Comp. & Maths. with Appls. Vol. 11, No. 7/8, pp. 655–665, 1985
Printed in Great Britain.

0097–4943/85 $3.00 + .00

THE NUMERICAL SOLUTION OF THE HELMHOLTZ EQUATION FOR WAVE PROPAGATION PROBLEMS IN UNDERWATER ACOUSTICS

A. BAYLISS
Exxon Corporate Research, Annandale, NJ 08801, U.S.A.

C. I. GOLDSTEIN
Brookhaven National Laboratory, Upton, NY 11973, U.S.A.

and

E. TURKEL
Tel-Aviv University, Israel, and Institute for Computer Applications in Science and Engineering, Hampton, VA 23665, U.S.A.

Abstract—The Helmholtz Equation $(-\Delta - K^2n^2)u = 0$ with a variable index of refraction, n, and a suitable radiation condition at infinity serves as a model for a wide variety of wave propagation problems. A numerical algorithm has been developed and a computer code implemented that can effectively solve this equation in the intermediate frequency range. The equation is discretized by using the finite element method, thus allowing for the modeling of complicated geometries (including interfaces) and complicated boundary conditions. A global radiation boundary condition is imposed at the far-field boundary that is exact for an arbitrary number of propagating modes.

The resulting large, nonselfadjoint system of linear equations with indefinite symmetric part is solved by using the preconditioned conjugate-gradient method applied to the normal equations. A new preconditioner based on the multigrid method is developed. This preconditioner is vectorizable and is extremely effective over a wide range of frequencies, provided the number of grid levels is reduced for large frequencies. A heuristic argument is given that indicates the superior convergence properties of this preconditioner. The relevant limit to analyze convergence is for K increasing and a fixed prescribed accuracy level. The efficiency and robustness of the numerical algorithm are confirmed for large acoustic models, including interfaces with strong velocity contrasts.

1. INTRODUCTION

In this paper we describe a numerical method for approximately solving the Helmholtz equation

$$\Delta u + K^2n^2(x, y)u = 0. \tag{1.1}$$

Equation (1.1) with a suitable radiation condition at infinity describes both the propagation and scattering of time-harmonic waves in general geometries. We will restrict the application of (1.1) to problems that occur in underwater acoustics. Therefore u will represent the acoustic pressure, $n(x, y)$ the index of refraction, and K the wave number ($= 2\pi f/c_0$, where f is the frequency and c_0 is a reference sound speed). The region of interest will be a duct or waveguide containing inhomogeneities and interfaces. The numerical method is based on combining a finite-element discretization with a recently developed iterative method for solving the resulting system of linear equations[1]. We observe that this numerical method is also applicable to three-dimensional problems as well as vector formulations of (1.1), such as those describing elastic wave propagation[2].

Various computational techniques have been applied to simplified propagation models, including parabolic equation and normal-mode type methods, asymptotic methods, and others. For a survey of various models and computational methods, see [3]. Although each of these techniques is effective under suitable assumptions, there are many important problems for which it is necessary to treat the complete wave propagation model in the low-to-intermediate frequency range. Such models can include full-angle propagation and backscattering. This can occur, for example, when the ocean bottom must be taken into account or a layer of ice is present on the ocean surface.

There are several difficulties associated with the numerical solution of the full acoustic propagation model (1.1). The solution must often be resolved over many wavelengths, leading to very large systems of linear equations. Recent results have shown that as the frequency increases (or, equivalently, as the size of the model increases), the number of points/wavelength must be increased[4]. Thus, the number of equations increases faster than quadratically in K. A finite-element method has recently been developed that dramatically reduces the number of equations in regions where little backscattering is present[5]. The resulting matrices will be indefinite and also nonselfadjoint due to the radiation boundary condition. Furthermore, effective radiation boundary conditions to be imposed at a finite boundary must be developed. Various alternative approaches for dealing with these difficulties are discussed in [5].

We have developed a numerical algorithm and implemented a computer code that can effectively solve (1.1). The equation is discretized using the finite-element method, thus allowing considerable flexibility in modeling complex geometries. The resulting linear system of equations is solved iteratively by using the preconditioned conjugate gradient method applied to the normal equations[1]. This method requires relatively little storage (i.e. storage does not have to be allocated for the bandwidth) and is well suited and efficient for large problems. A global radiation boundary condition is imposed at the far-field boundary. This boundary condition is based on a modal expansion of the far-field solution that is exact for an arbitrary number of propagating modes (see [6] and [7]).

The effectiveness of the iterative method depends on the choice of the preconditioner. We consider preconditioners based on a splitting of the discrete Laplacian. In previous papers[1, 8] we investigated preconditioners based on SSOR and ADI. In this paper we describe a preconditioner based on a version of the multigrid method (introduced in [16]), which we have found to be extremely effective over a wide range of frequencies, provided the number of grid levels is decreased as the frequency increases. We shall see that this preconditioner is considerably more efficient than the SSOR and ADI preconditioners employed in [1] and [8]. This preconditioner has the additional advantage of being vectorizable, since a relaxation scheme based on a red black ordering[9] is used.

