UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Identification of Discontinuous Coefficients from Elliptic Problems Using Total Variation Regularization

Tony F. Chan
Xue-Cheng Tai

August 1997

CAM Report 97-35

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90095-1555

IDENTIFICATION OF DISCONTINUOUS COEFFICIENTS FROM ELLIPTIC PROBLEMS USING TOTAL VARIATION REGULARIZATION

TONY F. CHAN * AND XUE-CHENG TAI†

Abstract. We propose several formulations for recovering discontinous coefficient of elliptic problems by using total variation (TV) regularization. The motivation for using TV is its well-established ability to recover sharp discontinuities. We employ an augmented Lagrangian variational formulation for solving the output-least-squares inverse problem. In addition to the basic output-least-squares formulation, we introduce two new techniques to handle large observation errors. First, we use a filtering step to remove as much of the observation error as possible. Second, we introduce two extensions of the output-least-squares model; one model employs observations of the gradient of the state variable while the other utilizes the flux. Numerical experiments indicate that the combination of these two techniques enables us to successfully recover highly discontinous coefficient even under observation errors as high as 100% in the L^2 norm.

1. Introduction. Consider the partial differential equations

$$\begin{cases} -\nabla \cdot (q(x)\nabla u) = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases} \tag{1}$$

$$\begin{cases} u_t - \nabla \cdot (q(x)\nabla u) = f(x,t) & \text{in } \Omega \times [0,T], \\ u = 0 & \text{on } \partial\Omega, \\ u(x,0) = u_0(x) & \text{in } \Omega. \end{cases} \tag{2}$$

We want to use observations of the solution u to recover the coefficient $q(x)$. We shall especially treat the case that $q(x)$ has discontinuities. This problem arises in many industrial applications, for example, reservoir and underground water investigations, [8], [20], [2]. Such inverse problems are ill-posed. The common approach is to use output-least-squares method with certain regularization techniques. When the coefficient $q(x)$ is assumed to be smooth, successful numerical methods are available, see Ito and Kunisch [11], and Kunisch and Tai [16]. However, when the coefficients have discontinuities, the usual regularization methods will smear out the sharp jumps or introduce Gibbs oscillations at the locations of the discontinuities and thus prevent us from getting accurate estimations; see Lin and Ramirez[17] for some one dimensional numerical results with H^1–norm regularizations.

In this work, we shall use a special regularization technique, namely, using the total variation (TV) norms to regularize the coefficient. This will allow the coefficient to have jumps and at the same time will discourage the oscillations that normally appear in the computations. We note that the TV-norm regularization has been used successfully in image processing [1], [4], [19], [18], [21]. For recent works that use TV-norm regularization techniques to identify discontinuous diffusion coefficient, we refer to [22], [6], [10].

We shall consider three kinds of observation data for identifying the coefficient:

(1) Observation for the solution u is available at some points.

(2) Observations for the solution u and its gradient ∇u are available at some points.

(3) Observations for the solution u and the velocity (or flux) $q\nabla u$ are available at some points.

* Department of Mathematics, University of California, Los Angeles, 405 Hilgard Avenue, Los Angeles, CA 90095-1555. Email: chan@math.ucla.edu, URL: http://www.math.ucla.edu/~chan. The work of this author is partially supported by the Office of Naval Research under contract ONR N00014-96-1-0277 and by the National Science Foundation under grant DMS-9626755.

† Department of Mathematics, University of Bergen, Alleg. 55, N-5007 Bergen, Norway. Email: Tai@mi.uib.no. URL: http://www.mi.uib.no/~tai. The work of this author is supported by the Research Council of Norway, under contract SEP-115837/431.

Except for some special situations, see [12], we must have enough many points of observations in order to have a good estimation for $q(x)$. Therefore, we shall assume that we have observations at sufficiently many points to recover all the detailed structures of the coefficient $q(x)$. Otherwise, we can only expect to recover some approximate structures of the coefficient $q(x)$.

We employ an augmented Lagrangian variational formulation for solving the output-least-squares inverse problem, by which the inverse problem is transformed into a nonlinear minimization problem in both u and q. This approach has been quite successful in inverse parameter estimation problems with smooth coefficient; see for example [11], [16]. In particular, the efficiency is usually much higher than other optimization techniques such as Newton or Levenberg-Marquardt methods because the augmented Lagrangian technique can exploit the special bilinear nature of the equation constraint.

As mentioned earlier, the identification problem is ill-posed and in numerical simulations, a small observation error can produce a large computational error. Even in the case of very little observation error, oscillations can appear in the estimated coefficient. Most of the works in the literature deal with *small* observation errors or use very *coarse* mesh to reduce the instability and oscillatory effect of the observation errors. In this work, we shall introduce several special techniques for treating large observation errors. First, we use a filtering step to remove as much of the observation error as possible. Second, we introduce two extensions of the output-least-squares model; one model employs observations of the gradient of the state variable while the other utilizes the flux. Numerical experiments indicate that the combination of these two techniques enables us to successfully recover highly discontinous coefficient even under observation errors as high as 100% in the L^2 norm.

We remark that when the coefficient $q(x)$ is not depending on time, the same kind of techniques used to estimate $q(x)$ from equation (1) can be used to estimate $q(x)$ from (2), see [13]. So we shall concentrate on the identification of $q(x)$ from (1) in the rest of this work.

The organization of the paper is as follows. In the next section, we present the three variants of the output-least-squares formulation of the inverse problem. In Section 3, we define the corresponding augmented Lagrangian formulations. The noise removing techniques are introduced in Section 4. Numerical results will be presented in Sections 5.

2. Formulation of the Identification Problems.

2.1. Output-Least-Squares Approach. In order to present our discussions in a more general setting involving continuous functions, we interpolate the point observations to get distributed observations. Correspondingly, we shall treat the following three cases:

(1) We have an observation $u_d \in L^2(\Omega)$ for the solutions u. We note that this condition is weaker than required of the solution u.

(2) We have observations $u_d \in L^2(\Omega)$, $\vec{u}_g \in (L^2(\Omega))^2$ for the solution u and its gradient.

(3) We have observation $u_d \in L^2(\Omega)$, $\vec{u}_v \in (L^2(\Omega))^2$ for the solution u and velocity $q\nabla u$.

As the observations may contain large random observation errors, and also due to the lack of proper boundary conditions for the coefficient $q(x)$, it is not preferable to use direct methods for solving the inverse problems. Instead, we shall search for a coefficient $q(x)$ which produces a solution $u(x)$ that has the smallest distance to the observations, subject to certain regularity conditions on $q(x)$.

Let the set K denote the set of admissible coefficients. In the following, the set K is taken to be

$$K = \{q| \quad q \in L^\infty(\Omega), \quad 0 < k_1 \leq q(x) \leq k_2 < \infty\},$$

with k_1 and k_2 known *a priori*. Let $e(q,u) = 0$ represents the equation constraint (1) in a suitable space; see §2.3. Finally, let $u(q)$ be the solution to $e(q,u) = 0$ for a given q.

