'_{\alpha-}\\&+\tfrac{1}{2}(\alpha-(j-1)h)^2f''_{\alpha-}+\cdots\}\\&-hd(x_j-\alpha)\{-(\alpha-jh)f'_{\alpha-}\\&+\tfrac{1}{2}(\alpha-jh)^2f''_{\alpha-}+\cdots\}\\&-hd(x_{j+1}-\alpha)\{((j+1)h-\alpha)f'_{\alpha+}\\&+\tfrac{1}{2}((j+1)h-\alpha)^2f''_{\alpha+}+\cdots\}\\&-hd(x_{j+2}-\alpha)\{((j+2)h-\alpha)f'_{\alpha+}\\&+\tfrac{1}{2}((j+2)h-\alpha)^2f''_{\alpha+}+\cdots\},\end{aligned}\tag{50}$$

where α lies between x_j and x_{j+1}. For our example, we choose $\alpha=x_j+h/2$ and the $2h$ wide hat function which gives $d(x_{j-1}-\alpha)=0$, $d(x_j-\alpha)=1/2h$, $d(x_{j+1}-\alpha)=1/2h$, and $d(x_{j+2}-\alpha)=0$. This particular value of α will give the maximum error. The error then becomes

$$\begin{aligned}E&=-\frac{1}{2}\left\{-\frac{h}{2}f'_{\alpha-}+\frac{1}{2}\left(\frac{h}{2}\right)^2f''_{\alpha-}+\cdots\right\}\\&\quad-\frac{1}{2}\left\{\frac{h}{2}f'_{\alpha+}+\frac{1}{2}\left(\frac{h}{2}\right)^2f''_{\alpha+}+\cdots\right\}\\&=-\frac{h}{4}\{f'_{\alpha+}-f'_{\alpha-}\}+\cdots\\&=O(h).\end{aligned}\tag{51}$$

The error is first order whenever f' is discontinuous at α.

On the other hand, suppose we choose the linear extrapolation function with $\varepsilon=0$. For this we have

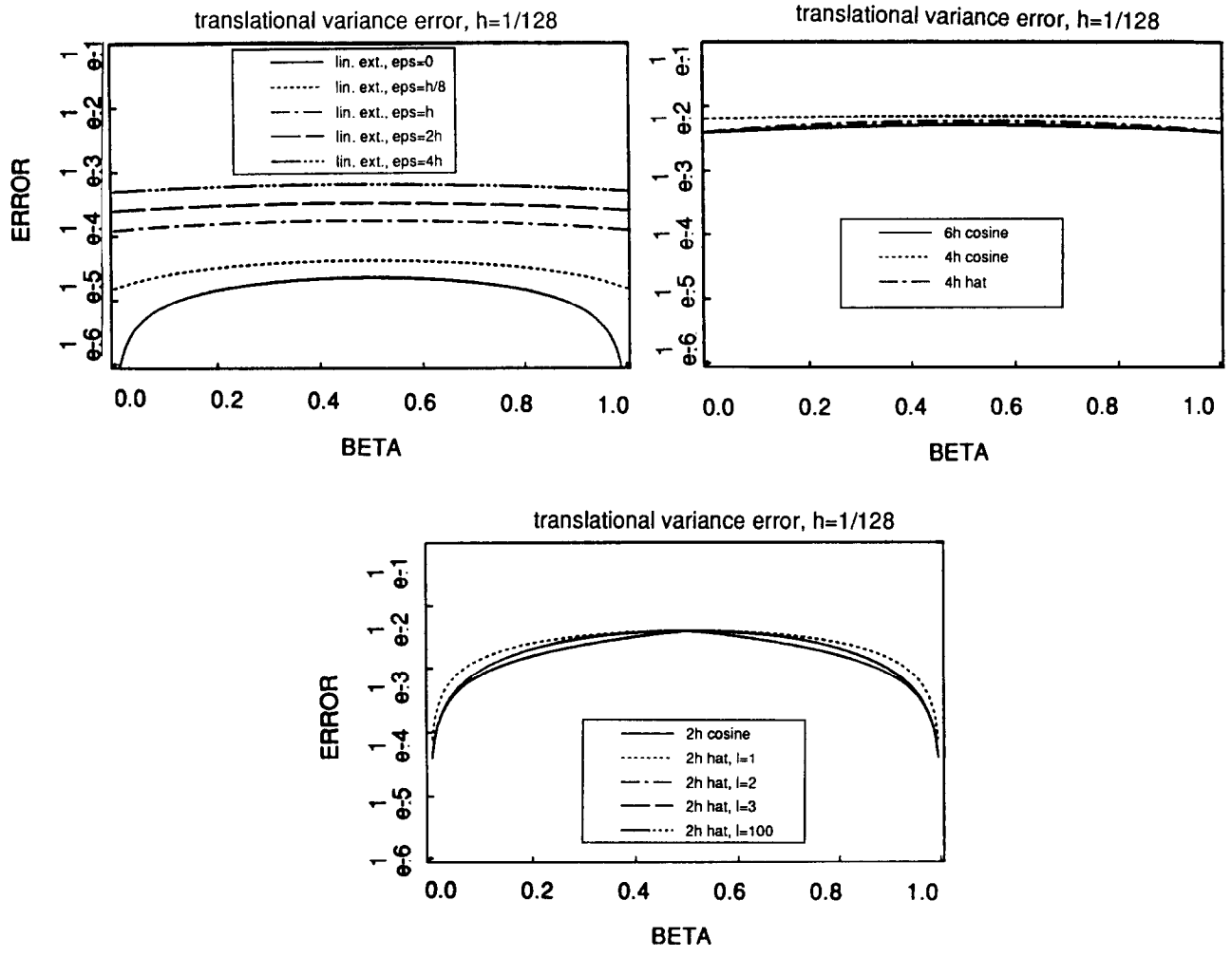

FIG. 4. Discrete delta function errors due to translation variance.

