JOURNAL OF COMPUTATIONAL PHYSICS 72, 372–382 (1987)

# Efficiency Improvement of GPST Inversion Algorithm

Y. M. CHEN AND M. S. LIU

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

Received March 21, 1986; revised October 20, 1986

To demonstrate the possibility of improving the efficiency of the generalized pulse-spectrum technique (GPST) inversion algorithm by implementing special re-structuralization and high level parallelism into the system of discretized Fredholm integral equations of the first kind, a simple two-parameter inverse problem of two-dimensional linear evolution equation is considered. Numerical simulations are carried out to test the feasibility and to study the general characteristics of the improved GPST without real measurement data. It is found that the improved GPST is not only as robust as the standard GPST but also, possessing the speedup, very close to the estimated one by performing the computational complexity analysis based upon FLO count. © 1987 Academic Press, Inc.

## INTRODUCTION

The so-called "generalized pulse-spectrum technique" (GPST) inversion algorithm, a versatile iterative numerical algorithm for solving multi-parameter inverse problems of a system of partial differential equations, has been thoroughly tested on a large class of small- to medium-sized inverse problems in numerous applications since 1974 [1–20] and GPST is found to be quite efficient and stable (robust). Moreover, it has been pointed out [21, 22] that GPST is basically a combination of a Newton-like method and te Tikhonov regularization method when the inverse problems are formulated as functional minimizing problems in some function spaces, and the convergence proofs of GPST for special cases can be found also in [21, 22]. Recently, many researchers in solving inverse problems have also treated the inverse problems as problems in functional minimizing by using different methods in nonlinear optimization, e.g., the Dividon–Flectcher–Powell method, the steepest descent method, and other Newton-like methods. They have achieved the same accuracy and stability but not the same efficiency. Theoretically, the efficiency of a numerical algorithm depends on the intrinsic mathematical structure of the algorithm, but in practice it depends even more on the skill of software implementation of the algorithm which is very much computer-architecture- and compiler-dependent. As a possible example of this, the AT & T Bell Laboratories'



0021-9991/87 $3.00

implementation of the Karmarkar algorithm for solving problems in linear programming at the present seems to achieve higher efficiency than all of the others' implementations.

As it stands now, the efficiency of the existing GPST will not lend itself to be useful for solving large-sized inverse problems. Hence there is an urgent need of improving the efficiency of the existing GPST inversion algorithm. Here we choose to achieve this purpose by preserving the basic mathematical structure of GPST and improving its implementation. The major "bottleneck" in the existing GPST lies in the numerical effort for solving the system of discretized Fredholm integral equations of the first kind, a $Z \times KM$ full matrix equation, by using the Tikhonov regularization method. Here $K$ is the number of unknown parameters, $M$ is the number of grid points, and $Z$ is the number of data (measurement) points in both space and frequency (spectrum) domains or the number of data points in both space and time (pulse) domains. Hence the FLO (floating point arithmetic operation) count for solving this matrix equation by using the Tikhonov regularization method is of $O(ZK^2M^2) + O(K^3M^3)$, which is extremely large for the large-sized inverse problems.

In this paper, the above-mentioned bottleneck is eliminated by introducing two modifications in the structure of GPST. One of them is based upon the re-structuralization of the system of discretized Fredholm integral equations of the first kind so that only a special narrow-banded sparse matrix equation needs to be solved by using the Tikhonov regularization method. Although the general philosophy of re-structuralization is problem-independent, its details are not only problem-dependent but also data-point-dependent. The other modification is based the introduction of parallelism into the system of discretized Fredholm integral equations of the first kind so that the original large matrix equation is replaced by a system of smaller decoupled matrix equations which can be solved either simultaneously on individual processors of a multi-processor computer or sequentially on a single-processor computer.

For the purposes of simplicity and comparison, an outline of the standard GPST for solving simple two-parameter inverse problems of two-dimensional linear evolution (wave and diffusion) equations is given in the next section. Then the newly improved version of GPST is presented. Next, numerical simulations are carried out on a single-processor computer UNIVAC 1100/81 to test the feasibility and to study the intrinsic characteristics of this new version of GPST with computer-generated data. Finally, discussion of the numerical results and estimation of the possible speedup are given in the last section.

