JOURNAL OF COMPUTATIONAL PHYSICS 50, 193–208 (1983)

A Numerical Algorithm for Solving Inverse Problems of Two-Dimensional Wave Equations

Y. M. CHEN

Department of Applied Mathematics and Statistics,
State University of New York,
Stony Brook, New York 11794

AND

J. Q. LIU

Department of Applied Mathematics,
Harbin Institute of Technology,
Harbin, People's Republic of China

Received April 20, 1982; revised August 10, 1982

The new iterative numerical algorithm of the pulse–spectrum technique (PST) is extended and developed to solve inverse problems of two-dimensional linear wave equations. It has the practical advantages of having the necessary data measured on a portion of the boundary only and no geometric limitation on the testing objects. Numerical simulations on several simple examples are carried out to test the feasibility and to study the general characteristics of this technique without the real measurement data. It is found that PST does give excellent results even with a very coarse computational grid and it is as robust as in the one-dimensional case.

INTRODUCTION

Inverse problems of wave equations involve the reconstruction of the coefficients characterizing the inhomogeneity of the propagating medium from experimental measured wave motions. Usually the solution of an inverse problem is not unique and does not depend continuously on the given data. Its applications can be found in many areas of engineering and geophysics, e.g., nondestructive evaluation of materials in engineering and the determination of the Earth's interior structure from reflection seismic data in geophysics. For the one-dimensional inverse problems, although there are many different methods, very few of them can be generalized in practice to solve higher dimensional inverse problems. In this paper, the pulse-spectrum technique (PST) is extended and further developed to solve inverse problems of two-dimensional linear wave equations in an arbitrary finite domain.

The basic idea of PST is that data are measured in the time domain with compact support and the synthesis of the unknown coefficient is carried out numerically in the

0021-9991/83 $3.00

complex frequency domain by an iterative algorithm. The PST was first introduced by Tsien and Chen [1] for solving an idealized velocity inverse problem in fluid dynamics. Then it was further developed by Chen and Tsien [2] to have the capability of handling noisy, poorly distributed, and inadequately measured data. It also was used to solve an inverse problem in electromagnetic wave propagation by Tsien and Chen [3]. Later it was extended successfully to solve inverse problems of a nonlinear acoustic wave equation by Hatcher and Chen [4]. Recently it has been used successfully by Chen and Weng [5] to synthesize nonuniform transmission lines. Moreover, the discretized version of this iterative algorithm under idealized conditions has been proved to converge quadratically [6], which is quite efficient from the numerical computation point of view. In a different direction, the PST has also been modified to solve one-dimensional and two-dimensional inverse problems of linear diffusion equations with great success [7, 8].

In this paper, PST is presented and extended for solving inverse problems of a two-dimensional linear wave equation and its feasibility is demonstrated by simple examples with calculations done on very coarse computational grids. No claims that either the results here are optimal or that a completely adaptable computer code has been developed are made here. However, it is found that PST fares very well with regard to the following four practical criteria for the evaluation of any numerical method:

(a) Universality criterion: Can a numerical method which is effective in one-space-dimensional problems be extended with similar success into higher-space-dimensional applications? Can a solution method which is effective for solving inverse problems of one type of equations, e.g., hyperbolic or parabolic type, be extended to solve the inverse problems of the other type of partial differential equations with similar success and minimum efforts?

(b) Economy of data acquisition criterion: The numerical method should be able to keep the difficulties and the cost expenditure of acquiring or measuring the necessary data for a successful calculation to minimum.

(c) Economy of programming effort criterion: The numerical method should be as close to the nondedicated program as possible, for existing practices of programming new dedicated numerical methods for every special types of problems can be unacceptably costly in many practical circumstances. Furthermore, the computer code should also contain as many as possible of the modules where the canned subroutines can be readily called upon.

(d) Economy of computing cost criterion: The numerical method should keep the cost of IO and CPU times and memory storage to minimum.

For simplicity, the formulation of the inverse problem of a linear two-dimensional wave equation is presented and the basic numerical algorithm of PST is given in the next section. Then numerical simulations are carried out to test the feasibility and to study the intrinsic characteristics of this numerical algorithm without the real

measurement data. Finally, a comprehensive discussion of the numerical results, their implication in actually implementing this computational algorithm, and the merits of PST are given.

Numerical Algorithm (PST)

Consider the following initial-boundary value problem of a two-dimensional linear wave equaion,

$$\begin{aligned} &\partial\{k(x,y)\,\partial u/\partial x\}/\partial x + \partial\{k(x,y)\,\partial u/\partial y\}/\partial y - \rho\partial^2 u/\partial t^2 \\ &\quad = 0, \qquad (x,y)\in\Omega, \quad 0<t<\infty, \\ &\qquad u(x,y,0) = \partial u(x,y,0)/\partial t = 0, \\ &(\partial u/\partial n + \alpha_\lambda u)|_{\Gamma_\lambda} = f_\lambda(x,y,t), \qquad \lambda = 1,2,\ldots,\Lambda, \end{aligned} \tag{1}$$

where Ω is a bounded region in xy space and $\Gamma = \Gamma_1 + \Gamma_2 + \cdots + \Gamma_\Lambda$ is the boundary of Ω.

Here the inverse problem is to determine the unknown coefficient $k(x,y)$ from the known coefficient ρ, the known boundary conditions $f_\lambda(x,y,t)$, $\lambda = 1, 2,\ldots, \Lambda$, and the additionally measured auxiliary data,

$$u|_{\tilde{\Gamma}} = h(x,y,t), \tag{2}$$

where $\tilde{\Gamma}$ is a portion of Γ_λ on which u is not given by (1), and both the $f_\lambda(x,y,t)$'s and $h(x,y,t)$ are Laplace transformable.