We close this section by outlining the remainder of the paper. In Sec. 2 we describe a wave-propagation model including an interface with a discontinuity in both the sound speed and density. We also describe the global radiation boundary condition, a finite-element method for solving this boundary value problem, and an efficient implementation of the global boundary condition with the finite-element method. In Sec. 3 we describe the iterative solution method and multigrid preconditioner. We also describe a new way of analyzing convergence of the iterative method, based on an accuracy condition developed in [4] relating the frequency and mesh size. Numerical results will be presented in Sec. 4, demonstrating the efficiency of the numerical method. We summarize our conclusions in Sec. 5.

2. CONTINUOUS AND DISCRETE MODEL

We shall describe our numerical method with respect to the following model problem. Consider (1.1) in a bounded portion of a two-dimensional semi-infinite rectangular waveguide. Hence, our computational domain D is given by $D = \{(x, y): 0 \leq x \leq \pi, 0 \leq y \leq \pi\}$. We assume that there is an interface Γ dividing D into two subregions, D_1 and D_2. Furthermore, suppose that the density ρ is piecewise constant with $\rho = \rho_1$ and D_1 and $\rho = \rho_2$ in D_2. Our propagation model is now given by the following boundary value problem:

$$
\begin{aligned}
&\text{(a)} \quad (-\Delta - K^2 n^2(x, y))u(x, y) = 0 \quad \text{in } D, \\
&\text{(b)} \quad \partial u(x, 0)/\partial y = 0, \\
&\text{(c)} \quad u(x, \pi) = 0, \\
&\text{(d)} \quad \partial u(0, y)/\partial x = g(y), \\
&\text{(e)} \quad \partial u(\pi, y)/\partial x = T(u), \\
&\text{(f)} \quad u_1(x, y) = u_2(x, y) \quad \text{for } (x, y) \text{ on } \Gamma, \\
&\text{(g)} \quad \rho_1^{-1}(\partial u_1(x, y)/\partial n) = \rho_2^{-1}(\partial u_2(x, y)/\partial n) \quad \text{for } (x, y) \text{ on } \Gamma,
\end{aligned}
\tag{2.1}
$$

where $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2$ denotes the Laplacian in Cartesian coordinates. The boundary operator T in (2.1)(e) is chosen to model the outflow of energy and will be discussed in detail below. In (2.1)(f) and (g), u_i denotes the restriction of the pressure u to D_i, $i = 1, 2$, and $\partial u_i/\partial n$ denotes the normal derivative on Γ pointing into D_2.

Remark 2.1

The Dirichlet boundary condition (2.1)(c) is a pressure release condition valid on the ocean surface. Condition (2.1)(b) models a rigid bottom, although a more general impedance condition could be implemented without difficulty. The forcing term in (2.1)(d) could readily be imposed as a Dirichlet condition instead of a Neumann condition. Furthermore, we could just as easily consider other coordinate systems (e.g. cylindrical coordinates) other than Cartesian. Finally, note that the method can be applied to problems with nonrectangular boundaries (see Ref. [6]). Based on our previous experience with complicated geometries[8], we do not anticipate a serious degradation in the numerical results.

We next define the radiation boundary operator, T, appearing in (2.1)(e). Consider the semi-infinite rectangular waveguide $\Omega = \{(x, y): 0 \leq x < \infty, 0 \leq y \leq \pi\}$. Assuming that $n(x, y) \equiv 1$ for $x \geq x_\infty$, we easily see that the outgoing solution of (2.1) can be expressed as

$$u(x, y) = \sum_{j=1}^{\infty} a_j \sqrt{\frac{2}{\pi}}\, e^{l_j x} \cos \sigma_j y \quad \text{for } x \geq x_\infty,$$

where $\sigma_j = j + 1/2$ and

$$l_j = \begin{cases} i\sqrt{K^2 - \sigma_j^2} & \text{for } K > \sigma_j, \\ -\sqrt{\sigma_j^2 - K^2} & \text{for } K < \sigma_j. \end{cases}$$

Thus the far-field solution is composed of a finite number of propagating modes (l_j imaginary) and an infinite number of evanescent modes (l_j real) that decay exponentially as $x \to \infty$. Set $f_j(y) = \sqrt{2/\pi} \cos \sigma_j y$ and note that

$$a_j = e^{-l_j x_\infty} \langle u, f_j \rangle_{\Gamma_\infty} \equiv e^{-l_j x_\infty} \int_0^\pi u(x_\infty, y) f_j(y)\, dy, \quad j = 1, 2, \ldots,$$

where $\Gamma_\infty \equiv \{(x, y): x = x_\infty, 0 \leq y \leq \pi\}$. We may now define our global boundary operator, as in [6] and [7], by

$$T(u) = \sum_{m=1}^{M} \langle u, f_m \rangle_{\Gamma_\infty} l_m f_m(y), \tag{2.2}$$

where M is chosen large enough to account for all propagating modes and those evanescent modes which are not small enough to be neglected at $x = x_\infty$.