## Generalized Pulse-Spectrum Technique (GPST)

For reasons of simplicity and pure economics, we consider the following simple test inverse problems with the special form of unknown coefficient $a(x)\,b(y)$.

$$\partial\{a(x)\,b(y)\,\partial u/\partial x\}/\partial x+\partial\{a(x)\,b(y)\,\partial u/\partial y\}/\partial y-\left\{\begin{array}{c}\partial^2u/\partial t^2\\ \text{or}\\ \partial u/\partial t\end{array}\right\}=0,\quad \begin{array}{l}0<x,\,y<1,\\ 0<t<\infty,\end{array}$$

$$u(x,\,y,\,0)=\partial u(x,\,y,\,0)/\partial t=0 \qquad \text{or} \qquad u(x,\,y,\,0)=0, \qquad 0<x,\,y<1,$$

$$\partial u(0,\,y,\,t)/\partial x=\partial u(1,\,y,\,t)/\partial x=\partial u(x,\,0,\,t)/\partial y$$

$$=\partial u(x,\,1,\,t)/\partial y=0, \qquad 0<t<\infty, \tag{1}$$

with sources $u(x,\,y,\,t)=f(x,\,y,\,t)$, $(x,\,y)\in\Omega_1\subset 0<x,\,y<1$, $0<t<\infty$, and measurement $u(x,\,y,\,t)=h(x,\,y,\,t)$, $(x,\,y)\in\Omega_2\subset 0<x,\ y<1$, $\Omega_1\cap\Omega_2=0$. The re-structuralization will be quite different for other types of inverse problems.

The first step of GPST calls for the Laplace transformation of (1) so that the entire system is transformed from the space-time domain to the space-complex frequency domain. Hence the initial-boundary value problem of the hyperbolic or parabolic system is transformed into the following boundary value problem of an elliptic system,

$$\partial\{a(x)\,b(y)\,\partial U/\partial x\}/\partial x+\partial\{a(x)\,b(y)\,\partial U/\partial y\}/\partial y-\left\{\begin{array}{c}s^2\\ \text{or}\\ s\end{array}\right\}U=0, \qquad \begin{array}{l}0<x,\,y<1,\\ 0\leqslant s<\infty,\end{array}$$

$$\partial U(0,\,y,\,s)/\partial x=\partial U(1,\,y,\,s)/\partial x=\partial U(x,\,o,\,s)/\partial y$$

$$=\partial U(x,\,1,\,s)/\partial y=0, \qquad 0\leqslant s<\infty,$$

$$U(x,\,y,\,s)=F(x,\,y,\,s), \qquad (x,\,y)\in\Omega_1, \qquad 0\leqslant s<\infty,$$

and (2)

$$U(x,\,y,\,s)=H(x,\,y,\,s), \qquad (x,\,y)\in\Omega_2, \qquad 0\leqslant s<\infty,$$

where $U(x,\,y,\,s)$, $F(x,\,y,\,s)$, and $H(x,\,y,\,s)$ are the Laplace transforms of $u(x,\,y,\,t)$, $f(x,\,y,\,t)$, and $h(x,\,y,\,t)$, respectively. Now, the two-parameter inverse problem is to determine $a(x)$ and $b(y)$ from (2), $F(x,\,y,\,s)$, and $H(x,\,y,\,s)$.

Next, the iteration begins by setting

$$\begin{aligned} U_{n+1}&=U_n+\delta U_n, & a_{n+1}&=a_n+\delta a_n,\\ b_{n+1}&=b_n+\delta b_n, & n&=0,\,1,\,2,\,3,\ldots,\end{aligned} \tag{3}$$

where $a_0$ and $b_0$ are the initial guesses for the corresponding unknown coefficients, and the $\delta$-terms are smaller than their corresponding non-$\delta$-terms in some norms.

Upon substituting (3) into (2) and neglecting terms of order $\delta^2$ and higher, one obtains the same linear system for $U_n$ as that for $U$ except with the additional subscript "$n$,"

$$\partial\{a_n(x)\, b_n(y)\, \partial U_n/\partial x\}/\partial x + \partial\{a_n(x)\, b_n(y)\, \partial U_n/\partial y\}/\partial y - \begin{pmatrix} s^2 \\ \text{or} \\ s \end{pmatrix} U_n = 0,$$

$$0 < x, y < 1, \qquad 0 \leqslant s < \infty,$$

$$\partial U_n(0, y, s)/\partial x = \partial U_n(1, y, s)/\partial x = \partial U_n(x, 0, s)/\partial y$$
$$= \partial U_n(x, 1, s)/\partial y = 0, \qquad 0 \leqslant s < \infty, \tag{4}$$