The PST calls for the Laplace transformation of (1) and (2) so that the entire system is transformed from the time domain to the complex frequency domain, and the corresponding system is

$$\begin{aligned} &\partial\{k(x,y)\,\partial v/\partial x\}/\partial x + \partial\{k(x,y)\,\partial v/\partial y\}/\partial y - \rho s^2 v = 0, \qquad (x,y)\in\Omega, \\ &(\partial v/\partial n + \alpha_\lambda v)|_{\Gamma_\lambda} = F_\lambda(x,y,s), \qquad \lambda = 1,2,\ldots,\Lambda, \end{aligned} \tag{3}$$

$$v|_{\tilde{\Gamma}} = H(x,y,s), \tag{4}$$

where $v(x,y,s)$, $F_\lambda(x,y,s)$, and $H(x,y,s)$ are the Laplace transformations of $u(x,y,t)$, $f_\lambda(x,y,t)$, and $h(x,y,t)$, respectively.

Now, the inverse problem is to determine $k(x,y)$ from ρ, $F_\lambda(x,y,s)$, $\lambda = 1, 2,\ldots, \Lambda$, and $H(x,y,s)$.

The iterative numerical algorithm begins by setting

$$v_{n+1} = v_n + \delta v_n, \qquad k_{n+1} = k_n + \delta k_n, \qquad n = 0, 1, 2, 3,\ldots, \tag{5}$$

where $k_0(x, y)$ is the initial guess for the unknown coefficient $k(x, y)$, $\|k_n\| > \|\delta k_n\|$ and $\|v_n\| > \|\delta v_n\|$, and $k_0|_\Gamma = k|_\Gamma$. Upon substituting (5) into (3) and neglecting terms of order δ^2 and higher, one obtains a system for v_n,

$$\begin{aligned} \partial(k_n \partial v_n/\partial x)/\partial x + \partial(k_n \partial v_n/\partial y)/\partial y - \rho s^2 v_n = 0, \qquad & (x, y) \in \Omega, \\ (\partial v_n/\partial n + \alpha_\lambda v_n)|_{\Gamma_\lambda} = F_\lambda(x, y, s), \qquad & \lambda = 1, 2, \ldots, \Lambda, \end{aligned} \tag{6}$$

and a system for δv_n,

$$\begin{aligned} &\partial(k_n \partial \delta v_n/\partial x)/\partial x + \partial(k_n \partial \delta v_n/\partial y)/\partial y - \rho s^2 \delta v_n \\ &\quad = -\partial(\delta k_n \partial v_n/\partial x)/\partial x - \partial(\delta k_n \partial v_n/\partial y)/\partial y, \\ &(\partial \delta v_n/\partial n + \alpha_\lambda \delta v_n)|_{\Gamma_\lambda} = 0, \qquad \lambda = 1, 2, \ldots, \Lambda. \end{aligned} \tag{7}$$

By using the method of Green's function, elliptic partial differential equation (7) can be changed to a Fredholm integral equation of the first kind which relates $\delta k_n(x, y)$ to $\delta v_n(x, y, s)$ as

$$\begin{aligned} &\iint_\Omega G_n(x, y, x', y', s)\{\partial(\delta k_n \partial v_n/\partial x')/\partial x' + \partial(\delta k_n \partial v_n/\partial y')/\partial y'\}\, dx' dy' \\ &\quad = -\delta v_n(x, y, s), \end{aligned} \tag{8}$$

where $G_n(x, y, x', y', s)$ is the Green's function of the differential operator in (7). Moreover, $v_{n+1}(x, y, s)$ at the right hand side of (8) can be replaced by $v(x, y, s)$. Upon setting (x, y) at $\tilde{\Gamma}$, one obtains from (4) and (8) a Fredholm integral equation of the first kind for $\delta k_n(x, y)$ as

$$\begin{aligned} &\iint_\Omega G_n|_{\tilde{\Gamma}}\, \{\partial(\delta k_n \partial v_n/\partial x')/\partial x' + \partial(\delta k_n \partial v_n/\partial y')\, \partial y'\}\, dx'\, dy' \\ &\quad = -H(x, y, s)|_{\tilde{\Gamma}} + v_n|_{\tilde{\Gamma}} \end{aligned} \tag{9}$$

or defined as

$$\iint_\Omega K_n(\tilde{\Gamma}, s, x', y')\, \delta k_n\, dx'\, dy' = g_n(\tilde{\Gamma}, s). \tag{9'}$$

Equations (5), (6), and (9) form the basic structure for each iteration in the iterative numerical algorithm of PST. First, a numerical integration subroutine is used to evaluate the Laplace transforms $F_\lambda(x, y, s)$ and $H(x, y, s)$ at $s = s_\sigma$, $\sigma = 1, 2, 3, \ldots, \Sigma$. Then these discrete values will be used to solve (6) and (9) numerically.

Boundary value problem (6) and the Green's function of (7) can be solved numerically by simply using the following first-order finite difference method. Assuming that Ω can be approximated by a collection of small quadrilaterals

(triangles as a special case) where the ratio of the maximum dimension to the minimum dimension is of order one and each computational grid point is denoted by a pair of numbers (x, y), the finite difference approximation at an interior point (x_i, y_j) is derived by considering the area integration of an element area centered at (i, j) (Fig. 1) as

$$\iint_{\Omega_{ij}} \{(k_n \partial v_n/\partial x)/\partial x + \partial(k_n \partial v_n/\partial y)/\partial y - s_\sigma v_n\}\, dx\, dy = 0. \tag{10}$$

By Green's formula, (10) becomes

$$\sum_{l=1}^{4} \int_{\Gamma_{ijl}} k_n \partial v_n/\partial n\, d\gamma - \iint_{\Omega_{ij}} s_\sigma v_n\, dx\, dy = 0. \tag{11}$$