Remark 2.2

When only very few propagating modes are important, a simpler local boundary operator can be efficiently implemented[1]. For example, if only the mth mode, $e^{l_m x} f_m(y)$, is propagating for $x \geq x_\infty$, the global operator T defined by (2.2) could be replaced by the impedance boundary operator given by

$$B(u) = (\partial/\partial x - l_m)u. \tag{2.2'}$$

However, as the number of significant modes increases, the order of the local boundary operator increases.

We next discretize the continuous model, (2.1), using the finite-element method. We first

observe, using integration by parts, that the solution u of (2.1) satisfies the following variational problem:

$$a(u, v) = -\rho_2\langle g, v\rangle_{\Gamma_0} \equiv -\rho_2 \int_0^{\pi} g(0, y)\overline{v(0, y)}\, dy \quad \forall\, v \text{ in } H^E \tag{2.3}$$

(assuming $\Gamma_0 \subset D_1$), where $\Gamma_0 \equiv \{(x, y): x = 0, 0 \leqslant y \leqslant \pi\}$, and H^E is the space of continuous complex-valued functions defined by

$$H^E \equiv \left\{ v: \sum_{j=1}^{2} \int_{D_j} (|v|^2 + |\nabla v|^2)\, dx\, dy < \infty \text{ and } v(x, \pi) = 0 \right\}.$$

The bilinear form $a(v, w)$ is defined by

$$\begin{aligned} a(v, w) \equiv \rho_2 \int_{D_1} (\nabla v \cdot \nabla \bar{w} - K^2 n^2 v \bar{w})\, dx\, dy \\ + \rho_1 \left(\int_{D_2} (\nabla v \cdot \nabla \bar{w} - K^2 n^2 v \bar{w})\, dx\, dy \right. \\ \left. - \oint_{\Gamma_2} T(v) \bar{w}\, dy \right) \quad \forall v, w \text{ in } H^E. \end{aligned} \tag{2.4}$$

To discretize problem (2.3), we introduce a family of finite-dimensional subspaces $S^h \in H^E$ such that S^h becomes dense in H^E as $h \to 0$. The approximate solution, u^h in S^h, of (2.1) or (2.3) is defined by

$$a(u^h, v^h) = -\rho_2\langle g, v^h\rangle_{\Gamma_0} \quad \forall v^h \text{ in } S^h. \tag{2.5}$$

It can generally be proved that $u^h \to u$ as $h \to 0$. For a comprehensive treatment of the finite-element method, see [10] or [11].

Remark 2.3

It is frequently convenient to replace the equations obtained from (2.5) by another system, where all terms that are multiplied by K^2 are lumped in the diagonal of the matrix. For a discussion of mass lumping, see Ref. [12]. All numerical results in Section 4 were obtained by using a lumped mass matrix.

Typically, the finite-element spaces S^h consist of sufficiently smooth piecewise polynomials of some fixed degree defined on a partitioning of D into simple subsets with diameter of order $O(h)$. We have implemented and tested a finite-element algorithm based on continuous piecewise linear functions defined on right triangles. Introducing a basis, $\{\phi_j\}$, for the finite-element space S^h (with dimension $N = N(h)$) in the usual way[10, 11], the approximate solution u^h in S^h may be expressed as

$$u^h(x, y) = \sum_{j=1}^{N} q_j \phi_j(x, y), \tag{2.6}$$

where $q_j = u^h(P_j)$. Substituting (2.6) into (2.5) with v^h given by ϕ_j, $j = 1, \ldots, N$, we obtain the following system of equations for the unknown column vector $\mathbf{q} = (q_1, \ldots, q_N)^T$:

$$A\mathbf{q} = \mathbf{g}, \tag{2.7}$$

where $\mathbf{g} = (g_1, \ldots, g_N)^T$, with $g_j = -\rho_2\langle g, \phi_j\rangle_{\Gamma_0}$, $j = 1, \ldots, N$, and the matrix A is given by

$$A = (a_{ij}), \quad a_{ij} = a(\phi_i, \phi_j), \quad i, j = 1, \ldots, N. \tag{2.8}$$

We close this section by describing a method for efficiently implementing the global boundary operator T (consisting of M modes). We start with Eqs (2.7) and let N_y denote the number of grid points in the y-direction. Note that the iterative solution method to be described in Sec. 3 merely requires matrix multiplications of the form $A\mathbf{x}$, with $\mathbf{x} = (x_1, \ldots, x_N)^T$, so that the matrix A need not be stored. It follows readily by using an appropriate ordering for the vertices and (2.2), (2.4) and (2.8) that for each matrix multiplication $A\mathbf{x}$ we must evaluate

$$y_i = \sum_{j=1}^{N_y} \langle T(\phi_j), \phi_i \rangle_{\Gamma_x} x_j, \quad i = 1, \ldots, N_y. \tag{2.9}$$

