A LEVEL-SET APPROACH FOR INVERSE PROBLEMS INVOLVING OBSTACLES

FADIL SANTOSA*

Abstract. An approach for solving inverse problems involving obstacles is proposed. The approach uses a level-set method which has been shown to be effective in treating problems of moving boundaries, particularly those that involve topological changes in the geometry. We develop two computational methods based on this idea. One method results in a nonlinear time-dependent partial differential equation for the level-set function whose evolution minimizes the residual in the data fit. The second method is an optimization that generates a sequence of level-set functions that reduces the residual. The methods are illustrated in two applications: a deconvolution problem and a diffraction screen reconstruction problem.

Key Words. Inverse problems, level-set method, Hamilton-Jacobi equations, surface evolution, optimization, deconvolution, diffraction.

---

* Advanced Computing Research Institute, Cornell University, Ithaca, NY 14853, `santosa@cs.cornell.edu`. Permanent address: Department of Mathematical Sciences and Department of Mechanical Engineering, University of Delaware, Newark, DE 19716, Research partially supported by the Air Force Office of Scientific Research under grant F49620-93-I-0500, the Department of Energy under grant DE-FG02-94ER25225, the National Science Foundation under grant DMS-9210489, and in part by the Cornell Theory Center which receives major funding from the National Science Foundation and IBM corporation, with additional support from New York State and members of its Corporate Research Institute.

1. Inverse problems involving obstacles. There is a host of inverse problems wherein the desired unknown is a region in $I\!R^2$ or $I\!R^3$. The region is possibly multiply connected or consisting of several subregions. A classical example is the inverse scattering problem for an obstacle (see Colton and Kress [2]). Other examples include a problem in mine detection (Friedman [4]), reconstruction of a diffraction screen (Sondhi [10]; Magnanini and Papi [7]).

A common goal in these problem is to determine the set of an unknown characteristic function given remotely measured data. Abstractly, they can be posed as:

Find D in the equation

$$A(u) = g, \tag{1a}$$

where

$$u(x) = \begin{cases} u_{\text{int}} & \text{for} \quad x \in D \\ u_{\text{ext}} & \text{for} \quad x \notin D \end{cases} . \tag{1b}$$

Here, g and u represent the data and the model parameters of the problem, respectively. The operator $A(\cdot)$, the forward map, is a map from model to the data.

For the mine detection problem, u_{int} is the conductivity of the mine while u_{ext} is the conductivity of the surrounding medium. In the diffraction screen reconstruction problem, $u_{\text{int}} = 1$ and $u_{\text{ext}} = 0$ are prescribed. In both problems, the desired unknown is the domain D.

In the case of inverse scattering by an obstacle, A is the map to the farfield pattern from a scatterer D for a given set of incident waves. For this particular example, u_{ext} is the sound speed of the (exterior) propagating medium. Instead of defining u_{int}, we prescribe boundary conditions (sound-soft or sound-hard) on the wave field on ∂D (see [2]).

Indeed, we believe that obstacle problems arise naturally in situations when the desired unknown is a priori assumed to be a characteristic function. More generally, we would know u_{ext} while u_{int} may be unknown. However, we believe methods can be devised whereby both the domain D and the value u_{int} are determined simultaneously from the data.

We note that in the case where $A(\cdot)$ is governed by a partial differential equation, there are effective methods for computing the action of A. These include standard methods boundary element methods and finite element methods, and a version of finite difference methods called 'the immersed interface methods' of Leveque and Li [6].

Rather than being specific about the nature of the mapping $A(\cdot)$ and the spaces for the model and the data, we choose to talk in generalities and keep the discussion at the formal level. The purpose of this work is to demonstrate that level-set method naturally lends itself to obstacle inverse problems.

2. The level-set approach. The level-set method was developed by Osher and Sethian [9] for problems involving the motion of curves and surfaces. A particular advantage of this approach is the ability of the method to track the motion through topological changes.

Recently, the level-set approach has been used develop methods to solve a segmentation problem arising in computer vision (see Maladi, Sethian and Vermuri [8]; Kichenassamy, Kumar, Olver, Tannenbaum and Yezzi [5]). The methods, sometimes refered to as 'snakes' or 'active contour models', have been shown to be quite versatile and powerful.

One attractive attribute of the level-set method is that it gives a natural way of describing closed curves, particularly, those that sequentially change following a certain rule. Consider a 2-dimensional problem. Suppose the characteristic set of interest is D. The boundary of D is described by a function $\phi(x)$

$$\partial D = \{x \ : \ \phi(x) = 0\}.$$

In the level-set approach, we generate a sequence of functions $\phi_k(x)$ such that

$$D_k \to D,$$

where $\partial D_k = \{x \ : \ \phi_k(x) = 0\}$. Note that k may be a continuous parameter, representing time, if we consider evolution.

In an obstacle reconstruction problem, we can think of the variable u as the desired unknown. The function u in (1b), can be given a level-set description as follows. We introduce an unknown function $\phi(x)$ which is tied to u in

$$u(x) = \begin{cases} u_{\text{int}} & \text{for} \quad \{x \ : \ \phi(x) < 0\} \\ u_{\text{ext}} & \text{for} \quad \{x \ : \ \phi(x) > 0\} \end{cases} . \tag{2}$$

Several advantages of this mode of representing the unknown through the function $\phi(x)$ becomes apparent:

1. No a priori assumptions about the connectedness (topology) of D need to be made. That is, D could be made up of several disconnected subregions. Moreover, D could be multiply connected.
2. No a priori assumptions on the nature of D need to be made, e.g., one often finds in the scattering literature the assumption that the unknown obstacle is star-shaped.

On the other hand, one immediately sees that in using this description, a linear inverse problem becomes nonlinear. This is due to the nonlinear dependence of u on ϕ.

Under this description, the inverse problem stated in (1) becomes:

Find $\phi(x)$ *in*

$$u(x) = \begin{cases} u_{\text{int}} & \{x \ : \ \phi(x) < 0\} \\ u_{\text{ext}} & \{x \ : \ \phi(x) > 0\} \end{cases} ,$$

such that

$$A(u) = g.$$

We will next describe two approaches for finding $\phi(x)$ in such a problem. The first one is based on a time evolution, leading to a Hamilton-Jacobi equation. The second is based on optimization. We note that our motivation for proposing this approach is due to its flexibility in describing obstacles. We do not expect this formulation to help in making an illposed problem better behaved.