The line integral of (11) can be approximated by

$$\begin{aligned}
\int_{\Gamma_{ij1}} k_n \partial v_n/\partial n\, d\gamma &\sim \frac{k_{ni,j} + k_{ni+1,j}}{2\cos\theta_1} \left\{ \frac{v_{ni+1,j} - v_{ni,j}}{\gamma_{(i,j)(i+1,j)}} \right. \\
&\quad \left. - \frac{v_{ni,j+1} - v_{ni,j-1}}{2\gamma_1} \sin\theta_1 \right\} \gamma_1, \\
\int_{\Gamma_{ij2}} k_n \partial v_n/\partial n\, d\gamma &\sim \frac{k_{ni,j} + k_{ni,j+1}}{2\cos\theta_2} \left\{ \frac{v_{ni,j+1} - v_{ni,j}}{\gamma_{(i,j+1)(i,j)}} \right. \\
&\quad \left. - \frac{v_{ni-1,j} - v_{ni+1,j}}{2\gamma_2} \sin\theta_2 \right\} \gamma_2, \\
\int_{\Gamma_{ij3}} k_n \partial v_n/\partial n\, d\gamma &\sim \frac{k_{ni,j} + k_{ni-1,j}}{2\cos\theta_3} \left\{ \frac{v_{ni-1,j} - v_{ni,j}}{\gamma_{(i-1,j)(i,j)}} \right. \\
&\quad \left. - \frac{v_{ni,j-1} - v_{ni,j+1}}{2\gamma_3} \sin\theta_3 \right\} \gamma_3, \\
\int_{\Gamma_{ij4}} k_n \partial v_n/\partial n\, d\gamma &\sim \frac{k_{ni,j} + k_{ni,j-1}}{2\cos\theta_4} \left\{ \frac{v_{ni,j-1} - v_{ni,j}}{\gamma_{(i,j-1)(i,j)}} \right. \\
&\quad \left. - \frac{v_{ni+1,j} - v_{ni-1,j}}{2\gamma_4} \sin\theta_4 \right\} \gamma_4,
\end{aligned} \tag{12}$$

and the area integral of (11) can be approximated by

$$\iint_{\Omega_{ij}} s_\sigma v_n\, dx\, dy \sim s_\sigma v_{nij} \Delta_{ij}, \tag{13}$$

where the θ's are the angles between the normal vectors and local axes of the computational grid measured counterclockwise (Fig. 1); the γ_l's are the lengths of the

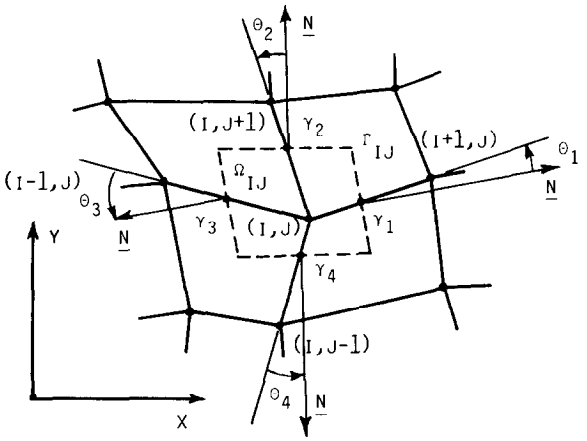


FIG. 1. Computational grid of the finite difference method at an interior point (i, j).

Γ_{ijl}'s; $\gamma_{(i,j)(p,q)}$ is the distance between (i,j) and (p, q); and Δ_{ij} is the area of the small quadrilateral Ω_{ij}.

At the boundary, the normal derivative is approximated by a different finite difference scheme which does not use the phantom zones. Assuming that $p = 1$ corresponds to the boundary (Fig. 2), the approximation is

$$\partial v_n/\partial n|_\Gamma \sim \frac{4v_{n2,q} - 3v_{n1,q} - v_{n3,q}}{\gamma_{(1,q)(2,q)} + \gamma_{(2,q)(3,q)}} \cos\theta + \frac{v_{n1,q+1} - v_{n1,q-1}}{\gamma_{(1,q)(1,q+1)} + \gamma_{(1,q-1)(1,q)}} \sin\theta. \tag{14}$$

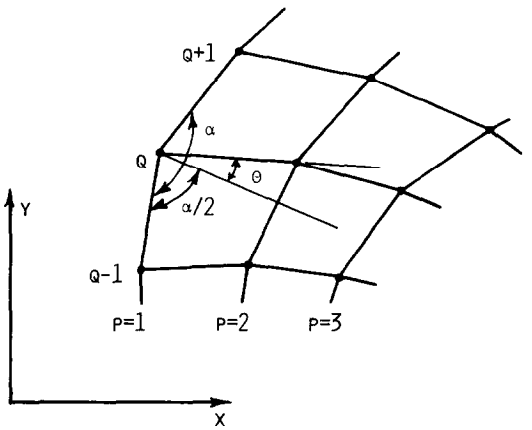


FIG. 2. Computational grid for the finite difference approximation of the normal derivative at a boundary point $(1, q)$.