$d(x_{j-1}-\alpha)=-1/4h$, $d(x_j-\alpha)=3/4h$, $d(x_{j+1}-\alpha)=3/4h$, and $d(x_{j+2}-\alpha)=-1/4h$. The error then becomes

$$\begin{aligned} E=&\left\{\frac{3h}{8}f'_{\alpha-}-\frac{9h^2}{32}f''_{\alpha-}+\cdots\right\}\\ &+\left\{-\frac{3h}{8}f'_{\alpha-}+\frac{3h^2}{32}f''_{\alpha-}+\cdots\right\}\\ &+\left\{\frac{3h}{8}f'_{\alpha+}+\frac{3h^2}{32}f''_{\alpha+}+\cdots\right\}\\ &+\left\{-\frac{3h}{8}f'_{\alpha+}-\frac{9h^2}{32}f''_{\alpha+}+\cdots\right\}\\ &=O(h^2). \end{aligned} \tag{52}$$

The other discrete delta functions give similar results. The numerical results are thus supported by the analysis. However, we really only have half the picture so far. The other half has to do with translation invariance.

The issue of translation invariance can be understood with the following problem. Imagine now we are evaluating the right-hand side of (44) for a given grid with a fixed $f(x)$ and α. Also assume α is located between two grid points. Now suppose we translate the grid to the right a small distance such that the fixed point α is still between the same two grid points. When we evaluate the right-hand side of (44), we now obtain a different approximation to $f(\alpha)$. The difference in the results is due solely to the translation of the grid.

In our use of discrete delta functions in the extension and restriction operations, we would like the error due to translation to be as small as possible. To test for this effect, we introduce a parameter β, with $0\leqslant\beta\leqslant 1$, into our specification of the grid. Our grid points then become $x_{j+\beta}=(j+\beta)h$, and we look at the error

$$E=(\alpha)-\sum_j hd(x_{j+\beta}-\alpha)f(x_{j+\beta}). \tag{53}$$

We look at four mesh sizes, $h=1/N$, $N=16$, 32, 64, 128, with β varying from 0 to 1, and all of the discrete delta functions listed above. Figure 4 shows the results for three cases for $N=128$: linear extrapolation functions, the wide functions $4h$ hat, $4h$ and $6h$ cosines, and the narrow functions $2h$ hat and $2h$ cosine. In the upper left of this figure we can see two distint types of errors. For example, the smallest error, that of linear extrapolation ($\varepsilon=0$), has zero error for $\beta=0$ and $\beta=1$, while in the range $0.2<\beta<0.8$, the error is fairly constant. On the other hand, in the upper right, Peskin's $4h$ cosine has a larger error for all β, but the error is relatively constant over the whole range $0\leqslant\beta\leqslant 1$. In the lower left of the figure, we see that the narrow functions all behave similarly. In fact, we cannot distinguish any difference between the $2h$ cosine and the $2h$ hats with $l=2$, 3, 100. We also know from the analysis above that the extrapolation functions are $O(h^2)$ while the interpolation functions are $O(h)$. A desirable feature would be to have both $O(h^2)$ behavior along with a flat profile shown by, for example, the $4h$ cosine. Then, when we solve a problem with a fixed h and a moving source term, additional errors due to translational variance will be minimized. Bu perturbing the linear extrapolation function by a small parameter, we can achieve small translation variance, as shown in the upper left of Fig. 4 for the cases $\varepsilon=h/8$, h, $2h$, $4h$ (ε has to be a function of h to maintain second-order accuracy). In addition we also keep the desired $O(h^2)$ behavior which is shown in Fig. 5.

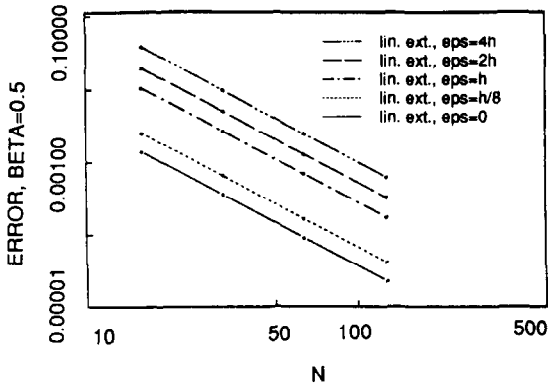

FIG. 5. Discrete delta function error as a function of grid size at $\beta=0.5$.

From the analysis and numerical results presented in this section, we now know how to more accurately perform extension and restriction operations in one dimension. These ideas carry over in a straight forward manner to two dimensions.

4. APPLICATIONS

In this section, we present several case studies of the cochlea using our improved method. We start comparing the semi-implicit and the fully-implicit method in order to show that in fact the fully-implicit method is an improvement over the semi-implicit method. This is followed by a mesh refinement study where we look at a particular boundary configuration and solve our problem on grids 64 by 16, 128 by 32, and 256 by 64. This is followed by a study of transient response of the model. Finally, we examine how well our model compares with experimental data.

4.1. *Semi-implicit vs Fully-implicit*

We test our improved method by looking at the following example. We used a 64×64 point periodic fluid domain. Shown in Fig. 6 is the fluid domain along with the 201-point immersed boundary. The outer boundaries are rigid while the interior boundary is flexible and represents the basilar

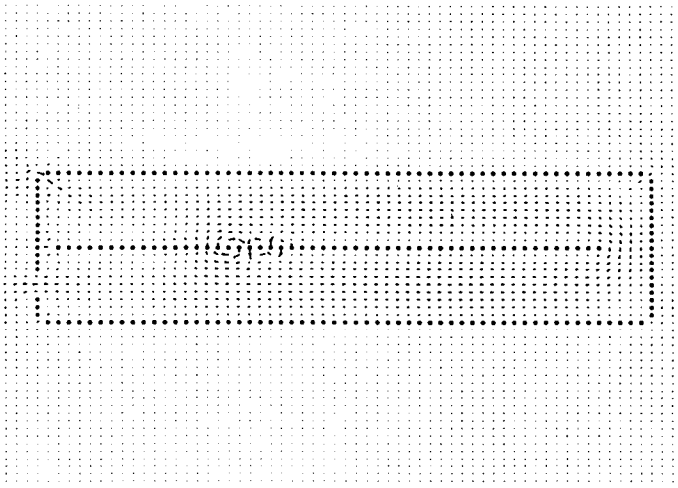

FIG. 6. Fluid domain with the 201-point cochlea.

