International Journal of Computer Mathematics
Vol. 87, No. 7, June 2010, 1630–1637

An iterative algorithm based on level set method for the shape recovery problem

Qing Liu* and Xiao-Liang Cheng

Department of Mathematics, Zhejiang University, Hangzhou, P.R. China

(*Received 10 July 2008; revised version received 02 August 2008; accepted 07 August 2008*)

In this paper, we propose an iterative algorithm based on the level set method for the shape recovery problem. We use a suitable preconditioner for the artificial time-dependent system for the level set formulation and propose an iterative algorithm of the level set function. We prove the convergence of our algorithm under some hypothesis. Numerical experiments show the efficiency of the algorithm.

Keywords: level set method; inverse problem; iterative algorithm; shape recovery problem; finite element method

2000 AMS Subject Classifications: 05A05; 30C70; 49Q10; 57N25; 57N50

1. Introduction

Let $\Omega \subset \mathbf{R}^d$, $d = 2, 3$. Consider the Poisson equation

$$\begin{cases} -\Delta u = m(\mathbf{x}), & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \end{cases} \tag{1}$$

where $m(\mathbf{x})$ is the characteristic function

$$m(\mathbf{x}) = \begin{cases} m_2, & \mathbf{x} \in D \subset \Omega, \\ m_1, & \mathbf{x} \neq D, \end{cases} \tag{2}$$

with two constants $m_1 > m_2$.

Suppose we have data measurements up to the noise level $z(\mathbf{x})$ of the solutions (1) and (2) on domain Ω and $z(\mathbf{x}) = 0$ on $\partial\Omega$, we want to recover the shape of domain D such that

$$Q(\tilde{u}(\mathbf{x})) = \min_D Q(u(\mathbf{x})), \tag{3}$$

where $Q(u(\mathbf{x})) = 1/2 \int_\Omega (u(\mathbf{x}) - z(\mathbf{x}))^2 \mathrm{d}\mathbf{x}$ and $\tilde{u}(\mathbf{x})$ is the solution of Equations (1) and (2). It is an inverse problem and is highly ill-posed since u is of two-order derivatives-smooth. The

*Corresponding author. Email: korien03@163.com

ISSN 0020-7160 print/ISSN 1029-0265 online

DOI: 10.1080/00207160802410095
http://www.informaworld.com

challenge in solving this problem comes from the fact that we do not know the topology of D aforehand.

As $m(\mathbf{x})$ is the function with discontinuities, the level set method is a powerful tool for treating these problems. The level set method has been widely used to reconstruct the shape problem such as the potential problem [4,5,7], electrical impedance tomography [1,3], eigenvalue optimization problems in the shape design [6,9], and elliptic inverse problems with discontinuous coefficients [2].

In this paper, we propose an iterative algorithm to solve problem (3). We use the finite element method or the finite difference method for solving the forward problem (1) when $m^{(n)}(\mathbf{x}) = m(\psi^{(n)}(\mathbf{x}))$ is given. Then we update the level set function $\psi^{(n)}(\mathbf{x})$ to $\psi^{(n+1)}(\mathbf{x})$ by iteration. The idea comes from the fact that the level set function $\psi(\mathbf{x})$ is the solution of a steady-state equation for the artificial time-dependent system. We use the forward Euler method to reach the steady-state solution. In the next section, we derive this time-dependent system for the level set function for the general inverse problem. Then, we derive the algorithm for problems (1)–(3) and show its convergence in Section 3. We demonstrate our algorithm by showing some numerical results in Section 4.

2. Level set method

As described in [4], we consider the optimization problem

$$\min_m \frac{1}{2}\|F(m) - b\|^2, \tag{4}$$

where $F(m)$ is a vector function of vector m and the components of m take m_1 or m_2, b is the given measurement of $F(m)$ up to noise level, and $\|\cdot\|$ is the least square norm. A direct application of the output least squares method to solve this problem typically runs into trouble. Often one approximately solves the optimization problem by the Tikhonov-type regularization

$$\min_m \frac{1}{2}\|F(m) - b\|^2 + \beta R(m), \tag{5}$$

where $R(m)$ is a regularization term, and $\beta > 0$ is the regularization parameter. There are some different choices about β discussed in [10].

Generally, in the literature, the optimization problem (5) can be written as the steady-state equation for the artificial time-dependent problem

$$\begin{aligned} M(m)\frac{\partial m}{\partial t} &= -[J^{\mathrm{T}}(F(m) - b) + \beta R'(m)], \\ m(0) &= m_0, \end{aligned} \tag{6}$$

where $J = \partial F/\partial m,\ t \geq 0$, is the artificial time variable, and the preconditioner M is positive definite.