Corresponding to each of the three cases, we shall solve:

$$\textbf{(P1)} \qquad \min_{e(q,u)=0,\ q\in K} \frac{1}{2}\|u(q) - u_d\|^2_{L^2(\Omega)} + \beta R(q)\ ,$$

$$\textbf{(P2)} \qquad \min_{e(q,u)=0,\ q\in K} \frac{1}{2}\|u(q) - u_d\|^2_{L^2(\Omega)} + \frac{1}{2}\gamma\|\nabla u(q) - \vec{u}_g\|^2_{L^2(\Omega)} + \beta R(q)\ ,$$

$$\textbf{(P3)} \qquad \min_{e(q,u)=0,\ q\in K} \frac{1}{2}\|u(q)-u_d\|^2_{L^2(\Omega)} + \frac{1}{2}\gamma\|q\nabla u(q)-\vec{u}_v\|^2_{L^2(\Omega)} + \beta R(q)\ ,$$

where $R(q)$ is a regularization functional used to control the regularity of $q(x)$. In all our experiments, we use $\gamma = 1$ for simplicity. A judicious choice of γ, possibly depending on h, can potentially improve the efficiency significantly.

2.2. Total Variation Regularization. If the coefficient is continuous, $R(q) = \|q\|^2_{H^2(\Omega)}$ or $\|q\|^2_{H^1(\Omega)}$ are commonly used as the regularization term. In [16], existence, uniqueness and convergence have been proved for such kinds of regularization. However, if the coefficient has large jumps, the use of H^2 or H^1–regularization will smear out the sharp discontinuities or produce Gibbs oscillations. In this work, we shall take as regularization functional the following:

$$TV(q) = \int_\Omega |\nabla q| dx,$$

where $TV(q)$ is the total variation of q; see Ziemer [23] and Giusti [9] for definitions. When q is not differentiable, $|\nabla q|$ is understood as a measure, see p. 111 of [23].

However, the TV-norm functional is not differentiable with respect to q. For numerical purpose, we introduce

$$R(q) = \int_\Omega \sqrt{|\nabla q|^2 + \varepsilon}\ dx. \tag{3}$$

This functional is well defined for $q \in H^1(\Omega)$. However, for convenience, in our numerical simulations we use piecewise constants to approximate q and consequently q is not in $H^1(\Omega)$ and $R(q)$ is not well defined. In Section §5, we shall show that there exists a modified version of $R(q)$ that does approximate $TV(q)$ as ϵ goes to zero.

2.3. The Equation Constraint. For any given $q \in K$ and $u \in H^1_0(\Omega)$, we shall define the linear operators: $A_q : H^1_0(\Omega) \to H^{-1}(\Omega)$, $B_u : K \to H^{-1}(\Omega)$ by:

$$\begin{aligned} A_q u &= -\nabla\cdot(q\nabla u), \\ B_u q &= -\nabla\cdot(q\nabla u). \end{aligned}$$

For any $q \in K$, A_q is a homomorphism from $H^1_0(\Omega)$ to $H^{-1}(\Omega)$. Let C be any homomorphism from $H^{-1}(\Omega)$ to $H^1_0(\Omega)$, e.g. in our algorithms we use the inverse Laplacian operator with homogeneous Dirichlet boundary conditions for C. Moreover, we assume that C is chosen such that (Cv, w), which is the dual action of w on Cv, defines an inner product for $v,\ w \in H^{-1}(\Omega)$. Correspondingly, the operator C should satisfy:

$$(Cv, w) = (v, Cw)\ , \quad \forall v,\ w \in H^{-1}(\Omega)\ ,$$

$$(v, w)_{H^{-1}(\Omega)} = (Cv, w)\ , \quad \forall v,\ w \in H^{-1}(\Omega)\ , \tag{4}$$

$$\|v\|^2_{H^{-1}(\Omega)} = \sqrt{(Cv, v)}\ , \quad \forall v \in H^{-1}(\Omega)\ . \tag{5}$$

We now define the equation constraint as:

$$\begin{aligned} e(q,u) &= C(-\nabla\cdot(q\nabla u) - f) \\ &= C(A_q u - f) \\ &= C(B_u q - f). \end{aligned}$$

In the following sections, we use A^*, B^* and C^* to denote the corresponding adjoint operators of A, B and C respectively. In the discrete approximations, A, B and C are matrices and A^*, B^* and C^*

are the corresponding transposes. Let P_h and S_h be the finite element spaces approximating $q(x)$ and $u(x)$ respectively (see [5]). Then the matrix A_q corresponds to the following standard discrete elliptic operator:

$$(A_q u, v) = (q\nabla u, \nabla v), \quad \forall u,\ v \in S_h \subset H_0^1(\Omega).$$

The matrix B_u now corresponds to:

$$(B_u q, v) = (\nabla u \cdot \nabla v, q), \quad \forall q \in P_h, \quad \forall v \in S_h,$$

which generally is not square nor symmetric. The matrix C is symmetric positive definite and corresponds to the homomorphism operator that maps $H^{-1}(\Omega)$ to $H_0^1(\Omega)$. The most straightforward homomomorphism from $H^{-1}(\Omega)$ to $H_0^1(\Omega)$ is the inverse of the Laplace operator and this is what we use in our implementation. More specifically, let D be the discrete Laplacian matrix corresponds to:

$$(Du, v) = (\nabla u, \nabla v), \quad \forall u,\ v \in S_h \subset H_0^1(\Omega),$$

then, we take $C = D^{-1}$.

2.4. Physical Settings in Industrial Applications. The estimation problems we consider here arise in many industrial applications. Two most closely related applications are groundwater flow simulations and oil reservoir simulations. If equations (1) and (2) are used to describe the groundwater motion, then u represents the piezometrical head of ground water in Ω; the function f characterizes the sources and sinks in Ω. The filtration (transmissivity) coefficient q is, physically, positive and piecewise smooth with possible discontinuities on some surfaces in Ω.

Equations (1) and (2) can also be used to describe the fluid flow of a one phase oil reservoir. In such a case, u is the pressure related to the flow in a heterogeneous reservoir; q is called the absolute permeability which is related to the permeability of the medium and other physical parameters such as the viscosity of the fluid. For a one-phase reservoir, the estimation problem **(P1)** corresponds to measuring the pressure to recover the absolute pemeability. The estimation problem **(P2)** corresponds to measuring both the pressure and its gradient to recover q. For the estimation problem **(P3)**, we need to measure both the pressure and the velocity of the fluid to recover q.

