On the estimation of discontinuous solutions of parameter identification problems

Stefan Kindermann and Andreas Neubauer
Industrial Mathematics Institute, Johannes Kepler University, A-4040 Linz, Austria
E-mail: kindermann@indmath.uni-linz.ac.at, neubauer@indmath.uni-linz.ac.at

Abstract. We consider the identification of a diffusion parameter in a second order elliptic equation in two dimensions by interior measurements. The diffusion parameter is assumed to have discontinuities. For its reconstruction we propose two regularization algorithms – one is based on a moving grid approach and the other on a Hamilton-Jacobi type equation. The numerical realization via an inexact iteratively regularized Gauss-Newton method shows that these algorithms can effectively identify the discontinuities.

1. Introduction
We want to identify a possibly discontinuous diffusion coefficient $\tilde{\gamma}$ defined on a domain $\Omega \subset \mathbb{R}^2$ in the equation

$$-\text{div}(\tilde{\gamma} \nabla u) = f, \quad u\big|_{\partial \Omega} = 0,$$

(1)

where $f \in L^2(\Omega)$ is given, from noisy measurements $u^\delta$ of $u$ with $\|u^\delta - u\| \leq \delta$.

We assume that positive constants $\gamma_1, \gamma_2$ exist such that $\gamma_1 \leq \tilde{\gamma}(x) \leq \gamma_2$ almost everywhere. We want to find that part of $\tilde{\gamma}$ which differs from a constant background diffusivity which we assume to be 1, without loss of generality. Hence, our unknown is $\gamma$ with $\tilde{\gamma} = 1 + \gamma$. In what follows, we denote the nonlinear operator that maps $\gamma = \tilde{\gamma} - 1$ to the solution of (1) by $F$. The problem we are faced with is then the ill-posed equation

$$F(\gamma) = u^\delta.$$

(2)

We will always assume attainability in the following, i.e., for exact data $u$ there exists a $\gamma^\dagger$ for which (2) holds.

In many cases a standard regularization method using Hilbert space norms (see, e.g., [5]), implemented for example by Tikhonov regularization, does not give satisfactory results, since the discontinuities either are smeared out if regularization is too strong or oscillations occur when the regularization norm is too weak.

In particular for discontinuous solutions, the use of the BV-seminorm has been quite effective for this kind of problem (cf., e.g., [4, 11]). However, the BV-functional suffers from the drawback that it is not differentiable, so usually a differentiable approximation to this functional is used. Moreover, BV-regularization shows the so-called staircasing effect [16], i.e., regularized solutions have the tendency to become piecewise constant.
We will present two new approaches that turned out to be quite effective in estimating discontinuous diffusion coefficients, the moving grid approach [10] in Section 2 and the Hamilton-Jacobi approach [6] in Section 3. Numerical results are shown in Section 4.

2. The moving grid approach

This approach is inspired from the work on regularization by curve and surface representations (cf. [7, 8, 9, 13, 14]), where the discontinuous functions are regarded as curves or surfaces and regularization is applied to their parameterizations. In the discretized case this has the effect that the unknown $\gamma$ is always defined on a grid, which changes with the iteration and is adapted to the regularized solution. The change is defined via optimization problems. Unfortunately, in higher dimensions, these optimization functionals are rather flat, leading to slow convergence.

The resulting grid has the property that the mesh size is small wherever the solution exhibits discontinuities, as expected. So instead of computing the grid via optimization it seems more efficient to adapt the mesh size directly to the smoothness of $\gamma$. This adaption can be implemented efficiently by the deformation method. Combined with standard algorithms from regularization theory in Hilbert spaces one still obtains good results for discontinuous solutions with a small extra amount of recalculating the grid in each step.

By our method we may use a discretization of $\gamma$ that is smoother than $\gamma$ itself, for example continuous ansatz functions even for discontinuous $\gamma$. Since the grid will be adaptive to the solution, the grid size will be small wherever $\gamma$ has jumps, and this compensates the approximation error at the nonsmooth parts.

