INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 8, Number 3, Pages 410–426

A SEMI-IMPLICIT BINARY LEVEL SET METHOD FOR SOURCE RECONSTRUCTION PROBLEMS

CHUNXIAO LIU AND SHENGFENG ZHU

(Communicated by Xue-Cheng Tai)

Abstract. The aim of this paper is to investigate the application of a semi-implicit additive operator splitting scheme based binary level set method to source reconstruction problems. We reformulate the original model to be a new constrained optimization problem under the binary level set framework and solve it by the augmented Lagrangian method. Then we propose an efficient gradient-type algorithm based on the additive operator splitting scheme. The proposed algorithm can create new holes during the evolution. Topological changes can be handled automatically and complex geometry can be recovered under a certain amount of noise in the observation data. Numerical examples are presented to show the effectiveness and efficiency of our method.

Key Words. source reconstruction, binary level set method, augmented Lagrangian method, additive operator splitting.

1. Introduction

Let $U \subset \mathbb{R}^2$ be an open bounded domain and $\mathcal{K}_{ad}$ be a class of admissible compact subsets of U. Consider the following nonlinear output-least-squares problem

$$\min_{\Omega \in \mathcal{K}_{ad}} F(\Omega), \quad F(\Omega) = \frac{1}{2} \int_M |u(\Omega) - u^*|^2 \mathrm{d}x, \tag{1}$$

where the observation data u^* defined on a set $M \subset U$ or $M \subset \partial U$ typically contains noise. The relationship between u and Ω is given by the elliptic equation

$$-\Delta u = \chi_\Omega \ \text{ in } U \tag{2}$$

subject to homogeneous Dirichlet boundary conditions on ∂U, where χ_Ω is the characteristic function of Ω, i.e.,

$$\chi_\Omega = \begin{cases} 1 & \text{in } \Omega, \\ 0 & \text{in } U \setminus \Omega. \end{cases}$$

Given the noisy observation u^* of u, the aim is to find the optimal shape of Ω which satisfies the state equation (2) and fits the observation data best.

The source reconstruction problem (1) belongs to shape recovery (cf. [3, 4, 5, 6, 7, 13]), which is a very popular and challenging field of inverse problems. Other examples of shape reconstruction and identification include diffraction screen (cf. [19, 25]), identification of cavities (cf. [1, 3]), electrical impedance tomography (cf. [8, 10, 14]), etc. For solving such problems, one needs to find a mechanism to represent the shape and follow its evolution. Many effective and efficient methods

Received by the editors January 20, 2010 and, in revised form, November 26, 2010.
2000 *Mathematics Subject Classification.* 49N45, 65N21.

have been motivated. Hettlich and Rundell [13] solved an inverse source problem from measurements of the Neumann boundary values of u on ∂U. But their numerical methods based on boundary variation and shape derivative calculation [27] could only reconstruct a regular annular obstacle and failed to recover non-simply connected shapes. In [12], a phase-field method based on the Ginzburg-Landau regularization was employed for the solution of an inverse conductivity problem. Multi-connected inclusions can be recovered by the implicit representation of the shape. However, thousands of iterations were required for solving the gradient descent flow by the explicit Euler time stepping. The level set method originally proposed by Osher and Sethian [24] for interface evolution and tracking has been applied in many fields [22]. The interface is represented implicitly by the zero level set of a Lipschitz continuous function. Effective difference schemes for the Hamilton-Jacobi equation can be implemented on fixed grids. Moreover, certain types of shape and topological changes, such as merging, splitting and developing sharp corners can be handled automatically. The combination of level set methods with shape sensitivity analysis [27] has been widely applied to many shape recovery problems (see, e.g., [3, 5, 8, 9, 10, 14, 25]) and optimal shape design (see, e.g., [2, 23]). We refer to see the surveys [4, 7, 28] and the references therein. For such problems, the interface is often evolved with a given velocity obtained by calculating the shape gradient of the objective functional.

However, as pointed out in [2, 6], the shape gradient based level set method can not create new holes, which may cause the algorithm to get stuck at local shapes with fewer holes than the optimal geometry. The reconstructed results therefore largely depend on the initial guess. The topology derivative [26] proposed for hole nucleation has been combined with shape derivative in level set methods to solve inverse source problems in [6], where numerical examples demonstrated that the new algorithm can recover certain shapes that the shape gradient based level set method fails to. But the topology sensitivity analysis is generally quite complicated. Newton-type level set methods, such as the Gauss-Newton method (cf. [25]) and the Levenberg-Marquardt approach (cf. [5]), can have a decrease in the number of iterations compared with gradient-type methods, but they require the inversion of a large dense sensitivity matrix in each iteration, which is computationally slow.

Explicit schemes for the level set based gradient flow typically suffer from the Courant-Friedrichs-Lewy (CFL) stability condition [22], which means that an iterative algorithm requires rather many iterations to reach a stationary state. In order to accelerate convergence by reducing the number of iterations, the semi-implicit additive operator splitting (AOS) scheme [17, 32] was incorporated into the traditional level set method [18] and the piecewise constant level set approach [31] for efficiently solving structural topology optimization. This scheme avoids the CFL restriction and is unconditionally stable. It treats all the spatial variables in a symmetrical way. At each iteration, the computational effort for solving the tridiagonal linear systems by the fast Thomas algorithm is moderate. It has locally second order of accuracy and globally first order of accuracy.

The level set approach of binary type was firstly proposed in [16] with application to image segmentation and later employed to the solution of elliptic coefficient inverse problems [20] and shape optimization [34]. The binary level set method (BLSM) is closely related to the phase-field method as pointed out in [16]. Similar as the multiple level set method [8, 28], the BLSM also requires N level set functions to represent up to 2^N subregions. In the BLSM, however, interfaces are identified implicitly by the discontinuities of the binary level set functions (BLSFs) taking

only two values 1 and -1 at convergence. The BLSM moves the level set functions towards 1 or -1 at every mesh point rather than evolving the interface during the iteration. It enjoys a hole nucleation mechanism with no need to analyze the complex topology sensitivity during the evolution and hence is more independent of the initial guess than the level set method. Furthermore, it is more efficient since the re-initialization process is eliminated.

In this paper, we incorporate the AOS scheme into the BLSM to efficiently solve the shape reconstruction problem (1). The outline of this paper is as follows. In Section 2, basic formulations of the BLSM are given. Then a new constrained minimization problem is set up under the BLSM framework for the original model problem. In Section 3, we employ the augmented Lagrangian method, which is a robust approach for constrained optimization [21], to solve the new problem and present the Uzawa algorithm. Furthermore, the AOS scheme is employed to efficiently solve the level set evolution equation. Numerical results are given in Section 4. Finally, concluding remarks are made in Section 5.