3. The Augmented Lagrangian Methods. For the equations we consider in this work, the equation constraint $e(q, u)$ is in a bilinear form. The augmented Lagrangian approach posed in [16] reduces the minimization problems to a system of coupled algebraic equations. Efficient iterative methods can be used to solve these equations.

We shall treat each of the three variational formulations in turn.

3.1. Augmented Lagrangian for (P1). In this case, only L^2-observations are available. The augmented Lagrangian method is used to enforce the equation constraint:

$$e(q, u) = 0.$$

For any $r > 0$, let us define the augmented Lagrangian functional as:

$$L_r(q, u, \lambda) = \frac{1}{2}\|u - u_d\|^2_{L^2(\Omega)} + \beta R(q) + \frac{r}{2}\|e(q, u)\|^2_{H_0^1(\Omega)} + (\lambda, e(q, u))_{H_0^1(\Omega)}.$$

When the equation constraint takes the specific structures as in §2.3, the existence of a saddle point for L_r is known, see [16]. Moreover, if $(\tilde{q}, \tilde{u}, \tilde{\lambda})$ is a saddle point for L_r, then $(\tilde{q}, \tilde{u})$ is a minimizer for **(P1)**. We shall use the following algorithm to search for a saddle point for L_r over $K \times H_0^1(\Omega) \times H_0^1(\Omega)$ in an iterative way. The main idea is to alternatively solve the minimization problem in one of the two variables u and q assuming the other is known. A few steps of this alternating minimization procedure is then followed by an updating step on the Lagrange multiplier λ.

Algorithm 1.

Step 1 Choose $u_0 \in H_0^1(\Omega), \lambda_0 \in H_0^1(\Omega)$ and $r > 0$.

Step 2 Set $u_n^0 = u_{n-1}$. For $k = 1, 2, \ldots, k_{max}$, do:

Step 2.1 Solve $q_n^k = \arg\min_{q \in K} L_c(q, u_n^{k-1}, \lambda_{n-1})$ which gives [7]:

$$\left(\beta R'(q_n^k) + rB^*_{u_n^{k-1}} C(B_{u_n^{k-1}} q_n^k - f) + B^*_{u_n^{k-1}} \lambda_{n-1},\ \phi - q_n^k\right) \geq 0, \quad \forall \phi \in K. \tag{6}$$

Step 2.2 Solve $u_n^k = \arg\min_{u \in H_0^1(\Omega)} L_c(q_n^k, u, \lambda_{n-1})$, which gives:

$$u_n^k - u_d + rA_{q_n^k} C(A_{q_n^k} u_n^k - f) + A_{q_n^k} \lambda_{n-1} = 0\ . \tag{7}$$

Step 3 Set $u_n = u_n^k, q_n = q_n^k$, and update λ_n as

$$\lambda_n = \lambda_{n-1} + re(q_n, u_n). \tag{8}$$

It is easy to see that the above algorithm is the Uzawa gradient method for searching for a saddle point for L_r which satisfies:

$$\frac{\partial Lr}{\partial u} = 0,\ \frac{\partial Lr}{\partial q} = 0,\ \frac{\partial Lr}{\partial \lambda} = 0.$$

We shall use $u = 0$ for the boundary condition of (7). The variational form of (6) implies a Neumann boundary conditionn for q_n^k.

If we use finite element approximations in our computations, $A_{q_n^k}$ is a matrix that depends on q_n^k, $B_{u_n^k}$ is a matrix that depends on u_n^k. If the finite elements are quasi-uniform, we can use inverse inequalities to show that total variation norm is equivalent to the Sobolev norms $H^k(\Omega)$, $k = 1, 2$. The equivalent constant depends on the mesh size h. Making use of this equivalence, we can employ the same techniques as in [16] to show that there is a unique saddle point for L_r if the observation error is sufficiently small, i.e. $\|\tilde{u} - u_d\| << 1$. Moreover, the iterative solution of Algorithm 1 converges to the saddle point with a linear convergence rate. However, if the observation error is large, we can only show as in [5] that a subsequence of $\{(q_n, u_n, \lambda_n)\}$ converges to a saddle point.

In our simulations, we use a simple projection method to handle the variational inequality (6), i.e. we find q_n^k which is the solution of

$$\beta R'(q_n^k) + rB^*_{u_n^{k-1}} C(B_{u_n^{k-1}} q_n^k - f) + B^*_{u_n^{k-1}} \lambda_{n-1} = 0, \tag{9}$$

and then set

$$q_n^k = \max(k_1, \min(q_n^k, k_2)).$$

In our simulations, the constant k_1 is taken to be very small and k_2 is taken to be very large. In most of our tests, the solution of (9) is in the interior of the admissible coefficient set K. This same technique is used to solve the variational inequalities of the algorithms that will be proposed later.

3.2. Augmented Lagrangian for (P2). When we have H^1-observations, we shall enforce the equation constraint $-\nabla \cdot (q \nabla u) = f$ in the H^{-1}-norm that we introduced earlier in (4) and (5). The augmented Lagrangian functional is defined as:

$$L_r(q, u, \lambda) = \frac{1}{2}\|u - u_d\|^2_{L^2(\Omega)} + \frac{1}{2}\gamma\|\nabla u - \vec{u}_g\|^2_{L^2(\Omega)} + \beta R(q)$$

$$
\begin{aligned}
&+ \quad \frac{r}{2}\| -\nabla\cdot(q\nabla u) - f\|^2_{H^{-1}(\Omega)} \\
&+ \quad (\lambda, -\nabla\cdot(q\nabla u) - f)_{H^{-1}(\Omega)}, \\
&\qquad \text{for } q \in K, \quad u \in H^1_0(\Omega),\ \lambda \in H^{-1}(\Omega).
\end{aligned}
$$

Using (4) and (5), it is easy to calculate that:

$$
\begin{aligned}
\frac{\partial Lr}{\partial q} &= \beta R'(q) + rB_u^* C(B_u q - f) \\
&+ B_u^* C\lambda, \\
\frac{\partial Lr}{\partial u} &= u - u_d - \gamma\nabla\cdot(\nabla u - \vec{u}_g), \\
&+ rA_q^* C(A_q u - f) \\
&+ A_q^* C\lambda \\
\frac{\partial Lr}{\partial \lambda} &= -\nabla\cdot(q\nabla u) - f.
\end{aligned}
$$

We shall solve:

$$
\frac{\partial Lr}{\partial q} = 0, \tag{10}
$$

$$
\frac{\partial Lr}{\partial u} = 0, \tag{11}
$$

$$
\frac{\partial Lr}{\partial \lambda} = 0. \tag{12}
$$

by an iterative procedure similar to that for **(P1)**.