$$U_n(x, y, s) = F(x, y, s), \qquad (x, y) \in \Omega_1, \qquad 0 \leqslant s < \infty,$$

and

$$U_n(x, y, s) = H(x, y, s), \qquad (x, y) \in \Omega_2, \qquad 0 \leqslant s < \infty,$$

and a similar linear system for $\delta U_n$,

$$\partial\{a_n(x)\, b_n(y)\, \partial \delta U_n/\partial x\}/\partial x + \partial\{a_n(x)\, b_n(y)\, \partial \delta U_n/\partial y\}/\partial y - \begin{pmatrix} s^2 \\ \text{or} \\ s \end{pmatrix} \delta U_n$$
$$= -\partial\{[a_n(x)\, \delta b_n(y) + \delta a_n(x)\, b_n(y)]\, \partial U_n/\partial x\}/\partial x$$
$$-\partial\{[a_n(x)\, \delta b_n(y) + \delta a_n(x)\, b_n(y)]\, \partial U_n/\partial y\}/\partial y,$$

$$0 < x, y < 1, \qquad 0 \leqslant s < \infty, \tag{5}$$

$$\partial \delta U_n(0, y, s)/\partial x = \partial \delta U_n(1, y, s)/\partial x = \partial \delta U_n(x, 0, s)/\partial y$$
$$= \partial \delta U_n(x, 1, s)/\partial y = 0, \qquad 0 \leqslant s < \infty,$$

$$\delta U_n(x, y, s) = 0, \qquad (x, y) \in \Omega_1, \qquad 0 \leqslant s < \infty,$$

and

$$\delta U_n(x, y, s) = 0, \qquad (x, y) \in \Omega_2, \qquad 0 \leqslant s < \infty.$$

By using the method of Green's function, setting $(x, y) \equiv \mathbf{x}$ at $\Omega_2$, and replacing $U_{n+1}(x, y, s)$ by $U(x, y, s)$ and then $H(x, y, s)$, one obtains from (5) a system of Fredholm integral equations of the first kind,

$$\int_0^1 \int_0^1 G_n(\mathbf{x}, \mathbf{x}', s)|_{\mathbf{x}_i \in \Omega_2} \{\partial[(a_n\, \delta b_n + \delta a_n b_n)\, \partial U_n/\partial x']/\partial x'$$
$$+ \partial[(a_n\, \delta b_n + \delta a_n b_n)\, \partial U_n/\partial y']\}\, dx'\, dy'$$
$$= \{U_n(\mathbf{x}, s) - H(\mathbf{x}, s)\}|_{\mathbf{x}_i \in \Omega_2}, \qquad i = 1, 2, 3, \ldots, I, \tag{6}$$

where $G_n(\mathbf{x}, \mathbf{x}', s)$ is the Green's function of (5).

Equations (3), (4), and (6) form the basic structure of each iteration of the standard GPST inversion algorithm. By using the simple center finite difference scheme, one can solve for $U_n$ and $G_n$ for a fixed set of $\{s_j\}$, $j = 1, 2, 3, \ldots, J$. Similarly,

Eq. (6) can be discretized simply by using the trapezoidal rule on the same spatial computation-grid and for the same set of $\{s_j\}$ to become an ill-conditioned linear algebraic system,

$$\mathbf{R}_n \cdot \begin{pmatrix} \delta \mathbf{a}_n \\ \delta \mathbf{b}_n \end{pmatrix} = \mathbf{F}_n, \tag{7}$$

which is solved by using the Tikhonov regularization method such as one solves

$$(\mathbf{R}_n^T \cdot \mathbf{R}_n + \alpha \mathbf{I}) \cdot \begin{pmatrix} \delta \mathbf{a}_n \\ \delta \mathbf{b}_n \end{pmatrix} = \mathbf{R}_n^T \cdot \mathbf{F}_n, \tag{8}$$

where $\mathbf{R}_n$ is a $IJ \times 2M^{1/2}$ rectangular matrix with $M$ being the number of spatial grid points.

In essence, each cycle of iteration consists basically of first solving (4) (symmetric matrix with half bandwith $M^{1/2}$) $J$ times and then solving (8) once. By using the direct matrix solvers, e.g., Gaussian elimination and LU decomposition, the asymptotic FLO count for solving (4) $J$ times is of $O(JM^2)$/iteration, for forming (8) it is $4IJM$/iteration, and for solving (8) it is $(8/3)\, M^{3/2}$/iteration. Hence the total asymptotic FLO count per iteration of the standard GPST inversion algorithm is $O\{M(\frac{8}{3}M^{1/2} + 4IJ + JM)\}$ which is definitely too large for large $M$. In general, the unknown coefficient $k(x, y) \neq a(x)\, b(y)$; then the corresponding total asymptotic FLO count per iteration is $O\{M^2(\frac{1}{3}M + IJ + J)\}$.