The Green's function of (7) satisfies (10) also except that the right-hand side contains an additional term

$$\begin{aligned} \iint_{\Omega_{ij}} \delta\{\gamma_{(i,j)(p,q)}\}\, dx\, dy &= 1, \qquad \text{for} \quad (i,j) = (p,q), \\ &= 0, \qquad \text{for} \quad (i,j) \neq (p,q). \end{aligned} \tag{15}$$

Hence the discretized $v_n(x, y, s)$ and $G_n(x, y, x', y', s)$ satisfy the same linear algebraic system except with a different right-hand side. For computational efficiency, one can solve for $v_n(x, y, s_\sigma)$ and $G_n(\tilde{\Gamma}, x', y', s_\sigma)$ simultaneously by solving the following linear system,

$$\mathsf{A}_n(k_n, s_\sigma) \cdot \{\mathbf{V}_n(s_\sigma), \mathbf{G}_n(\tilde{\Gamma}, s_\sigma)\} = \{\mathbf{B}(s_\sigma), \mathbf{C}(\tilde{\Gamma}, s_\sigma)\}, \tag{16}$$

where $\mathsf{A}_n(k_n, s_\sigma)$ is the known pentadiagonal matrix from discretizing elliptic partial differential equation (6), $\mathbf{V}_n(s_\sigma)$ is the vector with all $v_{ni,j}(s_\sigma)$'s as its components, $\mathbf{G}_n(\tilde{\Gamma}, s_\sigma)$ is the vector with all $G_n(\tilde{\Gamma}, x_i', y_j', s_\sigma)$'s as its components, the known vector $\mathbf{B}(s_\sigma)$ comes from the boundary condition of (7), and the known vector $\mathbf{C}(\tilde{\Gamma}, s_\sigma)$ comes from the boundary condition of (7) and the location of the Dirac delta function.

The Fredholm integral equation of the first kind (9) can be discretized by simply using the rectangle rule and the derivatives in the integrand are approximated by the following finite difference approximation,

$$\begin{aligned} \frac{\partial \delta k_n}{\partial x'} &\sim \frac{1}{\sin(\phi_{ij} - \psi_{ij})} \left\{ \frac{\delta k_{ni+1,j} - \delta k_{ni-1,j}}{\gamma_{(i,j)(i+1,j)} + \gamma_{(i,j)(i-1,j)}} \sin \phi_{ij} \right. \\ &\quad \left. - \frac{\delta k_{ni,j+1} - \delta k_{ni,j-1}}{\gamma_{(i,j)(i,j+1)} + \gamma_{(i,j)(i,j-1)}} \sin \psi_{ij} \right\}, \\ \frac{\partial \delta k_n}{\partial y'} &\sim \frac{1}{\sin(\phi_{ij} - \psi_{ij})} \left\{ \frac{-\delta k_{ni+1,j} + \delta k_{ni-1,j}}{\gamma_{(i,j)(i+1,j)} + \gamma_{(i,j)(i-1,j)}} \cos \phi_{ij} \right. \\ &\quad \left. - \frac{\delta k_{ni,j+1} - \delta k_{ni,j-1}}{\gamma_{(i,j)(i,j+1)} + \gamma_{(i,j)(i,j-1)}} \cos \psi_{ij} \right\}, \end{aligned} \tag{17}$$

where ϕ_{ij} and ψ_{ij} are the average angles of the local computational grid coordinates with respect to the x-axis (Fig. 3).

In a similar manner, the partial derivatives $\partial v_n/\partial x'$, $\partial v_n/\partial y'$, $\partial^2 v_n/\partial x'^2$, and $\partial^2 v_n/\partial y'^2$ are approximated by the same central difference scheme. Hence integral equation (9) is reduced to a linear algebraic system,

$$\mathsf{M}_n(v_n, G_n, s_\sigma) \cdot \delta\mathbf{K}_n(x, y) = \mathbf{S}_n(s_\sigma), \tag{18}$$

where the known matrix $\mathsf{M}_n(v_n, G_n, s_\sigma)$ comes from the discretization of the integral and it is an ill-conditioned full matrix with each row containing a different s_σ, the

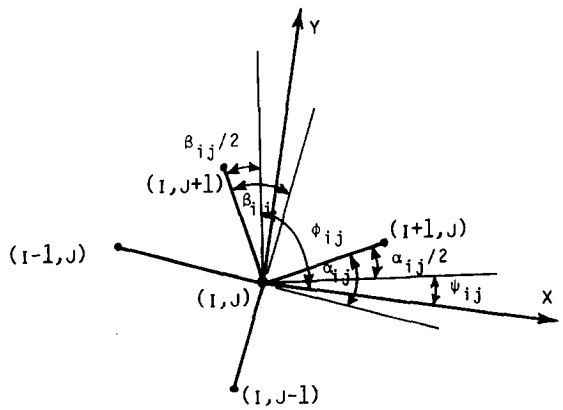
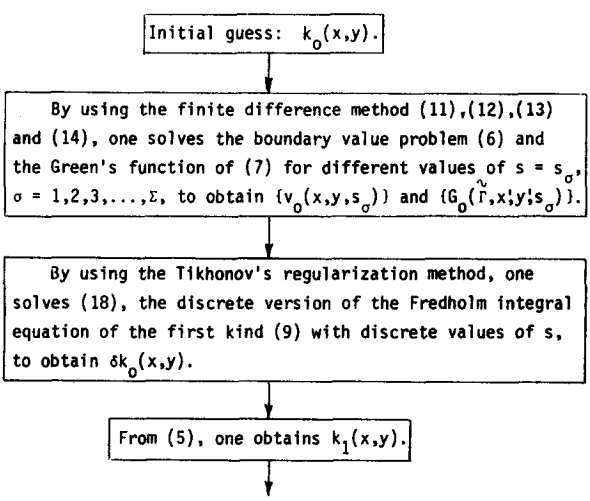


FIG. 3. Definitions of ϕ_{ij} and ψ_{ij} in the finite difference approximation of derivatives in the integrand of the integral equation at (i,j).

unknown vector $\delta\mathbf{K}_n(x, y)$ consists of all $\delta k_{ni,j}$ as its components, and the known vector $\mathbf{S}_n(s_\sigma)$ comes from the discretization of the right-hand side of (9) with components containing the corresponding complex frequency parameters s_σ, $\sigma = 1, 2, 3, \ldots, \Sigma$.