2. Binary level set method for the model problem

Let us first present the essential formulations of the BLSM [16, 20]. Assume that a simply connected open bounded domain $\hat{\Omega} \subset \mathbb{R}^2$ is divided into two subregions Ω_1 and Ω_2 by a closed curve Γ in $\hat{\Omega}$ such that $\Omega_1 \cup \Omega_2 \cup \Gamma = \hat{\Omega}$ and $\Omega_1 \cap \Omega_2 = \emptyset$. By the traditional level set method, the subregions are represented implicitly by a Lipschitz continuous function $\phi : \hat{\Omega} \mapsto \mathbb{R}$ satisfying

$$\begin{cases} \phi(x) > 0 \ \ \forall\ x \in \Omega_1, \\ \phi(x) = 0 \ \ \forall\ x \in \Gamma, \\ \phi(x) < 0 \ \ \forall\ x \in \Omega_2. \end{cases} \tag{3}$$

Different from the level set method, the BLSM aims to identify subregions using a discontinuous BLSF ϕ defined as

$$\phi(x) = \begin{cases} 1 \ \ \forall\ x \in \Omega_1, \\ \kappa \ \ \forall\ x \in \Gamma, \\ -1 \ \ \forall\ x \in \Omega_2, \end{cases} \tag{4}$$

where $\kappa \in (-1, 1)$. Then a piecewise constant function $\rho(x)$ that equals to c_1 in Ω_1 and c_2 in Ω_2 can be expressed as

$$\rho = \frac{1}{2}\big[c_1(\phi + 1) - c_2(\phi - 1)\big]. \tag{5}$$

Following [16], we can use two BLSFs ϕ_1 and ϕ_2 to represent a function having four constant values as

$$\begin{aligned} \rho = & \frac{1}{4}\big[c_1(\phi_1 + 1)(\phi_2 + 1) - c_2(\phi_1 + 1)(\phi_2 - 1) \\ & - c_3(\phi_1 - 1)(\phi_2 + 1) + c_4(\phi_1 - 1)(\phi_2 - 1)\big]. \end{aligned} \tag{6}$$

More generally, using N BLSFs $\{\phi_j\}_{j=1}^N$, we can represent up to 2^N subregions $\{\Omega_j\}_{j=1}^{2^N}$. We introduce the following notations: Φ and $\mathbf{c}$ are vectors given by $\Phi = \{\phi_1, \phi_2, ..., \phi_N\}$ and $\mathbf{c} = \{c_1, c_2, ..., c_{2^N}\}$. For $j = 1, 2, ..., 2^N$, let $(b_1^{j-1}, b_2^{j-1}, ..., b_N^{j-1})$ be the binary representation of $j - 1$, i.e., $b_i^{j-1} = 0$ or 1. Also we define

$$s(j) = \sum_{i=1}^{N} b_i^{j-1} \quad \text{and} \quad \psi_j = \frac{1}{2^N}(-1)^{s(j)} \prod_{i=1}^{N} (\phi_i + 1 - 2b_i^{j-1}). \tag{7}$$

Any function f having up to 2^N piecewise constant values can be written as

$$f(\Phi, \mathbf{c}) = \sum_{j=1}^{2^N} c_j \psi_j(\Phi). \tag{8}$$

Then (5) and (6) are two special cases corresponding to $N = 1$ and $N = 2$, respectively.

The BLSM requires each level set function to converge to 1 or -1 at every point in the domain, i.e., $\phi_i^2 = 1\ \forall i$, which we refer as the binary constraints in the following. When these constraints are fulfilled, we have supp $(\psi_j) = \Omega_j$, $\psi_j = 1$ in Ω_j and zero elsewhere. Thus, the supports of different basis functions $\{\psi_j\}_{j=1}^{2^N}$ are non-overlapping, i.e., supp $\psi_j \cap$ supp $\psi_i = \emptyset\ \forall\ i \neq j$, and $\cup_{j=1}^{2^N}$supp $\psi_j = \hat{\Omega}$, which prevents vacuum. We can see that ψ_j is actually the characteristic function of Ω_j. Then we obtain the following formulations for calculating the length of the boundary of Ω_j and the area of Ω_j respectively as

$$|\partial\Omega_j| = \int_{\hat{\Omega}} |\nabla\psi_j| \mathrm{d}x \quad \text{and} \quad |\Omega_j| = \int_{\hat{\Omega}} \psi_j \mathrm{d}x, \tag{9}$$

where $\int_{\hat{\Omega}} |\nabla\psi_j| \mathrm{d}x$ denotes the total variation (TV) semi-norm of ψ_j. Then the total length of the interfaces of all subregions is given by

$$|\Gamma| = \frac{1}{2} \sum_{j=1}^{2^N} \int_{\hat{\Omega}} |\nabla\psi_j| \mathrm{d}x. \tag{10}$$

However, the TV term is not differentiable w.r.t. ϕ. In numerical implementation, we use the smooth approximation

$$|\nabla\psi_j| \approx \sqrt{(\partial_x\psi_j)^2 + (\partial_y\psi_j)^2 + \epsilon}, \tag{11}$$

where $\epsilon > 0$ is a small parameter and the partial differential operators ∂_x and ∂_y are approximated by finite differences. The parameter ϵ should be chosen to be small enough so as to sharply preserve discontinuities and large enough so as not to cause numerical instability.

Now we apply the binary level set formulations to our model problem (1). Then we have $N = 1$, $\hat{\Omega} = U$, $\Omega_1 = \Omega$ and $\Omega_2 = U \setminus \Omega$. Associated with the domain Ω, we introduce a BLSF ϕ satisfying

$$\phi = \begin{cases} 1 & \text{in } \Omega, \\ -1 & \text{in } U \setminus \Omega. \end{cases}$$

Therefore, the characteristic function of Ω is given by $\chi_\Omega = (\phi + 1)/2$. Then (2) is equivalent to

$$\begin{cases} -\Delta u = \frac{1}{2}(\phi + 1) & \text{in } U, \\ u = 0 & \text{on } \partial U. \end{cases} \tag{12}$$

From (10), the length of interface turns to

$$|\Gamma| = \frac{1}{2} \int_U |\nabla\phi| \mathrm{d}x. \tag{13}$$

Due to the ill-posedness of the original problem (1), we use a regularization approach and incorporate the length term (13) into the minimization functional. This is commonly known as the TV regularization term, which allows discontinuities in the recovery and restricts the length of the interface (see e.g. [9, 20, 29]).

In this paper, we study the mildly ill-posed case when $M = U$ for simplicity. The minimization problem (1) can therefore be reformulated as the following regularized constrained one w.r.t. ϕ:

$$\min_{\phi} \ \mathcal{J}(\phi) = F(\phi) + \alpha \int_U |\nabla\phi| \mathrm{d}x \quad \text{subject to } K(\phi) = 0, \tag{14}$$

where

$$F(\phi) = \frac{1}{2}\int_U |u(\phi) - u^*|^2 \mathrm{d}x \ \text{ and } \ K(\phi) = \phi^2 - 1.$$

The regularization parameter $\alpha > 0$ measures the trade-off between a good fit and a regularized solution. The selection of the regularization parameter is a research topic on its own. Some existing approaches such as the unbiased predictive risk estimator method, the discrepancy principle and the L-curve method are well-studied. For additional information on the regularization parameter selection, we refer to see the books [11, 30] and the references therein.

3. Optimization method

In this section, we will use the augmented Lagrangian method to solve the constrained minimization problem (14). This robust technique has shown effective for solving many kinds of inverse problems (see, e.g., [8, 9, 15, 20]). A gradient-type algorithm is proposed. We use the semi-implicit AOS scheme rather than explicit scheme to solve the evolution equation for speeding up convergence.