## Efficiency Improvement of GPST

To improve the efficiency of the existing GPST, we elect to achieve this goal by preserving the basic mathematical structure of GPST and improving its implementation. From the asymptotic FLO count of the existing GPST given in the previous section, it is clear that in general the "bottleneck" lies in the formation and Gaussian elimination of (8). Hence our attention is focused on the elimination of this bottleneck by introducing two modifications into the detailed structure of the existing GPST.

The first step is to restructure the system of discretized Fredholm integral equations of the first kind so that the new matrix equation (7) is either smaller in size or possessing some preferred special structures. However, it is difficult to achieve this by examining the system of discretized Fredholm integral equations of the first kind as they are. A better way is to examine the discretized partial differential equation (5) from which the system of Fredholm integral equations of the first kind are derived. Here the discretized equation (5) is

$$\begin{aligned} &\delta a_{p+1} b_q (U^J_{p+1,q} - U^J_{p-1,q}) \\ &\quad + \delta a_p \{4 b_q (U^J_{p+1,q} + U^J_{p-1,q} + U^J_{p,q+1} + U^j_{p,q-1} - 4U^J_{p,q}) \\ &\quad + (b_{q+1} - b_{q-1})(U^J_{p,q+1} - U^j_{p,q-1})\} - \delta a_{p-1} b_q (U^j_{p+1,q} - U^J_{p-1,q}) \end{aligned}$$

$$
\begin{aligned}
&+\delta b_{q+1}a_p(U^j_{p,q+1}-U^j_{p,q-1})-\delta b_{q-1}a_p(U^j_{p,q+1}-U^j_{p,q-1})\\
&+\delta b_q\{4a_p(U^j_{p+1,q}+U^j_{p-1,q}+U^j_{p,q+1}+U^j_{p,q-1}-4U^j_{p,q})\\
&+(a_{p+1}-a_{p-1})(U^j_{p+1,q}-U^j_{p-1,q})\}\\
&=\{4(\Delta x)^2\begin{pmatrix}(s^j)^2\\ \text{or}\\ s^j\end{pmatrix}+16a_pb_q\}\,\delta U^j_{p,q}\\
&+a_p(b_{q+1}-4b_q-b_{q-1})\,\delta U^j_{p,q-1}-a_p(b_{q+1}+4b_q-b_{q-1})\,\delta U^j_{p,q+1}\\
&+b_q(a_{p+1}-4a_p-a_{p-1})\,\delta U^j_{p-1,q}-b_q(a_{p+1}+4a_p-a_{p-1})\,\delta U^j_{p+1,q},\\
&\qquad\qquad p,q=2,3,4,\ldots,P-1,\ j=1,2,3,\ldots,J, \qquad (9)
\end{aligned}
$$

where $P-1=1/\Delta x=1/\Delta y$, $U^j_{p,q}\equiv U\{(p-1)\,\Delta x,\ (q-1)\,\Delta y,\ s_j\}$, and the subscript "$n$" is omitted for convenience. It is clear that (9) is a linear relationship between $(\delta\mathbf{a}_n, \delta\mathbf{b}_n)$ and $\delta U^j_{n,p,q}$.

The standard GPST calls for setting $\delta U^j_{n,p,q}=H^j_{p,q}-U^j_{n,p,q}$ at the grid points in (9) only. Now the modification in the GPST algorithm calls for further approximations by setting $\delta U^j_{n,p,q}\equiv 0$ elsewhere for their smallness when the initial guess is close enough to the exact solution and neglecting equations of (9) corresponding to non-data grid points for their roles being less important in solving the inverse problem. We hope that the additional errors caused by these additional approximations will be small and automatically taken care of by the iterative process of GPST in the form of slower convergence. With the above procedure, one can pack (9) into a very compact sparse matrix equation. However, the compact form depends very much on the number and the location of the data measurement points. For the purpose of demonstration, the data points are assumed to be $(p,q)=(2,2)$, $(3,2)$, $(4,2)$,..., $(P-1,2)$, $(P-1,3)$, $(P-1,4)$,..., $(P-1,P-1)$; this choice of data points $(p,q)$ is not part of our algorithm. Then Eq. (9) has the compact form