3. Formal calculation of variations. In order to find the dependence of the forward map on small changes on the obstacle boundary, we need to calculate the variation of u caused by a variation in ϕ. To facilitate this calculation, let x be a point on the curve $\partial D = \{x \ : \ \phi(x) = 0\}$. Suppose $\phi(x)$ is perturbed by a small variation $\delta\phi(x)$. Let δx be the resulting variation of the point x. The variation results in the region D to become a new region denoted by D'.

By taking the variation of the equation $\phi(x) = 0$, we find

$$\delta\phi + \nabla\phi \cdot \delta x = 0. \tag{3}$$

Observe that the unit outward normal at x is

$$n(x) = \frac{\nabla\phi(x)}{|\nabla\phi(x)|}.$$

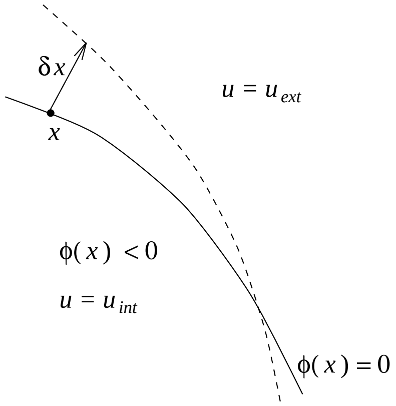

FIG. 1. *The geometry of the variation of the curve* $\{x\ :\ \phi(x) = 0\}$ *under a variation* $\delta\phi(x)$.

Next consider $u + \delta u$. In the accompanying Figure 1, observe that at x, since the surface is moving out, δu will make u in the region between x and $x + \delta x$ become u_{int}. Therefore, at these points $\delta u = u_{\text{int}} - u_{\text{ext}}$.

Consider the inner product of δu with a test function $f(x)$. We have, formally,

$$< \delta u, f >:= \int_{\mathbb{R}^2} \delta u(x)\, f(x) dx = \int_{D \cap D'} \delta u(x)\, f(x) dx.$$

The value of $\delta u(x)$ is either plus or minus of $(u_{\text{int}} - u_{\text{ext}})$. Since δx is infinitesimal, we can simplify the inner product to

$$< \delta u, f >= \int_{\partial D} (u_{\text{int}} - u_{\text{ext}})\, \delta x \cdot n(x)\, f(x)\, ds(x),$$

where $ds(x)$ is the incremental arclength. Roughly speaking, the expression $(\delta x \cdot n(x)\, ds(x))$ is the incremental area over which u varies at x. Therefore, integrating it over the entire boundary ∂D will lead to the inner product of δu with f.

We can now identify δu from the last expression. It is actually a measure on the curve ∂D. Its strength is the product of $(u_{\text{int}} - u_{\text{ext}})$ with normal component of δx. Therefore, we have

$$\delta u = (u_{\text{int}} - u_{\text{ext}})\, \frac{\nabla \phi(x)}{|\nabla \phi(x)|} \cdot \delta x \bigg|_{x \in \partial D}. \tag{4}$$

Note that the dot product induces the correct sign for δu. Consulting Figure 1, we note that δu will have the same sign as $(u_{\text{int}} - u_{\text{ext}})$ at the point labeled x. On the other hand, the opposite sign will occur if the normal component of δx is in the opposite direction of the normal.

4. An evolution approach. We propose to solve the inverse obstacle problem by deriving an evolution equation for $\phi(x)$. The equation will have the property that as 'time' $t \to \infty$, we arrive at a function $\phi(x)$ such that the associated $u(x)$ in (2) is a solution to the inverse problem.

Let t represent time, then the function ϕ depends on on both x and t. We use the notation

$$\partial D(t) = \{x\ :\ \phi(x,t) = 0\}.$$

Therefore, we associate an evolving region with $\phi(x,t)$.

We will solve the inverse problem in the leastsquares sense, that is, we seek the minimizer of

$$F(u) := \frac{1}{2} ||A(u) - g||_2^2. \tag{5}$$

The minimal requirement for the variation of $\phi(x,t)$ is that $F(u)$ be a decreasing function of t.

To this end, let us assume that each point $x \in \partial D(t)$ moves perpendicular to the surface. That is, the variation δx satisfies

$$\delta x = \alpha(x,t)\, \frac{\nabla \phi}{|\nabla \phi|}. \tag{6}$$

Here, $\alpha(x,t)$ can be viewed as the velocity of the surface at x. Substituting this expression in (4), we obtain

$$\delta u = (u_{\text{int}} - u_{\text{ext}})\ \alpha(x,t)\,|_{x\in\partial D(t)}\,. \tag{7}$$

The directional derivative of the function $F(u)$ in the direction δu is given by

$$\delta F(u) = < J(u)^T (A(u) - g), \delta u >,$$

where $J(u)$ is the Jacobian of $A(u)$ at u. Recalling the form of δu in (7) and inserting it in the above, we obtain

$$\delta F(u) = \int_{\partial D(t)} \left[J(u)^T (A(u) - g) \right]\ (u_{\text{int}} - u_{\text{ext}})\ \alpha(x,t)\, ds(x). \tag{8}$$

Assuming that $u_{\text{int}} > u_{\text{ext}}$, we arrive at a natural choice of $\alpha(x,t)$. In order for $\delta F(u)$ to be negative, we choose

$$\alpha(x,t)|_{\partial D(t)} = -J(u)^T (A(u) - g)\Big|_{\partial D(t)}\,. \tag{9}$$

To determine the equation for $\delta\phi(x,t)$, we use (3) and (6)

$$\delta \phi = -\nabla\phi \cdot \delta x = -\nabla \phi \cdot \left(\alpha(x,t)\, \frac{\nabla\phi}{|\nabla\phi|} \right) = -\alpha(x,t)\, |\nabla \phi|.$$