To solve (14), the Fréchet derivative of $\mathcal{J}$ w.r.t. ϕ denoted by $\mathcal{J}'(\phi)$ is required. The weak formulation of (2) is to find $u \in H_0^1(U) \equiv \{u \in L^2(U) | \nabla u \in L^2(U), u = 0 \text{ on } \partial U\}$ such that

$$\int_U \nabla u \cdot \nabla v \mathrm{d}x = \frac{1}{2}\int_U (\phi + 1) v \mathrm{d}x \quad \forall v \in H_0^1(U). \tag{15}$$

Let us assume that the state variable u is Fréchet differential at ϕ. Differentiating u on both sides of (15) w.r.t. ϕ in a direction $\psi \in C_0^\infty(U)$, we obtain

$$\int_U \nabla\big(u'(\phi)\psi\big) \cdot \nabla v \mathrm{d}x = \int_U \frac{1}{2}\psi v \mathrm{d}x \quad \forall v \in H_0^1(U). \tag{16}$$

Let us introduce the adjoint problem of (12) as

$$\begin{cases} -\Delta z = u - u^* & \text{in } U, \\ z = 0 & \text{on } \partial U. \end{cases} \tag{17}$$

Then the weak formulation of (17) reads: Find $z \in H_0^1(U)$ such that

$$\int_U \nabla z \cdot \nabla w \mathrm{d}x = \int_U (u - u^*) w \mathrm{d}x, \quad \forall w \in H_0^1(U). \tag{18}$$

On the other hand, we have

$$\begin{aligned} F'(\phi)\psi &= \int_U (u - u^*) u'(\phi)\psi \mathrm{d}x \\ &= \int_U \nabla z \cdot \nabla\big(u'(\phi)\psi\big) \mathrm{d}x = \int_U \frac{1}{2} z\psi \mathrm{d}x, \end{aligned} \tag{19}$$

where in the last two equalities we have used (18) and (16) by setting $w = u'(\phi)\psi$ and $v = z$, respectively.

Thus, we obtain from (19)

$$F'(\phi) = \frac{1}{2} z. \tag{20}$$

Denote $R(\phi) := \int_U |\nabla\phi| \mathrm{d}x$, then the derivative of $R(\phi)$ at ϕ in the direction ψ reads:

$$\begin{aligned} R'(\phi)\psi &= \frac{\mathrm{d}}{\mathrm{dt}} R(\phi + t\psi)\Big|_{t=0} \\ &= \int_U \frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla\psi \mathrm{d}x = -\int_U \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right)\psi \mathrm{d}x, \end{aligned}$$

which gives

$$R'(\phi) = -\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right). \tag{21}$$

Then the Fréchet derivative of $\mathcal{J}(\phi)$ w.r.t. ϕ is given by combining (20) and (21):

$$\mathcal{J}'(\phi) = \frac{1}{2}z - \alpha\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right). \tag{22}$$

By the augmented Lagrangian method, the constrained optimization problem (14) is transformed to an unconstrained one as follows:

$$\min_{\phi}\max_{\lambda} \quad \mathcal{L}_\mu(\phi, \lambda) = \mathcal{J}(\phi) + \int_U \lambda K(\phi) \mathrm{d}x + \frac{\mu}{2}\int_U K^2(\phi) \mathrm{d}x, \tag{23}$$

where the Lagrange multiplier $\lambda \in L^2(U)$. The penalization parameter $\mu > 0$ can be fixed as a large number or increased slowly with a small initial value. Maximizing w.r.t. λ, the constraint must be fulfilled and we get the minimization of the original problem.

At a saddle point of $\mathcal{L}_\mu$, we have

$$\begin{aligned} 0 = \frac{\partial \mathcal{L}_\mu}{\partial \phi} &= \mathcal{J}'(\phi) + \lambda K'(\phi) + \mu K'(\phi)K(\phi) \\ &= \frac{1}{2}z - \alpha\nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|}\right) + 2\lambda\phi + 2\mu\phi(\phi^2 - 1), \\ 0 = \frac{\partial \mathcal{L}_\mu}{\partial \lambda} &= K(\phi). \end{aligned} \tag{24}$$

We use an iterative algorithm to solve the nonlinear system (24). Denote by ϕ^k and λ^k the kth ($k = 0, 1, 2, \cdots$) iterations for ϕ and λ, respectively. Starting from the initial guesses ϕ^0 and λ^0, we get ϕ^{k+1} and λ^{k+1} from ϕ^k and λ^k successively.

First, we consider the updating of ϕ for a fixed λ^k. We introduce an artificial time variable $t \geqslant 0$ and solve the gradient descent flow

$$\begin{cases} \dfrac{\partial\phi}{\partial t}(x,t) = -\dfrac{\partial \mathcal{L}_\mu}{\partial\phi}(\phi, \lambda^k) & \text{in } U \times \mathbb{R}^+, \\ \phi(x,0) = \phi^0(x) & \text{in } U \end{cases} \tag{25}$$

to the steady state. Then we get $\partial\phi/\partial t = 0$, which implies that $\partial\mathcal{L}_\mu/\partial\phi = 0$. In numerical implementation, we can discretize (25) by the explicit Euler scheme as follows

$$\phi^{k+1} = \phi^k - \Delta t^k \frac{\partial \mathcal{L}_\mu}{\partial\phi}(\phi^k, \lambda^k), \tag{26}$$

where $\Delta t^k > 0$ is the time step. We can use a line search method to find the optimal step size in each iteration. Alternatively, we may choose a small fixed time step Δt by the trial and error method considering time consuming of the line search strategy for PDE constraint optimziation.

If we use an explicit scheme such as (26), the time step must be small enough to satisfy the CFL condition for stability. Then it requires rather many iterative steps

to solve (25) to a steady state. In this work, we use a semi-implicit AOS scheme to solve (25) for reducing the number of iterations required using explicit schemes.

The governing equation of (25) can be written as

$$\frac{\partial \phi}{\partial t} = \alpha \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) + \mathscr{F}(\phi), \tag{27}$$

where

$$\mathscr{F}(\phi) = -\frac{1}{2} z - 2\lambda\phi - 2\mu\phi(\phi^2 - 1).$$

We consider the semi-implicit discretization of (27). Assume that a rectangular domain U is divided by a uniform mesh of spacing h in both spatial directions. The grid points are arranged as a vector $[x_1, \cdots, x_{N_p}]^{\mathrm{T}}$, where N_p is the number of grid points. Let Δt be the temporal step size. Denote by ϕ_i^k the approximation of $\phi(x,t)$ at (x_i, t_k) with $t_k = k\Delta t$. The half-point difference and linear interpolation give the following semi-implicit scheme

$$\frac{\phi_i^{k+1} - \phi_i^k}{\Delta t} = \alpha \sum_{j \in \mathcal{N}(i)} \frac{\left(\frac{1}{|\nabla \phi|}\right)_j^k + \left(\frac{1}{|\nabla \phi|}\right)_i^k}{2} \frac{\phi_j^{k+1} - \phi_i^{k+1}}{h^2} + \mathscr{F}(\phi_i^k), \tag{28}$$

where $\mathcal{N}(i)$ denotes the four neighbors of the grid point x_i. It should be noted that we have used the homogeneous Neumann boundary condition for (27). The domain U is extended such that (28) is well defined for the original boundary points. Here, straightforward finite difference implementations would give rise to problems when $|\nabla u|$ vanishes in the 4-neighborhood. These problems can be avoided if we replace $\frac{1}{2}\left(\left(\frac{1}{|\nabla \phi|}\right)_j^k + \left(\frac{1}{|\nabla \phi|}\right)_i^k\right)$ in (28) by its harmonic counterpart:

$$\begin{aligned} \frac{\phi_i^{k+1} - \phi_i^k}{\Delta t} &= \alpha \sum_{j \in \mathcal{N}(i)} \frac{2}{(|\nabla \phi|)_j^k + (|\nabla \phi|)_i^k} \frac{\phi_j^{k+1} - \phi_i^{k+1}}{h^2} + \mathscr{F}(\phi_i^k) \\ &= \alpha \sum_{l \in \{x,y\}} \sum_{j \in \mathcal{N}_l(i)} \frac{2}{(|\nabla \phi|)_j^k + (|\nabla \phi|)_i^k} \frac{\phi_j^{k+1} - \phi_i^{k+1}}{h^2} + \mathscr{F}(\phi_i^k), \end{aligned} \tag{29}$$

where $\mathcal{N}_l(i)$ denotes the two neighbors of x_i in the direction l. Denote by $\mathscr{A}_x$ and $\mathscr{A}_y$ the one dimensional operators in the direction x and y, respectively. Then the discrete semi-implicit scheme (29) can be rewritten compactly in matrix-vector notation as

$$\frac{\phi^{k+1} - \phi^k}{\Delta t} = \sum_{l \in \{x,y\}} \mathscr{A}_l(\phi^k)\phi^{k+1} + \mathscr{F}(\phi^k), \tag{30}$$

where the matrix $\mathscr{A}_l(\phi^k) = (a_{ijl}(\phi^k))$ with

$$a_{ijl}(\phi^k) = \begin{cases} \dfrac{2\alpha}{h^2\left[(|\nabla \phi|)_j^n + (|\nabla \phi|)_i^n\right]} & j \in \mathcal{N}_l(i), \\ -\displaystyle\sum_{j \in \mathcal{N}_l(i)} \dfrac{2\alpha}{h^2\left[(|\nabla \phi|)_j^n + (|\nabla \phi|)_i^n\right]} & j = i, \\ 0 & \text{else.} \end{cases}$$

The AOS scheme then solves the following equations in parallel

$$\frac{\phi_l^{k+1} - \phi^k}{\Delta t} = 2\mathscr{A}_l(\phi^k)\phi_l^{k+1} + \mathscr{F}(\phi^k),\ l \in \{x,y\}, \tag{31}$$

which leads to two linear systems

$$\left[I - 2\Delta t\mathscr{A}_l(\phi^k)\right]\phi_l^{k+1} = \phi^k + \Delta t\mathscr{F}(\phi^k),\ l \in \{x, y\}. \tag{32}$$

Either tridiagonal coefficient matrix of the two systems in (32) is strictly diagonally dominant. Therefore, we can solve (32) efficiently by the Thomas algorithm and obtain the solutions

$$\phi_l^{k+1} = \left[I - 2\Delta t\mathscr{A}_l(\phi^k)\right]^{-1}\left[\phi^k + \Delta t\mathscr{F}(\phi^k)\right],\ l \in \{x, y\}. \tag{33}$$

Then the AOS scheme determines the final solution ϕ^{k+1} by

$$\phi^{k+1} = \frac{1}{2}\left(\phi_x^{k+1} + \phi_y^{k+1}\right). \tag{34}$$

Secondly, the updating scheme for the Lagrange multiplier λ follows the Uzawa algorithm [16]:

$$\lambda^{k+1} = \lambda^k + \mu^k K(\phi^k). \tag{35}$$

The penalty parameter μ can be set as a large constant during the iterations. To improve the convergence, a continuation in μ is adopted. We choose to update μ by

$$\mu^{k+1} = \eta\mu^k \tag{36}$$

with a small initial $\mu^0 > 0$, where the constant η is slightly larger than 1. To ensure stability, the penalty parameter should be limited by a prescribed upper bound $\mu_{\max}$.

Unifying all the above schemes, we present the following algorithm.

Algorithm 1 Semi-implicit AOS scheme based Uzawa algorithm

Set $k = 0$. Initialization: ϕ^0, λ^0, μ^0.
- Compute the state u^k by solving (12).
- Compute the adjoint z^k by solving (17).
- Compute $\partial\mathcal{L}_\mu/\partial\phi$ by (24).
- Update ϕ^k by solving (25) using the AOS scheme.
- Update λ by (35).
- If $\mu^k \leqslant \mu_{\max}$, update μ by (36); else $\mu^k = \mu_{\max}$.
- Iterate again if necessary; $k = k + 1$.

The binary constraint $K(\phi) = 0$ can also be treated with the simple penalty method. But it is hard to satisfy the constraint approximately unless the penalty parameter is set sufficiently large. Too large values of it will severely restrict the time step and may cause the algorithm unstable. By the augmented Lagrangian method, however, there is no need to set the penalty parameter very large to guarantee the convergence of the algorithm and the constraint can be satisfied very well. We choose a small μ^0 and increase it in a continuation way. The value of μ is small at the first stage of the evolution before the optimal topology of Ω is determined. With increasing μ, the weight for the constraint becomes larger and ϕ is forced to be 1 and -1 at every grid point in the domain.

4. Numerical results

We will present three numerical examples to illustrate the performance of the proposed algorithm with MATLAB. We choose the computational domain $U = (-1, 1)\times(-1, 1)$ and divide it into a uniform rectangular mesh with 64×64 elements. At each iteration, both of the state equation and adjoint problem are solved by the linear finite element approximation. We set $\epsilon = 10^{-3}$ by trial and error.

To test the effectiveness of our algorithm, we add noise to the exact finite element grid function u and obtain the observed data u^* as in [29], i.e.,

$$u^* = u + \sigma \frac{\|u\|_{L^2(U)}}{\|r^*\|_{L^2(U)}} r^*,$$

where σ is the noise level and r^* is a grid function with nodal values being the uniform random distribution in $[-1, 1]$ with zero mean value. Without any a-priori knowledge about the topology and shape to be reconstructed, the grid function associated with ϕ^0 is always chosen to be -1, i.e., the initial guess for Ω is an empty set. All two-dimensional illustrations of obstacle shapes in the following are shown with zero contour (i.e., $\kappa = 0$) as the interface. When the level set function converges to 1 and -1, there are four possible cases for the cells where the interface cuts through as shown in Fig. 1. This minor defect on the sub-cell resolution of the interface is a common problem in level set approaches of piecewise constant type while the traditional level set method does not have [33, 34]. Our algorithm leads to a zigzag interface compared to a relatively smooth interface obtained by the continuous level set method.

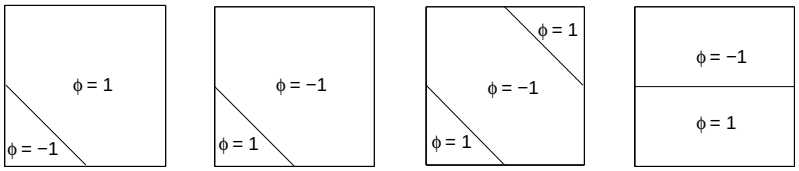


FIGURE 1. Four possible cases for cells straddling the interface by the MATLAB subrutine "contourf".

