

# Adaptive and stochastic algorithms for EIT and DC resistivity problems with piecewise constant solutions and many measurements

Kees van den Doel and Uri M. Ascher \*

September 28, 2011

## Abstract

This article develops fast numerical methods for the practical solution of the famous EIT and DC-resistivity problems in the presence of discontinuities and potentially many experiments or data. Based on a Gauss-Newton (GN) approach coupled with preconditioned conjugate gradient (PCG) iterations, we propose two algorithms. One determines adaptively the number of inner PCG iterations required to stably and effectively carry out each GN iteration. The other algorithm, useful especially in the presence of many experiments, employs a randomly chosen subset of experiments at each GN iteration that is controlled using a cross validation approach. Numerical examples demonstrate the efficacy of our algorithms.

## 1 Introduction

The elliptic PDE

$$\nabla \cdot (\sigma(\mathbf{x})\nabla u) = q(\mathbf{x}), \quad \mathbf{x} \in \Omega, \quad (1)$$

where  $\Omega \subset \mathbb{R}^d$  for  $d = 2$  or  $d = 3$  and  $\sigma(\mathbf{x}) \geq \hat{\sigma} > 0$  for all relevant  $\mathbf{x}$ , arises in many applications. Here we consider it under homogeneous Neumann boundary conditions and wish to recover  $\sigma(\mathbf{x})$  from measurements of  $u(\mathbf{x})$  on the boundary  $\partial\Omega$ .

Such a problem arises in electrical impedance tomography (EIT) [8, 7, 5, 27, 4] and in DC resistivity calculations [34, 28]. It is well-known that if the entire Dirichlet-to-Neumann map (equivalently, the boundary data  $u|_{\partial\Omega}$  expressed in terms of the source

---

\*Dept. of Computer Science, University of British Columbia, Vancouver, Canada [kvdoel/ascher@cs.ubc.ca](mailto:kvdoel/ascher@cs.ubc.ca). This work was supported in part by NSERC Discovery Grant 84306.

$q$  for any  $q(\mathbf{x})$  restricted to  $\partial\Omega$ ) is given then  $\sigma(\mathbf{x})$  can be uniquely recovered provided that it is sufficiently smooth on the domain  $\Omega$  (see, e.g., [5, 27, 4]).

In practice, data comes with noise, the forward model is idealized, and in heterogeneous media  $\sigma(\mathbf{x})$  may have jump discontinuities. Indeed, medical tomography machines based on this model are still scarce, mining companies do collect data from boreholes that are expensive to dig, and geophysicists often resort to a more general electromagnetic data inversion [17]. Still, some practical observations can be related to the idealized theory. In particular, better reconstructions can often be obtained with more experiments, each corresponding to a specific right hand side in (1), and the problem of reconstructing accurately does get harder when  $\sigma(\mathbf{x})$  has steep gradients.

Suppose then that we have  $s$  experiments. For each  $i$ ,  $1 \leq i \leq s$ , there are a given source  $q^i(\mathbf{x})$  and measured data  $\mathbf{b}_i$  of  $u$ . We seek a function  $m(\mathbf{x})$  that relates to  $\sigma(\mathbf{x})$  in one of the following two ways.

1. We employ a smooth transformation such as  $\sigma = m^{-1}$  (i.e.,  $m$  is resistivity), or  $\sigma = e^m$  (which automatically yields a positive  $\sigma$ ). To focus attention on other aspects, we consider here only the latter. Note that  $\sigma$  and  $m$  are in the same Sobolev space.

Thus, the problems

$$\begin{aligned} \nabla \cdot (e^m \nabla u^i) &= q^i, \quad i = 1, \dots, s, \\ \frac{\partial u^i}{\partial \nu} \Big|_{\partial\Omega} &= 0, \end{aligned} \quad (2)$$

are discretized on a staggered grid with width, or resolution,  $h$ , as described in [2], and the constant null-space is removed in a standard way.

2. Occasionally it is reasonable to assume that the sought conductivity function  $\sigma(\mathbf{x})$  takes only one of two values,  $\sigma_I$  or  $\sigma_{II}$ , at each  $\mathbf{x}$ . Viewing one of these as a background value, the problem is that of shape optimization. Such an assumption greatly stabilizes the inverse problem [1]. In [9, 10] we considered a level set function representation for the present problem, where we write  $\sigma = e^{P(m;0)} = e^{P(m)}$ , with

$$P(\xi; h) = \frac{\ln \sigma_I - \ln \sigma_{II}}{2} \tanh(\xi/h) + \frac{\ln \sigma_I + \ln \sigma_{II}}{2}. \quad (3)$$

The function  $P(m; h)$  depends on the resolution, or grid width  $h$ . It is a scaled and mollified version of the Heaviside step function, and its derivative magnitude is at most  $O(\frac{|\ln \sigma_I - \ln \sigma_{II}|}{h})$ . Thus, as  $h \rightarrow 0$  the sought function  $m(\mathbf{x})$  satisfying

$$\begin{aligned} \nabla \cdot (e^{P(m)} \nabla u^i) &= q^i, \quad i = 1, \dots, s, \\ \frac{\partial u^i}{\partial \nu} \Big|_{\partial\Omega} &= 0, \end{aligned} \quad (4)$$

has bounded first derivatives, whereas  $\sigma(\mathbf{x})$  is generally discontinuous.