Any $\alpha(x,t)$ satisfying (9) will produce a $\delta\phi$ that reduces $F(u)$. We choose $\alpha(x,t)$ to be

$$\alpha(x,t) = -J(u)^T (A(u) - g), \tag{10}$$

i.e., α is the extension of $-J(u)^T(A(u) - g)$. This leads to

$$\delta\phi = \left[J(u)^T (A(u) - g) \right]\ |\nabla\phi|.$$

The initial value problem for $\phi(x,t)$ is a Hamilton-Jacobi system

$$\frac{\partial \phi}{\partial t} = \left[J(u)^T (A(u) - g) \right]\ |\nabla\phi|, \tag{11a}$$

$$\phi(x,0) = \phi_0(x). \tag{11b}$$

Our claim is that starting from some initial surface $\phi_0(x)$, the evolution described by (11) will lead to a solution of the inverse problem. A simple choice for $\phi_0(x)$ in (11b) is such that $D(t \to \infty) \subset\subset D(0)$, although this is not crucial. We note that

1. The evolution is such that $F(u)$ is nonincreasing; that is, $\frac{\partial}{\partial t} F(u) \leq 0$.
2. If a solution u exists, then at the solution $A(u) = g$, so that $\partial\phi/\partial t = 0$ at the solution.
3. We can view the evolution as a flow in the steepest descent direction for the residual $F(u)$.

It should be emphasized here that we have no theoretical foundations that support our assertion that this is a method for solving the obstacle inverse problem. However, we have sufficient computational experience, which will be reported in this paper, to convince us that this method can be used to compute an approximate solution to the inverse problem.

5. An optimization approach. In the optimization approach, we generate a sequence of surfaces $\phi_k(x)$ whose associated function $u(x)$ represent descent directions for the functional $F(u)$. For this purpose, it is convenient to use the notation

$$\partial D_k = \{x \ : \ \phi_k(x) = 0\}.$$

Suppose we are currently at $\phi_k(x)$. An update $\delta\phi(x)$ is needed to generate $\phi_{k+1}(x)$. The process of finding $\delta\phi(x)$ is somewhat of a reverse of the procedure described in the last section.

We start with a Gauss-Newton approach (see for example Dennis and Schnable [3]) for minimizing the nonlinear leastsquares functional $F(u)$. The descent update for this approach is given by $\delta u'$ where

$$J(u)^T J(u)\delta u' = J(u)^T (g - A(u)). \tag{12}$$

Accepting this as an update for u, we need to find the associated update for ϕ. Note that $\delta u'$ here is defined over all of $\mathbb{R}^2$.

Next, consider the expression for δu in (7), which we rewrite here for convenience

$$\delta u = (u_{\text{int}} - u_{\text{ext}}) \ \alpha(x) \, |_{x\in\partial D_k} \, .$$

Implicit in this representation is the assumption that the points on ∂D_k move perpendicular to it. Now, we take the trace of $\delta u'$ in the Gauss-Newton update on ∂D_k and equate it with the righthand side of the above equality

$$\delta u'|_{x\in\partial D_k} = (u_{\text{int}} - u_{\text{ext}}) \ \alpha(x) \, |_{x\in\partial D_k} \, .$$

Rearranging, we find that

$$\alpha(x) \, |_{x\in\partial D_k} = \frac{\delta u'|_{x\in\partial D_k}}{u_{\text{int}} - u_{\text{ext}}}. \tag{13}$$

A way to interpret this is to say that $\alpha(x)$ satisfying (13) is related to the Gauss-Newton update of the functional $F(u)$.

A natural choice for $\alpha(x)$ is to extend the above relation to all points on $\mathbb{R}^2$

$$\alpha(x) = \frac{\delta u'}{u_{\text{int}} - u_{\text{ext}}}. \tag{14}$$

This choice, along with (3), leads to the update for $\delta\phi(x)$ given by

$$\delta\phi(x) = -\frac{\delta u'}{u_{\text{int}} - u_{\text{ext}}} |\nabla\phi_k(x)|. \tag{15}$$

We can now show that this is a descent direction for $F(u)$. To do so, we substitute (14) into the cost function variation formula in (8). We find that

$$\delta F(u) = -\int_{\partial D_k} \left[J(u)^T(A(u) - g)\right] \left[J(u)^T J(u)\right]^{-1} \left[J(u)^T(A(u) - g)\right] \, ds(x),$$

which is negative if $J(u)^T J(u) > 0$. Therefore, the variation $\delta\phi(x)$ corresponds to a variation in $F(u)$ that is negative.

A Gauss-Newton algorithm for the obstacle problem is

0. Choose $\phi_0(x)$; set $k = 0$.
1. Compute associated u; if $F(u) >$ tolerance, do
   i. Compute
$$J(u)^T J(u) \delta u' = J(u)^T (g - A(u)).$$
   ii. Compute
$$\delta\phi(x) = -\frac{\delta u'}{u_{\text{int}} - u_{\text{ext}}} |\nabla \phi_k(x)|.$$
   iii. Set $k = k + 1$, update $\phi_{k+1}(x) = \phi_k(x) + \delta\phi(x)$.

We do not have theoretical results that show that this algorithm will converge to the solution of the inverse obstacle problem. Rather, the present discussion is meant to give a computational framework for obtaining an approximate solution. While no convergence result is known at this time, we do have numerical experience from which we can draw some conclusion.

6. Example 1: Deconvolution. As a first problem, let us consider a simple deconvolution in 2 dimensions. The operator for this problem is linear, and is given by

$$Au = \int G(x - x')\, u(x')\, dx'.$$

For simplicity, we consider the unknown function u to be

$$u(x) = \begin{cases} 1 & \text{for} \quad x \in D \\ 0 & \text{for} \quad x \notin D \end{cases}.$$

In this example, $G(x)$ is a Gaussian

$$G(x) = \exp\left(-\sigma |x|^2\right).$$

The operator A is compact, so we expect its discretization to exhibit illconditioning. Note that it is also self-adjoint. The Jacobian is equal A. In the inverse problem, we are given g and we wish to solve for $u(x)$ in

$$Au = g.$$

Implicit in the problem is that u is a characteristic function.