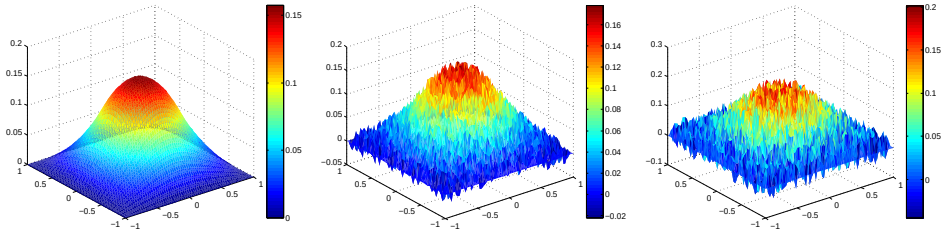
`Example 1:` We start with a simply connected example. The shape we want to reconstruct is a circle of radius 0.5 centered at the origin. The true shape is shown by the red dashed line in each reconstruction figure for comparison. We set $\alpha = 10^{-6}$, $\Delta t = 3$, $\mu^0 = 10^{-4}$, $\mu_{\max} = 10^{-1}$ and $\eta = 1.1$. In Fig. 2, we plot the true finite element solution u and its two random noisy observations. We show the converged level set function and the recovered shape for $\sigma = 0$ in Fig. 3, from which we see that the reconstructed shape matches the exact one very well. Then we present two reconstructed shapes in Fig. 4 with noise levels 20% and 40% to show the algorithm is not sensitive to noise for this simple example. The quantitative illustration is given in Fig. 5, which gives the curves of the residual, the L^1-error (i.e., L^1 distance between the indicator function of the computed shape and that of the exact shape) and the L^2 norm of $K(\phi)$.

Furthermore, we compare our algorithm with an explicit scheme (26) based algorithm for this example. To satisfy the CFL stability condition, we choose the time step as in [33]

$$\Delta t^k = ch \Big/ \left\| \frac{\partial \mathcal{L}_\mu}{\partial \phi}(\phi^k, \lambda^k, \mu^k) \right\|_{L^\infty}, \tag{37}$$

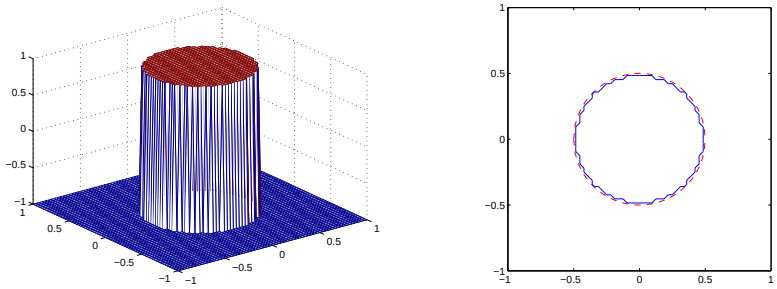
where $c > 0$ is a constant chosen by trial and error. Here we set $c = 0.6$. The comparison curves on the residual and L^1-error are plotted in Fig. 6 to illustrate the superiority of the semi-implicit iterative scheme. To achieve almost the same recovered results for $\sigma = 0$ and 40%, the explicit scheme requires more iterations and hence converges slower than our semi-implicit scheme as expected.

`Example 2:` This example is more difficult than Example 1 in shape recovery due to the multi-connectivity of the true domain $U \setminus \Omega$. The exact shape is a circle



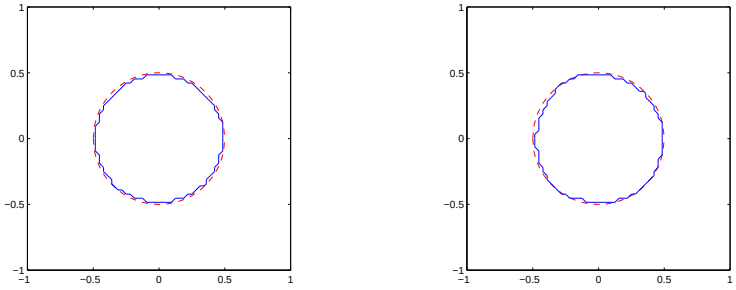
(a) True u (b) Observing u^* with $\sigma = 20\%$ (c) Observing u^* with $\sigma = 40\%$

FIGURE 2. Example 1: The true u and observation u^* with $\sigma = 20\%$ and 40%.



(a) Computed level set function (b) The discontinuity of ϕ

FIGURE 3. Example 1: The computed level set function and its discontinuity for $\sigma = 0$ after 150 iterations.

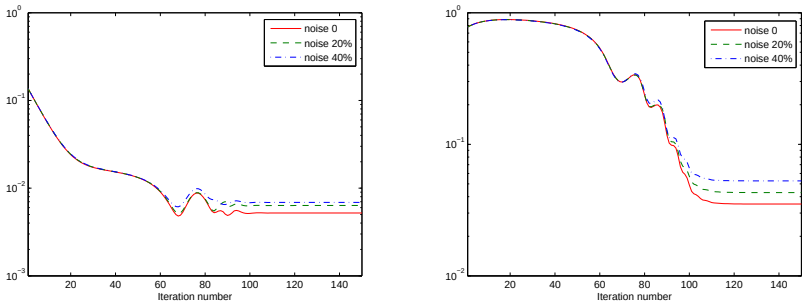


(a) σ= 20% (b) σ= 40%

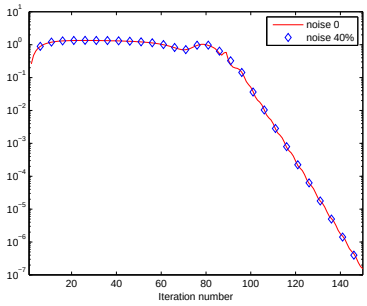
FIGURE 4. Example 1: The reconstructed shapes with $\sigma = 20\%$ and 40% after 150 iterations.

of radius 0.5 including a hole of radius 0.3, both centered at the origin. The exact finite element solution and two noisy observations are plotted in Fig. 7.

As shown in [6], the level set algorithm based on shape gradient can only recover the outer contour of the exact shape. For this example, here we show that our algorithm can lead to similar reconstruction quality as the shape-topology gradient



(a) Residual (b) L^1-Error



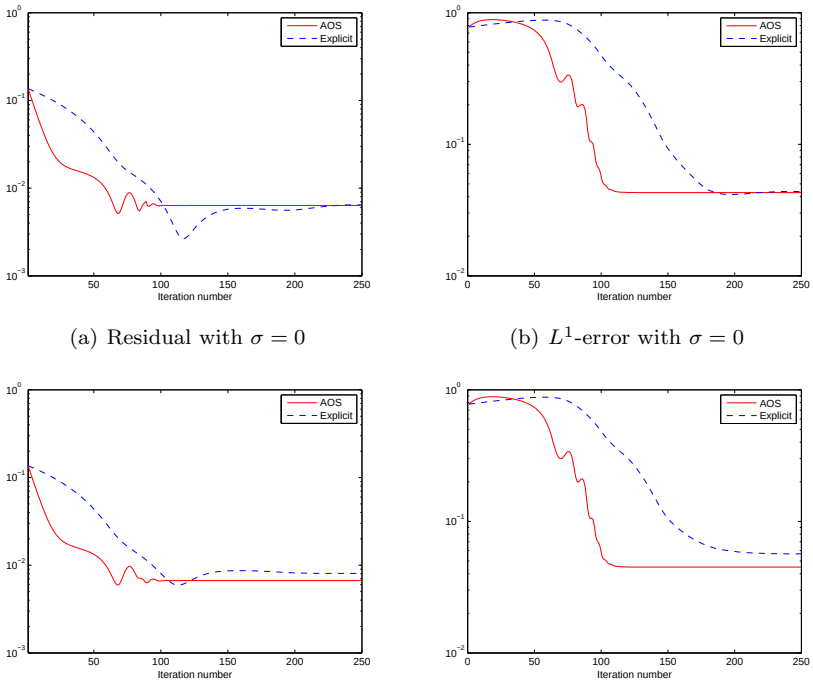
(c) $\|K(\phi)\|_{L^2(U)}$

FIGURE 5. Example 1: Convergence of residual, L^1-Error and $\|K(\phi)\|_{L^2(U)}$ for $\sigma = 0$, 20% and 40%.