membrane. On the lower left immersed boundary the gap represents the round window. The corresponding three points on the upper left boundary have a prescribed velocity and represent the piston-like motion of the stapes. The arrows in Fig. 6 show the fluid velocity field after 2048 time steps. The place on the basilar membrane that is undergoing maximum displacement can also be seen about midway down the basilar membrane at $x = 0.4$. Here we see a distinct fluid motion associated with the membrane vibration. Values of the physical parameters included: density $\rho = 1.0$ g cm^{-3}, and viscosity $\mu = 0.01$ g s^{-1} cm^{-1}. The time step was $\Delta t = 0.00025$ s. The stapes forcing frequencies we use are $\omega = 31.25$ Hz, 62.5 Hz, 125 Hz, and 250 Hz. The stiffness is $S_0 = 10^6$ dyne cm^{-3}, and $\lambda = -7.6$. For computational convenience we have nondimensionalized the space variable x so λ is dimensionless. The stapes velocity was $0.01\ (1 - \cos(2\pi\ \Delta t\ \omega))$. For the extension and restriction operations, we use the linear extrapolation version of the discrete delta function with $\varepsilon = 0$.

The travelling wave on the basilar membrane along with the stationary wave is shown in Fig. 7 for the semi-implicit method with $\omega = 125$ Hz. Figure 8 shows the upper half of the stationary envelopes for the fully-implicit method, for the semi-implicit method, and those from the asymptotic results from LeVeque, Peskin, and Lax [9]. The basilar membrane consists of 42 points, where five points at each end are treated as rigid. We include five rigid points at each end to show that the fully-implicit method does keep these rigid points stationary. We can also notice that the rise and decay of the stationary envelope is what we would expect. That is, the left side of the envelopes rise up slowly to the peak, while the right side decays rapidly away.

Also shown in this figure are the phase plots for all four frequencies tested. Here we see an increase in phase shift for a given membrane location. As we can see there is quite a difference between the phase shift for the semi-implicit method and for the fully-implicit method. We know from the asymptotic analysis of LeVeque, Peskin, and Lax [9] that the phase should be approximately

$$\phi(x) = \phi_0 - \left(\frac{2\rho\omega^2}{S_0\lambda}\right) e^{\lambda x}, \tag{54}$$

where ϕ_0 is a constant and x is the distance along the basilar membrane. From this equation we can see that for a fixed value of x, as we increase the frequency ω, the phase should increase as the square of ω. For the fully-implicit method we do see this. For example, at $x = 0.35$ we have the ratio $\phi_{125\,\mathrm{Hz}}/\phi_{250\,\mathrm{Hz}} = 4.45$, which is 11% higher than the expected value of 4. For $x = 0.45$ we have $\phi_{125\,\mathrm{Hz}}/\phi_{250\,\mathrm{Hz}} = 3.82$, which is 4.5% lower than the expected value of 4. If we compare this to the semi-implicit method, we see, for example, for $x = 0.42$ we have $\phi_{125\,\mathrm{Hz}}/\phi_{62.5\,\mathrm{Hz}} = 2$. Clearly, the phase relationship is not quadratic in ω.

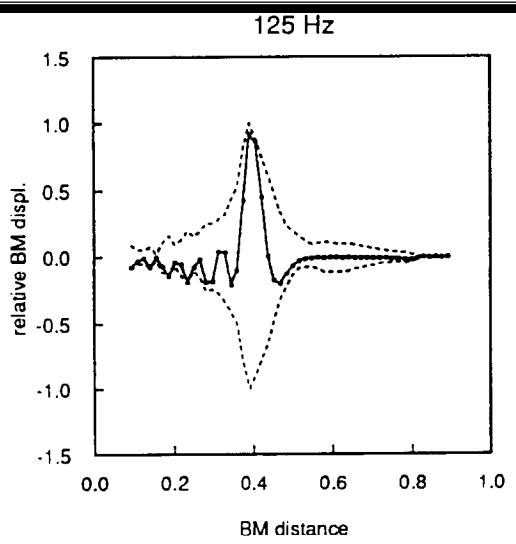

FIG. 7. Basilar membrane displacement with the semi-implicit method and $\omega = 125$ Hz.

To show the phase shift of points along the basilar membrane from the fully-implicit method, we look at a time trace of each point as shown in Fig. 9. In this figure, the time traces at the bottom correspond to the points nearest the stapes and the traces at the top of the figure correspond to the points farthest away from the stapes. We see in this figure that the points are oscillating at the same frequency but the farther from the stapes the more the phase shift. Another interesting feature of this figure is the presence of a long wavelength wave that initially travels down the basilar membrane and eventually dies out. This wave is most pronounced at the top of the figure. The phenomenon of a transient long wavelength wave is what is responsible for setting up the phase shift that occurs at steady state as the travelling waves move down the basilar membrane.