The inverse problem is to approximately recover  $m(\mathbf{x})$  from values of the  $u^i$  on the boundary. Thus, assume that  $m$  and  $u$  satisfying (2) or (4) have been discretized appropriately and reshaped into vectors  $\mathbf{m}$  and  $\mathbf{u}$ , respectively. Regardless of whether (2) or (4) is employed, we write the resulting discretized system for each experiment  $i$  as

$$A(\mathbf{m})\mathbf{u}^i = \mathbf{q}^i. \quad (5)$$

Furthermore, there are given projection matrices  $Q^i$  such that  $\mathbf{F}_i(\mathbf{m}) = Q^i\mathbf{u}^i = Q^iA(\mathbf{m})^{-1}\mathbf{q}^i$  predicts the  $i$ th data set. Defining

$$\mathbf{F} = (\mathbf{F}_1^T, \mathbf{F}_2^T, \dots, \mathbf{F}_s^T)^T, \quad \mathbf{b} = (\mathbf{b}_1^T, \mathbf{b}_2^T, \dots, \mathbf{b}_s^T)^T,$$

we seek an  $\mathbf{m}$  such that

$$\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\| \approx \|\boldsymbol{\epsilon}\|, \quad (6)$$

where  $\boldsymbol{\epsilon}$  is a zero-mean normally distributed noise whose level (say,  $\epsilon = \|\boldsymbol{\epsilon}\|$ ) may be known or estimated.<sup>1</sup> Amongst all candidates  $\mathbf{m}$  we select one with desired properties such as being smooth or piecewise constant. The selection of the desired solution to (6) and the stabilization of the solution process are achieved by regularization.

The classical Tikhonov formulation [35, 13] leads to the optimization problem

$$\mathbf{m}^* = \arg \min_{\mathbf{m}} \phi(\mathbf{m}; \beta) = \frac{1}{2} \|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|^2 + \beta R(\mathbf{m}), \quad (7)$$

with  $R$  a suitable regularization functional and  $\beta > 0$  the regularization parameter. A modified Gauss-Newton (GN) method for the optimization problem (7) can be written as follows. Iterating for  $n = 0, 1, \dots$ , set  $\mathbf{m} = \mathbf{m}_n$  at the  $n$ th iteration. Then

$$(J^T J + \beta_0 R''(\mathbf{m})) \delta \mathbf{m} = -\mathbf{g} \quad (8a)$$

$$\gamma = \arg \min_{\gamma} \phi(\mathbf{m} + \gamma \delta \mathbf{m}; \beta) \quad (8b)$$

$$\mathbf{m}_{n+1} \leftarrow \mathbf{m} + \gamma \delta \mathbf{m}. \quad (8c)$$

Here the gradient  $\mathbf{g}$  of  $\phi$  is given by

$$\mathbf{g} \equiv \nabla_{\mathbf{m}} \phi = J^T (\mathbf{F} - \mathbf{b}) + \beta R', \quad (9)$$

with  $J$  the Jacobian matrix of  $\mathbf{F}$ ,  $R' = \nabla_{\mathbf{m}} R$  the gradient of  $R$  and  $R'' = \nabla_{\mathbf{m}}^2 R$  the Hessian matrix. The step size is restricted to  $0 < \gamma \leq 1$  and  $\beta_0 \geq 0$  is another parameter; e.g.,  $\beta_0 = \beta$  to obtain damped GN with line search.

Notice next that the solution of (8a) can be rather expensive to carry out. In fact, even the evaluation of the misfit  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|$  requires  $s$  PDE solves, i.e.,  $s$

---

<sup>1</sup> Throughout this article we use the  $\ell_2$  vector norm unless otherwise specified.

inversions of (5). The sensitivity matrix  $J$  consists of  $s$  blocks  $J_i$ , each of which costing many PDE solves to construct [10, 16]. Direct methods for solving the linear system (8a) require the very expensive formation of  $J^T J$ , and this does not scale well at all. In the sequel we consider exclusively solution methods involving preconditioned conjugate gradient (PCG) iterations, so only matrix-vector products involving  $J$  are encountered. Let us assume that only  $s_n$  experiments, which may be combinations of the given  $s$  experiments, are used in the  $n$ th iteration: this is elaborated upon further in Section 5. If the number of inner PCG iterations required to obtain  $\delta \mathbf{m}$  in the  $n$ th iteration is  $M_n$  then the GN step requires the solution of  $2s_n M_n$  PDEs.<sup>2</sup> Assuming that  $l_n$  additional PDE solves per experiment are required (e.g., for the weak line search), the total cost is  $(2M_n + l_n)s_n$  PDE solves per outer iteration  $n$ . For  $N$  outer iterations we obtain the work estimate (i.e., the number of PDE solves)

$$W = \sum_{n=1}^N (2M_n + l_n)s_n. \quad (10)$$

The purpose of this article is to propose novel methods for keeping this work estimate as low as possible while still obtaining credible high quality solutions in the presence of potential discontinuities in  $\sigma$ . We propose two algorithms, the second of which incorporating the first, and demonstrate their potential power.

In Sections 3 and 4 we set  $s_n = s$ , i.e., all experiments are used at each outer iteration  $n$ . The method then requires  $O(s \sum_{n=1}^N (2M_n + l_n))$  PDE solves, and our goal is to keep this number small, achieving moderate  $N$  and  $M_n$  with  $l_n \approx 1$ . In Section 3 we motivate and describe an algorithm for the “best” choice of  $M_n$  at each outer iteration. Numerical examples demonstrating the efficacy of this algorithm are offered in Section 4.