If we apply the forward Euler discretization to Equation (6) with a special choice of time step, in fact, it coincides with a preconditioned steepest descent method for minimization problem (5). However, these methods are not known for their efficiency [4]. The main reason is the fact that m is piecewise constant with m_1 and m_2, but it cannot keep this property from equation (6). Vogel [10], pointed out that the system (6) cannot reach the steady-state solution when $M = I$ and $\beta = 0$.

As m is discontinuous, we introduce a smoother level set function $\psi(\mathbf{x})$. We consider m as the function of ψ such that

$$m(\mathbf{x}) = H(\psi(\mathbf{x})), \tag{7}$$

with

$$H(\psi) = \begin{cases} m_2, & \psi \leq 0, \\ m_1, & \psi > 0. \end{cases} \tag{8}$$

Following [4,6,8], we usually apply the regularization $R(\psi)$ to ψ and obtain

$$\min_{\psi} \frac{1}{2}\|\hat{F}(\psi) - b\|^2 + \beta R(\psi), \quad \hat{F}(\psi) = F(m) = F(H(\psi)). \tag{9}$$

If we get the solution ψ, then we get the shape of D from the definition

$$D = \{\mathbf{x} \in \Omega | \psi(\mathbf{x}) \leq 0\}.$$

We know that the solution of Equation (9) should be the steady-state solution of the following system:

$$\begin{aligned} M(\psi)\frac{\partial \psi}{\partial t} &= -[\hat{J}^{\mathrm{T}}(\hat{F}(\psi) - b) + \beta R'(\psi)], \\ \psi(0) &= \psi_0, \end{aligned} \tag{10}$$

where $\hat{J} = \partial \hat{F}/\partial \psi$. To compute this derivative, we use the chain rule and the computation of the derivative $\partial m/\partial \psi$. To do this, we use an approximation function to H, for example, for small $\varepsilon > 0$,

$$H_\varepsilon(s) = \frac{m_1 - m_2}{\pi} \tan^{-1}\left(\frac{s}{\varepsilon}\right) + \frac{m_1 + m_2}{2}. \tag{11}$$

Obviously,

$$\lim_{\varepsilon \to 0} H_\varepsilon(s) = \begin{cases} m_2, & s < 0, \\ m_1, & s > 0. \end{cases}$$

There are some different choices of preconditioner and regularization terms to get some different algorithms based on the formulation (6) and (10), see [4,5,7,10] and reference therein.

Here, we will not use the regularization terms and the derivative $\partial m/\partial \psi$, so we propose an algorithm based on Equation (6) with suitable $M(m)$. We let $M(m)$ such that

$$M(m)\frac{\partial m}{\partial t} \approx \frac{\partial \psi}{\partial t}.$$

In fact we let $M(m) = \lim_{\varepsilon \to 0} H_\varepsilon^{-1}(m)$ in each interval $[t_n,\ t_{n+1}]$. Thus we get the iteration of the level set function ψ by the forward Euler method for system (6) with $\beta = 0$,

$$\psi^{(n+1)} = \psi^{(n)} - \tau[J^{\mathrm{T}}(F(m^{(n)}) - b)], \quad m^{(n)} = H(\psi^{(n)}), \tag{12}$$

where $\tau > 0$. The advantages of the iteration (12) are that we can update the level set function ψ to overcome the discontinuity of m, and we use Equation (6) (not Equation (10)) to avoid the computation of the derivatives of $\partial m/\partial \psi$.

In the next section, we apply this algorithm to solve Equations (1)–(3) and prove the convergence of the algorithm for this problem.

3. Iterative algorithm

Let $P_1, P_2, \ldots, P_N$ be the grid nodes in Ω. First we use the finite element method or finite difference method to approximate the forward Poisson Equation (1), and denote the discrete Poisson operator $-\Delta$ as $N \times N$ matrix A.

Denote

$$\vec{\psi} = (\psi(P_1), \psi(P_2), \ldots, \psi(P_N))^{\mathrm{T}},$$
$$\vec{z} = (z(P_1), z(P_2), \ldots, z(P_N))^{\mathrm{T}},$$
$$\vec{m} = (m(P_1), m(P_2), \ldots, m(P_N))^{\mathrm{T}}.$$

So problem (3) turns into

$$\min_{\vec{\psi}} \frac{1}{2}\|A^{-1}\vec{m} - \vec{z}\|^2, \quad \vec{m} = H(\vec{\psi}),$$

where $m_i = H(\psi_i),\ i = 1, 2, \ldots, N$.