This idea of a moving grid has been used for numerical computations in PDEs (see, e.g., [18]). For instance, in the framework of hyperbolic equations the development of shock waves and the corresponding lack of smoothness of the solutions may not be handled efficiently by a fixed grid. In the context of ill-posed problems, the moving grid approach has been successfully applied in [12] to linear integral equations.

A moving grid is an adaptive grid obtained via a transformation of a fixed, uniform one. Therefore, we need a transformation function $\phi$ which is one-to-one and onto on $\Omega$

$$\phi : \Omega \rightarrow \Omega.$$  

A sufficient condition for a $C^1$-function $\phi$ to fulfill these conditions is that

$$\phi(\partial \Omega) = \partial \Omega \quad \text{and} \quad \det D\phi > 0 \quad \text{in} \, \Omega.$$  

If $c$ is defined on $\Omega$ on a fixed uniform grid, then we can find an approximation to $\gamma$ on an adaptive grid by $c(\phi^{-1})$, where $\phi$ is an appropriate transformation function (again defined on a uniform grid).

We briefly describe the main ideas of our algorithm. In each step we find an approximation $c_n(\phi_n^{-1})$ of the solution, which is obtained by applying a regularization method to the equation

$$F(c(\phi_n^{-1})) = u^\delta$$  

with fixed $\phi_n$. In the next step we compute an error estimator, which measures the smoothness of $c_n(\phi_n^{-1})$. Then we calculate a new transformation function $\phi_{n+1}$ depending on this error estimation.

Thus, the general steps of the algorithm look as follows:

1. Start with a uniform grid and the identity as transformation function $\phi_0(\xi) = \xi \in \Omega \subset \mathbb{R}^2$, $n = 0$.
2. Compute $c_n$ by regularizing the equation $F(c(\phi_n^{-1})) = u^\delta$ with respect to $c$ and set $\gamma_n := c_n(\phi_n^{-1})$.  

3. If a stopping criterion is satisfied, set $\gamma = \gamma_n$; otherwise:
4. Update the transformation function

$$\phi_{n+1} = T(\phi_n, c_n),$$

where $T$ is the method of choice to define the moving grid; go to step 2.

In step 2 a prominent regularization method is Tikhonov regularization. However, it turned out that iterative regularization methods are more efficient for parameter estimation problems. We implemented several ones (see [10]). Best results were obtained with an inexact iteratively regularized Gauss-Newton method.

To describe the ideas we introduce some notation. In step 2 we have to solve (5), where $\phi_n$ is kept fixed and $c$ is the unknown. We use $F_{\phi_n}(c) := F(c(\phi_n^{-1}))$, and $F'_{\phi_n}(c)$ for the Fréchet-derivative with respect to $c$, and $F^*_{\phi_n}(c)$ for its adjoint (in the space $H^1_0$).

If the equation (5) is regularized via the iteratively regularized Gauss-Newton method, then $c_n$ is approximated by $c_{n,k}$ and the new update for $c_{n,k+1}$ is defined by the equation

$$(F'_{\phi_n}(c_{n,k})^*F_{\phi_n}(c_{n,k}) + \alpha_k I)(c_{n,k+1} - c_{n,k}) = -F'_{\phi_n}(c_{n,k})^*(F_{\phi_n}(c_{n,k}) - u^\delta) + \alpha_k c_{n,k}.$$  

As initial values we use $c_{n,0} = \gamma_{n-1}(\phi_n)$, $c_{0,0} = 0$.

The sequence $\{\alpha_k\}$ plays the role of a regularization parameter and can be chosen as a geometrically decaying sequence. For the iteratively regularized Gauss-Newton iteration, convergence and convergence rates have been proven in [1].