For the function $\phi(x)$, we choose an initial function

$$\phi_0(x) = a - b \exp\left(-\nu |x|^2\right).$$

This gives level curves which are circles. We select this form for $\phi_0(x)$ because in solving the initial value problem (11) numerically, we will impose homogeneous Neumann condition on $\phi(x,t)$, and $\phi_0(x)$ already satisfies this condition (approximately).

The specifics of the numerical example are as follows:

1. Domain: $[-1, 1] \times [-1, 1]$.
2. Constant: $a = 0.1$, $b = 1$, and $\nu = 10$.

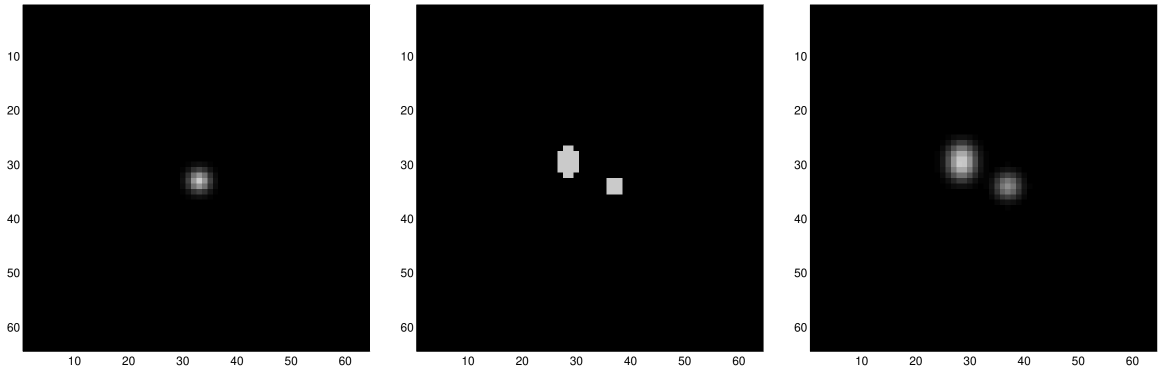

FIG. 2. *(a) The Gaussian blurring used. (b) The domain to be reconstructed. (c) Data for inversion.*

3. Convolution: $\sigma = 275.625$; in the computation $g(x)$ is zeroed for $|x| > 0.095$ (see Figure 2a).

We discretize the problem as follows. On the plane, we label node points by i and j, the corresponding coordinates are

$$x_{ij} = (-1 + (i-1)h, -1 + (j-1)h)\,, \quad i, j = 1, \cdots, n,$$

where $h = 2/(n-1)$. We keep track of the function $\phi(x)$ at the node points. The unknown function $u(x)$ is assumed to be piecewise constant taking values of either 1 or 0 over a pixel of size $h \times h$. We denote the value of $u(x)$ over the pixel centered at x_{ij} by u_{ij}. The rule for evaluating u is

$$u_{ij} = \begin{cases} 1 & \text{for} \quad \phi(x_{ij}) \leq 0 \\ 0 & \text{for} \quad \phi(x_{ij}) > 0 \end{cases}.$$

Therefore, in our implementation, we resolve the boundary of a region D to one pixel size.

The operator A can be viewed as a linear tranformation taking an $n \times n$ matrix to an $n \times n$ matrix. The evaluation can be done rapidly using Matlab's [1] convolution functions.

Evolution. For the numerical solution of (11), we use the method outlined in Osher and Sethian [9]. The main effort is the computation of the term

$$\alpha = -A(Au - g)$$

for a given u. The evolution of $\phi(x,t)$ is through

$$\frac{\partial \phi}{\partial t} = -\alpha\,|\nabla \phi|.$$

Crucial to the success is the time-step size in order to guarantee descent at each time step. We view the time step as a step length in optimization, and allow it to be shortened as needed.

Optimization. In implementing the optimization approach we need to solve (12) for $\delta u'$. We do this calculation using a conjugate gradient method. Since A is self-adjoint, we actually solve

$$A\,\delta u' = (g - Au).$$

The gradient of $\phi_k(x)$ is computed using the Osher and Sethian's discretization as in the evolution method. These details incorporated in an algorithm similar to that outlined in Section 5. Care must be taken in choosing the steplength β in order to insure proper descent.

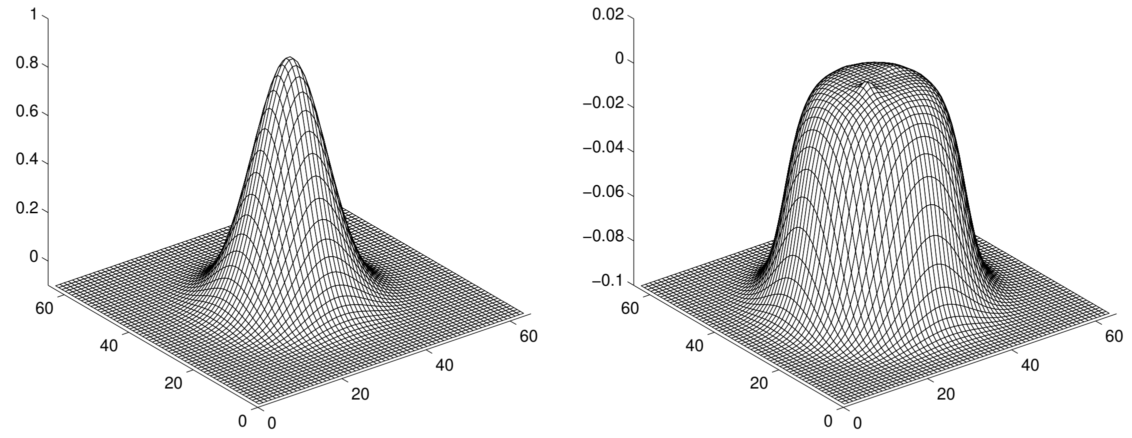

FIG. 3. *(a) Graph of the negative of $\phi_0(x)$. (b) Graph of the negative of the final $\phi(x)$.*