Then we apply the iteration (12) to problems (1)–(3) to get the following algorithm.

ALGORITHM 3.1

(1) For $n = 0$, initial guess value $\vec{\psi}^{(0)}$ is given;
(2) For $n = 0, 1, 2, \ldots,$ and parameter $\tau > 0$, we update

$$\vec{\psi}^{(n+1)} = \vec{\psi}^{(n)} - \tau[(A^{-1})^{\mathrm{T}}(A^{-1}\vec{m}^{(n)} - \vec{z})],$$
$$\vec{m}_i^{(n)} = H(\psi_i^{(n)}), \quad i = 1, 2, \ldots, N.$$

We can get the shape of D,

$$D = \lim_{n\to\infty}\{P_j : \psi^{(n)}(P_j) < 0\}.$$

To analyse the convergence of sequence $\vec{\psi}^{(n)}$, we will consider the following algorithm using function $H_\varepsilon(\cdot)$ to replace function $H(\cdot)$.

ALGORITHM 3.1′

(1) For $n = 0$, initial guess value $\vec{\psi}^{(0)}$ is given;
(2) For $n = 0, 1, 2, \ldots$, and parameter $\tau > 0$ and $\varepsilon > 0$, we update

$$\vec{\psi}^{(n+1)} = \vec{\psi}^{(n)} - \tau[(A^{-1})^{\mathrm{T}}(A^{-1}\vec{m}^{(n)} - \vec{z})],$$
$$\vec{m}_i^{(n)} = H_\varepsilon(\psi_i^{(n)}), \quad i = 1, 2, \cdots, N.$$

Let $\vec{\psi}^{(n)}$ be the sequence of Algorithm 3.1′

$$\vec{e}^{(n+1)} = \vec{\psi}^{(n+1)} - \vec{\psi}^{(n)},$$

then

$$\vec{e}^{(n+1)} = \vec{e}^{(n)} - \tau(A^{-1})^{\mathrm{T}}A^{-1}[H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)})].$$

As $G = (A^{-1})^{\mathrm{T}} A^{-1}$ is a symmetric positive definite matrix, we have the relation

$$\begin{aligned}\|\vec{e}^{(n+1)}\|^2 &= (\vec{e}^{(n)} - \tau G[H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)})],\ \vec{e}^{(n)} - \tau G[H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)})] \\ &= \|\vec{e}^{(n)}\|^2 - 2\tau(G[H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)})],\ \vec{e}^{(n)}) + \tau^2\|G[H_\varepsilon(\psi^{(n)}) \\ &\quad - H_\varepsilon(\psi^{(n-1)})]\|^2. \end{aligned} \tag{13}$$

By the property of function $H_\varepsilon(\cdot)$ in Equation (11), we get

$$H_\varepsilon(\psi_i^{(n)}) - H_\varepsilon(\psi_i^{(n-1)}) = \frac{m_1 - m_2}{\pi} \cdot \frac{\varepsilon}{\varepsilon^2 + \xi_i^2}(\psi_i^{(n)} - \psi_i^{(n-1)}),$$

where ξ_i is between $\psi_{\varepsilon,i}^{(n)}$ and $\psi_{\varepsilon,i}^{(n-1)}$. We suppose $\|\psi^n\|$ are bounded, for fixed $\varepsilon > 0$. Then we have

$$\frac{m_1 - m_2}{\pi} \cdot \frac{\varepsilon}{\varepsilon^2 + \xi_i^2} \geq c^* > 0. \tag{14}$$

Thus, we can get

$$H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)}) = \Lambda \vec{e}^{(n)},$$

where Λ is a diagonal matrix with positive entries as in Equation (14). Thus, there exists a constant c_1 independent of $\tau,\ n$ such that

$$(G[H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)})],\ \vec{e}^{(n)}) \geq c_1\|\vec{e}^{(n)}\|^2.$$

Obviously, we have the inequality, for constant c_2 independent of $\tau,\ n$,

$$\|G[H_\varepsilon(\psi^{(n)}) - H_\varepsilon(\psi^{(n-1)})]\|^2 \leq c_2\|\vec{e}^{(n)}\|^2.$$

So we get

$$\|\vec{e}^{(n+1)}\|^2 \leq (1 - 2c_1\tau + c_2\tau^2)\|\vec{e}^{(n)}\|^2.$$

For sufficient small parameter $0 < \tau < 2c_1/c_2$, we obtain

$$\|\vec{e}^{(n+1)}\|^2 \leq \rho\|\vec{e}^{(n)}\|^2$$

with $\rho < 1$.