Instead of solving the system above exactly, we used a conjugate gradient method to approximate the exact solution. That means that $c_{n,k+1} = c_{n,k} + \nu_l$, where $\nu_l$ denotes the $l$th step of a CG-method applied to the equation

$$(F'_{\phi_n}(c_{n,k})^*F_{\phi_n}(c_{n,k}) + \alpha_k I)\nu = -F'_{\phi_n}(c_{n,k})^*(F_{\phi_n}(c_{n,k}) - u^\delta) + \alpha_k c_{n,k}.$$  

In the CG method the operator on the left-hand side has to be applied to functions $\nu$. This means that we have only to calculate directional derivatives and hence it is not necessary in the discretized version to compute and store the matrix corresponding to the operator $F'_{\phi_n}(c_{n,k})^*F_{\phi_n}(c_{n,k})$. Since for the evaluation of $F, F'h, F''z$ we always have to solve a PDE, we limited the number of CG-iterations to save computation time.

For the stopping rule in step 3 of our algorithm we used the well-known discrepancy principle by Morozov (cf., e.g., [5]).

We now turn to step 4 of the algorithm. As transformation $T$ we used the deformation method. It provides direct control over the cell size. We briefly describe the main ideas following [18].

Given a positive monitoring function $m(\zeta, t) > 0$ depending on the space variable $\zeta$ and time $t$, we want to construct a deformation function $\phi(\xi, t)$ such that

$$\det D\phi(\xi, t) = m(\phi(\xi, t), t), \quad \phi(\xi, 0) = \phi_{\text{init}}(\xi), \quad \xi \in \Omega, \quad t > 0,$$

where $\xi \in \Omega$.  

In numerical computations for PDEs $m$ usually describes the smoothness of the solutions. If $m$ is small – indicating lack of smoothness – then the volume of a grid element will be small too. A necessary solvability condition for $m$ is the normalization property

$$\int_\Omega (m(\zeta, t)^{-1} - 1) d\zeta = 0.$$
Although (6) is a highly nonlinear PDE there is an elegant algorithm to solve it (cf. [2]).

First, we define a velocity field \( v(\zeta, t) \) by

\[
\begin{align*}
\text{div} \, v(\zeta, t) &= -\frac{\partial}{\partial t} m(\zeta, t), & \zeta \in \Omega, \ t \geq 0, \\
\langle v(\zeta, t), n(\zeta, t) \rangle &= 0 & \zeta \in \partial \Omega, \ t > 0,
\end{align*}
\]

where \( n(\zeta, t) \) denotes the unit outward normal to \( \partial \Omega \). \( v \) may be calculated by the gradient \( v = \nabla w \) of the solution of the Neumann problem (for fixed \( t \geq 0 \))

\[
\begin{align*}
\Delta w(\zeta, t) &= -\frac{\partial}{\partial t} m(\zeta, t), & \zeta \in \Omega, \\
\frac{\partial}{\partial n} w &= 0 & \text{on } \partial \Omega.
\end{align*}
\]

Note that the solvability condition of the Neumann problem is satisfied by the normalization property (7). A function \( \phi(\xi, t) \) satisfying (6) is obtained as solution of the system of ODEs (for fixed \( \xi \in \Omega \))

\[
\begin{align*}
\frac{d}{dt} \phi(\xi, t) &= v(\phi(\xi, t), t) m(\phi(\xi, t), t), & t > 0, \\
\phi(\xi, 0) &= \phi_{\text{ini}}(\xi).
\end{align*}
\]

In our case, \( t \) plays the role of a homotopy parameter connecting the initial grid at \( t = 0 \) with the final grid satisfying (6) at \( t = 1 \).

In each iteration in step 4 of our algorithm, we compute the deformation function \( \phi_{n+1} \) by

\[
\det D\phi_{n+1}(\xi) = m_n(\phi_{n+1}(\xi)).
\]

In our numerical realization we choose

\[
m_n(\zeta) := \frac{C_n}{(1 + \beta |\nabla \gamma_n(\zeta)|^2)^{\frac{1}{2}}}, \quad \gamma_n(\zeta) = c_n(\phi_n^{-1}(\zeta)),
\]

\( \beta > 0 \) being a fixed parameter, and \( C_n \) such that the normalization property (7) holds.