$$\mathbf{E}_n\cdot\begin{pmatrix}\delta\mathbf{a}_n\\ \delta\mathbf{b}_n\end{pmatrix}=\mathbf{L}_n, \qquad (10)$$

where $\mathbf{E}_n$ is a $J(2P-5)\times 2P$ sparse matrix of the following structure,

$$\mathbf{E}_n=\begin{Bmatrix}\mathbf{A}_n^{j=1} & \mathbf{B}_n^{j=1}\\ \mathbf{C}_n^{j=1} & \mathbf{D}_n^{j=1}\\ \\ \mathbf{A}_n^{j=2} & \mathbf{B}_n^{j=2}\\ \mathbf{C}_n^{j=2} & \mathbf{D}_n^{j=2}\\ \vdots & \vdots\\ \\ \mathbf{A}_n^{j=J} & \mathbf{B}_n^{j=J}\\ \mathbf{C}_n^{j=J} & \mathbf{D}_n^{j=J}\end{Bmatrix}, \qquad (11)$$

with $\mathbf{A}_n^j$ being $(P-2)\times P$ upper tri-diagonal matrices, $\mathbf{B}_n^j$ being $(P-2)\times P$ tri-column matrices with first three columns nonzero, $\mathbf{C}_n^j$ being $(P-3)\times P$ tri-column matrices with last three columns nonzero, $\mathbf{D}_n^j$ being $(P-3)\times P$ upper tri-diagonal matrices, and $\mathbf{L}_n$ is a $J(2P-5)$-dimensional full vector.

Per iteration, the asymptotic FLO count for forming the regularized normal equation (10) is $24JM^{1/2}$ which is much smaller compared to $8JM^{3/2}$ ($I\sim 2P=2M^{1/2}$) for the standard GPST. However, the asymptotic FLO count for solving the regularized normal equation is still the same.

Next, a simple parallelism is introduced into the system (10) to further improve the efficiency. This can be achieved by splitting (10) into two uncoupled equations,

$$\begin{aligned}\mathbf{E}_{1n}\cdot\delta\mathbf{a}_n&=\mathbf{L}_n,\\ \mathbf{E}_{2n}\cdot\delta\mathbf{b}_n&=\mathbf{L}_n,\end{aligned}\tag{12}$$

where

$$\mathbf{E}_{1n}=(\mathbf{A}_n^1,\mathbf{C}_n^1,\mathbf{A}_n^2,\mathbf{C}_n^2,\ldots,\mathbf{A}_n^J,\mathbf{C}_n^J)^{\mathrm{T}}$$

and

$$\mathbf{E}_{2n}=(\mathbf{B}_n^1,\mathbf{D}_n^1,\mathbf{B}_n^2,\mathbf{D}_n^2,\ldots,\mathbf{B}_n^J,\mathbf{D}_n^J)^{\mathrm{T}}.\tag{13}$$

This splitting can be achieved for any given set of $(p, q)$; different choices of $(p, q)$ will lead to different sparse structures of $\mathbf{E}_{1n}$ and $\mathbf{E}_{2n}$.

Then the regularized normal equation of (12) is

$$\begin{aligned}(\mathbf{E}_{1n}^{\mathrm{T}}\cdot\mathbf{E}_{1n}+\alpha_1\mathbf{I})\cdot\delta\mathbf{a}_n&=\mathbf{E}_{1n}^{\mathrm{T}}\cdot\mathbf{L}_n,\\ (\mathbf{E}_{2n}^{\mathrm{T}}\cdot\mathbf{E}_{2n}+\alpha_2\mathbf{I})\cdot\delta\mathbf{b}_n&=\mathbf{E}_{2n}^{\mathrm{T}}\cdot\mathbf{L}_n.\end{aligned}\tag{14}$$

It is clear that $\delta\mathbf{a}_n$ and $\delta\mathbf{b}_n$ can be solved either simulataneously on individual processors of a two-processor computer or sequentially on a single-processor computer.