By Equation (13) and the bounded hypothesis of sequence, we prove the convergent result.

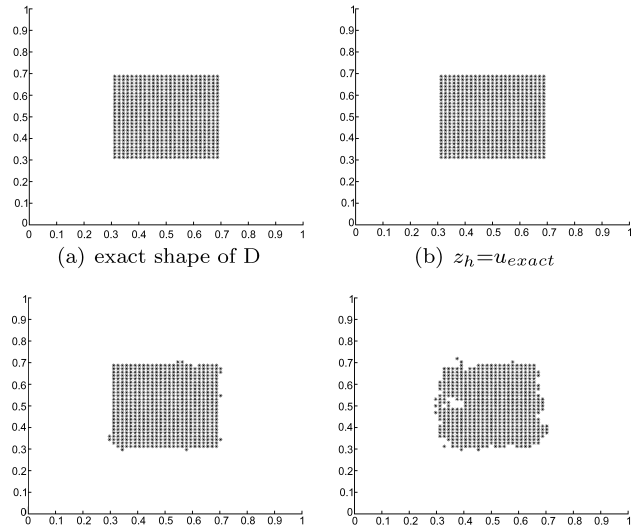
In the practical computation, we use Algorithm 3.1 because for small ε, the $H_\varepsilon(\cdot)$ is almost the same as $H(\cdot)$ for the grid function. We will perform some numerical experiments in the next section to show the efficiency of the algorithm.

4. Numerical results

We test Algorithm 3.1 for two examples with domain $\Omega = (0, 1) \times (0, 1)$, and $m_1 = 1,\ m_2 = 0$. The parameter τ is therefore set to 1 in all the computations presented here. The finite difference method is used to solve problems (1) and (2) on mesh $N = 64 \times 64$.

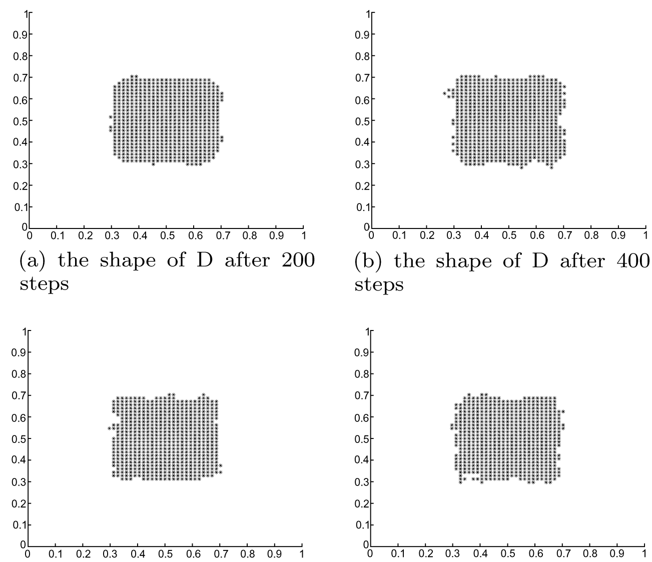
4.1 *Example 1*

First, we consider Example 1 as the exact shape of D, as shown in Figure 1(a). We first use the solution u_{exact} of problems (1) and (2) as z_h to recover the shape of domain D. Then we add some



(a) exact shape of D; (b) $z_h = u_{exact}$; (c) 2% noise to z_h; (d) 10% noise to z_h

Figure 1. Results with different noises: (a) exact shape of D; (b) $z_h = u_{\text{exact}}$; (c) 2% noise to z_h; (d) 10% noise to z_h.



(a) the shape of D after 200 steps; (b) the shape of D after 400 steps; (c) the shape of D after 600 steps; (d) the shape of D after 800 steps

Figure 2. Results with different iterative steps in the case of 10% noise: (a) the shape of D after 200 steps; (b) the shape of D after 400 steps; (c) the shape of D after 600 steps; (d) the shape of D after 800 steps.