Homogeneous Dirichlet boundary condition is used for (11). The boundary condition of (10) is implicitly contained in the variational form for $\frac{\partial Lr}{\partial q} = 0$. For example, if we take $R(q)$ as in (3), then in the discrete setting, the equation $\frac{\partial Lr}{\partial q} = 0$ can be written as:

$$
\begin{aligned}
\beta\,(R'(q), \phi) &+ r\,(B_u^* C(B_u q - f), \phi) \\
&+ (B_u^* C\lambda, \phi) = 0, \quad \forall \phi \in P_h,
\end{aligned}
$$

which implies a Neumann boundary condition for q. Once the matrices B_u and C have been fixed, we only need to get the matrix corresponding to $(R'(q), \phi)$ and solve the nonlinear equation:

$$
\beta R'(q) + rB_u^* C(B_u q - f) + B_u^* C\lambda = 0 \tag{13}
$$

to get q. After solving (13), we project q into set K as described before for **(P1)**. The matrix corresponds to $R'(q)$ depends on q. We use a similar technique to [4] to solve the nonlinear equation.

Algorithm 2.

Step 1 Choose $u_0 \in H^1_0(\Omega), \lambda_0 \in H^{-1}(\Omega)$ and $r > 0$.

Step 2 Set $u_n^0 = u_{n-1}$. For $k = 1, 2, \ldots, k_{max}$, do:

Step 2.1 Solve $q_n^k \in K$ from

$$
\begin{aligned}
\Big(\beta R'(q_n^k) &+ rB^*_{u_n^{k-1}} C(B_{u_n^{k-1}} q_n^k - f) \\
&+ B^*_{u_n^{k-1}} C\lambda_{n-1},\ \phi - q_n^k\Big) \geq 0, \quad \forall \phi \in K.
\end{aligned}
$$

Step 2.2 Solve $u_n^k \in H^1_0(\Omega)$ from

$$
\begin{aligned}
u_n^k - u_d &- \gamma\nabla\cdot(\nabla u_n^k - \vec{u}_g) \\
&+ rA^*_{q_n^k} C(A_{q_n^k} u_n^k - f) \\
&+ A^*_{q_n^k} C\lambda_{n-1} = 0.
\end{aligned}
$$

Step 3 Set $u_n = u_n^k, q_n = q_n^k$ and update the multiplier as

$$\lambda_n = \lambda_{n-1} + r(A_{q_n} u_n - f).$$

By assuming the observation error is sufficiently small, we can prove that there exists a unique saddle point for L_r and the iterative solution of Algorithm 2 converges to the saddle of L_r, see [16]. If the observation errors are large, we can only prove that there is convergent subsequence, see [5].

3.3. Augmented Lagrangian for (P3). Similar to **(P2)**, we shall use the same H^{-1}-norm to enforce the equation constraint. The augmented Lagrangian functional is now:

$$\begin{aligned} L_r(q,u,\lambda) &= \frac{1}{2}\|u-u_d\|^2_{L^2(\Omega)} \\ &+ \frac{1}{2}\gamma\|q\nabla u - \vec{u}_v\|^2_{L^2(\Omega)} + \beta R(q) \\ &+ \frac{r}{2}\|-\nabla\cdot(q\nabla u) - f\|^2_{H^{-1}(\Omega)} \\ &+ (\lambda, -\nabla\cdot(q\nabla u) - f)_{H^{-1}(\Omega)}, \\ &\qquad \text{for } q\in K, u\in H_0^1(\Omega), \lambda\in H^{-1}(\Omega). \end{aligned}$$

In a similar way as for **(P2)**, we can calculate that:

$$\begin{aligned} \frac{\partial Lr}{\partial q} &= \gamma(q\nabla u - \vec{u}_v)\cdot\nabla u + \beta R'(q) \\ &+ rB_u^* C(B_u q - f) \\ &+ B_u^* C\lambda, \\ \frac{\partial Lr}{\partial u} &= u - u_d - \gamma\nabla\cdot(q^2\nabla u - q\vec{u}_v) \\ &+ rA_q^* C(A_q u - f) \\ &+ A_q^* C\lambda. \end{aligned}$$

In the discrete setting, the variational equation for $\frac{\partial Lr}{\partial q} = 0$ is:

$$\begin{aligned} &\gamma(q\nabla u - \vec{u}_v, \phi\nabla u) + \beta(R'(q),\phi) \\ &+ r(B_u^* C(B_u q - f), \phi) \\ &+ (B_u^* C\lambda, \phi) = 0, \quad \forall\phi\in P_h. \end{aligned}$$

We use the following analogous algorithm to search for a saddle point for L_r:

Algorithm 3.

Step 1 Choose $u_0\in H_0^1(\Omega), \lambda_0\in H^{-1}(\Omega)$ and $r>0$.

Step 2 Set $u_n^0 = u_{n-1}$. For $k=1,2,\ldots,k_{max}$, do:

Step 2.1 Solve $q_n^k\in K$ from

$$\begin{aligned} \Big(&\gamma(q_n^k\nabla u_n^{k-1} - \vec{u}_v)\cdot\nabla u_n^{k-1} + \beta R'(q_n^k) \\ &+ rB^*_{u_n^{k-1}} C(B_{u_n^{k-1}} q_n^k - f) + B^*_{u_n^{k-1}} C\lambda_{n-1},\ \phi - q_n^k\Big) \geq 0, \quad \forall\phi\in K. \end{aligned}$$

Step 2.2 Solve $u_n^k\in H_0^1(\Omega)$ from

$$\begin{aligned} u_n^k - u_d &- \gamma\nabla\cdot(|q_n^k|^2\nabla u_n^k - q_n^k\vec{u}_v) \\ &+ rA^*_{q_n^k} C(A_{q_n^k} u_n^k - f) \\ &+ A^*_{q_n^k} C\lambda_{n-1} = 0. \end{aligned}$$

Step 3 Set $u_n = u_n^k, q_n = q_n^k$ and update the multiplier as

$$\lambda_n = \lambda_{n-1} + r(A_{q_n} u_n - f).$$

4. Noise Removal Preprocessing. In many practical applications, observations contain random noises. Let u be a function and u_d be its observation with random noises. From the observation u_d, we try to recover a more accurate function for u. In order to preserve the shape of the function and at the same time filter out the highly oscillatory noises, we use a total variation denoising procedure which solves:

$$\min_{v \in W_0^{1,1}(\Omega)} \left(\alpha TV(v) + \frac{1}{2}\|v - u_d\|^2_{L^2(\Omega)} \right).$$

If we replace $TV(v)$ by its approximation $R(v)$, this is equivalent to solving:

$$\begin{cases} -\alpha \nabla \cdot \left(\dfrac{\nabla u}{\sqrt{|\nabla u| + \varepsilon}} \right) + u - u_d = 0 & \text{in } \Omega \ , \\ u = 0 & \text{on } \partial\Omega \ . \end{cases} \tag{14}$$

Efficient numerical methods have been developed in [4] for solving this problem.