In this case, the asymptotic FLO count for forming (14) is still $24JM^{1/2}$/iteration. But the asymptotic FLO counts for solving (14) simulataneously or sequentially are $\frac{1}{3}M^{3/2}$/iteration or $\frac{2}{3}M^{3/2}$/iteration, respectively, which are improvements over that of the standard GPST by a factor of eight or four, respectively. However, the number of iterations for numerical convergence probably will increase. Finally, the speedup (efficiency improvement) of the improved GPST over the standard GPST on a two-processor computer can be estimated by

$$S_2\sim\frac{\frac{8}{3}M^{3/2}+8JM^{3/2}+JM^2}{\frac{1}{3}M^{3/2}+24JM^{1/2}+JM^2}.\tag{15}$$

For this special case, $k(x, y)=a(x)\,b(y)$, the speedup is diminutive due to the fact that the bottleneck lies in the effort of solving the initial-boundary value problem (4) and there has been no improvement made for this. On the other hand, the

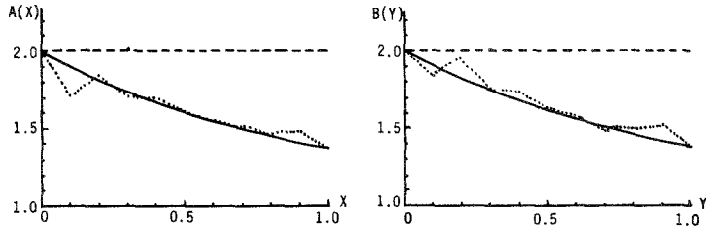


FIG. 1. Comparison of the calculated $a_N(x)$ and $b_N(y)$ $\cdots$ and the exact $a^*(x)$ and $b^*(y)$ ———, with the initial guesses $a_0(x)$ and $b_0(y)$ – – –

speedup of the improved GPST over the standard GPST in the backward calculation, solving for $\delta\mathbf{a}_n(x)$ and $\delta\mathbf{b}_n(y)$, can be estimated by

$$S_{2B} \sim (8 + 24J)/(1 + 72JM^{-1}), \tag{16}$$

i.e., $\text{Min}(M/3, 24J) \lesssim S_{2B} \lesssim \text{Max}(M/3, 24J)$, which is very good.

In general, the computational effort for forming and solving the regularized equation with full matrix in the backward calculation is much larger than solving the band sparse matrix several times in the forward calculation. The re-structuralization, which eliminates a few less-important equations corresponding to certain non-data grid points from (9) and sets $\delta U^J_{n,p,q} = 0$ at these grid points, will make the matrix in the regularized equation not only smaller but also sparse. Hence the major contribution to "$S$" will come from the backward calculation which may be substantially large.

## NUMERICAL SIMULATION

In order to test the feasibility and to study the general characteristics of the improved GPST computational algorithm for solving the special two-parameter inverse problems of the two-dimensional linear evolutional equations without real measurement data, the following numerical simulation procedure is carried out. A

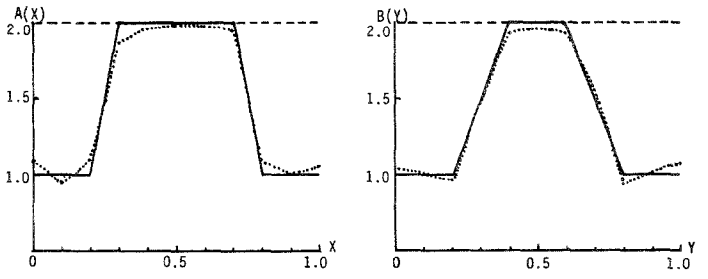


FIG. 2. Same as Fig. 1.

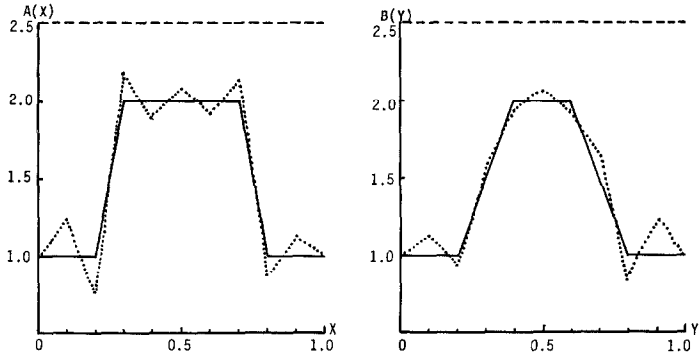


FIG. 3. Same as Fig. 1.