Since A_n is a symmetric, positive definite, narrow banded, and well-conditioned matrix, (16) can be solved by any modern efficient sparse matrix technique. However, M_n is either a rectangular matrix or an ill-conditioned square matrix; therefore Tikhonov's regularization method with second-order stabilizers [9] is used to solve (18). For (9′) the functional to be minimized is



SCHEME 1

$$\int_0^\infty \left\{ \iint_\Omega K_n(\tilde{\Gamma}, s, x', y')\, \delta k_n\, dx'\, dy' - g_n(\tilde{\Gamma}, s) \right\}^2 ds$$
$$+ \gamma \iint_\Omega \sum_{i=0}^{2} q_i(x', y') \sum_{j=0}^{i} \frac{\partial^j \delta k}{\partial x^\alpha \partial y^\beta}\, dx'\, dy' \tag{19}$$

with $q_0(x, y) \geqslant 0$, $q_1(x, y) \geqslant 0$, and $q_2(x, y) > 0$ continuous.

The essence of the first cycle of iteration is given in Scheme 1 and the procedure for other cycles is exactly the same. It is important to notice that each cycle of iteration consists basically of first solving direct boundary value problem (6) and the Green's function of (7) Σ times and then solving the Fredholm integral equation of the first kind (9) once.

Numerical Simulation

In order to test the feasibility and to study the general characteristics of the PST computational algorithm for solving two-dimensional inverse problems of the linear

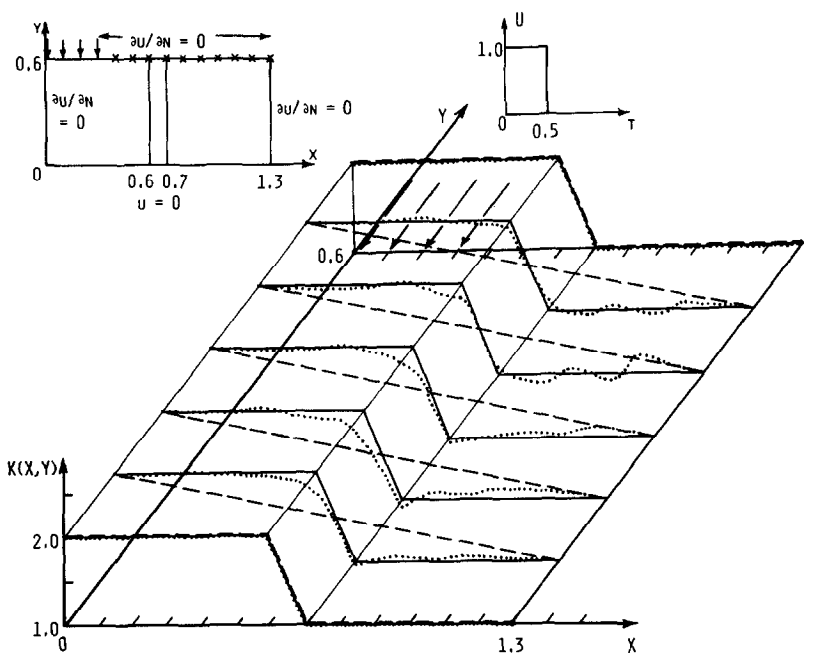


FIG. 4. The geometry of Ω and the boundary conditions are shown in the figure of the upper left-hand corner. A rectangular pulse of unit strength and duration 0.5 is applied where indicated by the arrows and $h(x, y, t)$ is measured at x's; $\Delta x = \Delta y = 0.1$. Comparison of the calculated $k_3(x, y)$ ($\cdots$) and the exact $k^*(x, y)$ (—) with the initial guess $k_0(x, y)$ (---) is also shown.

wave equation without real measurement data, the following numerical simulation procedure is carried out:

First, one chooses a $k^*(x, y)$ which is supposed to represent the correct unknown coefficient $k(x, y)$ and one also chooses the boundary functions $f_\lambda(x, y, t)$, $\lambda = 1, 2, ..., \Lambda$, which are supposed to represent a part of the measured data. Their Laplace transforms $F_\lambda(x, y, s)$ are numerically computed for a chosen discrete set of s, $\{s_\sigma\}$, $\sigma = 1, 2, 3, ..., \Sigma$. Then boundary value problem (3) is solved by using the finite difference method (11)–(14); thus one generates the rest of the supposedly measured auxiliary data $H(x, y, s_\sigma)|_{\tilde{\Gamma}}$, $\sigma = 1, 2, 3, ..., \Sigma$. Next, $k_0(x, y)$ is assumed. Hence upon solving (5), (6), and (9) numerically, $k_1(x, y)$ is obtained. Then in a similar manner $k_2(x, y)$ can be obtained. One continues this procedure until finally a numerical limit $k_N(x, y)$ is reached. Other than the truncation, round-off, numerical integration, and finite difference approximation errors in both generating the numerical data and computing $k_N(x, y)$, any norm of $k^*(x, y) - k_N(x, y)$ can be used as a criterion for evaluating the performance of the computational algorithm of PST.

The numerical simulation here is carried out for a class of $k^*(x, y)$ and $k_0(x, y)$. To demonstrate the capability of the computational algorithm of PST to handle the general geometry of the inverse problem, rectangles, quadrilaterals, and triangles are used for the domain Ω and various functions with different geometry are used for $k^*(x, y)$. For reasons of economy, very coarse computational grids are used in our computation. For simplicity, homogeneous boundary conditions are used everywhere except that a square pulse is applied to a small portion of the boundary Γ and $s = \sigma$.