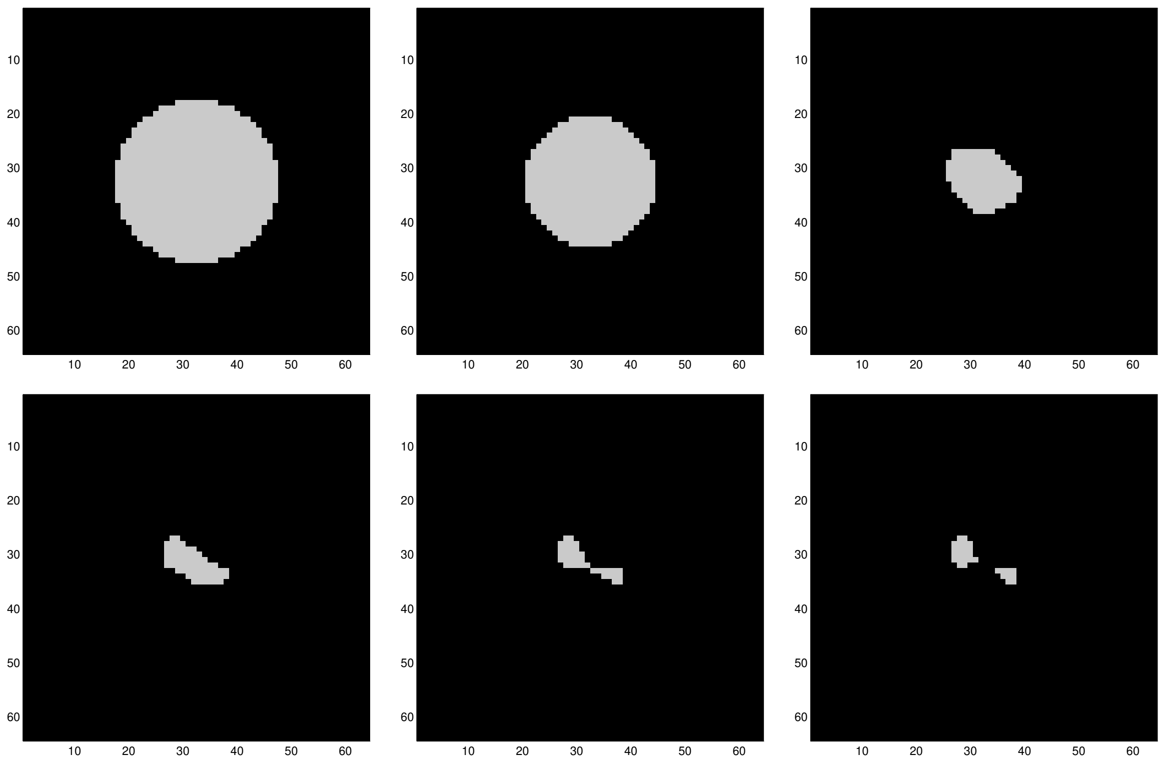

FIG. 4. *The evolution of the zero-level set though the Hamilton-Jacobi equation. Going from left to right beginning the top left, the initial level-set, the level set after 40 time steps, and subsequently, after 20 additional time steps.*

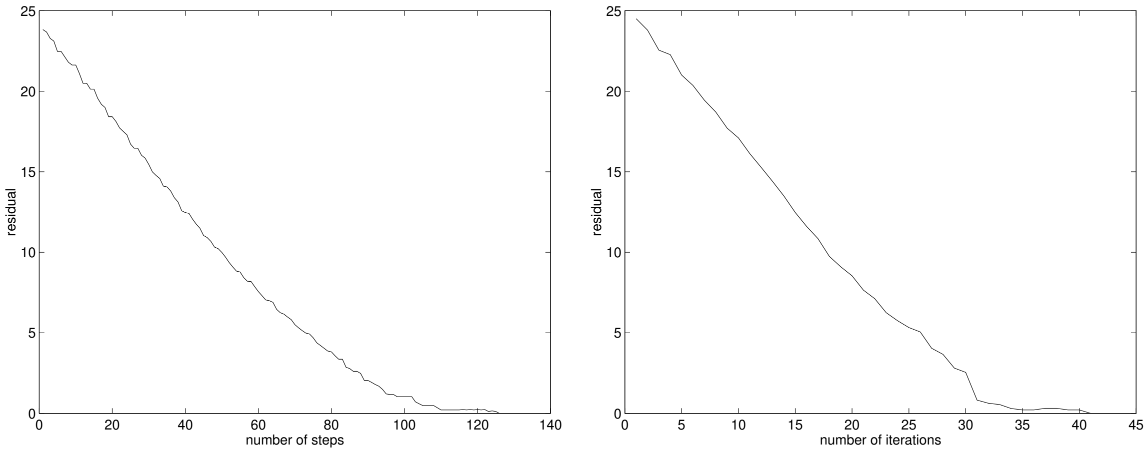

FIG. 5. *(a) Graph of the residual as a function of number of time steps for the evolution method. (b) Graph of the residual as a function of number of iterations for the optimization method.*

Results. We choose relatively small demonstration problems; $n = 64$. The discretized function G is displayed in Figure 2a. The domain we wish to reconstruct is shown in Figure 2b. Note that the domain is disjoint. Figure 2c is grey-level plot of the data.

We first present the result of the evolution method. Figures 3a and 3b show the graphs of the initial function $\phi_0(x)$ and the final function $\phi(x)$. Negative ϕ is plotted for clarity. The pictures in Figure 4 depict the evolution of the zero-level set through the Hamilton-Jacobi equation. The solution obtained is identical to the true domain in Figure 2b. The time step is fixed at 0.01.

The reduction in the residual is proportional to the norm of the function $\alpha = -A(Au - g)$. This residual is graphed for the evolution method in Figure 5a. The horizontal axis is the number of time steps.

We also obtained results using the optimization method. We only display the residual for as a function of iteration is shown in Figure 5b. The step length is $\beta = 0.01$ We took 10 conjugate gradient iterations in each inner loop of our optimization.