Equation (2.9) describes a full $N_y \times N_y$ matrix with elements $H_{ij} = \langle T(\phi_j), \phi_i \rangle_{\Gamma_x}$. If H were an arbitrary full matrix, the work involved in computing $A\mathbf{x}$ would be significantly larger than with the local boundary conditions, thus degrading the efficiency of the method. However, it is apparent from (2.2) that the continuous boundary operator $T(u)$ is of rank M, and this is also true for the finite-element approximation. Therefore $\mathbf{u} = (u_1, \ldots, u_{N_y})^T$ can be computed in $O(MN_y)$ operations.

In order to see this, assume for simplicity that $M = 1$. If we introduce the column vector $\mathbf{e} = (e_1, \ldots, e_{N_y})^T$, defined by $e_i = \langle \phi_i, f_1 \rangle_{\Gamma_x}$, then it can be readily seen that

$$H\mathbf{z} = l_1 \mathbf{e}\mathbf{e}^T\mathbf{z}, \tag{2.10}$$

where $\mathbf{z} = (z_1, \ldots, z_{N_y})^T$ is a vector ranging over the boundary points. If there are M modes in the boundary operator, then $H\mathbf{z}$ is a sum of M terms of the form (2.10).

3. SOLUTION METHOD

As seen in Sec. 2, a discretization of (2.1) leads to a system of equations

$$A\mathbf{x} = \mathbf{b}, \tag{3.1}$$

where A is typically quite large. Due to the properties of A already described, standard iterative methods are not applicable to the solution of (3.1)[13]. We have developed an iterative method for the solution of (3.1)[1], based on the preconditioned normal equations:

$$A'^*A'\mathbf{x}' = A'^*\mathbf{b}', \tag{3.2}$$

where $A' = Q_2^{-1}AQ_1^{-1}$, $\mathbf{x}' = Q_1\mathbf{x}$, $\mathbf{b}' = Q_2^{-1}\mathbf{b}$, and A'^* denotes the adjoint of A'. The preconditioning matrix $M^{-1} = Q_1^{-1}Q_2^{-1}$ was chosen in Refs. [1] and [8] to be an easily computed partial inverse of A_0, the positive definite matrix obtained by setting $K = 0$ in A. The conjugate gradient method is then applied to system (3.2) and is guaranteed to converge since the normal equations are positive definite symmetric. We refer to Ref. [1] for a detailed description of the conjugate gradient algorithm. The method requires only a small number of vector multiplications and additions, and it is only necessary to evaluate M^{-1}, A and A^* acting on a vector. Hence, no matrices need be inverted or stored, and the method is efficient, provided M^{-1} sufficiently reduces the number of iterations. In [1] and [8] we showed this to be the case when M^{-1} was chosen to be one or more sweeps of point SSOR or ADI applied to the discrete Laplacian, A_0.

Convergence of an iterative method is usually defined by letting the mesh size $h \to 0$ (hence, the number of equations $N \to \infty$) and keeping all other parameters fixed[13, 14]. However, in order to maintain accuracy as the wave number K increases, it is necessary to reduce h. Hence, K and h must be constrained by means of an accuracy condition[4]. In Ref. [4] we showed, e.g., that $K^{3+\alpha}h^2$ must be kept constant for continuous piecewise linear finite elements, where $\alpha \geqslant 0$ in general, but in many cases $\alpha = 0$. This implies that the number of points/wavelength must be increased as K increases. This is also the case as the domain size is increased.

In view of the above, a more relevant definition of convergence is obtained by prescribing a fixed accuracy (e.g. K^3h^2 fixed) and letting K increase. This has an important bearing on the

choice of preconditioner. In particular, preconditioners based on fast Laplace solvers are superior with respect to the standard definition of convergence but will have a very unfavorable growth rate as K increases. To illustrate this, we consider the following model problem on the unit square with Dirichlet boundary conditions:

$$-\Delta u - K^2(1 + i\delta/K)u = 0. \tag{3.3}$$

The dissipative term $i\delta K$ models the addition of a term, $\delta \mathrm{d}/\mathrm{d}t$, to the wave equation and is included to model the radiation condition.

We will study the convergence rate of the algorithm by analyzing the condition number κ of the matrix A'^*A'. It is well known[14] that, in general, the convergence rate of the conjugate gradient method applied to (3.2) is inversely proportional to $\Lambda \equiv \kappa^{1/2}$. Let A be the matrix corresponding to the standard five-point discretization of (3.3) (multiplied by h^2). Suppose that the preconditioning matrix M^{-1} is positive definite symmetric and commutes with A. If $Q_1 = Q_2 = M^{1/2}$, then Λ is given by $|\lambda_{\max}/\lambda_{\min}|$, where $\lambda_{\max}$ is the largest eigenvalue of A' (in magnitude), and $\lambda_{\min}$ is the smallest eigenvalue of A' (in magnitude). We easily see that the eigenvalues of A are given by

$$C_\alpha h^\alpha - K^2h^2(1 + i\delta/K),$$

where $0 \leqslant \alpha \leqslant 2$ and C_α is a constant.