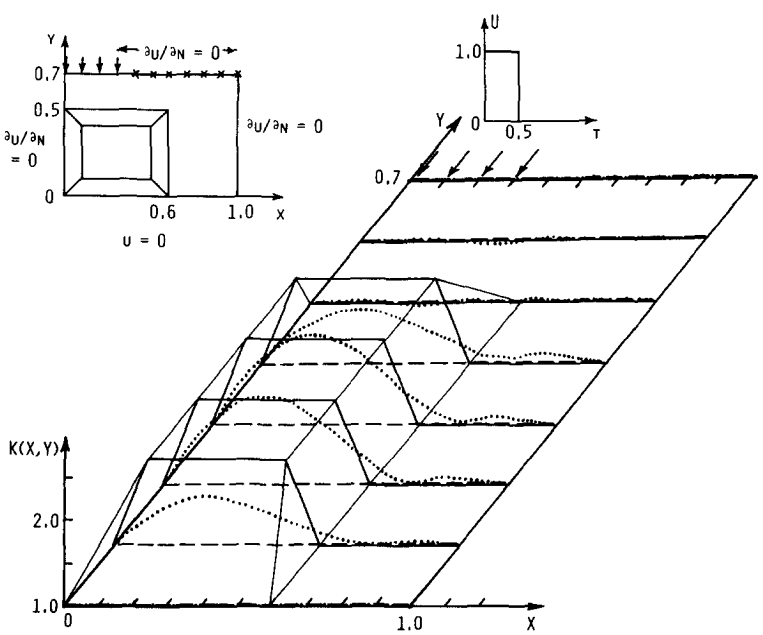


FIG. 5. Same description as that of Fig. 4.

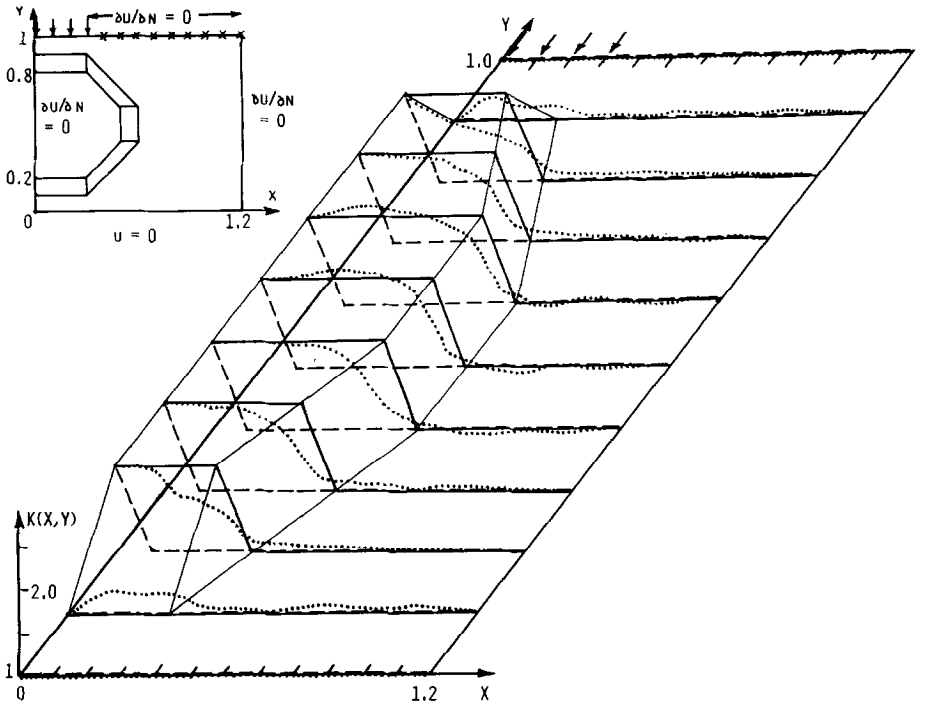


FIG. 6. Same description as that of Fig. 4 except the triangular zones.

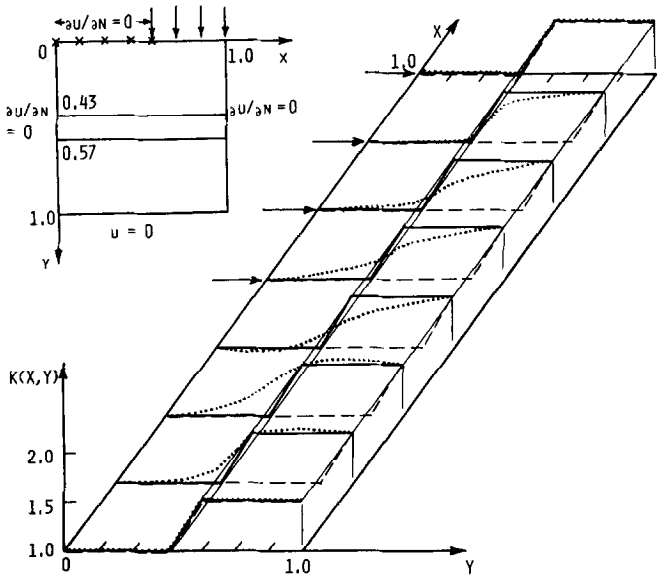


FIG. 7. The geometry of Ω and the boundary conditions are shown in the figure of the upper left-hand corner. The same rectangular pulse is applied where indicated by the arrows and $h(x, y, t)$ is measured at x's, $\Delta x = \Delta y = 0.143$. Comparison of the calculated $k_5(x, y)$ (···) and the exact $k^*(x, y)$ (—) with the initial guess $k_0(x, y)$ (---) is also shown.