It is perhaps not too surprising that the true D is reconstructed by both methods. What is important is that we were able to reconstruct a disjoint set of subregions. This gives an idea to the flexibility of the level-set approach for describing regions. We emphasize that in many inverse problems, we do not know a priori if the obstacle we wish to reconstruct consists of several subregions. Moreover, we need not make any assumptions on the form of the region (e.g., star-shaped, polygonal, etc.).

The reduction in the residual in both methods can be improved by choosing step sizes carefully, or implementing a line search. We defer addressing this issue. Our point is to demonstrate the use of a level-set approach on such obstacle inverse problems.

7. Example 2: Reconstruction of diffraction screen. The problem of diffaction screen reconstruction arises in acoustics and optics. In $I\!R^3$, let us set up the coordinates (x, z) where $x \in I\!R^2$. An opaque screen, with a cutout described by D is placed on the plane $z = 0$ (see Figure 6). A harmonic plane wave with wavenumber k propagates in the positive z direction. It hits the screen and the propagating field escapes through the cutout. We are given the values of the field at $z = \tau > 0$. The problem is to reconstruct the domain D from the data. This problem has been considered by Sondhi [10], and more recently by Magnanini and Papi [7].

We will consider this problem within the Kirchhoff approximation. Let $w(x, z)$ represent the

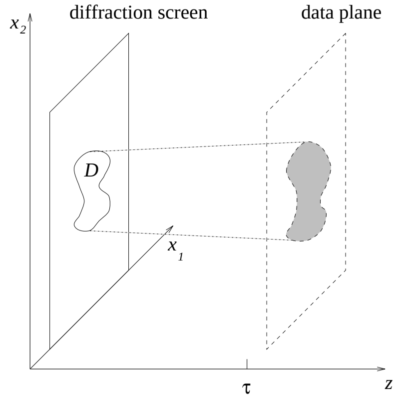

FIG. 6. *The diffraction screen reconstruction problem. A plane wave of wavenumber k is incident upon the screen at $z = 0$. We are given diffraction data at $z = \tau$. The problem is to reconstruct the cutout D in the screen.*

wave field (excess pressure in linear acoustics). Then w satisfies

$$\triangle w + k^2 w = 0, \quad z > 0, \quad \text{with} \;\; w(x, 0) = u(x). \tag{16}$$

Here, $u(x)$ is the characteristic function of the set D. Additionally, radiation condition is imposed on w. In the inverse problem, we are given

$$w(x, \tau) = g(x),$$

and we are asked to find D.

The problem is reposed in the Fourier domain. Let $\hat{w}(\xi, z)$ be the Fourier transform of $w(x, z)$

$$\hat{w}(\xi, z) = \int_{\mathbb{R}^2} w(x, z) \exp(-ix \cdot \xi) dx.$$

By taking the Fourier transform of (16), we obtain the solution to equation

$$\hat{w}(\xi, z) = \exp\left(iz\sqrt{k^2 - |\xi|^2}\right) \hat{u}(\xi).$$

Here $\hat{u}(\xi)$ is the transform of $u(x)$. In the inverse problem, the operator taking u to the data at $z = \tau$ is

$$Au = \exp\left(i\tau\sqrt{k^2 - |\xi|^2}\right) \hat{u}(\xi). \tag{17}$$

We view the data as being given in the Fourier domain, denoted by $\hat{g}(\xi)$. The inverse problem is to find D, the support of the characteristic function $u(x)$ in

$$Au = \hat{g}. \tag{18}$$

The problem, as can be seen, is linear. The only usual feature is that A takes a real function $u(x)$ to a complex function of ξ. The operator A is compact because of the exponential decay of the exponential multiplier (consult (17)). The instability has been studied in detail in [7].

The adjoint of A is given by

$$A^T v = \frac{1}{2\pi^2} \int_{\mathbb{R}^2} \exp\left(-i\tau\sqrt{k^2 - |\xi|^2}\right) v(\xi) \exp(i\xi \cdot x) d\xi.$$

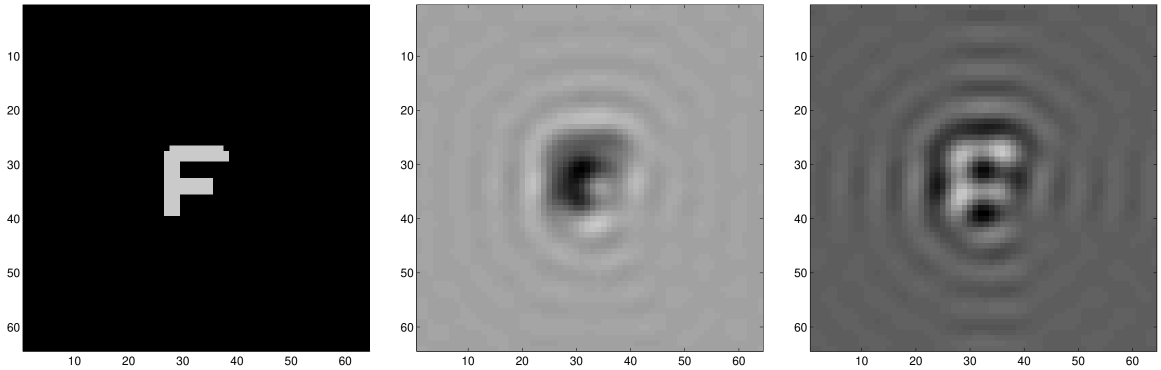

FIG. 7. *(a) The cutout to be reconstructed. (b) The real part of the diffraction data. (c) The imaginary part of the diffraction data.*

It takes a complex function of ξ to a real function.