In Section 5 we continue and expand in several ways on work reported in [18]. We propose an adaptive method for reducing the number of experiments  $s_n$  at each outer iteration, using random sampling and cross validation. When there are many experiments, say  $s$  is in the hundreds or thousands, the computational savings can be rather substantial. Our method is demonstrated numerically in Section 6.

## 2 Regularization approaches

For the more general problem (2) with  $\sigma$  differentiable, the most popular regularization  $R(\mathbf{m})$  in practice is a possibly weighted, discretized version of the functional

$$\int_{\Omega} |\nabla m|^2. \quad (11)$$

---

<sup>2</sup>This count does not take into account the cost of the preconditioner. The latter, discussed in detail in [10], involves inversion of  $R''$  which is a simple Poisson problem (see (11)). Thus, the preconditioner inversion cost is a small fraction of that of solving (5).

See for instance [17]. In the sequel we refer to the discretization of (11) as the *L2 regularization*. This has worked well also in the numerical examples reported in [28, 12].

However, it is well-known that this regularization also smears out discontinuities and is not useful when such solution features are present in  $m(\mathbf{x})$  and are important to reconstruct. A popular alternative is some variant of total variation [32, 3, 19] (TV), preferably using a Huber switching function with an adaptive switching parameter, which penalizes discontinuities more agreeably (essentially, it avoids attempts to integrate the square of a  $\delta$ -function across the discontinuity). However, it is harder to work with the latter alternative, and the results are occasionally far from certain for this highly ill-posed problem [3, 11]. Moreover, in many practical situations there is not enough data to determine if there actually are discontinuities present, and a TV regularization will not help.

If the piecewise constant assumption on  $\sigma(\mathbf{x})$  is justified then it is much better to use the *level set method*, i.e., (4) rather than (2). This usually allows utilization of (11) for the regularization functional, as the burden of handling the discontinuities shifts to the sharpening projection  $P$ . In this case the discontinuities of the model are built into the reconstruction as a priori information. It should be understood, however, that now the sensitivity matrix  $J$  (the Jacobian of  $\mathbf{F}$ ) involves multiplication by a derivative matrix  $P'$  containing many zeros but also elements of size  $O(\frac{|\ln \sigma_I - \ln \sigma_{II}|}{h})$ . The optimization problem thus becomes significantly harder than when using the L2 method. In the sequel we concentrate on recovering piecewise constant surface functions using the level set method, but occasionally employ also the L2 method as a sanity check.

In [9] the damped GN method (8) with  $\beta_0 = \beta > 0$  was found to perform poorly when using a level set method, even when solving (8a) by direct methods, due to the added nonlinearity by the sharpening function  $P$  of (3). Instead, a *dynamical regularization* method was advocated, with  $\beta = 0$  and a generalized Marquardt-type regularization of the inner linear system (8a), i.e., with a suitably chosen  $\beta_0 > 0$ . In [10] this was taken a step further, and the exact solution of the linear system (8a) was replaced by a small fixed number (in the range of 3 – 5) of PCG iterations. This has a regularizing effect, and it works with  $\beta_0 = 0$  (i.e., when applied directly to the singular system), rarely necessitating a line search. This method was found to work very well for noise levels around 3%. However, the magic number of inner iterations had to be determined by trial and error and in general it is problem dependent.

A related strategy to the dynamical regularization of [10] was proposed and analyzed as a general method in [29, 30], following [21]. In that method the number of inner iterations is considered variable, and it is determined by a sequence of tolerances on misfit linearizations (one for each outer iteration) which are considered user input to the algorithm, designed by trial and error. A convergence proof was given under some conditions on the sequence of tolerances and on the forward operator  $\mathbf{F}$ , which the level set approach does not satisfy.

### 3 Adaptive selection of the number of inner PCG iterations

In this section we develop a heuristic algorithm to select the number of inner PCG iterations at each outer iteration approximating (8). In general, as one may expect, for a very small number of inner iterations more, if cheaper, outer iterations are required. But we take here the approach of attempting to minimize the work while maximizing the yield of each outer iteration in the hope of keeping the required number of such iterations small. We assume the noise level  $\epsilon = \|\epsilon\|$  to be known, so a stopping criterion for the outer iteration is provided by (6) and the discrepancy principle is available to determine regularization parameters.

Given a current iterate  $\mathbf{m} = \mathbf{m}_n$ , for the present iteration to be the last one we want

$$\mathbf{F}(\mathbf{m} + \delta\mathbf{m}) = \mathbf{b} + \epsilon. \quad (12a)$$

We can rewrite this as

$$J\delta\mathbf{m} = \mathbf{b} - \mathbf{F}(\mathbf{m}) + \hat{\epsilon}, \quad (12b)$$

where

$$\hat{\epsilon} = \epsilon + \mathbf{E}, \quad (12c)$$

$$\mathbf{E} = J\delta\mathbf{m} - \mathbf{F}(\mathbf{m} + \delta\mathbf{m}) + \mathbf{F}(\mathbf{m}). \quad (12d)$$

Note that the GN method without regularization consists of solving the rank deficient problem (12b) without  $\hat{\epsilon}$  using linear least squares minimization. Thus, when we next linearize the problem by dropping  $\mathbf{E}$ , we are in effect solving an ill-posed inverse problem with “noise”  $\hat{\epsilon}$  instead of  $\epsilon$ . The expectation value of the norm of this new uncertainty obeys

$$\langle \|\hat{\epsilon}\|^2 \rangle = \langle \|\epsilon\|^2 \rangle + \|\mathbf{E}\|^2 \geq \langle \|\epsilon\|^2 \rangle.$$

(The equality in the expression above follows from the independence of the two terms forming  $\hat{\epsilon}$  and the assumption that  $\langle \epsilon \rangle = 0$ .) This suggests that when taking a step in an iterative method, additional regularization of the update  $\delta\mathbf{m}$  is required, beyond what we have put in to deal with the measurement noise  $\epsilon$ ; cf. [20].