We now consider K and h related by $K^3h^2 = \epsilon$. The eigenvalues are given by

$$C_\alpha K^{-3\alpha/2} - (\epsilon/K)(1 + i\delta/K). \tag{3.4}$$

The cases $\alpha = 0$ and $\alpha = 2$ give eigenvalues that are $O(1)$ and $O(K^{-1})$. If we precondition by a complete inverse of the discrete Laplacian, then the eigenvalues become

$$C_\alpha - \epsilon K^{3\alpha/2-1}(1 + i\delta/K). \tag{3.5}$$

The cases $\alpha = 0$ and $\alpha = 2$ give eigenvalues that are $O(1)$ and $O(K^2)$. By examining the case $\alpha = 2/3$ we see that the real parts of (3.4) and (3.5) can vanish. Therefore, in (3.4) we can have eigenvalues which are $O(K^{-2})$, and Λ can be as large as $O(K^2)$. Similarly, in (3.5) we can have eigenvalues which are $O(K^{-1})$, and Λ can be as large as $O(K^3)$, thus giving a very slow convergence rate for large K.

In the remainder of this section we construct a preconditioner based on a multigrid method—see Ref. [16] (applied to the discrete Laplacian, A_0)—that is much more efficient than the preconditioners considered in Refs. [1] and [8]. We begin by briefly describing a multigrid cycle. Consider a sequence of grids $G^0, \ldots, G^M$, with mesh size h_i on grid G^i given by $h_i = 2^{-i}h_0$, $i = 0, \ldots, M$, where $h_M = h$ and h_0 is independent of h. (Hence, the number of levels increases as $h \to 0$.) We obtain a sequence of equations

$$A^i\mathbf{x}^i = \mathbf{b}^i, \quad i = 1, \ldots, M,$$

in the same way as (3.1) by discretizing (2.1) as in Section 2 with mesh size h_i replacing h. We choose some relaxation scheme for each grid level, as well as some interpolation operator I_i^j from grid G^i to grid G^j.

To begin the cycle, we make r relaxation sweeps on the finest grid level, G^M, and then transfer the residual

$$\mathbf{R}^{M-1} = I_M^{M-1}\mathbf{R}^M = I_M^{M-1}(\mathbf{b}^M - \mathbf{A}^M\mathbf{x}^M)$$

to grid G^{M-1}. On this grid we obtain the equation

$$A^{M-1}\mathbf{v}^{M-1} = \mathbf{R}^{M-1}.$$

This process is repeated until we get to the coarsest grid G^0. On this grid we make l relaxation sweeps. To return to the finer grid, we again make l relaxation sweeps on G^0. We then calculate $\mathbf{u}^1$ by

$$\mathbf{u}^1_{\text{new}} = \mathbf{u}^1_{\text{old}} + I^1_0 \mathbf{v}^0.$$

We continue this process, using s relaxation sweeps on each successive grid up to the finest grid. The number of relaxation sweeps, r, s, and l will be discussed in the next section. The entire process defines one complete multigrid cycle. (See Ref. [15] for a more detailed description of the multigrid method and Ref. [16] for more details on the implementation of multigrid as a preconditioner.)

Since we are using the multigrid cycle as a preconditioning, we want the cycle to be positive definite and symmetric. Assuming a zero initial guess, this requires that all operations in the direction of finer grids are the adjoint of the operations in the direction of coarser grids. An efficient relaxation scheme for the Laplace equation is Gauss–Seidel with red black (RB) ordering[9]. This also makes the algorithm vectorizable. In view of symmetry considerations, we see that if RB is used when going to coarser grids, then BR must be used when returning to finer grids. Furthermore, we maintain symmetry by using linear interpolation for I^j_{j-1} and a full weighting for the fine-to-coarse residual transfer I^{j-1}_j[15].

It can be shown that a multigrid cycle reduces the error by a fixed amount, independent of h, and only requires $O(N)$ operations. Hence, a fixed number of multigrid cycles applied to the Laplacian acts as a fast Laplace solver. In view of our earlier remark concerning fast Laplace solvers, we expect that a multigrid cycle will be effective as a preconditioner for small K, but the number of iterations will grow rapidly as K increases, subject to the accuracy constraint. These conclusions are confirmed numerically in Ref. [16] in connection with a symmetric Helmholtz operator (i.e. a Neumann boundary condition replaces the radiation boundary condition).