Since we are using the monitoring function (11), this would require the inversion of \( \phi_n \). However, in [12] a variant of the above algorithm is described to circumvent this inversion. The idea is to include the transformation function \( \phi_n \) from the previous step by defining

\[
\phi_{n+1}(\xi) := \phi_n(\sigma(\xi, 1))
\]

with an unknown function \( \sigma(\xi, t) \), \( t \in [0, 1] \), such that (10) holds, i.e.,

\[
\det D\phi_{n+1}(\xi) = \det D\phi_n(\sigma(\xi, 1)) \det D\sigma(\xi, 1) = m_n(\phi_n(\sigma(\xi, 1))).
\]

This yields the following equation for \( \sigma \):

\[
\det D\sigma(\xi, 1) = \tilde{m}_n(\sigma(\xi, 1))
\]

with

\[
\tilde{m}_n(\xi) = \frac{m_n(\phi_n(\xi))}{\det D\phi_n(\xi)}.
\]
If we denote components of the deformation function in the $n$-th step by $a, b$, i.e., $\phi_n(\xi) = (a(\xi), b(\xi))$ and $\gamma_n(\xi) = c(\phi_n^{-1}(\xi))$, the chain rule yields $(\xi = (\xi_1, \xi_2))$:

$$m_n(\xi) = \frac{C_n}{\det D(a(\xi), b(\xi))(1 + \beta|\nabla \gamma_n(\phi_n(\xi))|^2)^{\frac{1}{2}}}$$

$$= \frac{C_n}{(a_1 b_{c_2} - a_{c_2} b_{1_1})^2 + \beta((b_{c_2} c_{1_1} - b_{1_1} c_{c_2})^2 + (a_{c_2} c_{c_2} - a_{c_2} c_{c_2})^2)^{\frac{1}{2}}}$$

We start with $\sigma(\xi, 0) = \xi$ and use the parameter $t \in [0, 1]$ to connect $\sigma(\xi, 0)$ with $\sigma(\xi, 1)$. The function $\sigma(\xi, t)$ is chosen to solve

$$\det \sigma(\xi, t) = \frac{1}{(1 - t) + t m(\sigma(\xi, t))}, \quad t \in [0, 1].$$

This equation has the form (6) and can be solved as above (see (8), (9)). Note that by the choice of how the right-hand side in (12) depends on $t$, $w(x, t)$ in (8) will not depend on $t$ and has to be solved only once in step 4.

For the numerical realization via finite elements see [10].

3. The Hamilton-Jacobi approach

The level set method [3, 17, 20] has turned out to be an interesting and efficient way of computing the evolution of curves and surfaces. In inverse problems this tool has been utilized to identify the set of discontinuities (jumps) of piecewise constant unknown functions, taking only two values. Adopting the Eulerian point of view on the evolution of surfaces, the level set method can be efficiently implemented by an equation of Hamilton-Jacobi type. The advantage of this method comes from the availability of sophisticated and efficient numerical schemes to solve them [15, 20].

Our motivation for using Hamilton-Jacobi equation arises as for the moving grid approach from the idea of regularization for curve and surface representation (cf. [7, 9, 14]). There discontinuous solutions of ill-posed problems are represented by curves and surfaces describing their graphs. By this approach we may work with continuous graph parameterizations instead of discontinuous functions. Contrary to the level set method we do not have to restrict ourselves to an a-priori assumption on the solutions such as being piecewise constant. It turns out that the formulation via a Hamilton-Jacobi equation arises quite naturally from a descent method for the error functional using curve and surface parameterization of the graph, i.e., we consider the functional

$$J(c, \phi) = \frac{1}{2}\|F(c(\phi^{-1})) - u^\delta\|^2$$

where, as above, $c$ is defined on $\Omega$ and $\phi$ is an appropriate diffeomorphism of $\Omega$.

We would like to define an evolution equation for $\gamma(. , t) = c(\phi^{-1}(., t), t)$ where $c_{t}, \phi_{t}$ are defined as steepest descent flow for (13). In the following we denote by $D\phi$ the Jacobian matrix of $\phi$ and by $|D\phi|$ the Jacobian determinant.