However, an estimate of  $\|\hat{\epsilon}\|$  is not known until we have solved this linearized problem, so we have to come up with a way to determine the correct amount of extra regularization a posteriori. Increasing  $\beta_0$  as  $\|\mathbf{E}\|$  increases amounts to a generalized trust region approach (see, e.g., [26]), but we need a way to determine this parameter.

Instead of increasing  $\beta_0$  it is more efficient to deploy iterative regularization [22, 23] to the inner linear system (8a), as we can rely on the regularizing properties of a finite number of PCG iterations. When applying PCG iterations starting at  $\delta\mathbf{m} = \mathbf{0}$  with the Laplacian preconditioner  $R''(\mathbf{m})$ , at each iteration  $\|\delta\mathbf{m}\|_{R''}$  increases, while the

linearized misfit  $\|\mathbf{J}\delta\mathbf{m}+\mathbf{F}-\mathbf{b}\|$  decreases [23]. The number of inner iterations  $M_n$  now takes on the role of a regularization parameter. To select  $M_n$  it is possible to use an estimate of  $\|\hat{\boldsymbol{\epsilon}}\|$ , but an approach that we found superior in practice is to attempt to maximize progress on the full nonlinear problem. In other words, the merit function remains  $\phi$  of (7). The challenge is to obtain sufficient decrease in  $\phi(\mathbf{m} + \delta\mathbf{m}(M_n))$  *without* evaluating all  $\phi(\mathbf{m} + \delta\mathbf{m}(j))$ ,  $j = 1, \dots, M_n$ .

The default step size is  $\gamma = 1$ . If  $\phi$  does not decrease even for  $M_n = 1$  then we perform a weak line search. We thus obtain the outer iteration (with  $\phi$  and  $\mathbf{g}$  depending on  $\beta$ )

$$\delta\mathbf{m}(M_n) = \text{PCG}(J^T J + \beta_0 R'', -\mathbf{g}, R'', M_n); \quad (13a)$$

$$M_n = \arg \min_M \phi(\mathbf{m} + \delta\mathbf{m}(M)) \quad (13b)$$

$$s.t. \|\mathbf{J}\delta\mathbf{m}(M) + \mathbf{g}\| > \|\boldsymbol{\epsilon}\|; \quad (13c)$$

$$\text{if } M_n > 0 : \gamma = 1, \text{ else} \quad (13d)$$

$$M_n = 1, \gamma = \arg \min_{\gamma} \phi(\mathbf{m} + \gamma\delta\mathbf{m}(M_n)); \quad (13e)$$

$$\mathbf{m}_{n+1} = \mathbf{m} + \gamma\delta\mathbf{m}(M_n). \quad (13f)$$

Here  $\text{PCG}(\hat{A}, \hat{\mathbf{b}}, \hat{C}, k)$  is the result of applying  $k$  PCG iterations to the linear symmetric positive definite system  $\hat{A}\hat{\mathbf{x}} = \hat{\mathbf{b}}$  starting at  $\hat{\mathbf{x}} = \mathbf{0}$ , with preconditioner  $\hat{C}$ .

As with line search methods the criterion for determining  $M_n$  can be quite rough. The most straightforward implementation would be to check reduction in  $\phi$  and condition (13c) after every PCG iteration, but this takes up to  $s(M_n - 1)$  additional PDE solves. Instead we just perform  $M_n$  iterations for our current guess of this value, possibly adjusted by early termination through (13c). We then check at the cost of  $s$  PDE solves if more reduction in  $\phi$  can be achieved with  $M_n - 1$  steps. This replaces a search for an appropriate  $\beta_0$ . If so then  $M_n$  is reduced until the best (smaller) value is found, if not then  $M_n$  is increased for the next outer iteration. The resulting method is described in detail in Algorithm 1. The numerical examples reported in Section 4 as well as in later sections suggest that this algorithm performs remarkably well. Moreover, since the inner iterations provide regularization, the method performs also very well when setting  $\beta_0 = 0$ , as the role of this parameter is now taken over by the PCG iterations, and we therefore use this latter value for all calculations reported in this paper.

The method as stated in Algorithm 1 can, depending on the parameters  $\epsilon$  and  $\beta$ , produce a variety of regularized solutions. If  $\beta$  is larger than a critical value  $\beta_\epsilon$ , such that the misfit of  $\mathbf{m}^*$  in (7) is larger than  $\epsilon$ , then in Algorithm 1 we will always have  $\mu > \epsilon$ , and the method terminates when the minimizer  $\mathbf{m}^*$  of  $\phi$  is found with accuracy determined by  $tol$ . If  $tol$  is small enough then this is just the classical Tikhonov regularized solution, with Algorithm 1 selecting specific outer iterations.