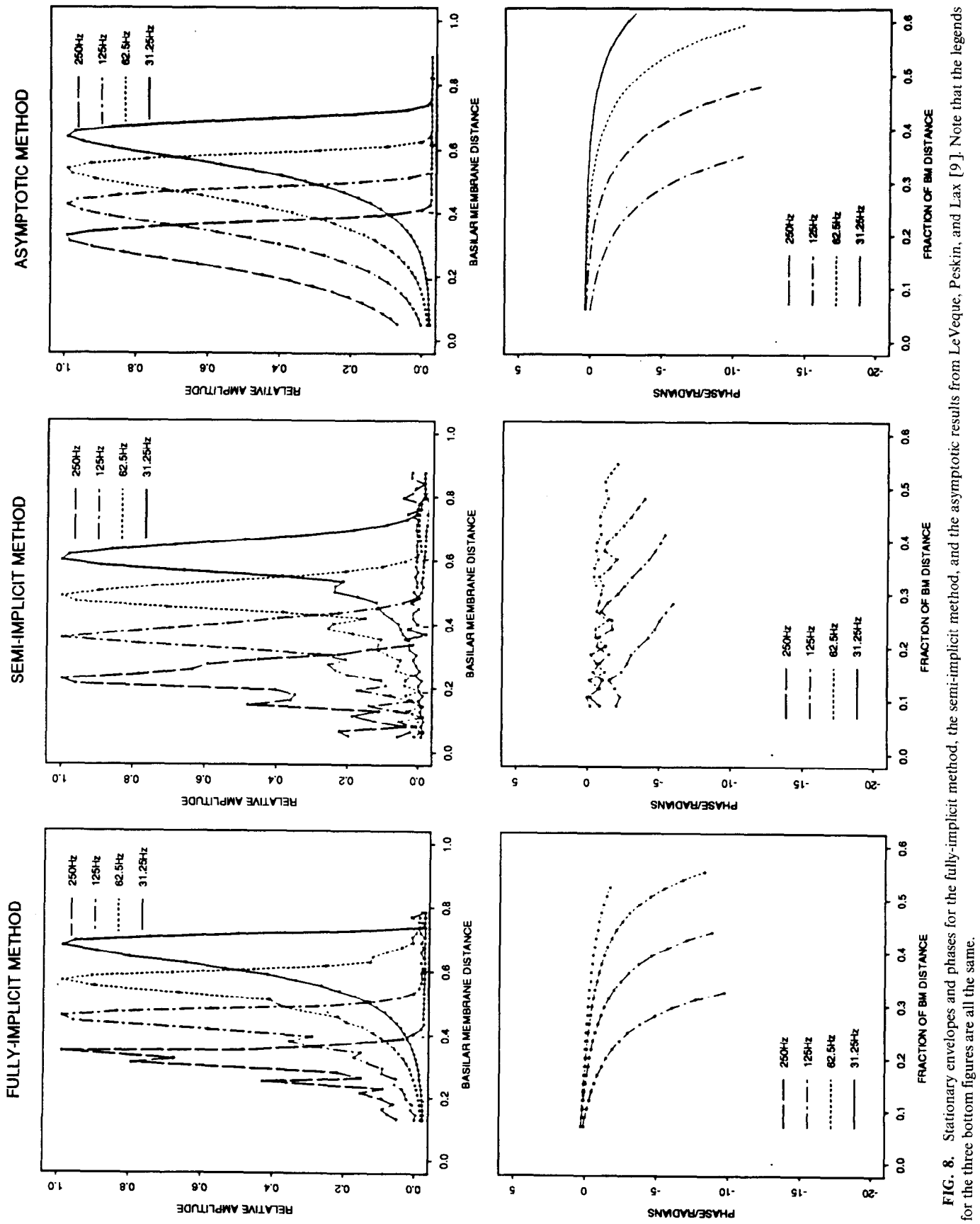

FIG. 8. Stationary envelopes and phases for the fully-implicit method, the semi-implicit method, and the asymptotic results from LeVeque, Peskin, and Lax [9]. Note that the legends for the three bottom figures are all the same.

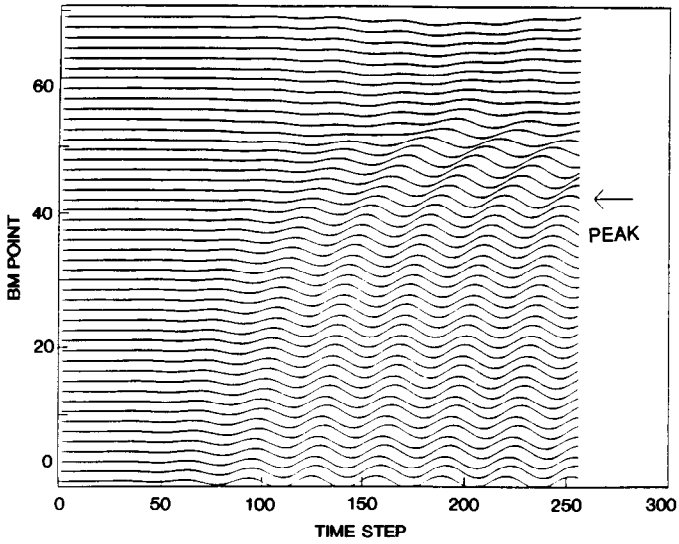

FIG. 9. Time traces for basilar membrane points for the 64 by 16 fluid mesh.

Further proof of the improvements the fully-implicit method gives can be seen in Fig. 10. Here we plot envelope peak location versus the log of the frequency. We show three lines, the semi-implicit, the fully-implicit, and the asymptotic results from LeVeque, Peskin, and Lax [9]. For the asymptotic formula,

$$x_p = \frac{1}{4\lambda}\log\left(\frac{S_0^4\lambda^2}{8\rho^3\mu}\right) - \frac{7}{4\lambda}\log(\omega), \tag{55}$$

where x_p is the peak location, the slope is -0.1475 and the intercept is 1.223. For the fully-implicit method the slope is -0.1443 and the intercept is 1.186. For the semi-implicit method the slope is -0.1736 and the intercept is 1.227.

All of these results show that the fully-implicit method is an improvement over the semi-implicit method.