The choice of α is important in removing the noises from the observations. If α is chosen too big, then the recovered function will be very smooth and far from the true u. If α is chosen too small, then the random noisies cannot be removed from the observations. In our simulations, we determine α such that the recovered function u satisfies

$$\|u - u_d\|_{L^2(\Omega)} = \sigma \ ,$$

where σ is the L^2–noise level which we assume to know *a priori*. For this, we use an implicit projection technique derived from one used in [19]. In [3], further techniques have been developed to solve this problem efficiently. Numerical evidence shows that it is not necessary to know σ exactly.

4.1. Noise Removing for (P1). Before we use Algorithm 1 to solve the inverse problem **(P1)**, we shall first use the above noise removing technique for u_d to get a denoised solution u of (14). Then, we use u as the observation data and also as the initial value for u_0 in Algorithm 1.

We remark that other regularizations, e.g. the H^1 norm, can also be used to denoise u_d, but our experience is that the TV technique works better in our experiments.

4.2. Noise Removing for (P2). As observations for both u and ∇u are available, we shall solve:

$$\begin{aligned} \min_{v \in H_0^1(\Omega)} \Big(\alpha R(v) \ &+ \ \frac{1}{2}\|v - u_d\|^2_{L^2(\Omega)} \\ &+ \ \frac{1}{2}\omega \|\nabla v - \vec{u}_g\|^2_{L^2(\Omega)} \Big) \end{aligned} \tag{15}$$

to get a denoised u. The choice of the parameter ω is not very important. In all our experiments, we use $\omega = 1$. Minimization (15) is equivalent to solving:

$$\begin{cases} -\alpha \nabla \cdot \left(\dfrac{\nabla u}{\sqrt{|\nabla u|^2 + \varepsilon}} \right) - \omega \nabla \cdot (\nabla u - \vec{u}_g) + u - u_d = 0 & \text{in } \Omega \ , \\ u = 0 & \text{on } \partial\Omega. \end{cases} \tag{16}$$

After solving (16), we can use three different ways to get a denoised gradient ∇u:

1. Solve (14) for each component of $\vec{u}_g$ and this removes the noises from $\vec{u}_g$ directly.
2. Using numerical differentiation to get the gradient of the solution of (16). When the coefficient has large jumps, this turns out to be a very bad choice.
3. The third approach is to find the minimizer for:

$$\begin{aligned} \min_{\vec{w} \in \mathbf{W}^{1,1}(\Omega)} \alpha TV(\vec{w}) \ &+ \ \frac{1}{2}\|\vec{w} - \nabla u_d\|^2_{\mathbf{L}^2(\Omega)} \\ &+ \ \frac{1}{2}\|\vec{w} - \vec{u}_g\|^2_{\mathbf{L}^2(\Omega)} \end{aligned} \tag{17}$$

In the above, we have:

$$TV(\vec{w}) = TV(w_1) + TV(w_2).$$

If we replace $TV(w_i)$ by $R(w_i)$, then minimizing (17) is equivalent to solving some partial differential equations. In our simulations, both the first and the third approaches have been used.

4.3. Noise Removing for (P3). For the inverse problem **(P3)**, we shall again solve (14) to remove the noises from u_d. To remove the noises from $\vec{u}_v$ we observe that:

$$-\nabla \cdot (q\nabla u) = f, \quad \text{and} \quad q\nabla u \approx \vec{u}_v \ .$$

Therefore, we shall find a minimizer for:

$$\begin{aligned} \min_{\vec{w}\in \mathbf{W}^{1,1}(\Omega)} \alpha TV(\vec{w}) \quad &+ \quad \frac{1}{2}\|\vec{w} - \vec{u}_v\|^2_{\mathbf{L}^2(\Omega)} \\ &+ \quad \frac{1}{2}\omega\| - \nabla \cdot \vec{w} - f\|^2_{\mathbf{L}^2(\Omega)} \ , \end{aligned}$$

which removes the noises from $\vec{u}_v$ and at the same time enforces the equation constraint.

5. Numerical Experiments. We first describe the discretization issues. Let $\Omega \subset R^n$, $n = 1, 2, 3$ be a bounded domain. We first divide Ω into finite elements $\mathcal{T}_h = \{e_i\}$. In all the simulations, a uniform mesh is used. The domain is discretised by simplicial elements, namely intervals, triangles and tetrahedrals in 1D, 2D and 3D respectively. Let S_h denote the piecewise linear finite element space over $\mathcal{T}_h$ with zero Dirichlet boundary value on $\partial\Omega$. Let P_h denote the piecewise constant finite element space over $\mathcal{T}_h$. The space S_h will be used to approximate u and the space P_h will be used to approximate q. Let e_i and e_j be any two elements of the finite element division $\mathcal{T}_h$ and $|\bar{e}_i \cap \bar{e}_j|$ be the $(n-1)$-dimensional measure of the interface between $\bar{e}_i$ and $\bar{e}_j$. For a given $q \in P_h$, we define q_i to be $q|_{e_i}$. Then it is easy to calculate that:

$$TV(q) = \int_\Omega |\nabla q| dx = \sum_{i<j} |q_i - q_j||\bar{e}_i \cap \bar{e}_j|.$$

Correspondingly, we define the discrete functional $R(q)$ as:

$$R(q) = \sum_{i<j} \left(\sqrt{\frac{|q_i - q_j|^2}{h^2} + \varepsilon} \right) |\bar{e}_i \cap \bar{e}_j| \cdot h \ .$$

In all our simulations, the value of ε is taken to be 0.01. With $R(q)$ thus defined, it is easy to form the matrix $R'(q)$.

The observations for our simulations are generated as follows. Let $f(x)$ be a given function. For any given $q \in P_h$, let $u \in S_h$ be the solution of:

$$(q\nabla u, \nabla v)_\Omega = (f, v)_\Omega, \ \forall v \in S_h.$$

The observation u_d is obtained from

$$u_d = u + \delta_1 * R_d * \|u\|_{L^2(\Omega)},$$

where δ_1 is a constant controlling the noise level and R_d is a vector with $\dim(R_d) = \dim(u_d)$ of uniformly distributed random numbers in $[-1/2, 1/2]$ with zero mean. The observations $\vec{u}_g$ and $\vec{u}_v$ are generated similarly by adding random noisies to their true finite element values, i.e.

$$\vec{u}_g = \nabla u + \delta_2 * R_d * \|\nabla u\|_{\mathbf{L}^2(\Omega)},$$

$$\vec{u}_v = q\nabla u + \delta_2 * R_d * \|q\nabla u\|_{\mathbf{L}^2(\Omega)}.$$

Our implementation is in Matlab with a machine precision about 10^{-15}. The random vectors R_d are generated by the Matlab function **rand.m** and their values change at every call. In all our experiments, we take $\delta_1 = \delta_2 = \delta$.