In order to obtain a slower growth in the number of iterations as K increases, we introduce the following idea. Consider the model problem (3.3) and suppose that a preconditioner is used which acts as an inverse of the discrete Laplacian (multiplied by h^2) on those eigenvectors corresponding to a limited part of its spectrum, say those eigenvalues of the form $O(h^\alpha)$ with $0 \leqslant \alpha \leqslant \alpha_0$. Furthermore, assume that the remaining eigenvectors are essentially unchanged by the preconditioner. Then the eigenvalues of A' are given by

$$\begin{aligned} &C_\alpha - \epsilon K^{3\alpha/2-1}(1 + i\delta/K) \quad \text{for } 0 \leqslant \alpha \leqslant \alpha_0, \\ &C_\alpha K^{-3\alpha/2} - (\epsilon/K)(1 + i\delta/K) \quad \text{for } \alpha_0 < \alpha \leqslant 2. \end{aligned} \tag{3.6}$$

It can be seen that if $\alpha_0 = 2/3$, then Λ is only $O(K)$.

If we now neglect the effects of the residual transfers and coarse-to-fine interpolations in the multigrid preconditioner and assume that the relaxation eliminates the highest frequencies of the error on each grid, then we can treat multigrid as a preconditioner of the previous form with eigenvalues of the preconditioned matrix given by (3.6). For simplicity assume that $h = 2^{-L}$ on the finest grid. Hence, we choose the coarsest grid, G_J, to be such that the highest frequencies on this grid are $O(h^{2/3})$. This gives

$$h_J = 2^{-J} = 2^{-2L/3}. \tag{3.7}$$

The numerical examples will demonstrate that, for $N = 128$, three levels are optimal. This is very close to the value predicted by (3.7). Either increasing or decreasing the number of levels results in a striking increase in the number of iterations. Furthermore, for $N = 200$ a simple modification of the above argument predicts that a coarsest grid of either 25 or 50 points should be optimal. In fact, we find that a coarsest grid of 50 points is optimal. Thus, although the above argument is only heuristic, it does correctly predict nearly optimal coarsest grids. A rigorous argument validating our heuristic result will be given elsewhere.

4. NUMERICAL RESULTS

In this section we present typical numerical results obtained by using the multigrid preconditioner described in the preceding section. All results were obtained for the problem described in Section 2. The results were obtained on a square of length π using piecewise linear elements on right triangles. There are N grid intervals in each direction, so that the number of equations is $(N + 1)^2$. In all cases the mass matrix is lumped.

The numerical examples are designed to illustrate the convergence properties of the algorithm when K increases and K and h are constrained by the accuracy requirement that K^3h^2 be fixed. As was indicated in Section 3, for large K the preconditioner will be most effective when only a small number of levels are used in the multigrid algorithm. The results demonstrate the sensitivity of the preconditioner to the number of levels. In addition, the numerical results illustrate the effect of boundary conditions on convergence properties of the algorithm and the robustness of the method as interfaces with strong contrast (causing large backscattering) are introduced. In examples 1–4 we assume no interface present, and $n(x, y) \equiv 1$ in (2.1).

We define convergence of the iterative method to mean that the normalized mean square norm of $M^{-1}r$ is less than 10^{-6}, where r is the residual. This quantity is naturally produced by the implementation of the preconditioned conjugate gradient algorithm. We have verified that monitoring the norm of r instead of $M^{-1}r$ causes only slight changes in the number of iterations. Note that our stopping criteria is more stringent than might be required in practice, where the truncation error is often sufficient to stop the iteration process.

In example 1 we consider (2.1) with the Neumann data (2.1d) chosen so that the exact solution is

$$u(x, y) = \exp\left(i\sqrt{K^2 - \tfrac{1}{4}}\, x\right) \cos(y/2). \tag{4.1}$$

The radiation boundary condition is the local condition, (2.2′), which is exact for this mode. Based on the argument of Section 3, we use only three grid levels. In addition, we use two relaxation sweeps on the coarsest grid and one sweep on the other grids. In Table 1 the number of iterations required for convergence is shown for different values of K and N. The last three entries in Table 1 show the number of iterations required for three frequencies for which $K^3h^2 = 0.5$ (corresponding to a normalized mean square error of about 7%). The first entry corresponds to a case solved by using SSOR as a preconditioner, which required 284 iterations for convergence[1]. The same problem required 290 iterations to converge when using ADI as a preconditioner (see Ref. [8]). It is apparent that on this simple problem, the multigrid preconditioner is more effective than SSOR and ADI. Furthermore, the growth in the number of iterations is slow as K increases with K^3h^2 fixed.