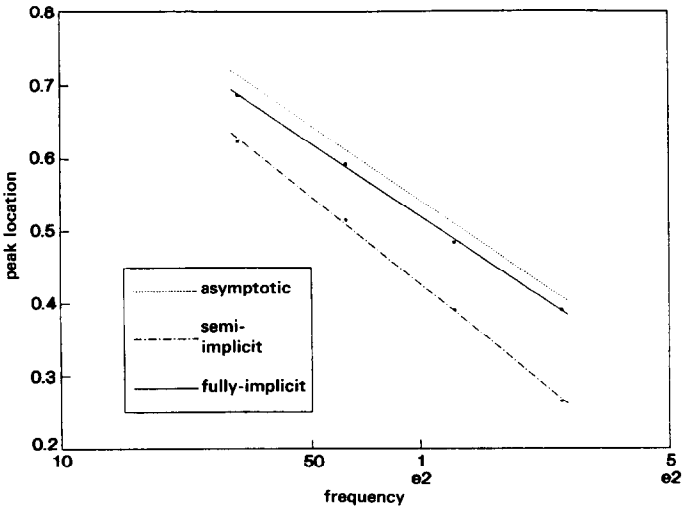

FIG. 10. Envelope peak location versus frequency.

4.2. *Mesh Refinement*

To actually determine the overall accuracy of our method, we tested a particular configuration of the cochlea on three different scales. The three fluid domains were 64 by 16, 128 by 32, and 256 by 64. For each of these three cases, we held the ratio of time step to space step, $\Delta t/h$, constant. For the 64 by 16 case, $\Delta t = 0.0005$ and $h = 1/64$. For the 128 by 32 case, $\Delta t = 0.00025$ and $h = 1/128$. For the 256 by 64 case, $\Delta t = 0.000125$ and $h = 1/256$. The stapes velocity for all three cases was $0.01\ (1 - \cos(2\pi t\omega))$. Other constants used were $S_0 = 10^6$ for the basilar membrane, $\lambda = -7.6$, $\rho = 1.0$, $\mu = 0.01$, and $\omega = 62.5$ Hz. For each of these three cases we used a combination of both fully-implicit points and semi-implicit points. The stapes and the basilar membrane were treated fully-implicitly while the rigid walls were treated semi-implicitly. In general, this combination of the two different types of points allows the critical points to be handled in the more stable fully-implicit manner, while the not so critical points such as rigid boundaries some distance from high flow regions can be handled in the less computationally intensive semi-implicit manner. The stiffness for the semi-implicit rigid points was 5×10^4. For all three cases the model was executed to the same final time of 0.128.

Figure 11 shows the velocity field at the final time for the 64 by 16 fluid mesh case. Also shown are the boundary points for our model cochlea. The stapes is at the upper left corner and the basilar membrane is running horizontally through the middle. The rigid outer walls are located at the far right and bottom of each figure. We can see quite well the velocity patterns near the location on the basilar membrane of maximum displacement.

To calculate an overall accuracy we look at how the maximum stationary envelope peak is converging to some final value. We assume a form for the error of $E = Ch^p$, where E is the difference between the envelope peak at a given mesh and the envelope peak of the true solution. Since we do not know the true magnitude we can still use this formula to calculate p. Let E_i be the error for mesh i, where $i = 1$ is the 64 by 16 mesh etc., then

$$\log_2\left(\frac{E_1 - E_2}{E_2 - E_3}\right) = p.$$

When we substitute in the values for E_i, we calculate a value for p of 1.75. We can conclude that we do have a second-order accurate method.

4.3. *Transient Response*

One of the main features of our model is the ability to analyze transient responses. Most previous mathematical models have only looked at steady state responses of the basilar membrane. One study done by Zias [20] used the steady state model of Steele and Taber [14] to generate

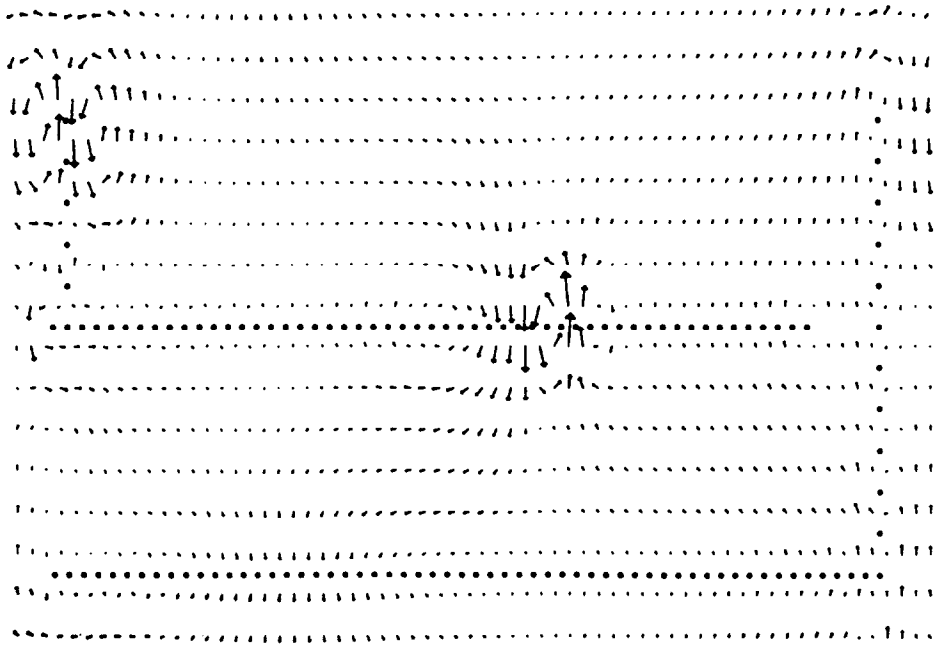

FIG. 11. Velocity field for the 64 by 16 fluid mesh at $t = 0.128$.