In identifying the coefficient $q(x)$, the proposed algorithms are very sensitive to the value of the regularization parameter β. This is true especially when the observation errors are large. In our simulations, we adjust the value of β so that the algorithms produce the best numerical solutions. In using Algorithm 1–3, it is not necessary to do many inner iterations between Step 2.1 and Step 2.2; we use only 3 iterations (i.e. $k_{max} = 3$). The initial value of the Lagrange multiplier λ is always taken to be $\lambda_0 = 0$.

5.1. One Dimensional Experiments. We begin with several experiments in 1D. The more modest computational demands of the 1D setting allows us to explore in more details several algorithmic variations.

Example 1. In this example, we try to identify a one dimensional piecewise constant q with several jump discontinuities from a L^2–observation by using the **(P1)** formulation. We take $f(x) = 1$. The true coefficient $q(x)$ is chosen as: (see Figure 1)

$$q(x) = \begin{cases} 0.5, & x \in [0, 1/5] \cup [2/5, 3/5] \cup [4/5, 1], \\ 2, & x \in [1/5, 2/5] \cup [3/5, 4/5]. \end{cases}$$

The interval $[0, 1]$ is divided into 200 elements, i.e. $h = 1/200$. The identified $q(x)$ for different noise levels are shown in Figure 1. Different values for β and r are used for each subfigure; see Table 1. We solve (14) to smooth the observation u_d. Then we use the smoothed u_d as the observation and also as the initial value for u in Algorithm 1. Since the TV functional $R(q)$ is nonlinear in q, we need an initial guess for q for the iterative solution of the equation in Step 2.1. The results in Figure 1 is computed with the initial value $q = 1$. Our experience is that the algorithm is very robust with respect to the choice of the initial value for q.

Table 1. The values of β and r for Figure 1

	$\delta = 0.2\%$	$\delta = 2\%$	$\delta = 10\%$	$\delta = 20\%$	$\delta = 60\%$	$\delta = 100\%$
$\beta =$	1e-4	3e-4	3e-4	3e-4	5e-3	5e-3
$r =$	1e2	1e2	1e2	1e2	1e2	1e2

We have run simulations for this example many times to see the identified q with different random observation errors. The conclusion is that if the observation error is more than 10%, then the location of the discontinuity cannot be recovered reliably. On the other hand, if the observation error is less than 1%, then we can always get a rather good estimation of $q(x)$, by which we mean that both the location of the discontinuity and the value of $q(x)$ is identified with an accuracy of less than 1% error.

Example 2. We identify the same coefficient $q(x)$ as in Example 1 by observing both the value of u and the gradient of u using formulation **(P2)**. The observation data with different noise levels are given in Figure 2. The identified q is shown in Figure 3. The values of β and r for different observation errors are summarized in Table 2.

Table 2. The values of β and r for Figure 3

	$\delta = 0.2\%$	$\delta = 2\%$	$\delta = 10\%$	$\delta = 20\%$	$\delta = 60\%$	$\delta = 100\%$
$\beta =$	1e-4	3e-4	3e-4	3e-4	5e-3	5e-3
$r =$	1e2	1e2	1e2	1e2	1e2	1e2

Tests with different observation errors show that the identified coefficient is rather accurate (less than 1% of error) when the observation error is less than 20%. The location of the discontinuities can be identified even with observation error up to 100% for the example considered here. However, the values of $q(x)$ is not accurately identified when the observation error is bigger than 20%.

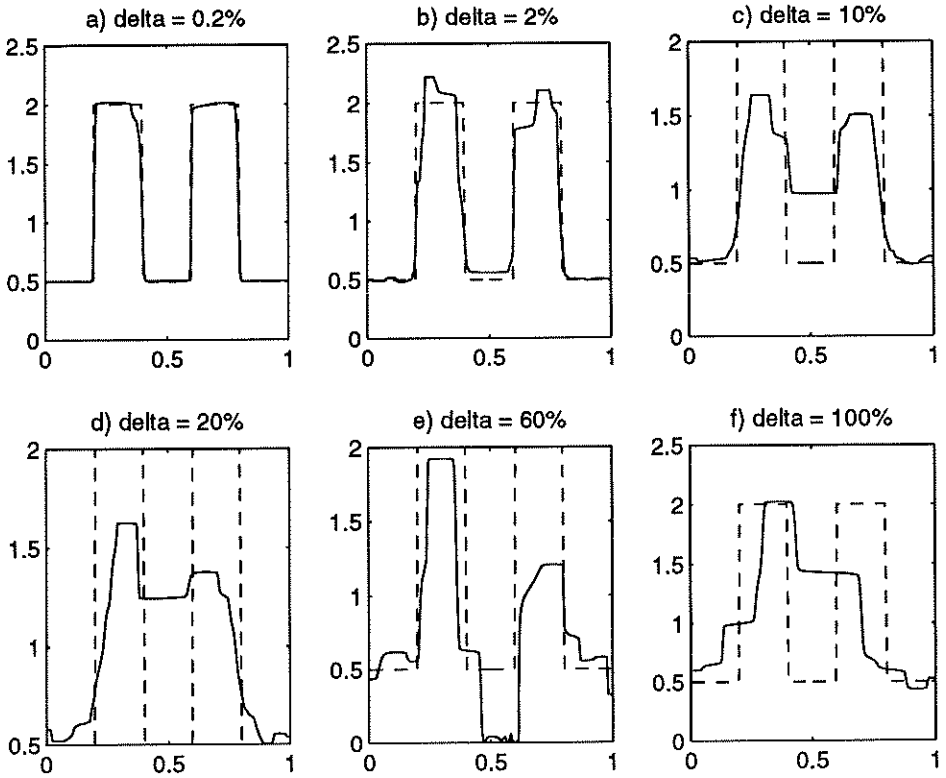

Figure 1. The identified coefficient with different noise levels by L^2–observation.

If the boundary conditon is known exactly, we have also done tests that show the observation for u is unnecessary, i.e. we can get the same accuracy for the identified coefficient only from the observation of the gradient of u.

Example 3. We now present results for formulation **(P3)**; i.e. we identify $q(x)$ by observing both u and $q\nabla u$. The identified q is shown in Figure 4. The values of β and r for different observation errors are given in Table 3.

Table 3. The values of β and r for Figure 4

	$\delta = 0.2\%$	$\delta = 2\%$	$\delta = 10\%$	$\delta = 20\%$	$\delta = 60\%$	$\delta = 100\%$
$\beta =$	1e-4	3e-4	3e-4	3e-4	5e-3	5e-3
$r =$	1e2	1e2	1e2	1e2	1e2	1e2

The performance does not seem to be as good as for formulation **(P2)**. In particular, in order to identify the location of discontinuity of $q(x)$ with an accuracy of 1%, we need the observation error to be much less than 2%.