---

**Algorithm 1** Adaptive selection of number of inner PCG iterations. The PCG method may terminate early (at iteration  $k \leq M$ ) by the discrepancy principle.

---

```

m = m0 (initial guess)
M = M0 (initial number of inner iterations)
ε = estimated absolute noise level
μ = ||F(m) - b||
φ =  $\frac{1}{2}\mu^2 + \beta R(\mathbf{m})$ 
α = 4/3; ρ = 3/4; tol = 10-6 (for example)
repeat
  g = JT(m)(F(m) - b) + βR'(m)
  δm0 = 0
  {δm1, ..., δmk} = PCG (JTJ + β0R'', -g, R'', M) (save all k iterations)
  M = k
  μ1 = ||F(m + δmM) - b||
  μ0 = ||F(m + δmM-1) - b||
  φ1 =  $\frac{1}{2}\mu_1^2 + \beta R(\mathbf{m} + \delta\mathbf{m}_M)$ 
  φ0 =  $\frac{1}{2}\mu_0^2 + \beta R(\mathbf{m} + \delta\mathbf{m}_{M-1})$ 
  φold = φ
  if φ1 < φ0 and φ1 < φ then
    m = m + δmM
    M = ⌊αM + 1⌋
    μ = μ1
    φ = φ1
  else
    while φ0 < φ1 or φ1 > φ do
      if M > 1 then
        M = M - 1
        μ1 = μ0
        φ1 = φ0
        μ0 = ||F(m + δmM-1) - b||
        φ0 =  $\frac{1}{2}\mu_0^2 + \beta R(\mathbf{m} + \delta\mathbf{m}_{M-1})$ 
      else
        δm1 = ρδm1
        μ1 = ||F(m + δm1) - b||
        φ1 =  $\frac{1}{2}\mu_1^2 + \beta R(\mathbf{m} + \delta\mathbf{m}_1)$ 
      end if
    end while
    m = m + δmM
    μ = μ1
    φ = φ1
  end if
until μ ≤ ε or (φold - φ)/φold < tol

```

---

If  $\beta = 0$  and the estimated noise level  $\epsilon$  is at least as large as the (unknown) true noise level then the method will terminate as soon as  $\mu \leq \epsilon$ , before the minimum of  $\phi$  is found. In this case we have a nonlinear version of what is called *iterative regularization* in the context of linear problems [22, 23]. Following [10] we refer to this as dynamical regularization. If  $0 < \beta < \beta_\epsilon$ , or *tol* is not very small, then we will still terminate before the minimum of  $\phi$  is found, but the regularization is now a hybrid of dynamical and Tikhonov regularization.

Employing dynamical regularization removes the need to search for the coefficient  $\beta$ . Furthermore, especially in the presence of appreciable noise, no optimization problem ever needs to be solved to high accuracy here. Thus we believe that dynamical regularization can be significantly more efficient in practice. On the other hand, recall that in sensitive applications the nonlinear problem gets more difficult to solve as  $\beta$  decreases. In fact, a continuation or homotopy method in the Tikhonov coefficient (called “cooling” by geophysicists) is often employed in practice, where  $\beta$  is gradually decreased. In the limit case of dynamic regularization where  $\beta = 0$ , the solution of the nonlinear problem may depend more on the initial iterate. Of course, a continuation method in  $\beta$  may be employed here as well.

## 4 Numerical examples involving Algorithm 1

In this section we study the DC resistivity problem on a unit square, in the hope of demonstrating the efficacy of the method outlined in Section 3. The inverse problem is to recover  $\sigma$ , which at each grid point takes on one of two specified values, from measurements of  $u^i$  on the boundary. Instead of the conductivity  $\sigma$  we shall display results for the resistivity  $\rho = 1/\sigma$ .

For Experiment  $i$ ,  $q^i$  consists of a positive point source on the left boundary and an opposite source on the right boundary, so

$$q^i(\mathbf{x}) = \delta_{\mathbf{x}, \mathbf{p}_L^i} - \delta_{\mathbf{x}, \mathbf{p}_R^i},$$

where  $\mathbf{p}_L^i$  and  $\mathbf{p}_R^i$  are located on the left and right boundaries. Different data sets are obtained by varying the positions  $\mathbf{p}_L^i$  and  $\mathbf{p}_R^i$  of the two opposing sources. We place the left source at  $K$  equidistant points including the corners, and similarly for the right source, in all possible combinations. This gives a total of  $s = K^2$  data sets. Voltage is measured on the entire boundary. A synthetic conductivity model is used to compute the data  $\mathbf{b}$ , which, unless indicated otherwise, is calculated on a grid that is twice as fine as the grid used for the reconstruction, and Gaussian noise is added to it. All solutions of (5) in this section, implementing PDE solves, were carried out using a direct method.

The conductivity distributions used to synthesize data consist of objects with conductivity  $\sigma_I = 1$  placed in a background of conductivity  $\sigma_{II} = 0.1$ . Two scenarios are envisioned. In the first scenario no prior knowledge about the conductivity is

assumed, and reconstructions are accordingly carried out for  $\mathbf{m}$  with  $\sigma = e^m$  using an L2 regularization. In the second scenario the bi-constant nature of the true solution, including the values of  $\sigma_I$  and  $\sigma_{II}$ , is assumed known, and reconstruction is obtained employing the level set method using the sharpening function (3). As can be expected, appropriately using the prior information results in more accurate reconstructions. Results are presented for both Tikhonov and dynamical regularizations in order to demonstrate that our technique applies to both methods.