Taking the Gateaux-derivative with respect to $c$ in direction $h_{1}$ we obtain (assuming differentiability)

$$\frac{d}{dt}J(c + th_{1}, \phi)|_{t=0} = \langle F'(c(\phi^{-1}))h_{1}(\phi^{-1}), F(c(\phi^{-1})) - u^\delta\rangle_{L^2}$$

$$= \langle h_{1}, [F'(c(\phi^{-1}))^*(F(c(\phi^{-1})) - u^\delta)](\phi)|D\phi\rangle_{L^2}$$

Thus the steepest descent direction is obtained for

$$h_{1} = -[F'(c(\phi^{-1}))^*(F(c(\phi^{-1})) - u^\delta)](\phi)|D\phi|.$$
On the other hand, taking the Gateaux-derivative with respect to $\phi$ in direction $h_2$ yields:

$$\frac{d}{dt} J(c, \phi + th_2) \bigg|_{t=0} = (r, \nabla c^T (\phi^{-1}) \frac{d}{dt} (\phi + th_2)^{-1}) \bigg|_{t=0}$$

where

$$r := F'(c(\phi^{-1}))^* (F(c(\phi^{-1})) - u^\delta).$$

(14)

Now since

$$0 = \frac{d}{dt} [(\phi + th_2)((\phi + th_2)^{-1})] \bigg|_{t=0} = h_2(\phi^{-1}) + [D(\phi^{-1})]^{-1} \frac{d}{dt} (\phi + th_2)^{-1} \bigg|_{t=0}$$

we obtain

$$\frac{d}{dt} (\phi + th_2)^{-1} \bigg|_{t=0} = -D(\phi^{-1})h_2(\phi^{-1}).$$

(15)

Hence,

$$\frac{d}{dt} J(c, \phi + th_2) \bigg|_{t=0} = -\langle r(\phi) | D\phi|, \nabla c^T D(\phi^{-1})(\phi)h_2 \rangle_{L^2}.$$ 

This suggests to choose as steepest descent direction

$$h_2 = r(\phi) | D\phi|, \nabla c^T D(\phi^{-1})(\phi)h_2 T \nabla c.$$ 

All together we obtain the following evolution equation

$$\left( \begin{array}{c} \phi_t \\ c_t \end{array} \right) = \lambda(s_1, s_2) r(\phi) | D\phi| \left( \begin{array}{c} [D(\phi^{-1})(\phi)]^T \nabla c \\ -1 \end{array} \right),$$

(16)

where $\lambda$ is an appropriate scaling function.

An evolution equation for $\gamma$ is obtained by the chain rule:

$$\gamma_t = c_t(\phi^{-1}) - \nabla c^T (\phi^{-1}) \frac{d}{dt} \phi^{-1}$$

Analogously to (15) we get

$$\gamma_t = c_t(\phi^{-1}) - \nabla c^T (\phi^{-1}) D(\phi^{-1}) \phi_t(\phi)^{-1}.$$ 

Together with (16) we arrive at

$$\gamma_t = -\lambda(\phi^{-1}, t) r(\phi) | D\phi| (\phi^{-1}) [(1 + (\nabla c^T)(\phi^{-1}) D(\phi^{-1}) D(\phi^{-1})^T (\nabla c)(\phi^{-1})].$$

Expressing $c$ via $\gamma$ using

$$\nabla \gamma = (D(\phi^{-1}))^T (\nabla c)(\phi^{-1})$$

and by the choice $\lambda = |D\phi^{-1}|(\phi)$ we end up with

$$\gamma_t = -r \sqrt{1 + |\nabla u|^2}.$$ 

Finally, giving the flow direction with respect to $c$ and $\phi$ different weights, this together with (14) yields

$$\gamma_t = -F'(\gamma)^* (F(\gamma) - u^\delta) \sqrt{\sigma + (1 - \sigma)|\nabla \gamma|^2}, \quad \sigma \in (0, 1].$$

Note that the limiting cases $\sigma = 0$ and $\sigma = 1$ correspond to the level set method and the Landweber iteration, respectively.
By an explicit time discretization we obtain an iteration procedure of the form
\[ \gamma_{n+1} = \gamma_n - \omega_n F'(\gamma_n)^* (F(\gamma_n) - u^\delta) \sqrt{\sigma + (1 - \sigma)|\nabla \gamma_n|^2}, \] (17)
where \( \omega_n \) controls the length of the chosen time-step.