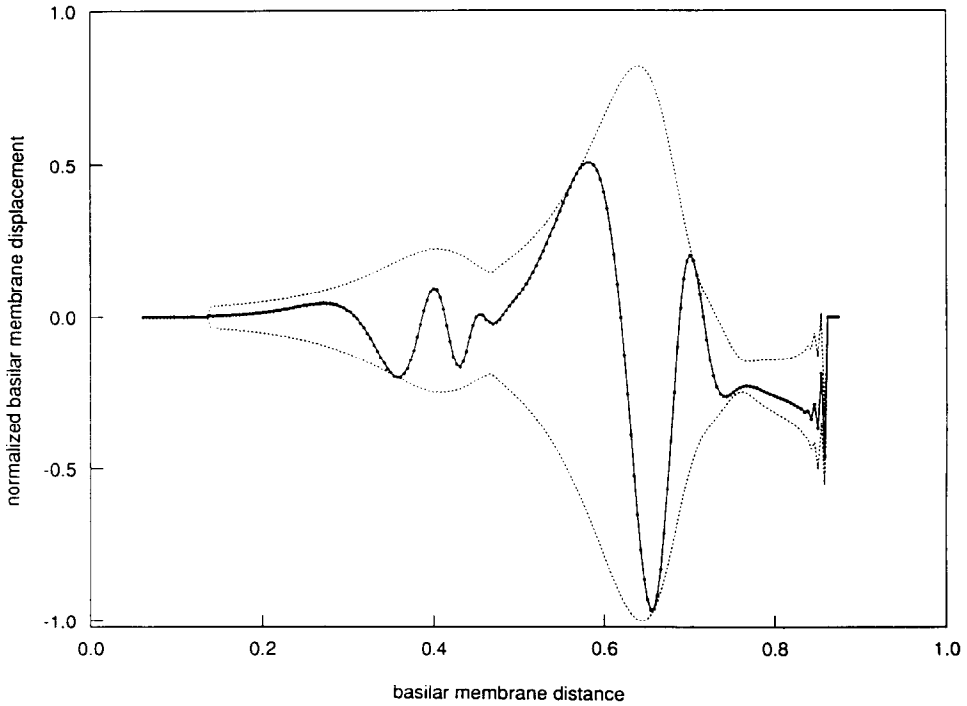

FIG. 12. Basilar membrane displacement for the transient signal case. The dashed line is the envelope swept out by the basilar membrane. The solid line is the position of the basilar membrane at $t = 0.512$.

normal mode shapes of the basilar membrane. These could then be combined in the correct manner and the inverse fast Fourier transform taken to give the time dependent solution to varying signals. This is the only transient response study of which we are aware. With our model we can see such features as the initial long wavelength wave that propagates down the basilar membrane that "sets up" the phase shift of the boundary points. We can also study that the basilar membrane behavior is like when signals of differing frequen-

We used the same fluid domain and boundary configuration for this study as we did for the mesh refinement study with a fluid grid of 256 by 64. We used $\Delta t = 0.00025$ and let the stapes oscillate at 250 Hz for 1024 time steps. After this we added an additional excitation of 50 Hz to the 250 Hz signal. Shown in Fig. 12 is the basilar membrane displacement at $t = 0.512$. Here we see two peaks, the leftmost corresponds to the 250-Hz signal and the rightmost to the 50-Hz signal. Shown in Fig. 13 is the normalized time trace of each of the boundary points. The time traces in the bottom of the figure show the higher frequency, 250 Hz, that are present for the whole time, and the traces in the top of the figure show the lower frequency, 50 Hz, that only appears for $t > 0.256$. We can also see at the top of this figure the long wavelength wave that is responsible for setting up the phase shift of the basilar membrane points.

These figures clearly demonstrate the power of our method to observe the transient behavior of the basilar membrane.

4.4. *Comparison to Experimental Data*

In this section we compare the results of our model to the experimental data of Rhode [12] taken from squirrel monkeys. In most experimental data such as that of Rhode [12], the measurement location on the basilar membrane is fixed and the frequency of input sound is varied. In this way many different frequency points can be taken with the measurement device fixed. In our computational model, we fix the frequency and calculate the amplitude and phase at every point on the membrane. This restricts how many frequencies we can measure simply due to the cost of computing. To run our 256 by 64 model with 502 points (229 of which are treated implicitly) on a Cray X-MP/48 for 1000 time steps takes about 10 min of Cray time. Therefore, only seven frequencies, 1, 4, 5, 6, 7, 7.5 and 8 kHz. For these tests we used the same configuration as that of the 256 by 64 mesh in the mesh refinement section. The stiffness profile was chosen so as to best match the amplitude data of Rhode [12]. We used $S_0 = 8 \times 10^9$ and $\lambda = 11.5$. For each of the five tests, we used $\Delta t = \omega^{-1}/16$ and ran each out to 512 time steps. The results at the same time t from each test were compared to those of Rhode [12]. As pointed out by Allen and Sondhi [2] who compare their model to the results of Rhode [12], no measurements of the parameters S_0 and λ have been made for the squirrel monkey. Allen and Sondhi [2] used $S_0 = 2.8 \times 10^9$ and $\lambda = 0.9 - 1.5$. If we rescale our value of λ to be on the same basis, we use $\lambda = 1.6$.

Figure 14 shows the results of our tests plotted with the experimental data of Rhode [12]. As can be seen, our data agrees quite well in the region we tested. We also compare the phase data as well. Shown in Fig. 15 is the phase data from Rhode [12] and our results. Our phase data was adjusted so that the 1-kHz point coincided with the 1-kHz point of Rhode [12]. Here we also can see good agreement. Notice that there is some discrepancy at 8 kHz, where we see our results are slightly higher than the experimental data. This may indicate the actual basilar membrane may undergo a larger phase shift than what would be expected from the ω^2 dependency we expect from the asymptotic results of LeVeque, Peskin, and Lax [9]. Even though the comparison of our results with the experimental data

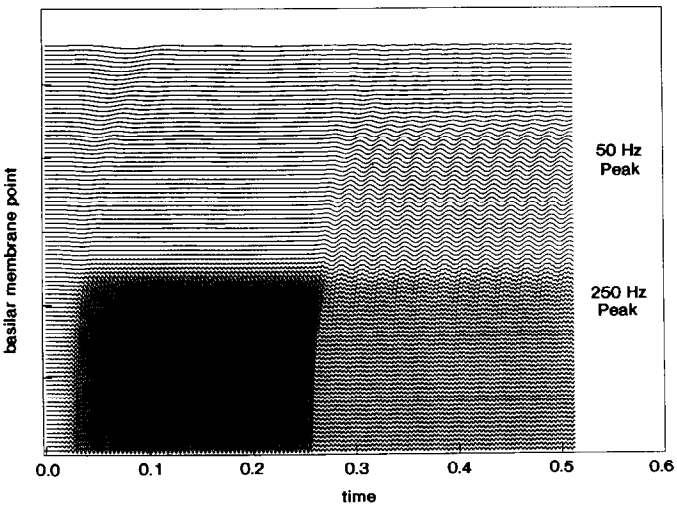

FIG. 13. Time traces for the transient signal case.