## 4.1 Example 1

Three square regions are placed as depicted in Figure 1(a). We employ  $s = 9$  experiments with  $K = 3$  sources and sinks, and compute on a  $65^2$  grid. The relative noise level is 2.5%, and we aim to reconstruct up to a misfit<sup>3</sup> of 3%.

For the Tikhonov L2 regularization we found by (laborious) trial and error that the solution converges to the desired misfit for approximately  $\beta = 6.5 \times 10^{-5}$ . The

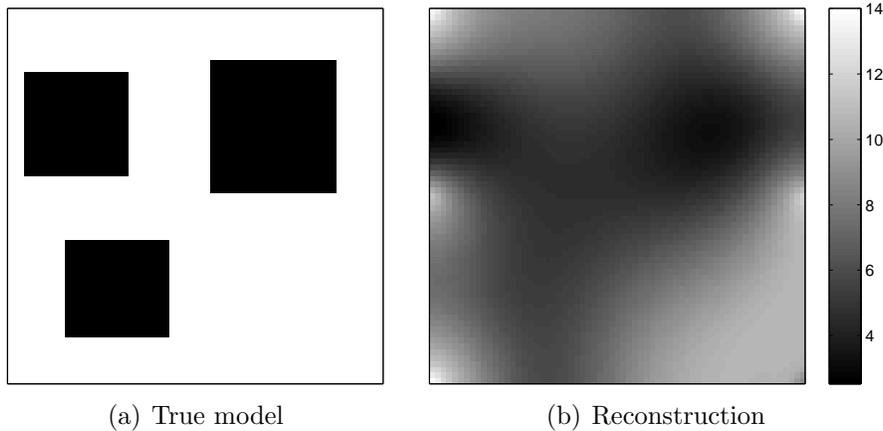


Figure 1: L2 reconstruction of the resistivity using Tikhonov regularization with  $\beta = 6.5 \times 10^{-5}$  (Example 1).

necessary conditions for (7) were solved first using the damped GN method (8) (i.e., setting  $\beta_0 = \beta$ ) with a fixed number  $M_n$  of inner PCG iterations, and then using Algorithm 1. The outer iteration was deemed to have converged if the relative decrease in objective function was less than  $10^{-3}$ .

For  $M_n$  fixed at 20 inner iterations, the reconstruction depicted in Figure 1 was obtained in 4 outer iterations, with a total of  $4 \times 2 \times 20 \times 9 = 1440$  PDE solves. For  $M_n = 3$  convergence was achieved in 26 outer iterations with 1404 PDE solves. Using Algorithm 1 the problem was solved in 4 iterations with  $(M_1, \dots, M_4) = (5, 8, 7, 10)$ ,

<sup>3</sup>The misfit is defined as  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|/\|\mathbf{b}\|$ . Throughout, when a noise level is mentioned in percentage, it is the relative rather than the absolute misfit that gets used.

a total of 540 PDE solves, which represents a substantial saving. The results are summarized in Table 1.

It may be objected that the reconstruction in Figure 1 looks quite bad, and indeed it does. Nevertheless, in a practical situation one may be interested at first in a very rough idea whether any desired structure is present, and only then possibly proceed with a larger experimental setting to refine the reconstruction. So, these types of reconstruction are performed quite often in practice.

$M_n$	$W$
20	1440
3	1404
adaptive	540

Table 1: Example 1. Work  $W$  (namely, the number of PDEs solved) for fixed and adaptive choices of  $M_n$ .

## 4.2 Example 2

Next, we apply the level set method (4) to the same problem with the additional information on the exact solution assumed known. By trial and error a value of  $\beta = 10^{-3}$  was found to result in the desired misfit for the Tikhonov method.

We also reconstruct using dynamical regularization, setting  $\beta = 0$  and terminating when the misfit drops below 3%. For the latter method, where the algorithm “attempts” to solve an underdetermined problem, the obtained solution depends more on the starting point, which plays the role of prior information, in a manner that the Tikhonov reconstruction does not.

We used three different starting configurations, two of which are depicted in Figure 2. The third initial guess is  $\mathbf{m} = \mathbf{0}$ , which may seem odd at first. However, in this case we interpret the function  $P$  defined in (3) as just a nonlinear transfer function. If the slope of the level set function stays small, values between  $\sigma_I$  and  $\sigma_{II}$  can be obtained for  $\sigma$ . In this case the reconstruction sharpens at each outer iteration, but always remains somewhat blurry. Since this starting guess has no bias as to where the objects to be “discovered” would be, it may be useful in practice. However, this is no longer a pure level set method. Figure 2 displays the obtained results.

It may be argued that the dynamical regularization result in Figure 2(d) is clearer than the Tikhonov result 2(c). But it may also be argued, ironically, that the experimental setting does not allow for a reliable, clear model approximation. It is also possible to threshold the image in Figure 2(f) for higher contrast results.