Summarizing our numerical experience, we can see that the observation for the gradient ∇u is preferable in situations that the observation errors are big. When the observation errors are small, all three formulations **(P1)**–**(P3)** can give accurate solutions. For a given level of observation errors, the identified coefficient by **(P2)** is more accurate than that of the other two formulations.

5.2. Two Dimensional Experiments. In two dimensions, the computational cost is increased dramatically and the efficient solution of the nonlinear equations becomes a more critical issue. Therefore, we only present a few examples with large observation errors. We will not dwell into the details of the numerical procedures in the present paper. We use a modified version of the method used by

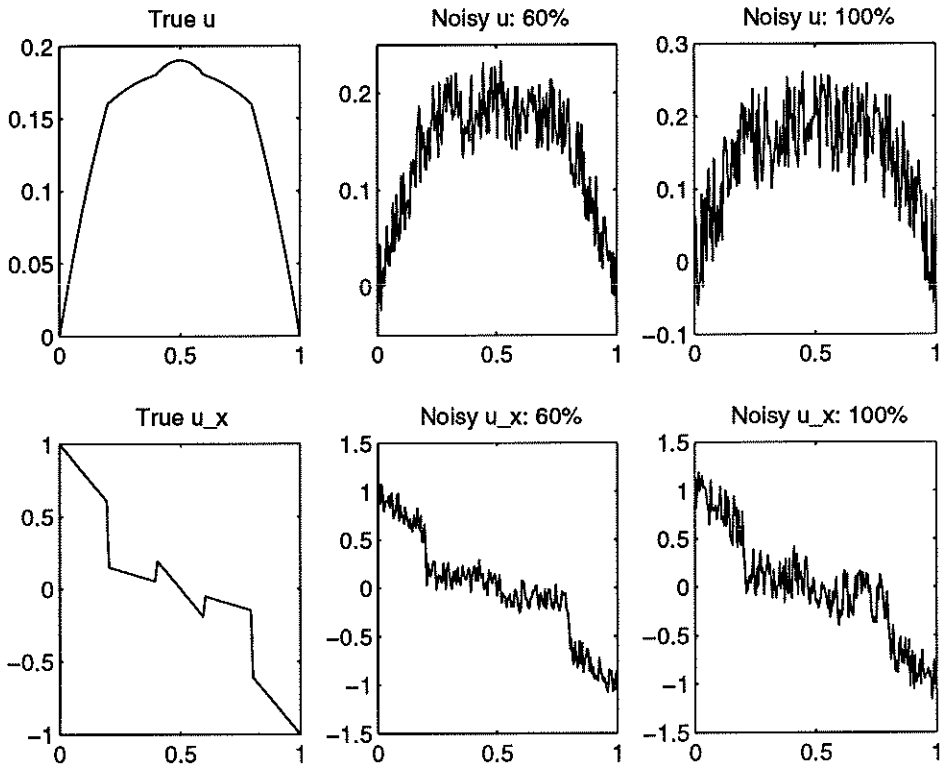

Figure 2. The observations with 60% and 100% noises.

Kunisch and Tai [15, 14] where the augmented Lagrangian method is combined with a nonoverlapping domain decomposition method. We emphasize that in this method, the domain decomposition method is not used as a preconditioner for the linearized elliptic operators that arise in our algorithms. Instead, the full inverse problem is solved on each subdomain simultaneously. We remark that in using the algorithm of [14], it is necessary to use Neumann boundary conditions on some part of $\partial\Omega$, see [14].

The specific values of the parameters used in our experiments are summarized in subfigure c) of Figures 5–7. In the figure, m is the number of subdomains used in each of the $x-$ and $y-$ directions, and n is the number of elements used in each of the $x-$ and $y-$ directions. The paramters c and rho correspond to the parameter r used in Algorithms 1–3.

Example 1. We identify a piecewise smooth coefficient from an observation of u with random errors using formulation **(P1)**. The true coefficient is $q(x,y) = c_1(x,y)e^{xy}$ where $c_1(x,y)$ is a discontinuous piecewise constant function with values $c_1 = 10$ or $c_1 = 1$, see Figure 5. Similar to the one-dimensional problems, the maximum tolerable observation error for L^2-observations is about 1%. The observation error for Figure 5 is $\delta = 1\%$.

Example 2. We identify the same q as in Example 1 by observations of u and ∇u using formulation **(P2)**. 100% observation errors are added, i.e. $\delta = 1 = 100\%$. The computed functions are given in Figure 6. The maximum tolerable observation error is about 100%.

Example 3. We identify the same q by observations of u and $q\nabla u$ by using formulation **(P3)**. 1% observation errors are added, i.e. $\delta = 0.01$. The computed functions are given in Figure 7. The maximum tolerable observation error is about 1%.

The conclusions that we can draw from the 2D experiments are similar to that of the 1D tests, namely that formulation **(P2)** can tolerate more observation errors than **(P1)** and **(P3)**.

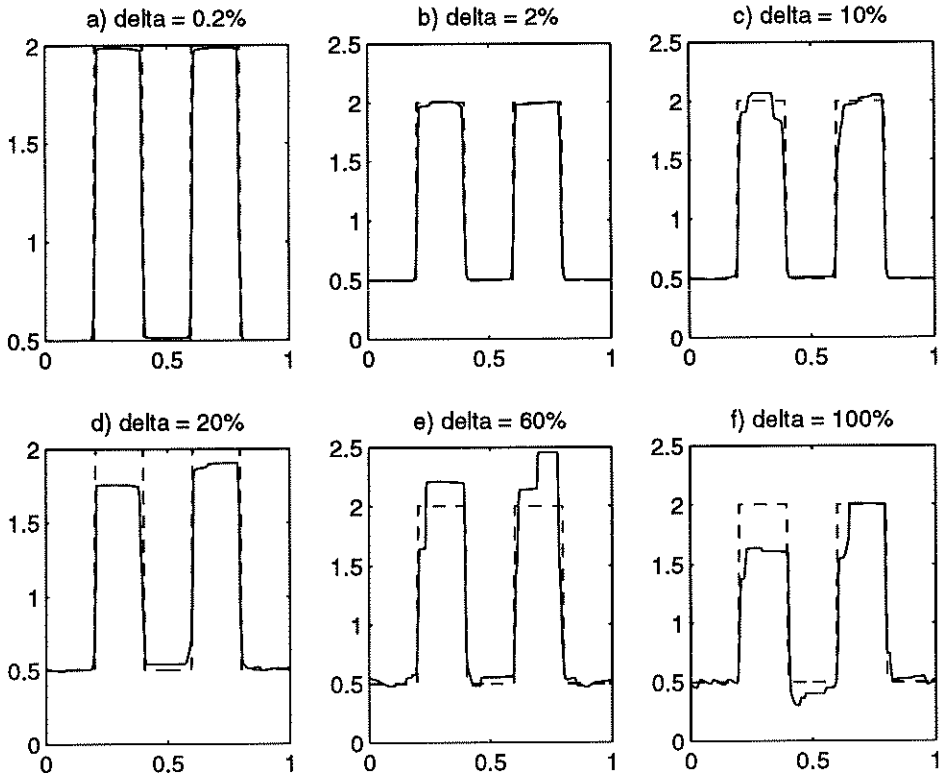

Figure 3. The identified coefficient with different noise levels by H^1–observation.