We use a similar discretization as for the convolution problem in Example 1. We restrict our attention to the node points

$$x_{ij} = (-1 + (i-1)h, -1 + (j-1)h)\,, \quad i, j = 1, \cdots, n,$$

where $h = 20\pi/(n-1)$. The function $\phi(x)$ are evaluated at the coordinates x_{ij}. As in the previous example, we resolve the boundary of D to one pixel width, so we assume $u(x)$ to be piecewise constant on the pixel, taking on values

$$u_{ij} = \begin{cases} 1 & \text{for} \quad \phi(x_{ij}) \leq 0 \\ 0 & \text{for} \quad \phi(x_{ij}) > 0 \end{cases} .$$

Evolution. For this method, we need to compute

$$\alpha = -A^T(Au - g).$$

This is done using the FFT facilities in Matlab [1]. Care must be taken to relate the FFT of a discretized function with the Fourier transform of a continuous function of x. Moreover, we must keep track of real and imaginary variables. The time-stepping algorithm for $\phi(x, t)$ is the same as the one used in Example 1. For this problem, we implemented a variable time step. The time step is halfed until decrease in the residual achieved or until a lower bound on the time step is reached. While this does not guarantee decrease at each step, it does restrict the size of residual increase when that occurs.

Optimization. At each iteration, we must solve the normal equation for a real $\delta u'$

$$A^T A\, \delta u' = A^T(g - Au).$$

This is done using a conjugate gradient algorithm. The remainder of the Gauss-Newton method is identical to the one in the previous example.

Results. We chose a cutout in the shape of the letter 'F' for our test (Figure 7a). The problem size is $n = 64$. Data is generated by taking the FFT of u corresponding to our cutout. We scaled the problem so that the frequency sampling is at 1/10 cycles per unit length. This corresponds to the image plane being a square of dimension 20π. The corresponding frequency domain data

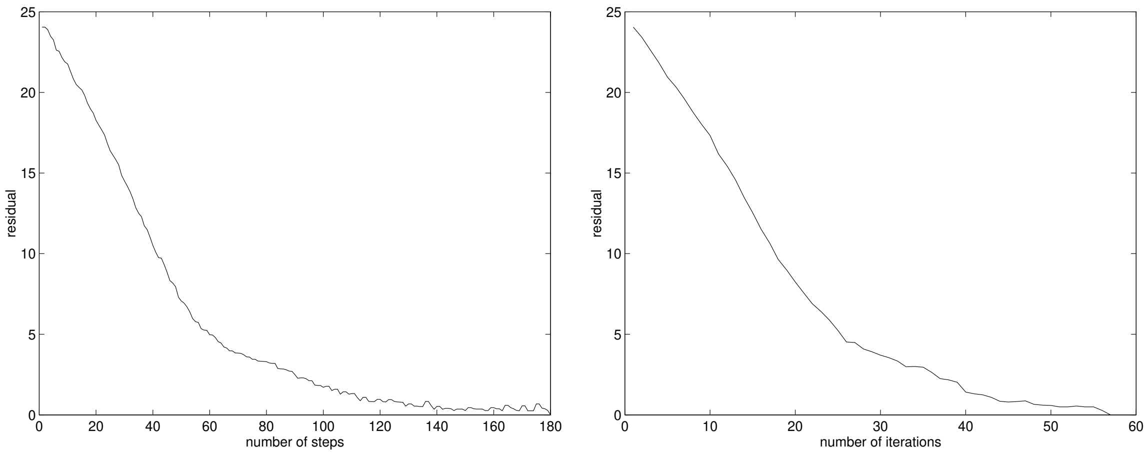

FIG. 8. *(a) Graph of the residual as a function of number of time steps (not time) for the evolution method. (b) Graph of the residual as a function of number of iterations for the optimization method.*

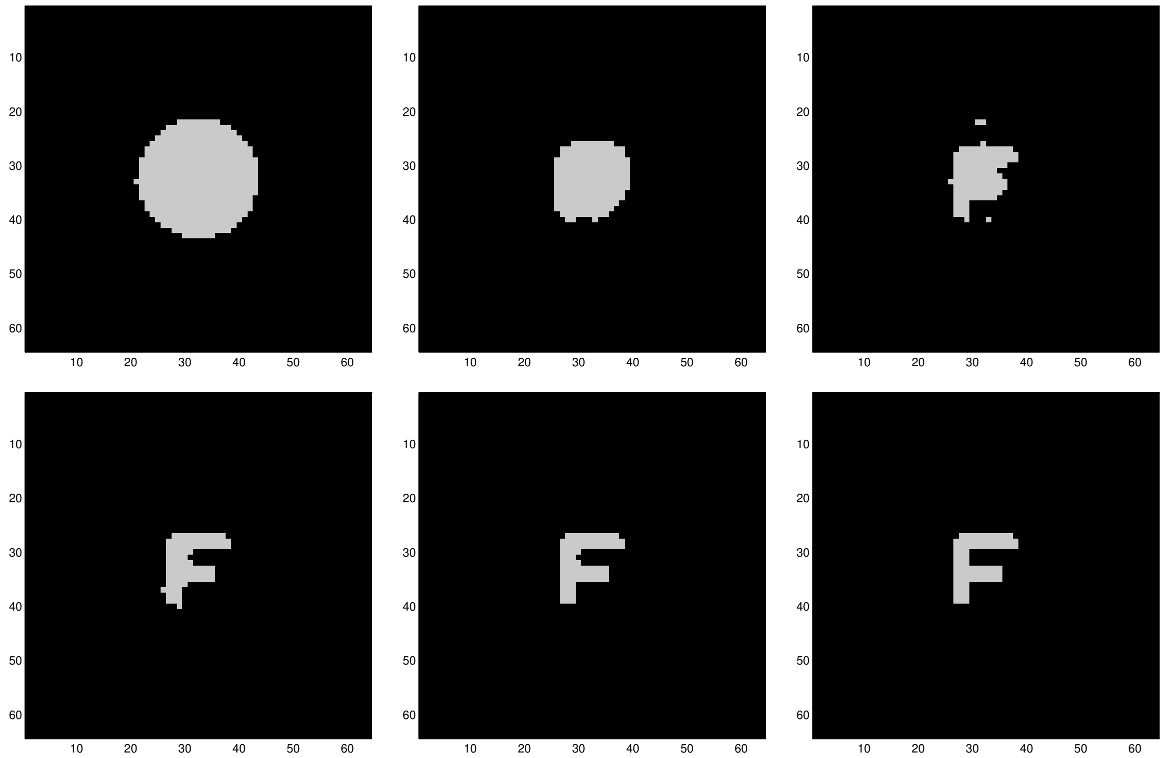

FIG. 9. *The evolution of the zero-level set though the optimization procedure. Going from left to right beginning the top left, the level-set after 10 iterations, after 20, 30, 40, 50 and 57 iterations. Note that the cutout has been reconstructed perfectly.*

$\hat{g}$ is calculated with $k = 1$ and with the measurement screen at $\tau = \pi$. To view the data, we take the inverse FFT of $\hat{g}$. The real and imaginary parts of g are displayed in Figures 7b and 7c. They correspond to the real and imaginary parts of the measured wave field. As can be seen, what is measured is a very blurred picture of the original cutout. The blurring gets more severe as the measurement screen is moved away from the cutout, and as the wavenumber k is decreased. However, increasing k has the effect of increasing the reverberations around the original cutout.