The corresponding work requirements are summarized in Table 2. It is apparent that if an optimal fixed  $M_n$  is selected then the adaptive Algorithm 1 performs just slightly better; its main advantage is that  $M_n$  is selected automatically. Note also

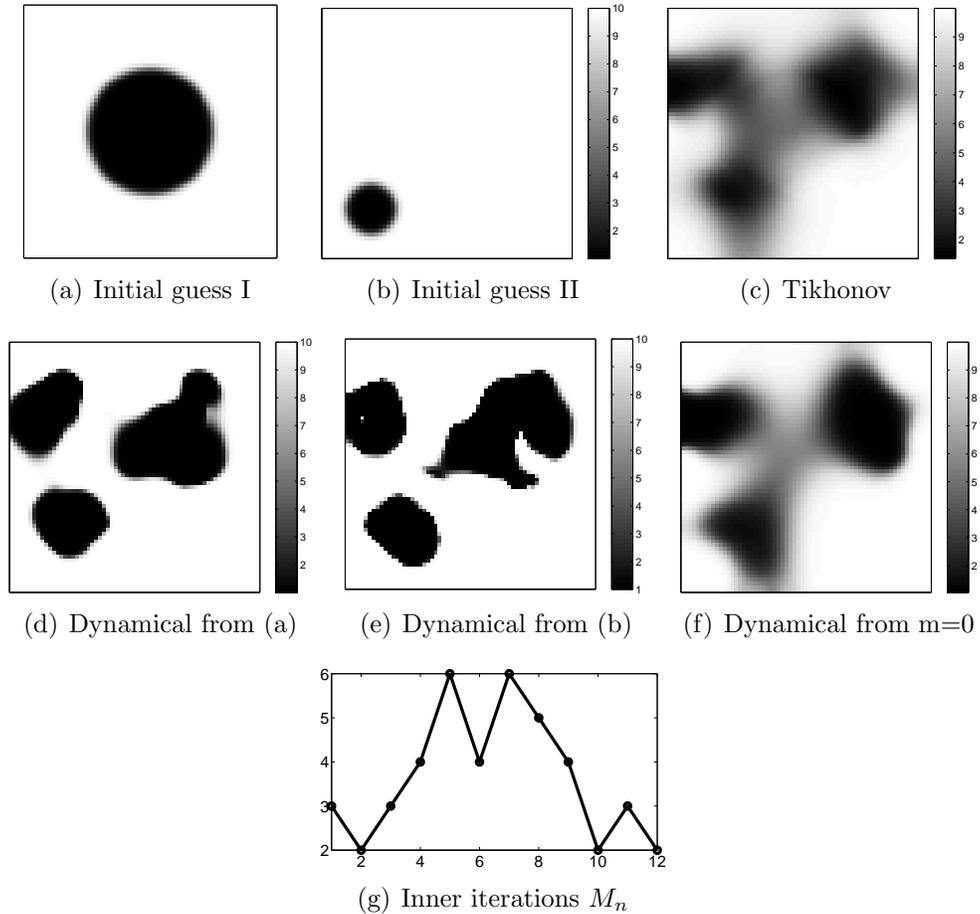


Figure 2: Example 2. (a,b) initial guesses; (c) reconstruction using Tikhonov; (d,e,f) reconstruction using dynamical regularization from the three different starting points; (g)  $M_n$  versus outer iteration  $n$  for the Tikhonov reconstruction.

that the performance difference between Tikhonov (with the given  $\beta$ ) and dynamic regularization is not significant.

### 4.3 Example 3

For lower noise levels the required number of inner iterations tends to increase, so we examine the performance of Algorithm 1 for the same setting as in Example 2 but with noise level 0.8%.

Figure 3 displays the reconstruction results. As this requires a more accurate forward model we employed a  $257^2$  grid for the reconstruction, with artificial data computed on a  $513^2$  grid. The starting configuration of the level set function was  $\mathbf{m} = \mathbf{0}$  and dynamical regularization was employed. The work tallies are summarized in Table 3.

$M_n(T)$	$W(T)$	$M_n(D)$	$W(D)$
2	1548		
5	900	5	1000
20	$\infty$	20	$\infty$
adaptive	792	adaptive	756

Table 2: Example 2. Work  $W$  for fixed and adaptive choices of  $M_n$  with Tikhonov (T) and dynamical (D) regularizations;  $\infty$  indicates no convergence.

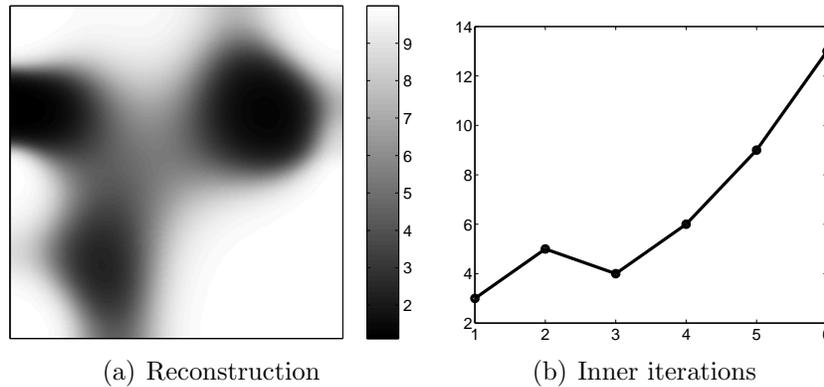


Figure 3: Example 3. Reconstruction result and adaptive iteration counts for the smaller noise level of 0.8%.

Note that the number of inner iterations goes up but reconstruction is not more accurate visually, as  $s = 9$  experiments are apparently insufficient to increase resolution. Here the value of  $M_n = 5$  yields significant inefficiency. The better  $M_n = 10$  is majorized by the adaptive algorithm, but again only by a factor of less than 2.