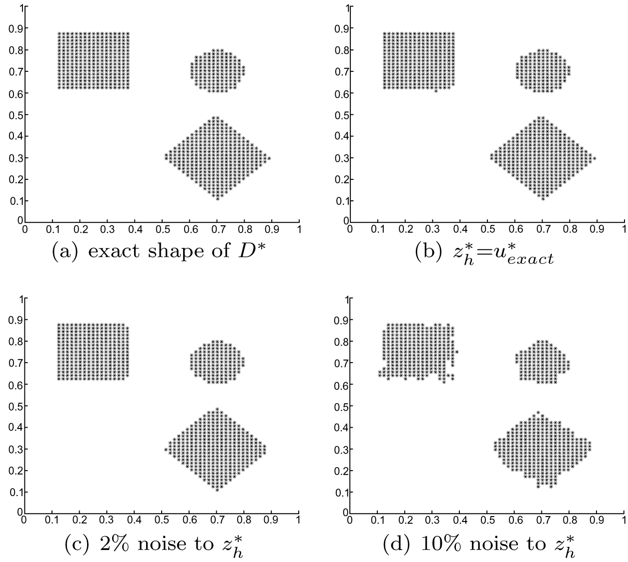


Figure 3. Results with different noises: (a) exact shape of D^*; (b) $z_h^* = u^*_{\text{exact}}$; (c) 2% noise to z_h^*; (d) 10% noise to z_h^*.

noises on this solution and repeat the recovery process. If the tolerance

$$\frac{\|u_{\text{exact}} - A^{-1}m^{(n)}\|}{\|u_{\text{exact}}\|} \geq 10^{-3} \tag{15}$$

or the number of iteration is more than 1000, the iteration will stop. The results are shown in Figure 1.

In order to interpret the process of our iterative algorithm, we present the shapes of D in the case of 10% noise after 200, 400, 600, and 800 iterative steps, respectively, in Figure 2.

4.2 *Example 2*

Then we consider Example 2 to indicate that our algorithm can also deal with the situation that has more than one square. Similarly, the exact shape of D^* is shown in Figure 3(a). The computational process and parameters are similar to those of Example 1. The results are shown in Figure 3.

5. Conclusion

This article provides a new idea to deal with the shape recovery problem. First, the efficiency of the level set method for the artificial time-dependent system and discontinuous terms inspires us to introduce the $H(\psi(\mathbf{x}))$ function, where $\psi(\mathbf{x})$ is a level set function to overcome the high ill-posedness caused by the discontinuities of $m(\mathbf{x})$. Secondly, different from the regularization method for the inverse problem, we construct an iterative algorithm with an appropriate preconditioner $M(m)$ to update the level set function $\psi(\mathbf{x})$, and then we can recover the shape D from its definition.

Acknowledgement

The project is supported by National Natural Science Foundation of China (Grant no. 10471129).

References

[1] L. Borcea, J.G. Berryman, and G.C. Papanicolaou, *High-contrast impedance tomography*, Inverse prob. 12 (1996), pp. 835–858.

[2] T.F. Chan and X.C. Tai, *Level set and total variation regularization for elliptic inverse problems with discontinuous coefficients*, J. Comput. Phys. 193 (2003), pp. 40–66.

[3] M. Cheney, D. Isaacson, and J.C. Newell, *Electrical impedance tomography*, SIAM Rev. 41 (1999), pp. 85–101.

[4] K. Doel and U.M. Ascher, *On level set regularization for highly ill-posed distributed parameter estimation problems*, J. Comput. Phys. 216 (2006), pp. 707–723.

[5] F. Frühauf, O. Scherzer, and A. Leitão, *Analysis of regularization methods for the solution of ill-posed problems involving discontinuous operators*, SIAM J. Numer. Anal. 43 (2005), pp. 767–786.

[6] E. Haber, *A multilevel, level-set method for optimizing eigenvalues in shape design problems*, J. Comput. Phys. 198 (2004), pp. 518–534.

[7] F. Hettlich and W. Rundell, *Iterative methods for the reconstruction of an inverse potential problem*, Inverse Probl. 12 (1996), pp. 251–266.

[8] A. Leitao and O. Scherzer, *On the relation between constraint regularization, level sets, and shape optimization*, Inverse Probl. 19 (2003), pp. L1–L11.

[9] S. Osher and F. Santosa, *Level set methods for optimization problems involving geometry and constraints I. frequencies of a two-density inhomogeneous drum*, J. Comput. Phys. 171 (2001), pp. 272–288.

[10] C. Vogel, *Computational Methods for Inverse Problem*, SIAM, Philadelphia, 2002.