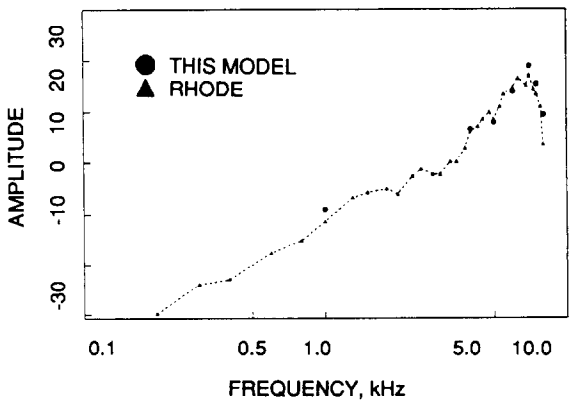

FIG. 14. The data from Rhode [12] is the ratio of basilar membrane displacement to malleus displacement normalized to 80 dB SPL while the data from this work is relative basilar membrane.

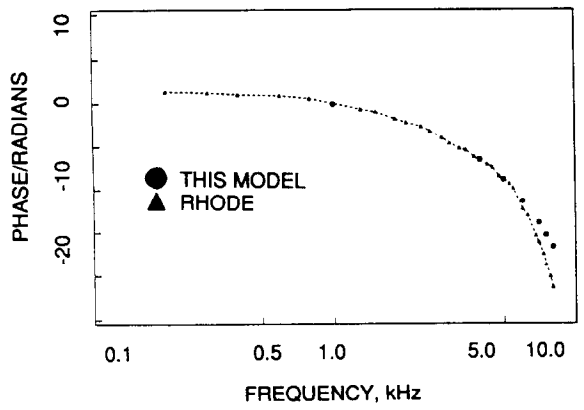

FIG. 15. Basilar membrane phase from Rhode [12] and this work.

appears good, this agreement needs to be qualified. Our tests were conducted with equal stapes velocity and then the basilar membrane displacement amplitude was normalized to equal power input. The data points were then shifted so that a curve drawn through our data would agree with the data from Rhode [12] as well as possible. This is the same as expressing our data relative to some fixed value such as 1 Å. The experimental data of Rhode [12] were collected at various sound pressure levels and then normalized to 80 dB SPL. What was then calculated was the ratio of basilar membrane amplitude to malleus amplitude. (The malleus is the bone of the middle ear that touches the eardrum.) This means that the experimental data reflects the nonlinear transfer function that exists between the eardrum and the oval window. No such nonlinear transfer function is present in our calculations of equal power input. In addition, the uncertainty in the values of S_0 and λ make the agreement between the two sets of data shown in Fig. 14 better than what might be expected from the rather coarse model that was used here.

5. CONCLUSIONS

We have presented a modified version of Peskin's original immersed-boundary method [11]. From the results of the previous section, we draw several conclusions.

The semi-implicit method takes advantage of the small movements of the immersed boundaries and linearizes the calculation of the forces exerted by the immersed boundary points. Numerical experiments show that this approach is suitable for situations where the immersed boundary is not in a flow regime that requires large forces to maintain its position. In situations where there are flow rates that do require the boundary forces to be large, we describe a fully-implicit method that can be used. This approach removes the instabilities of the semi-implicit method and makes modelling of such demanding cases as the cochlea possible. We show how to make the operations of extension and restriction more accurate. By analyzing a one-dimensional model problem, we see how to achieve second-order accurate solutions by choosing the appropriate discrete delta function.

When the modified method is applied to solving a two-dimensional model of the cochlea, we see that:

(1) the model can predict the biomechanical behavior of the cochlea such as travelling waves and stationary envelopes,

(2) the results from the model are in close agreement with the asymptotic results of LeVeque, Peskin, and Lax [9],

(3) the results from the model are in close agreement with the experimental results of Rhode [12],

(4) transient analysis is possible, which should be useful in speech analysis.

We have presented other results elsewhere [4] that also show that with a nonlinear stiffness for the basilar membrane, the model predicts the perception of decreasing pitch with increasing loudness, and that bone conduction can excite the basilar membrane in a manner similar to excitation by the stapes.

Even though the results demonstrate the success of the model, there are some aspects that have not been handled as well as necessary. One of these is the spatial resolution of the travelling wave in the envelope peak region. To adequately resolve the envelope shape at high frequencies requires many fluid grid and boundary points. The largest version of the model used here, with 256×64 grid points, is too small to adequately resolve the spatial wave over the whole audio frequency range. A larger mesh model is needed, say on the order of 1000×500. Another problem is the limitation of using two dimensions. By lumping the mass of the tectoral membrane and the Organ of Corti together with the basilar membrane, the model over-simplifies the geometry. In addition, the effect of the coiling of the cochlea on the travelling waves could not be investigated using only two dimensions. These problems can only be dealt with in a full three-dimensional model.