For the computations we also have to discretize the functions \( \gamma_n \). For a numerical realization based on a discretization scheme by Sethian [19] see [6].

Due to the nonlinearity of problem (1) we observed that (17) yields satisfying results, but shows rather slow convergence. Hence, we used a generalization with a Gauss-Newton-type velocity, i.e.,
\[ \gamma_{n+1} = \gamma_n - (F'(\gamma_n)^* F'(\gamma_n) + \alpha_n I)^{-1} (F'(\gamma_n)^* (F(\gamma_n) - u^\delta)) \sqrt{\sigma + (1 - \sigma)|\nabla \gamma_n|^2}, \] (18)
where the inversion is, as for the moving grid approach, replaced by a fixed number of CG-iterations.

4. Numerical results
For the numerical experiments we used \( f(x, y) = \sin(2\pi x) \sin(2\pi y) \) as right-hand side in (1). The following examples were considered.

- **Circle:** \( \gamma = 1 + 2 \chi_{B(0,55,0,45)}(0.3) \).
- **Ramp:** \( \gamma(x, y) = 1 + 2 \chi_{0.25≤x≤0.6} \chi_{0.2≤y≤0.8} \).
- **Moon:** \( \gamma = 1 + 2 (\chi_{B(0,55,0,5)}(0.3) (1 - \chi_{B(0,4,0,5)}(0.25))) \).

Here, \( B(x_0,y_0)(r) \) denotes the circle with midpoint at \( (x_0,y_0) \) and radius \( r \).

For all examples the data points were first computed using a fine uniform grid with \( N = 120 \) and then contaminated by random noise. This grid is much finer than the one used to calculate the regularized solutions (usually \( N = 40 \)). By this we avoid so-called inverse crimes, namely, to use the same setup for the calculation of the simulated data and the regularization itself.

The weight parameter \( \sigma \) in (18) was chosen iteration dependent according to the formula \( \sigma_n = \max(0.995^n, 0.1) \), i.e., at the beginning more emphasis was given to the function value and later on to the grid adjustment. For further numerical aspects see [6] and [10].

Figures 1 and 2 show that we can identify the location of discontinuities quite well. The ramp example exhibits that our algorithm does not suffer from the staircasing effect of several BV-regularizations. We observed that it is difficult to identify \( \gamma \) in regions where the gradient of \( u \) vanishes or is small. This effect can be expected, since \( \gamma \) is not identifiable at points where \( \nabla u = 0 \).

5. Discussion
A comparison of both approaches shows that the moving grid approach (in Lagrangian variables) has the advantage that an adaptive grid for \( \gamma \) is used which provides more flexibility in the approximation of the unknown solution. Obviously, for the same gridsize an adaptive method will in general yield better results. However, one has to use many gridlines if several areas of discontinuities occur. It might then happen that very long and thin patches occur yielding unsatisfactory results. Moreover, the stiffness matrix has to be recalculated in every step, since the grid is changing globally.

The advantage of the Hamilton-Jacobi approach (in Eulerian variables) is the use of a fixed grid for \( \gamma \). Hence, it is simpler to implement and the stiffness matrix has to be calculated only once. However, to get nice contour lines, one needs a much finer grid than for the moving grid approach.
Figure 1. Reconstruction of the three examples with the moving grid approach
Figure 2. Reconstruction of the three examples with the Hamilton Jacobi approach
It will be the aim of future research to combine the advantages of both methods. Thus, we want to work with a grid that can be locally adapted, this leads us to h-adaptivity. The penalty term will be chosen again inspired from regularization by surface representations.