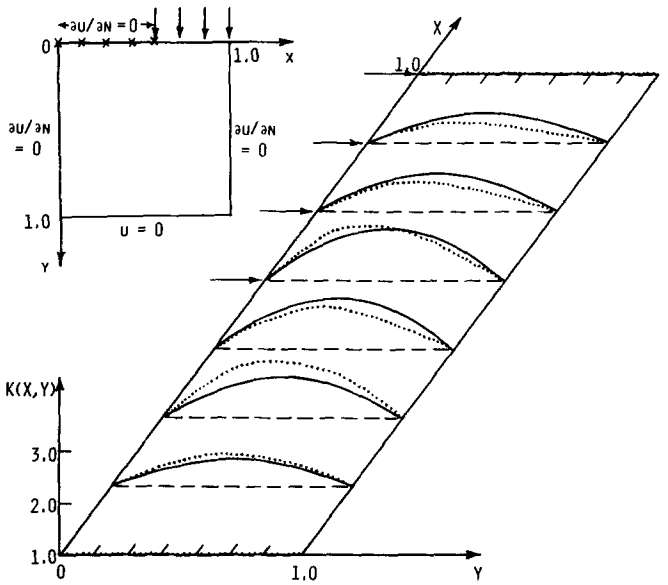


FIG. 8. Same description as that of Fig. 7.

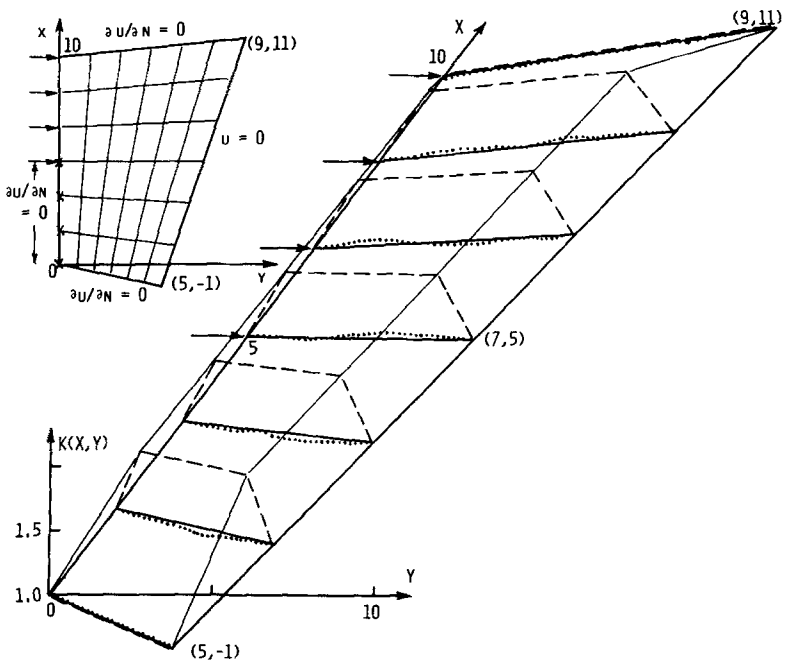


FIG. 9. The geometry of Ω, the boundary conditions, and the computational grid are shown in the figure of the upper left hand corner. The same rectangular pulse is applied as indicated by the arrows and $h(x, y, t)$ is measured at x's. Comparison of the calculated $k_5(x, y)$ (···) and the exact $k^*(x, y)$ (—) with the initial guess $k_0(x, y)$ (---) is also shown.

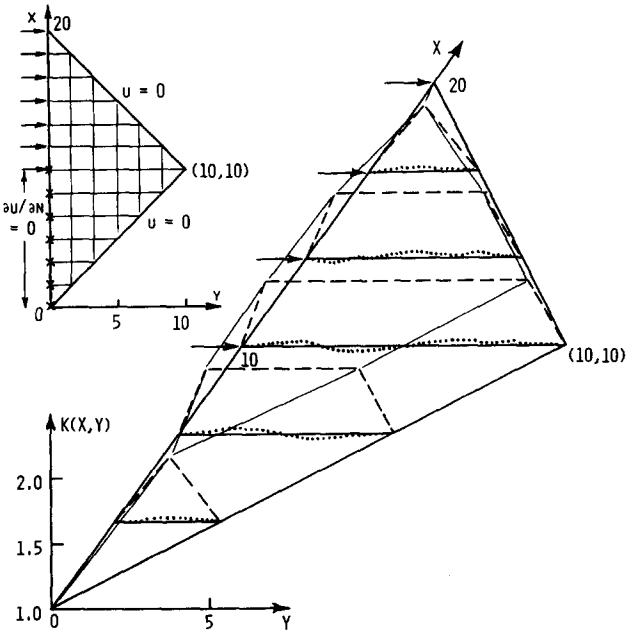


FIG. 10. Same description as that of Fig. 9.

$\sigma = 1, 2, 3, \ldots, 15$, are chosen in our computation. The numerical results are plotted in Figs. 4–10. The maximum norms of $k^*(x, y) - k_N(x, y)$ and $k^*(x, y) - k_0(x, y)$ for all cases can be estimated from the graphs in these figures. The L_2 norms, $I_n = \|k^*(x, y) - k_n(x, y)\|_2$, $n = 0, N$, for all cases are tabulated in Table I.

Here the numerical results in Figs. 4–10 are not the best available ones because the iteration procedure will stop as soon as $\Delta_{n+1} \equiv \|k_{n+1} - k_n\|_2 \leqslant 0.05$ (0.05 or smaller in Δ_n will make very little difference in plotting the numerical results). Then $k_N(x, y) \equiv k_{n+1}(x, y)$. From our past and present experience, all the I_n's in our numerical simulation for one-parameter inverse problems are monotonic decreasing functions of n when n is in the practical range, say $0 \leqslant n < 15$ or more. For reasons of economy, the I_n's are computed only for two simpler and smaller examples and their results are plotted in Fig. 11.