direct problem is solved for a given coefficient $k^*(x, y) = a^*(x)\, b^*(b)$ to obtain the simulated measurement data, and the improved GPST is used to see whether $a^*(x)$ and $b^*(y)$ are recovered or not. First, one chooses a $k^*(x, y)$ which represents the correct coefficient, and also one chooses the source $f(x, y, t)$ which represents a part of the measured data. Its Laplace transform $F(x, y, s)$ is numerically computed for a chosen discrete set of $s = s_j$, $j = 1, 2, 3, \ldots, J$. Then the boundary value problem of the positive-definite elliptic partial differential equation (2) is solved by using the simple center finite difference scheme; thus one generates the rest of the measurement data $H(x, y, s_j)$, $j = 1, 2, 3, \ldots, J$. Next, $k_0(x, y) = a_0(x)\, b_0(y)$ is chosen. Hence upon solving (3), (4), and (14) numerically, $k_1(x, y) = a_1(x)\, b_1(y)$ is obtained. $k_2(x, y) = a_2(x)\, b_2(y)$ can be obtained in a similar manner. One continues this procedure until finally a numerical limit $k_N(x, y) = a_N(x)\, b_N(y)$ is reached. Other than the truncation, round-off, numerical integration, and finite difference approximation errors in both generating the numerical data and computing $a_N(x)$ and $b_N(y)$, any norm of $a^*(x)\, b^*(y) - a_N(x)\, b_N(y)$ can be used as a criterion for evaluating the performance of the improved GPST inversion algorithm.

The numerical simulation here is carried out for a general class of $a^*(x)$ and $b^*(y)$, e.g., monotonic functions, piecewise-linear continuous functions, and oscillatory functions. For simplicity, a unit square is chosen as the spatial domain which is divided uniformly into one hundred square subdomains. A single point source $u(x, y, t) = e^{-t}$ is located in the center of the spatial domain. The data

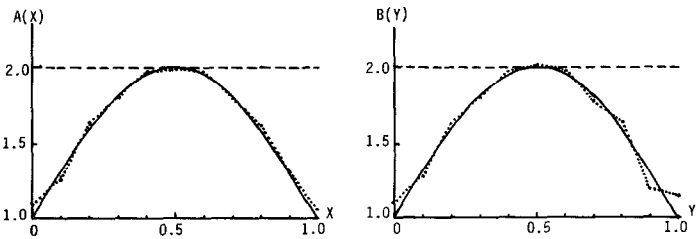


FIG. 4. Same as Fig. 1.

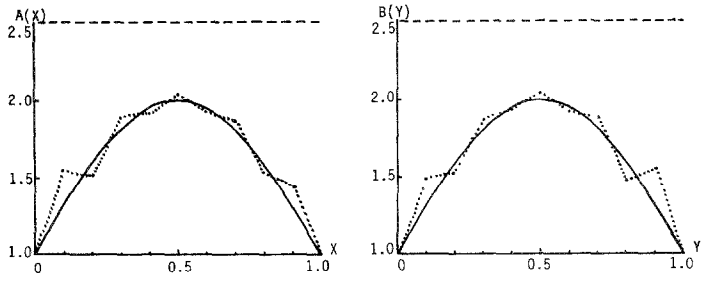


FIG. 5. Same as Fig. 1.

measurement points are located at the grid points $(x_p, y_q) \equiv (p, q) = (2, 2), (3, 2), (4, 2), \ldots, (10, 2), (10, 3), (10, 4), \ldots, (10, 10)$. The discrete values of $s$ for the diffusion equation used here are 1, 2, 3, 4, and 5, and the corresponding values for the wave equation are 1, $2^{1/2}$, $3^{1/2}$, 2, and $5^{1/2}$. The value for the regularization parameter $\alpha$ is 0.05 for all examples, although it is not the optimum value for every example.

It is found that CPU times on the UNIVAC 1100/81 for each iteration of the improved GPST and the standard GPST are approximately 3.5 and 5.2 s, respectively. The numerical results are plotted in Figs. 1–5. The number of iterations needed for the numerical convergence for all examples and their relative maximum pointwise errors are tabulated in the following table.

| Fig. # | $N$ | $\dfrac{\text{Max}\lvert a^*(x) - a_N(x)\rvert}{\text{Max}\lvert a^*(x)\rvert}$ | $\dfrac{\text{Max}\lvert b^*(y) - b_N(y)\rvert}{\text{Max}\lvert b^*(y)\rvert}$ |
|---|---|---|---|
| 1 | 5 | 0.09 | 0.95 |
| 2 | 5 | 0.06 | 0.04 |
| 3 | 10 | 0.12 | 0.11 |
| 4 | 5 | 0.05 | 0.07 |
| 5 | 10 | 0.11 | 0.12 |