Since the grid should only be fine where the discontinuities are, it will be necessary that one can also coarsen the grid during the iteration procedure.

Acknowledgments

This work was supported by the Austrian Science Foundation Funds under grant SFB F013/F1317.

References

[1] B. Blaschke, A. Neubauer, and O. Scherzer. On convergence rates for the iteratively regularized Gauss-Newton method. *IMA Journal of Numer. Anal.*, 17:421–436, 1997.
[2] P. Bochev, G. Liao, and G. dela Pena. Analysis and computation for adaptive moving grids by deformation. *Numer. Methods of PDEs*, 12:489–506, 1996.
[3] M. Burger. A level set method for inverse problems. *Inverse Problems*, 17:1327–1356, 2001.
[4] Z. Chen and J. Zou. An augmented Lagrangian method for identifying discontinuous parameters in elliptic systems. *SIAM J. Control Optim.*, 37:892–910, 1999.
[5] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Kluwer, Dordrecht, 1996.
[6] S. Kindermann. A new iterative regularization method using an equation of Hamilton-Jacobi type. SFB Report 03-9, SFB F013, University of Linz, 2003.
[7] S. Kindermann and A. Neubauer. Identification of discontinuous parameters by regularization for curve representations. *Inverse Problems*, 15:1559–1572, 1999.
[8] S. Kindermann and A. Neubauer. Estimation of discontinuous parameters of elliptic partial differential equations by regularization for surface representations. *Inverse Problems*, 17:789–803, 2001.
[9] S. Kindermann and A. Neubauer. Regularization for surface representations of discontinuous solutions of linear ill-posed problems. *Numer. Funct. Anal. Optim.*, 22:79–105, 2001.
[10] S. Kindermann and A. Neubauer. Parameter identification by regularization for surface representations via the moving grid approach. *SIAM J. Control Optim.*, 42:1416–1430, 2003.
[11] R. Luce and S. Perez. Parameter identification for an elliptic partial differential equation with distributed noisy data. *Inverse Problems*, 15:291–307, 1999.
[12] A. Neubauer. Estimation of discontinuous solutions of ill-posed problems by regularization for surface representations: numerical realization via moving grids. In Y.C. Hon, M. Yamamoto, J. Cheng, and J.Y. Lee, editors, *Recent Development in Theories and NUMERICS, International Conference on Inverse Problems*, pages 67–83, Singapore, 2003. World Scientific Publisher.
[13] A. Neubauer and O. Scherzer. Reconstruction of discontinuous solutions from blurred data. In R. L. Barbour, M. J. Carvlin, and M. A. Fiddy, editors, *Computational, Experimental, and Numerical Methods for Solving Ill-Posed Inverse Imaging Problems: Medical and Nonmedical Applications*, volume 3171 of *Proceedings of SPIE*, pages 34–41, Washington, 1997. SPIE.
[14] A. Neubauer and O. Scherzer. Regularization for curve representations: uniform convergence for discontinuous solutions of ill-posed problems. *SIAM J. Appl. Math.*, 58:1891–1900, 1998.
[15] S. Osher and J. A. Sethian. Fronts propagating with curvature depending speed: algorithms based on Hamilton-Jacobi formulations. *J. Comp. Physics*, 79:12–49, 1988.
[16] W. Ring. Structural properties of solutions of total variation regularization problems. *M2AN, Math. Model. Numer. Anal.*, 34:799–810, 2000.
[17] F. Santosa. A level-set approach for inverse problems involving obstacles. *ESAIM, Control Optim. Calc. Var.*, 1:17–33, 1996.
[18] B. Semper and G. Liao. A moving grid finite element method using grid deformation. *Numer. Methods of PDEs*, 11:603–615, 1995.
[19] J. A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci. USA 93*, 41591–1596, 1996.
[20] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, Cambridge, 1999.