REFERENCES

[1] L. Alvarez, P. L. Lions, and J. M. Morel. Image slective smoothing and edge detection by nonlinear diffusion. II. *SIAM J. Numer. Anal.*, 29:845–866, 1992.

[2] H. T. Banks and K. Kunisch. *Estimation techniques for distributed parameter systems.* Birkhäuser, Basel, 1989.

[3] P. Blomgren and T. F. Chan. Modular solvers for constrained image restoration problems. *Preprint*, 1997.

[4] T. F. Chan, G. H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. In *Proc. of ICAOS'96, 12 th Int'l Conf. on Analysis and Optimization of Systems: Images, Wavelets and PDE's, Paris, June 26-28, 1996, M. Berger et al (eds)*, number 219 in Lecture Notes in Control and Information Sciences, pages 241–252, 1996.

[5] T. F. Chan and X.-C. Tai. Augmented lagrangian and total variation methods for recovering discontinuous coefficients from elliptic equations. In M. Bristeau, G. Etgen, W. Fitzgibbon, J. L. Lions, J. Periaux, and M. F. Wheeler, editors, *Computational Science for the 21st Centur*, pages 597–607. John Wiley & Sons, 1997.

[6] D. C. Dobson and F. Santosa. An image enhancement technique for eletrical impedance tomography. *Inverse problems*, 10:317–334, 1994.

[7] I. Ekeland and R.Temam. *Convex analysis and variational problems.* North-Holland, Amsterdam, 1976.

[8] R. Ewing, editor. *The mathematics of reservoir simulation.* SIAM, Philadelphia, 1983.

[9] E. Giusti. *Minimal surfaces and functions of bounded variations.* Birkhäuser, 1984.

[10] S. Gutman. Identification of discontinuous parameters in flow equations. *SIAM J. Contr. Optim.*, 28:1049–1060, 1990.

[11] K. Ito and K. Kunisch. The augmented lagrangian method for parameter estimation in elliptic systems. *SIAM J. Control Optim.*, 28:113–136, 1990.

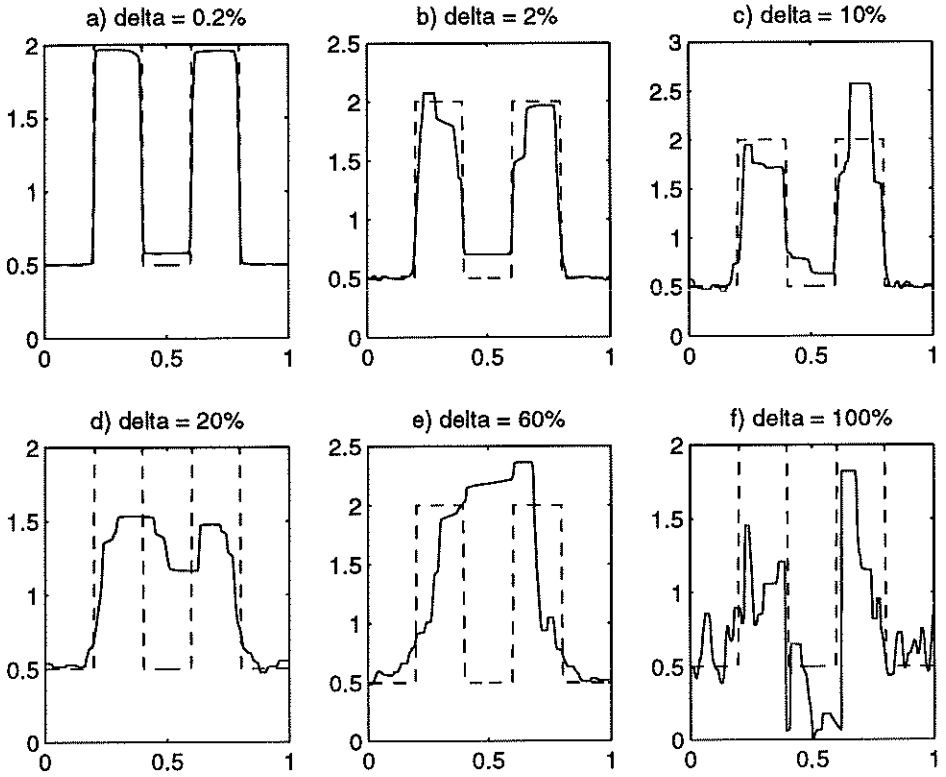

Figure 4. The identified coefficient with different noise levels by observing u and $q\nabla u$.

[12] C. Kravaris and J. H. Seinfeld. Identifibility of spatially-varying conductivity from point observation as an inverse sturm-liouville problem. *SIAM J. Contr. Optim.*, 24:522, 1986.

[13] K. Kunisch and G. Peichl. Estimation of a temporally and spatially varying diffusion coefficient in a parabolic system by an augmented Lagrangian technique. *Numer. Math*, 59:473–509, 1991.

[14] K. Kunisch and X.-C. Tai. Nonoverlapping domain decomposition methods for elliptic parameter estimation problems. *Preprint*, 1997.

[15] K. Kunisch and X.-C. Tai. Nonoverlapping domain decomposition methods for inverse problems. In M. Espedal D. Keyes and P. Bjørstad, editors, *Proceedings of the 9th international domain decomposition methods*. John Wiley and Sons, 1997.

[16] K. Kunisch and X.-C. Tai. Sequential and parallel splitting methods for bilinear control problems in Hilbert spaces. *SIAM J. Numer. Anal.*, 34:91–118, 1997.

[17] T. Lin and E. Ramirez. A numerical method for parameter identification of a two point boundary value problem. *Preprint*, 1997.

[18] S. Osher and L. I. Rudin. Feature-oriented image enhancement using shock filter. *SIAM J. Numer. Anal.*, 27:919–940, 1990.

[19] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[20] E. Vainikko. Courant elements for the inverse problem in the filtration coefficient. In K. Krizek, N. Neittaanmaki, and R. Stenberg, editors, *Finite element methods; fifty years of the Courant element*, volume 164 of *Lecture notes in pure and applied mathematics*, pages 451–461, New York, Hong Kong, 1994. Marcel Dekker.

[21] C. Vogel and M. Oman. Iterative methods for total variation dennoising. *SIAM J. Sci. Comp.*, 17:227–238, 1996.

[22] C. R. Vogel. Sparse matrix equations arising in distributed parameter identification. preprint, 1997.