## DISCUSSION

From the results of numerical simulation, it seems to be clear that the improved GPST is as robust and accurate as the standard GPST. Moreover, at least for the simple test inverse problem here, the results of numerical simulation have demonstrated three things: (I) The efficiency of the standard GPST is successfully improved by the introduction of re-structuralization and high level parallelism into the system of discretized Fredholm integral equations of the first kind; (II) the speedup can be estimated with reasonable accuracy by performing the computational complexity analysis based upon FLO counting, e.g., the actual $S_1$ per iteration is $\sim 1.49$ while the estimation of $S_1$ per iteration from the formula

corresponding to (15) is $\sim 1.72$; and (III) the number of iterations needed for the numerical convergence of the improved GPST is only slightly larger than that for the standard GPST, e.g., $\sim 25\%$.

There seems to be no reason that the above three phenomena should not be true for more general cases. Hence we believe that the introduction of the re-structuralization and high level parallelism will make GPST super efficient and the computational complexity analysis based upon FLO count can be realistically used as an important tool to estimate the efficiency/speedup in improving the GPST inversion algorithm before its implementation on a computer. In these regards, the efforts of programming the parallel-structured GPST (based upon the improvements presented in this paper) on a CRAY X-MP4 for various large scale practical applications have just begun, and an analysis of the speedup of the general parallel-structured GPST based upon the computational complexity analysis will be given in the near future.

## ACKNOWLEDGMENT

This research was supported in part by the National Science Foundation under Grant No. DMS-8512188.

## REFERENCES

1. D. S. TSIEN AND Y. M. CHEN, *Computing Methods in Nonlinear Mechanics* (Univ. of Texas, Austin, 1974), p. 935.
2. Y. M. CHEN AND D. S. TSIEN, *J. Comput. Phys.* **25**, 366 (1977).
3. D. S. TSIEN AND Y. M. CHEN, *Radio Sci.* **13**, 775 (1978).
4. Y. M. CHEN, International Symp. on Ill-posed Problems, Univ. of Delaware, Newark, Delaware, Oct. 2, 1979 (unpublished).
5. Y. M. CHEN AND J. Q. LIU, *J. Comput. Phys.* **43**, 315 (1981).
6. Y. M. CHEN AND X. T. WENG, *Circuit Theory Appl.* **10**, 69 (1982).
7. Y. M. CHEN AND Y. LIN, *Int. J. Numer. Meths. Eng.* **19**, 825 (1983).
8. R. P. HATCHER AND Y. M. CHEN, *SIAM J. Sci. Stat. Comput.* **4**, 149 (1983).
9. Y. M. CHEN AND J. Q. LIU, *J. Comput. Phys.* **50**, 193 (1983).
10. J. Q. LIU AND Y. M. CHEN, *SIAM J. Sci. Stat. Comput.* **5**, 255 (1984).
11. Y. M. CHEN AND J. Q. LIU, *J. Comput. Phys.* **53**, 429 (1984).
12. Y. M. CHEN AND G. Q. XIE, *Non-Destr. Test. Commun.* **1**, 125 (1984).
13. G. Q. XIE AND Y. M. CHEN, *Appl. Numer. Math.* **1**, 217 (1985).
14. Y. M. CHEN AND S. L. WANG, in *Review of Progress in Quantitative Nondestructive Evaluation* 4, edited by D. O. Thompson and D. E. Chimenti (Plenum, New York, 1985), p. 543.
15. Y. M. CHEN, *Geophys.* **50**, 1664 (1985).
16. Y. M. CHEN AND G. Q. XIE, *J. Comput. Phys.* **62**, 143 (1986).
17. C. H. QIU AND Y. M. CHEN, *Int. J. Solids Struct.* **22**, 901 (1986).
18. X. Y. LIU AND Y. M. CHEN, *SIAM J. Sci. Stat. Comput.* **8**, 436 (1987).
19. Y. M. CHEN, G. Q. XIE AND X. Y. LIU, *SIAM J. Numer. Anal.*, in press.
20. X. Y. LIU AND Y. M. CHEN, *Math. Comput.*, in press.
21. Y. N. TANG AND Y. M. CHEN, in *Advances in Computer Methods for Partial Differential Equations-V*, edited by R. Vichnevetsky and R. Stepleman (IMACS, 1984), p. 433.
22. Y. M. CHEN AND Y. N. TANG, in preparation.