The results of the evolution method is shown in Figure 8a. In our calculation, the initial time step is taken to be 0.314. A lower bound of 0.0063 is set. The lower bound is used at many of the time steps, several of them lead to a small increase in the residual. Eventually, the original cutout is reconstructed.

For the optimization method, we used step length $\beta = 0.01$, and performed 10 conjugate gradient iterations the inner loop. The plot of the residual versus number of outer iterations is displayed in Figure 8b. We also show the shapes of the zero-level sets of the function $\phi(x)$ at different points in the iteration (Figure 9). Note that the cutout is reconstructed by this method also.

8. A remark about instability and regularization. As we have previously alluded to, this mode of representing the unknown obstacle, albeit rather natural and convenient, does not alter the properties of an inverse problem. More particularly, an inverse problem that is illconditioned will remain so even if it is given a level-set description.

A natural regularization, for severely illposed problems where there is substantial noise in the data is curve shortening. Effective numerical methods for curve shortening flows or (inward) mean curvature flows has been proposed by Osher and Sethian [9]. Active contour modeling is based on such a flow. A multiplier to the curvature and an advection term are incorporated to allow segmentation of concave objects [8, 5]. In this spirit, we suggest an evolution of the type

$$\frac{\partial \phi}{\partial t} = (-\alpha + b)|\nabla \phi| + \lambda\, a\, |\nabla \phi| \left(\nabla \cdot \frac{\nabla \phi}{|\nabla \phi|} \right).$$

The constant λ is a regularization penalty term. The function a and b will need to be determined based on side information about the unknown characteristic function. For instance, if we know that the domain D consists of convex subregions, the choice $a = 1$ and $b = 0$ makes sense as these will force short contours for the solution.

The choice of the functions a and b will be crucial to the success of the regularization strategy. We plan to devote some effort in this direction in the future. The same idea can be applied to optimization methods by imposing a total variation penalty to the residual cost function.

9. Discussion. We have described a method for obtaining numerical solutions of inverse problems involving obstacles using a level-set strategy. Representing the unknown in such problems using a level-set function is quite natural and very flexible. Two computational schemes have been devised. One is based on time evolution, while the other is based on optimization. Both solution methods are demonstrated in numerical examples.

There are several other issues not addressed in this work.

1. *Choice of regularization for severely illposed problems.* We mentioned that an illposed problem will most likely remain so under a level-set description. Therefore, regularization is an important issue in problems involving noise in the data. We advocate using geometric type penalty terms such as arclength.
2. *Unknown jump.* In this work, we assumed that the jump in the unknown function $u(x)$ across the interface is given. In many problems arising in application, the

jump itself may be part of the problem. We believe it is possible to modify the present methods and devise a scheme whereby both the jump and the unknown domain are recovered. The success of such a strategy will most likely depend on overdeterminancy of the data.

3. *More complicated models.* We applied our methods to two relative simple linear inverse problems. It would be interesting to apply the same methods to nonlinear problems, such as the inverse scattering problem.
4. *Algorithms based on problem structure.* In this work, we did not use the properties of the parameter-to-data maps. We believe special algorithms that exploit certain structure of these maps are possible.
5. *Accelerating convergence.* We noted that the performance of our methods could be improved by incorporating a line search strategy. Indeed, we think that tools from optimization methods should be imported to this problem to enhance performance.
6. *3-dimensional problems.* The methods described here extends to 3-dimensions. The necessary computational tools has been developed by Osher and Sethian [9].

We hope to address some of the questions raised above in future work.

10. Acknowledgements. The author gratefully acknowledges pleasant and useful discussions on this work with Paul Steen (Cornell) and Satyanad Kichenessamy (Minnesota). This work was completed during the author's visit to the Advanced Computing Research Institute, Cornell Theory Center, Spring 1995. He thanks the director of the institute, Tom Coleman, for his support and interest.

REFERENCES

[1] *MATLAB: High-performance numeric computation and visualization software – Reference guide*, MathWorks, Natick, MA, 1992.

[2] D. Colton and R. Kress, *Inverse acoustic and electromagnetic scattering theory*, Springer-Verlag, Berlin, 1992.

[3] J. Dennis and R. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice-Hall, Englewood Cliffs, 1983.

[4] A. Friedman, *Detection of mines by electric measurements*, SIAM J. Appl. Math., 47 (1987), pp. 201–212.

[5] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, *Gradient flows and geometric active contour models*, preprint, (1994).

[6] R. LeVeque and Z. Li, *The immersed interface method for elliptic equations with discontinuous coefficients and singular sources*, SIAM J. Num. Analysis, 31 (1994), pp. 1019–1044.

[7] R. Magnanini and G. Papi, *An inverse problem for the helmholtz equation*, Inverse Problems, 1 (1985), pp. 357–370.

[8] R. Malladi, J. Sethian, and B. Vemuri, *Shape modeling with front propagation: a level set approach.* To appear in IEEE Trans. Pattern Anal. Machine Intell.

[9] S. Osher and J. Sethian, *Fronts propagation with curvature dependent speed: Algorithms based on hamilton-jacobi formulations*, Journal of Computational Physics, 56 (1988), pp. 12–49.

[10] M. Sondhi, *Reconstruction of objects from their sound-diffraction patterns*, J. Acoust. Soc. Am., 46 (1969), pp. 1158–1164.