In example 2 we consider the Neumann data as a source modeled by the derivative of a Gaussian centered at $y = \pi/2$. The radiation boundary condition is now given by (2.2) with the sum extending over all propagating modes and the first four evanescent modes. Results are given for different values of K with K^3h^2 fixed at 1.01 and 0.425. By examining modal solutions for these parameters, we believe that the first case corresponds to roughly 10% accuracy, and the second case corresponds to roughly 5% accuracy. The results in Table 2 correspond to three levels in the multigrid algorithm, with two relaxation sweeps on the coarsest grid and one sweep on the other grids. It is seen that the number of iterations grows close to linearly with K for K^3h^2 fixed. We also observe that for fixed grid size, the number of iterations appears to grow at a rate of $O(K^2)$. This shows that the evaluation of the effectiveness of a preconditioner for large K depends crucially on the relationship between K and h. In Table 3, results are given in

Table 1. Results for example 1

K	N	K^3h^2	Iterations
4.16	61	.197	24
5.92	65	.5	37
7.76	97	.5	43
9.4	129	.5	46

Table 2. Results for example 2

K	N	K^3h^2	Iterations
11.88	129	1.01	640
16	201	1.01	863
20	281	1.01	1059
8.9	129	.425	292
12	201	.425	472
15	281	.425	674

which four levels in the multigrid preconditioner and four relaxation sweeps on the coarsest grid are used. It is apparent that the convergence is considerably worse in this case. For the last two entries in Table 3, seven levels were used, and the degradation of the convergence is quite striking.

Due to space limitations, detailed comparisons of the multigrid preconditioner with SSOR and other preconditioners will be presented elsewhere. We simply state that SSOR for the case of $K = 20$ and $N = 281$ did not converge in 2200 iterations. Furthermore, for K^3h^2 fixed the number of iterations required for convergence is increasing at a rate greater than $O(K^{3/2})$. An operation count (counting additions and multiplications equally) indicates that each iteration with multigrid as a preconditioner requires about twice the work of an iteration using SSOR as a preconditioner. The $O(K)$ growth in the number of iterations as K increases makes the multigrid preconditioner significantly more efficient for large models.

In the remaining examples we use the same multigrid preconditioner as in example 1. In examples 3 and 4 we consider the effect of the radiation boundary condition on the number of iterations required for convergence. In example 3 the Neumann data (2.1d) is chosen so that the solution is (4.1), whereas in example 4 the Neumann data is the derivative of a Gaussian. In both examples $K = 16$ and $N = 201$. The radiation boundary condition is either the local boundary condition (2.2′) or the global boundary condition (2.2), accounting for 1, 5, 10 and 20 modes. The results are presented in Tables 4 and 5. In example 4 there are 16 propagating modes, so the only boundary condition that accounts for all of the modes is the last one. In example 3, however, there is only one mode in the solution, and all of these boundary conditions are nonreflecting on that mode.

The results in Table 4 indicate that the radiation boundary condition has a very small effect on the number of iterations required for convergence when one mode is present in the solution. In the case of the derivative of a Gaussian, however, the global boundary conditions that allow for reflections require considerably more iterations than the boundary condition that accounts for all propagating modes. It can be shown that if the global boundary condition (2.2) does not account for all of the propagating modes, the boundary value problem (2.1) can be singular or have eigenvalues very close to zero. This can certainly degrade the conditioning of the matrix. A comparison of Tables 4 and 5 also indicates that the number of iterations increases as the number of propagating modes increases, even if the radiation boundary condition is accurate for all of these modes.

In examples 5 and 6 we consider the effect of a rectangular interface with a piecewise constant index of refraction. In both cases the Neumann data is given by the derivative of a

Table 3. Results for example 2 using 4 levels and 4 relaxation sweeps on the coarsest grid

K	N	K^3h^2	Iterations
11.88	129	1.01	2511
16	201	1.01	>3900
20	281	1.01	>2000
8.9	129	.425	956
12	201	.425	1492
15	281	.425	1918
11.88†	129	1.01	3969
8.9†	129	.425	1516

†7 Levels and 2 Relaxation Sweeps on the Coarsest Grid

Table 4. Results for example 3 (one mode in solution)

Boundary Condition	Number of Modes in BC	Iterations
Local	—	81
Global	1	77
Global	5	80
Global	10	81
Global	20	81

Table 5. Results for example 4 (point source)

Boundary Condition	Number of Modes in BC	Iterations
Local	—	805
Global	1	1401
Global	5	1411
Global	10	1226
Global	20	863

Table 6. Results for example 5

K	N	Index of Refraction	Iterations
16	201	1	863
16	201	0.5	856
16	201	0.33	798
16	201	0.25	841

Table 7. Results for example 6

K	N	Index of Refraction	Iterations
16	201	1	863
16	201	0.5	943
16	201	0.33	1108
16	201	0.25	788
16	201	1.25	1082
20	281	1	1059
20	281	0.33	1269

Gaussian. In example 5 we consider an interface with $\pi/4 \leqslant x \leqslant \pi/2$ and $0 \leqslant y \leqslant \pi/4$, where the index of refraction, n, may be a constant other than 1. In example 6 the size of the region is extended, so that $\pi/4 \leqslant x \leqslant 3\pi/4$ and $0 \leqslant y \leqslant \pi/2$. In these examples n varies from 1 to 0.25 and in one case is 1.25. These contrasts would cause considerable backscattering, so that the parabolic equation method is expected to be inaccurate for these problems. The number of iterations required for convergence is shown in Tables 6 and 7 for these examples. The results indicate that the preconditioner is robust and can handle strong contrasts extending over relatively large regions.