based level set method [6]. We choose $\alpha = 10^{-6}$, $\Delta t = 5$, $\mu^0 = 10^{-7}$, $\mu_{\max} = 10^{-2}$ and $\eta = 1.1$. First, we present four intermediate iterations for $\sigma = 0$ in Fig. 8 to illustrate the topological changes during the evolution process. Then we plot in Fig. 9 the final computed level set function and its discontinuity. The reconstructed shapes with different noise levels are also shown in Fig. 10. We can see that our method can recover the inner contour of the exact shape. Compared with the reconstruction results in Example 1, we find that it can tolerate less noise than that in Example 1 as the geometry becomes more complex, which is to be expected. See Fig. 11 for the corresponding convergence history of the residual and L^1-error. Choose $c = 0.5$. Fig. 12 shows that the AOS scheme converges faster than the explicit scheme.

`Example 3:` Then we consider an example with Ω consists of two components: a circle and an ellipse. We set $\alpha = 10^{-6}$, $\Delta t = 5$, $\mu^0 = 10^{-6}$, $\mu_{\max} = 10^{-2}$ and $\eta = 1.05$. We first illustrate the evolution process with 5% noise in Fig. 13. See Fig. 14 and Fig. 15 for final reconstructions and curves of residual and L^1-error with different amounts of noise.

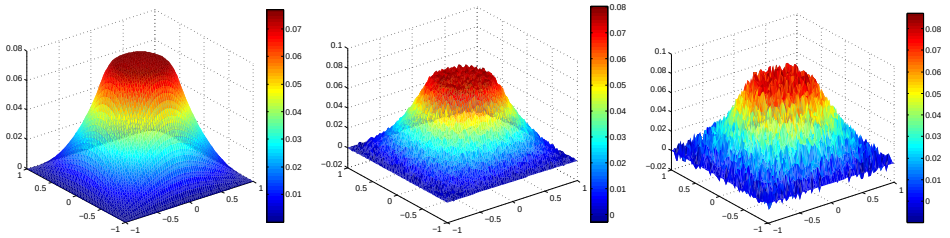
Finally, we investigate the influence of the regularization parameter. The same parameters have been used except for α. We set $\sigma = 15\%$. In Fig. 16, we plot the final reconstruction results for different choices of α. We can observe that a too small value of α ($=10^{-10}$) gives an oscillating curve and a larger value can lead to a smoother shape. However, a too high value ($\alpha = 10^{-4}$) will prevent shape splitting and the multi-connected obstacles are failed to recovered.



(a) Residual with $\sigma = 0$ (b) L^1-error with $\sigma = 0$

(c) Residual with $\sigma = 40\%$ (d) L^1-error with $\sigma = 40\%$

FIGURE 6. Example 1: Comparison of the AOS scheme and an explicit scheme on the convergence history of the residual and L^1-error.

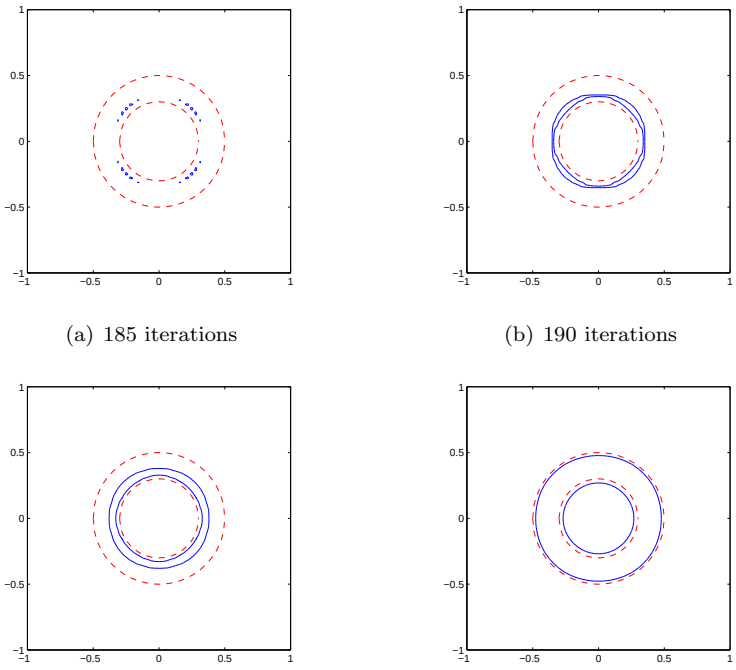


(a) True u (b) Observing u^* with $\sigma = 5\%$ (c) Observing u^* with $\sigma = 15\%$

FIGURE 7. Example 2: The true u and observation u^* with $\sigma = 5\%$ and 15%.

5. Conclusions

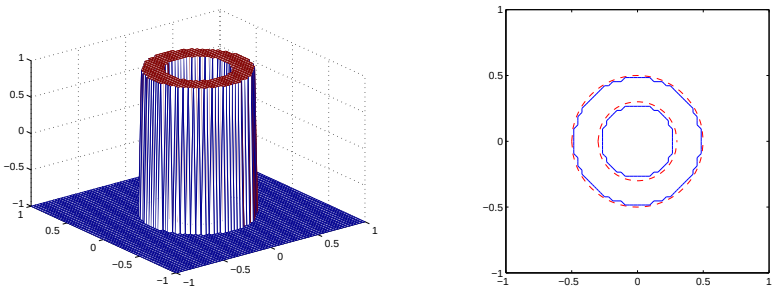
In this work, we have applied the BLSM to the source reconstruction problem. By embedding one level set function into the original problem, we reformulated it to be a new constrained optimization problem w.r.t. the level set function. Then we employed the augmented Lagrangian method to solve it. In numerical implementation, the semi-implicit AOS scheme was applied to relax the time step restriction, which improved the efficiency of the proposed gradient-type algorithm. As demonstrated in numerical experiments, our algorithm can handle complex topological



(a) 185 iterations (b) 190 iterations

(c) 200 iterations (d) 420 iterations

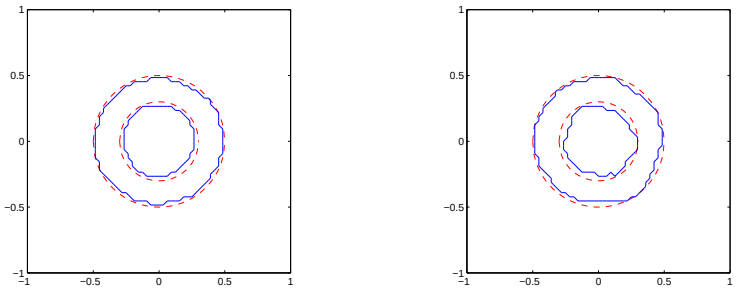
FIGURE 8. Example 2: The evolution process with $\sigma = 0$.