TABLE I

Fig. #:	4	5	6	7	8	9	10
N:	3	3	3	5	6	5	5
I_0:	2.32	4.47	5.40	2.12	3.50	2.50	2.50
I_N:	0.58	2.03	2.32	0.75	1.47	0.28	0.30

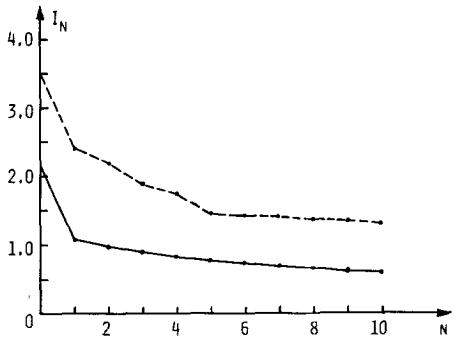


FIG. 11. The I_N's as functions of N, $N = 1, 2, \ldots, 10$, shown for the example in Fig. 7 (—) and the example in Fig. 8 (---).

DISCUSSION

Although only a small number of computational zones (both in Ω and s) are used in the numerical simulation here, the numerical results in Figs. 4–10 have demonstrated that the PST iterative numerical algorithm does give good results in solving two-dimensional inverse problems of the linear wave equation and it is as robust as for the one-dimensional case [1, 2, 3]. The accuracy of the numerical algorithm can be improved greatly if a larger number of computational zones in Ω are used; more efforts are made in computing each individual step and in discretization of the partial differential equations and the integral equations in the numerical algorithm; and a larger number of s_σ's are used and their values are properly chosen in solving the Fredholm integral equation of the first kind. Of course, it is counterproductive for one to increase overwhelmingly the number of computational zones in Ω, for it is well known that this will make the discretized version of the Fredholm integral equation of the first kind (18) more ill conditioned.

Mathematically, the inverse problem is a nonlinear problem regardless of whether the original partial differential equation is linear or not. Hence in general the solution of an inverse problem with a minimum number of constraints is not unique. To be sure, the PST iterative numerical algorithm is not a method for settling the question of the uniqueness of the solution of an inverse problem. This approximate solution is unique in the sense of being the closest one to the initial guess in the L_2 norm. Moreover, it is clear from our numerical simulation that for different initial guesses $k_0(x, y)$, the iterations converge to slightly different $k_N(x, y)$'s. However, if this numerical algorithm is reasonably robust, then any one of the approximate solutions will be an acceptable approximation. This numerical computation phenomenon can be attributed to the accumulation of those nonnegligible errors in computing each iterate. Here, most of these computational errors come from the regularization procedure in solving the ill-posed Fredholm integral equation of the first kind.

The PST iterative numerical algorithm can be extended to solve three-dimensional

inverse problems in a straightforward manner because the finite difference method or the finite element method is just as adaptable to solve any three-dimensional boundary value problem with arbitrary finite domain as to solve the two-dimensional boundary value problem, and Tikhonov's regularization method is also as adaptable to solve the three-dimensional Fredholm integral equation of the first kind as to solve the two-dimensional case. Moreover, since the synthesis procedure is carried out in the frequency domain where the governing equation is an elliptic partial differential equation coming either from the Laplace transformed hyperbolic equation or the Laplace transformed parabolic equation, the PST iterative numerical algorithm can be used to solve inverse problems of both hyperbolic and parabolic partial differential equations with trivial changes in the computer code of PST. Hence PST fares very well in regard to the universality criterion.

As it has been demonstrated in the previous sections, the measurement data are needed only at a portion of the boundary to solve two-dimensional inverse problems of the linear wave equation successfully. Hence the PST fares very well in regard to the economy of data acquisition criterion in comparison with other possible methods where measurement data are needed in the whole interior.

The programming for the PST is basically nondedicated because the changeover from solving the inverse problems of a class of hyperbolic partial differential equations to solving the inverse problems of a class of parabolic partial differential equations is simply a matter of changing a few coefficients (changing a few cards) in the elliptic equation solver. Moreover, one does not have to program a subroutine for the elliptic equation solver, for there is an abundance of finite difference and finite element computer codes for solving two- or three-dimensional elliptic partial differential equations available in the public domain. Hence the PST again fares very well in regard to the economy of programming effort criterion.

Finally, it seems also to fare rather well in regard to the economy of computing cost criterion. However, the actual computing costs depend very much on the particular computer hardwares and softwares, and one cannot be sure of this until a benchmark comparison test is performed.

Efforts in carrying out the generalization of the PST iterative numerical algorithm to solve two- and three-dimensional inverse problems of a system of coupled linear wave equations are under way and their results will be reported in the near future. Furthermore, the PST iterative numerical algorithm also can be generalized to determine several unknown coefficients of a system of partial differential equations simultaneously, and similar efforts have also been started.

REFERENCES

1. D. S. TSIEN AND Y. M. CHEN, *in* "Computaional Methods in Nonlinear Mechanics; Proc. Int. Conf. Comput. Meth. Nonlinear Mechs., The University of Texas, Austin," pp. 935–943. 1974.
2. Y. M. CHEN AND D. S. TSIEN, *J. Comput. Phys.* **25** (1977), 366.

3. D. S. TSIEN AND Y. M. CHEN, *Radio Sci.* **13** (1978), 775.
4. R. P. HATCHER AND Y. M. CHEN, "An iterative method for solving inverse problems of a nonlinear wave equation," *SIAM J. Sci. Stat. Comput.*, to appear.
5. Y. M. CHEN AND X. T. WENG, *Circuit Theory Appl.* **10** (1982), 69.
6. Y. M. CHEN, *in* "Proc. Int. Symp. Ill-Posed Problems: Theory and Practice," University of Delaware, Newark, October 1979.
7. Y. M. CHEN AND J. Q. LIU, *J. Comput. Phys.* **43** (1981), 315.
8. J. Q. LIU AND Y. M. CHEN, An iterative algorithm for solving inverse problems of two-dimensional diffusion equations, *SIAM J. Sci. Stat. Comput.*, to appear.
9. A. N. TIKHONOV AND V. Y. ARSENIN, "Solutions of Ill-Posed Problems," Wiley, *New York*, 1977.