APPENDIX

When we actually solve (27), we make the additional approximation $\mathbf{R}''\mathbf{E}'' = \beta\mathbf{I}$, where $\beta = 3/8h$. This approximation eliminates the need to solve a linear system for $\mathbf{F}$, since the matrix in (27) becomes diagonal. We derive this approximation as follows. The extension operation $\mathbf{EF}$ is defined by (15) and the restriction operation $\mathbf{Ru}$ by (16). Putting these together, we have for the kth element of a vector derived from the restriction-extension operation

$$(\mathbf{REF})_k = \sum_{i,j=1}^{N} h^2 D_{ij}(\mathbf{X}_k) \sum_{l=1}^{m} \Delta s\, D_{ij}(\mathbf{X}_l)\, \mathbf{F}_l$$

$$= \sum_{l=1}^{m} \Delta s \left\{ \sum_{i,j=1}^{N} h^2 D_{ij}(\mathbf{X}_k)\, D_{ij}(\mathbf{X}_l) \right\} \mathbf{F}_l$$

$$= \sum_{l=1}^{m} \Delta s\, g(\mathbf{X}_k - \mathbf{X}_l)\, \mathbf{F}_l, \tag{56}$$

where $g(\mathbf{X}_k - \mathbf{X}_l) = \sum_{i,j=1}^{N} h^2 D_{ij}(\mathbf{X}_k)\, D_{ij}(\mathbf{X}_l)$. The value of g only depends on the separation between $\mathbf{X}_l$ and $\mathbf{X}_k$. We also know $g(\mathbf{X}_l - \mathbf{X}_k)$ is zero when $\|\mathbf{X}_l - \mathbf{X}_k\| \geq 4h$ because of the way the D_{ij} were defined. If we assume $\mathbf{F}$ is constant over the region where g is nonzero, then we obtain

$$(\mathbf{REF})_k = \mathbf{F}_k \sum_{l=1}^{m} \Delta s\, g\,(\mathbf{X}_k - \mathbf{X}_l).$$

If we further assume the spacing between each boundary point is the same and equal to Δs, then letting $\Delta s \to 0$ allows us to approximate the sum over l as an integral

$$(\mathbf{REF})_k \approx \mathbf{F}_k \int_{-4h}^{4h} \sum_{i,j=1}^{N} h^2 \left\{ \frac{1}{4h}\left(1 + \cos\left(\frac{\pi}{2h}(x_k - ih)\right)\right)\right\}$$

$$\times \left\{ \frac{1}{4h}\left(1 + \cos\left(\frac{\pi}{2h}(y_k - jh)\right)\right)\right\}$$

$$\times \left\{ \frac{1}{4h}\left(1 + \cos\left(\frac{\pi}{2h}(x(s) - ih)\right)\right)\right\}$$

$$\times \left\{ \frac{1}{4h}\left(1 + \cos\left(\frac{\pi}{2h}(y(s) - jh)\right)\right)\right\} ds.$$

Letting $x_k = y_k = y(s) = 0$ and $x(s) = s$ and integrating over s from $-4h$ to $4h$ gives

$$(\mathbf{REF})_k \approx \beta \mathbf{F}_k, \tag{57}$$

where $\beta = 3/8h$.

ACKNOWLEDGMENTS

The author is deeply grateful to Randall J. LeVeque for his help and guidance throughout this project. Thanks also to Dr. John Bell of Lawrence Livermore Labs for his fluid solver, Charlie Peskin of the Courant Institute for his inspiration and guidance, and Lisa Fauci of Tulane University for her help. This research was supported at the University of Washington Applied Mathematics Department in part by NSF Grants DMS-8601363 and DMS-8657319. Cray computer time was provided through NSF at the San Diego Supercomputer Center and by Boeing Computer Services.

REFERENCES

1. J. B. Allen, in *Mathematical Modeling of the Hearing Process—Proceedings, Troy, NY 1980*, edited by H. M. H. and L. A. Rubenfeld (Springer-Verlag, Berlin/Heidelberg, 1981), p. 1.
2. J. B. Allen and M. M. Sondhi, *J. Acoust. Soc. Am.* **66**, No. 1, 123 (1979).
3. J. B. Bell, P. Colella, and H. M. Glaz, *J. Comput. Phys.* **85**, 257 (1989).
4. R. P. Beyer, Ph.D. thesis, University of Washington, December 1989.
5. C. K. Birdsall, *Plasma Physics via Computer Simulation* (McGraw-Hill, New York, 1985).
6. A. J. Chorin, *Math. Comput.* **22**, 745 (1968).
7. L. J. Fauci, Ph.D. thesis, New York University, October 1986.
9. R. J. LeVeque, C. S. Peskin, and P. D. Lax, *SIAM J. Appl. Math.* **48**, 191 (1988).
10. C. H. Loh, *J. Acoust. Soc. Am.* **74**, 95 (1983).
11. C. S. Peskin, *J. Comput. Phys.* **25**, 220 (1977).
12. W. S. Rhode, *J. Acoust. Soc. Am.* **49**, No. 4 (Part 2), 1218 (1971).
13. W. S. Rhode, *J. Acoust. Soc. Am.* **67**, No. 5, 1696 (1980).
14. C. R. Steele and L. A. Taber, *J. Acoust. Soc. Am.* **65**, 1007 (1979).
15. C. R. Steele and J. G. Zias, *J. Acoust. Soc. Am.* **77**, 1849 (1985).
16. M. A. Viergever, in *Peripheral Auditory Mechanisms—Proceedings, Boston 1985*, edited by J. B. Allen, J. L. Hall, A. Hubbard, S. T. Neely, and A. Tubis (Springer-Verlag, Berlin/Heidelberg, 1986), p. 63.
17. M. A. Viergever, *J. Acoust. Soc. Am.* **64**, 1048 (1978).
18. M. A. Viergever, *Mechanics of the Inner Ear—A Mathematical Approach* (Delft Univ. Press, Delft, 1980).
19. G. von Békésy, *Experiments in Hearing* (McGraw-Hill, New York, 1960).
20. J. G. Zias, in *Peripheral Auditory Mechanisms—Proceedings, Boston 1985*, edited by J. B. Allen, J. L. Hall, A. Hubbard, S. T. Neely, and A. Tubis (Springer-Verlag, Berlin/Heidelberg, 1986), p. 73.
21. J. Zwislocki, *J. Acoust. Soc. Am.* **67**, 1679 (1980).