(a) Computed level set function (b) The discontinuity of ϕ

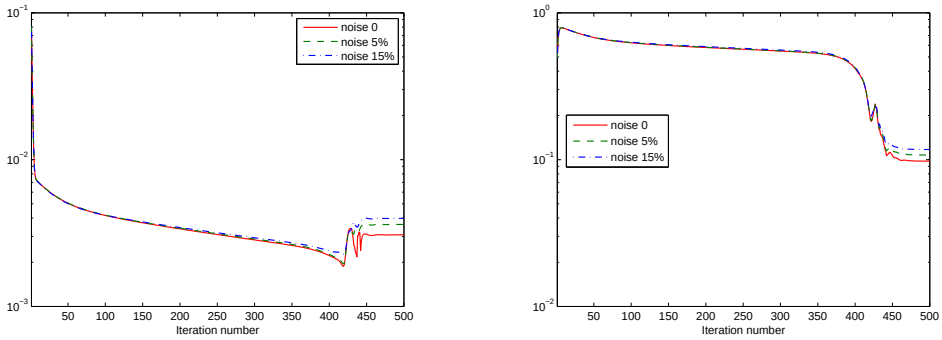
FIGURE 9. The computed level set function and its discontinuity with $\sigma = 0$ after 500 iterations.

changes automatically and largely avoid getting stuck at local minimizers during the evolution. We can achieve good reconstruction quality for the shape that the traditional level set method fails to. Moreover, reconstructions from observations with different noise levels were compared to show the robustness of our algorithm for the mildly ill-posed model problem. We will try to use the AOS and other efficient semi-implicit schemes to solve the gradient descent flows for more ill-posed nonlinear shape recovery problems from boundary measurements.



(a) σ= 5% (b) σ= 15%

FIGURE 10. Example 2: The reconstructed shapes for $\sigma = 5\%$ and 15%.

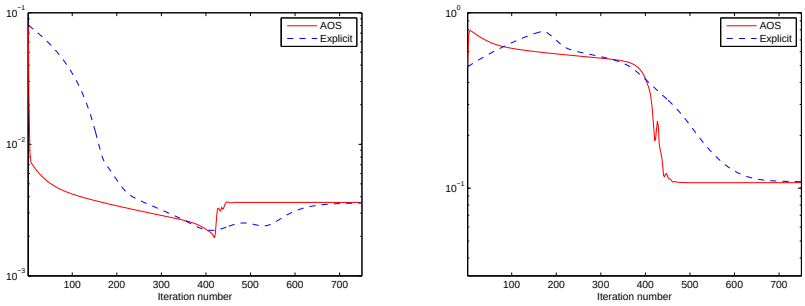


(a) Residual (b) L^1-Error

FIGURE 11. Example 2: Convergence of residual and L^1-Error for $\sigma = 0$, 5% and 15%.

References

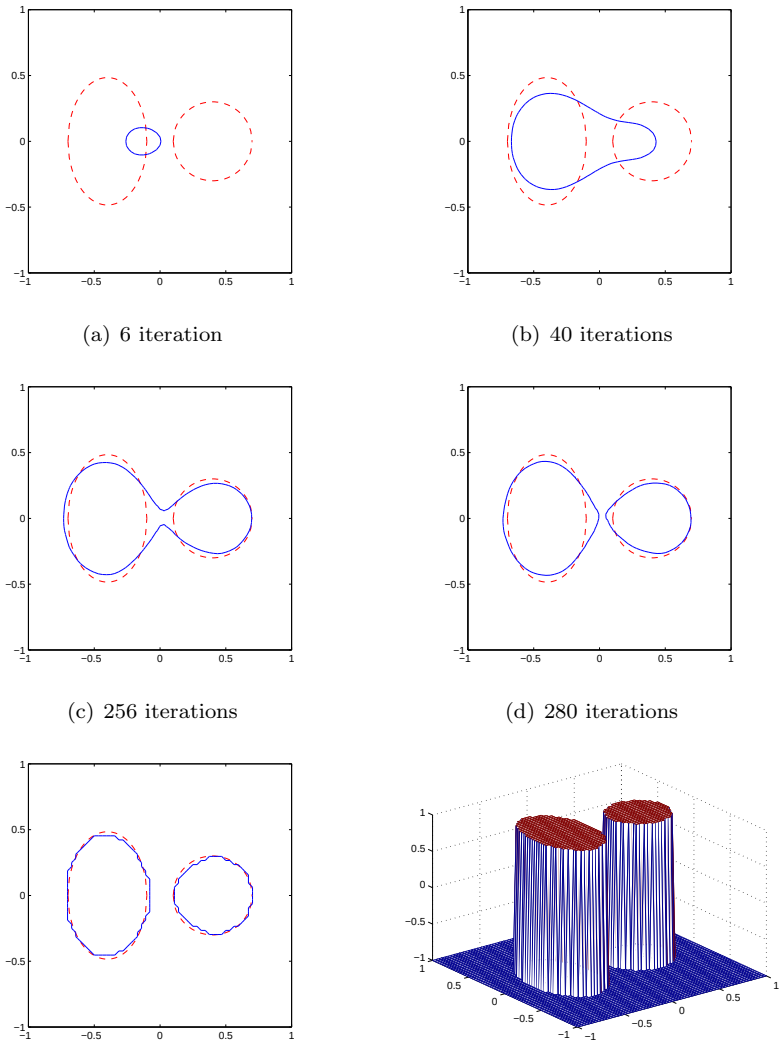
[1] G. Alessandrini and L. Rondi, Optimal stability for the inverse problem of multiple cavities, J. Diff. Eqns., 176 (2001) 356-386.
[2] G. Allaire, F. Jouve and A. Toader, Structural optimization using sensitivity analysis and a level-set method, J. Comput. Phys., 194 (2004) 363-393.
[3] M. Burger, A level set method for inverse problems, Inverse Problems, 17 (2001) 1327-1355.
[4] M. Burger, A framework for the construction of level set method for shape optimization and reconstruction, Interfaces and Free Boundaries, 5 (2003) 301-329.
[5] M. Burger, Levenberg-Marquardt level set methods for inverse obstacle problems, Inverse Problems, 20 (2004) 259-282.
[6] M. Burger, B. Hackl and W. Ring, Incorporating topological derivatives into level set methods, J. Comput. Phys., 194 (2004) 344-362.
[7] M. Burger and S. Osher, A survey on level set methods for inverse problems and optimal design, Eur. J. Appl. Math., 16 (2005) 263-301.
[8] 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) 40-66.
[9] T.F. Chan and X.-C. Tai, Identification of discontinuous coefficients from elliptic problems using total variation regularization, SIAM J. Sci. Comput., 25 (2003) 881-904.
[10] E.T. Chung, T.F. Chan and X.-C. Tai, Electrical impedance tomography using level set representation and total variational regularization, J. Comput. Phys., 205 (2005) 357-372.



(a) Residual with $\sigma = 5\%$ (b) L^1-error with $\sigma = 5\%$

FIGURE 12. Example 2: Comparison of the AOS scheme and an explicit scheme on the convergence history of the residual and L^1-error.