5. CONCLUSIONS

We have described a general method to solve Helmholtz-type equations for an intermediate range of frequencies. The iterative method is based on obtaining an effective preconditioner which enables the solution to be obtained in a relatively small number of iterations. The relevant limit to analyze the convergence properties of a preconditioner is for K increasing and a fixed prescribed accuracy level. In this regime the number of iterations increases at a rate greater

than $O(K^2)$ when using a complete multigrid cycle or other methods based on fast solvers as a preconditioner. This is very unsatisfactory for large frequencies. For SSOR the number of iterations increases at a rate greater than $O(K^{3/2})$, which is also unfavorable for large K.

The use of multigrid as a preconditioner with a restricted number of levels gives a rate of increase of $O(K)$, thus resulting in a significantly more effective algorithm. This is demonstrated by both a heuristic argument and by numerical results. Using this method we have been able to solve two-dimensional problems with up to 10 wavelengths in each direction and with more than 78,000 unknowns in a reasonable number of iterations. For example, with a sound speed of 5000 ft/s and a frequency of 10 Hz, this corresponds to a square of length 5000 ft. On an IBM 3033 computer with double-precision arithmetic, the large model (i.e. $K = 20$ and 281×281 unknowns) required four hours of computer time, and the smaller model ($K = 16$ and 201×201 unknowns) required 98 minutes of computer time. The computational effort is significantly reduced when using the truncation error instead of 10^{-6} as a stopping criterion for the iterative method. Furthermore, both the storage requirements and computation time are greatly reduced if we approximate the magnitude of the pressure instead of the pressure. This is sufficient for many applications. Problems with strong velocity contrast do not appear to significantly degrade the performance of the numerical algorithm. Thus, this method may be suitable for efficient computation of the full acoustic model in cases where one-way propagation models and other approximate models would be inaccurate.

Acknowledgements—This work was supported in part by the Applied Mathematical Sciences subprogram of the Office of Energy Research, U.S. Department of Energy, under contract No. DE-AC02-76CH00016. One of the authors was partially supported by the National Aeronautics and Space Administration under NASA Contract NAS1-17130 while the author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23665.

REFERENCES

1. A. Bayliss, C. I. Goldstein and E. Turkel, An iterative method for the Helmholtz equation. *J. Comput. Phys.* **49**, 443 (1983).
2. L. Brekhovskikh, *Waves in Layered Media*. Academic Press, New York (1960).
3. J. A. DeSanto, Editor, *Ocean Acoustics*. Springer-Verlag, New York (1979).
4. A. Bayliss, C. I. Goldstein and E. Turkel, On accuracy conditions for the numerical computation of waves. *J. Comput. Phys.*, to appear.
5. C. I. Goldstein, "The numerical solution of underwater acoustic propagation problems using finite difference and finite element methods." Naval Res. Lab. Report No. 8820 (1984).
6. G. J. Fix and S. P. Marin, Variational methods for underwater acoustic problems. *J. Comput. Phys.* **28**, 253 (1978).
7. C. I. Goldstein, A finite element method for solving Helmholtz type equations in waveguides and other unbounded domains. *Math. Comp.* **39**, 309 (1982).
8. A. Bayliss, C. I. Goldstein and E. Turkel, Preconditioned conjugate gradient methods for the Helmholtz equation, in *Elliptic Problem Solvers*. II (Edited by G. Birkhoff), Vol. 2, pp. 233–243. Academic Press, New York (1984).
9. A. Brandt, Multigrid solvers on parallel computers in *Elliptic Problem Solvers* (Edited by M. Schultz), pp. 39–84. Academic Press, New York (1981).
10. G. Strang and G. Fix, *An Analysis of the Finite Element Method*. Prentice-Hall, Englewood Cliffs, N.J. (1973).
11. O. C. Zienkiewicz, *The Finite Element Method in Engineering Science*. McGraw-Hill (1971).
12. R. L. Kuhlemeyer and J. Lysmer, Finite element method accuracy for wave propagation problems. *J. Soil Mech. Found. Div. Proc. ASCE*, **99SM5**, 421 (1973).
13. D. M. Young, *Iterative Solution of Large Linear Systems*. Academic Press, New York (1971).
14. H. C. Ellman, "Iterative methods for large, sparse, nonsymmetric systems of linear equations." Yale U., Dept. Comp. Science, Research Report 229 (1981).
15. A. Brandt, Multi-level adaptive solutions to boundary value problems. *Math. Comp.* **31**, 333 (1977).
16. J. Gozani, A. Nachshon and E. Turkel, Conjugate gradient coupled with multigrid for an indefinite problem, in *Advances in Computer Methods for Partial Differential Equations*. V (Edited by R. Vichnevetsky and R. Stepleman). IMACS (1984).