[11] H. Engl, M. Hanke, and A. Neubauer, Regularization of Inverse Problems, Kluwer, Dordrecht, 1996.
[12] F. Frühauf and H. Grossauer, Solving constraint ill-posed problems using Ginzburg-Landau regularization functionals, J. Inv. Ill-Posed Problems, 16 (2008) 35-49.
[13] F. Hettlich, W. Rundell, Iterative methods for the reconstruction of an inverse potential problem, Inverse Problems, 12 (1996) 251-266.
[14] K. Ito, K. Kunisch and Z. Li, Level-set function approach to an inverse interface problem, Inverse Problems, 17 (2001) 1225-1242.
[15] K. Kunisch and X.-C. Tai, Sequential and parallel splitting methods for bilinear control problems in Hilbert spaces, SIAM J. Numer. Anal., 34 (1997) 91-118.
[16] J. Lie, M. Lysaker and X.-C. Tai, A binary level set model and some applications to Mumford-Shah image segmentation, IEEE Trans. Image Process., 15 (2006) 1171-1181.
[17] T. Lu, P. Neittaanmäki and X.-C. Tai, A parallel splitting up method and its application to Navier-Stokes equations, Appl. Math. Lett., 4 (1991) 25-29.
[18] J. Luo, Z. Luo, L. Chen, L. Tong and M.Y. Wang, A semi-implicit level set method for structural shape and topology optimization, J. Comput. Phys., 227 (2008) 5561-5581.
[19] R. Magnanini and G. Papi, An inverse problem for the Helmholtz equation, Inverse Problems, 1 (1985) 357-370.
[20] L.K. Nielsen, X.-C. Tai, S.I. Aanonsen and M. Espedal, A binary level set model for elliptic inverse problems with discontinuous coefficients, Int. J. Numer. Anal. Mod., 4 (2007) 74-99.
[21] J. Nocedal and S. Wright, Numerical Optimization, Springer-Verlag, New York, 1999.
[22] S. Osher and R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer, 2003.
[23] 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) 272-288.
[24] S. Osher and J.A. Sethian, Fronts propagation with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations, J. Comput. Phys., 79 (1988) 12-49.
[25] F. Santosa, A level-set approach for inverse problems involving obstacles, ESAIM: Control Optim. Calc. Var., 1 (1996) 17-33.
[26] J. Sokołowski and A. Żochowski, On the topological derivative in shape optimization, SIAM J. Control Optim., 37 (1999) 1251-1272.
[27] J. Sokołowski and J.-P. Zolesio, Introduction to Shape Optimization: Shape Sensitivity Analysis, Springer, Heidelberg, 1992.
[28] X.-C. Tai and T. F. Chan, A survey on multiple level set methods with applications for identifying piecewise constant functions, Int. J. Numer. Anal. Mod., 1 (2004) 25-48.
[29] X.-C. Tai and H. Li, A piecewise constant level set method for elliptic inverse problems, Appl. Numer. Math., 57 (2007) 686-696.
[30] C. Vogel, Computational Methods for Inverse Problems, SIAM, Philadelphia, 2002.
[31] P. Wei and M.Y. Wang, Piecewise constant level set method for structural topology optimization, Int. J. Numer. Meth. Eng., 78 (2009) 379-402.



(a) 6 iteration

(b) 40 iterations

(c) 256 iterations

(d) 280 iterations

(e) 600 iterations

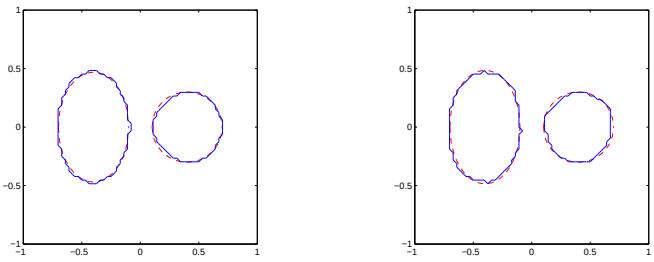
(f) The final computed level set function

FIGURE 13. Example 3: The evolution process with $\sigma = 5\%$.

[32] J. Weickert, B.H. Romeny and M.A. Vergever, Efficient and reliable schemes for nonlinear diffusion filtering, IEEE Trans. Image Process., 7 (1998) 398-409.
[33] S. Zhu, Q. Wu, C. Liu, Variational piecewise constant level set methods for shape optimization of a two-density drum, J. Comput. Phys., 229 (2010) 5062-5089.
[34] S. Zhu, C. Liu, Q. Wu, Binary level set methods for topology and shape optimization of a two-density inhomogeneous drum, Comput. Methods Appl. Mech. Eng., 199 (2010) 2970-2986.

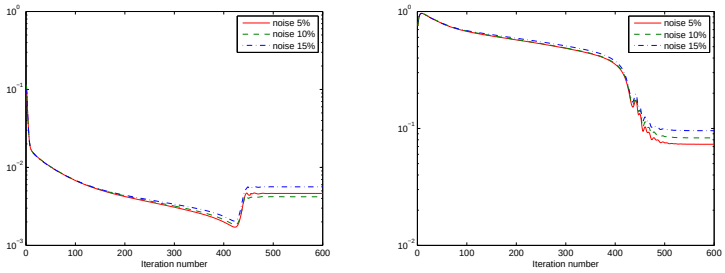
Center of Mathematical Sciences, Zhejiang University, Hangzhou 310027, Zhejiang, P.R. China
E-mail: xxliu198431@126.com

Department of Mathematics, Zhejiang University, Hangzhou 310027, Zhejiang, P.R. China
E-mail: shengfengzhu@zju.edu.cn



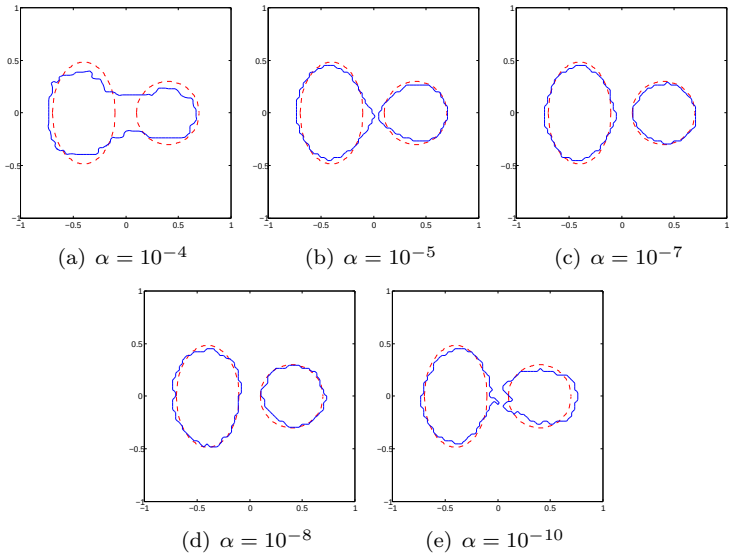
(a) σ= 10%, $\alpha = 10^{-7}$ (b) σ= 15%, $\alpha = 10^{-7}$

FIGURE 14. Example 3: The reconstructed optimal shapes for $\sigma =$ 10% and 15%.



(a) Residual (b) L^1-Error

FIGURE 15. Example 3: Convergence of residual and L^1-Error for $\sigma = 5\%$, 10% and 15%.



(a) $\alpha = 10^{-4}$ (b) $\alpha = 10^{-5}$ (c) $\alpha = 10^{-7}$

(d) $\alpha = 10^{-8}$ (e) $\alpha = 10^{-10}$

FIGURE 16. Example 3: The effect of regularization parameter on the reconstruction with $\sigma = 15\%$.