$M_n$	$W$
5	19089
10	1260
adaptive	720

Table 3: Example 3. Work  $W$  for fixed and adaptive choices of  $M_n$ .

Viewing the results in Tables 2 and 3 together clearly demonstrates the advantage of our adaptive algorithm.

## 5 A stochastic method for reducing the number of experiments per iteration

Must we solve PDEs of the form (5)  $s$  times at each outer iteration  $n$ ? Let us first make two general observations in this regard. The first is that it is certainly easy to imagine, especially if we have a very large number of experiments  $s$ , that they may be severely redundant. Let us refer to a situation where using only every  $l$ th experiment for some  $l \gg 1$  still yields acceptable results, as “embarrassing redundancy”. Gaining major savings in a situation like that is of course rather welcome, but is far simpler than what we have in mind, so we assume no such redundancy and attempt to verify this in the experiments below.

The second observation is that at the earlier outer iterations  $n$ , using a relatively small number of data sets  $s_n$  would possibly suffice to improve the approximate solution  $\mathbf{m} = \mathbf{m}_n$  towards convergence. Indeed, recall that faster convergence orders in Newton-like methods kick in only when  $\mathbf{m}$  is already close to a local solution. Once we get closer to the target the time comes to use more experiments, but even then there is the uncertainty in data measurements and the entire mathematical model that limits practically achievable accuracy. Practical heuristic ideas involving variable  $s_n$  have been around for a while [6].

Below we consider a stochastic method for selecting  $s_n$  experiments that are combinations of the  $s$  given ones and a practical cross-validation algorithm to control the size of  $s_n$ . Note that since  $s_1$  is typically chosen very small, and  $s_N$  is expected to be of the order of  $s$  for a moderate number of outer iterations  $N$ , the sequence  $\{s_n\}$  is expected to grow rapidly as a function of  $n$ .

Let us rewrite the objective function (7) as

$$\phi(\mathbf{m}; \beta) = \frac{1}{2} \sum_{i=1}^s \|\mathbf{F}_i(\mathbf{m}) - \mathbf{b}_i\|^2 + \beta R(\mathbf{m}), \quad (14)$$

and introduce a stochastic variable  $\mathbf{w} = (w^1, \dots, w^s)$ . The probability distribution of  $\mathbf{w}$  is chosen to satisfy

$$\langle w^i w^j \rangle = \delta^{ij}, \quad (15)$$

where  $\langle \rangle$  denotes the expectation value as before. We can now write (14) as

$$\phi(\mathbf{m}; \beta) = \frac{1}{2} \langle \left\| \sum_{i=1}^s w^i (\mathbf{F}_i(\mathbf{m}) - \mathbf{b}_i) \right\|^2 \rangle + \beta R(\mathbf{m}), \quad (16)$$

and consider approximating the expectation value at outer iteration  $n$  by random sampling from a set of  $s_n$  vectors  $\mathbf{w}$ , with  $s_n \leq s$ .

The method of *simultaneous random sources* [33, 25, 31, 24, 18] selects the  $w^i$ 's to be randomly  $\pm 1$ , so that (15) is satisfied. A different single realization (i.e.,  $s_n = 1$

for all  $n$ ) is chosen at each outer iteration, combined with an averaging procedure. This leads to a method called *stochastic approximation* (SA) in [18]. That paper goes on to consider a *stochastic averaging approximation* (SAA) technique, where a smallish number of random realizations of  $\mathbf{w}$  (that is far less than  $s$ ) is used in all outer iterations.

Such methods can be very efficient if the data in different experiments are measured at the same locations, i.e.,  $Q^i = Q$ , for in that case we have

$$\sum_{i=1}^s w^i \mathbf{F}_i = \sum_{i=1}^s w^i Q^i A(m)^{-1} \mathbf{q}^i = QA(m)^{-1} \left( \sum_{i=1}^s w^i \mathbf{q}^i \right), \quad (17)$$

which can be computed with a single PDE solve per realization of  $\mathbf{w}$ . However, if the receivers are different for different experiments, as is a common situation in practice (recall for instance the experimental setup described in Section 4), then up to  $s$  PDE solves are required to compute  $\sum_{i=1}^s w^i \mathbf{F}_i$ , and the method is no longer efficient.

We propose a different yet somewhat related approach, where (16) is estimated by random sampling using a different set of samples for  $\mathbf{w}$  at each outer iteration. In a typical iteration we choose the vectors  $\mathbf{w}$  to each consist of a vector with a single nonzero component at a random location which has value  $s$ . Such a distribution also satisfies (15) and allows the computation of  $\sum_{i=1}^s w^i \mathbf{F}_i$  with a single PDE solve per realization. Clearly, this choice boils down to selecting a random subset of the given experiments at each outer iteration.

Furthermore, cross validation is employed to control the number of samples required. The function  $\phi(\mathbf{m}; \beta)$  is estimated based on a set  $\mathcal{S}$  of  $s_n$  random samples of  $\mathbf{w}$ , and a GN step is performed. Then we cross-validate the step by checking that (16) estimated on a different set of  $s_n$  random samples also decreases. If it does then we conclude that  $s_n$  samples were sufficient; if not then we increase  $s_n$ , potentially until finally all the available experiments are employed.

We start the outer iteration with a small set  $\mathcal{K}$  of randomly drawn experiments, for instance with just 2 elements. At iteration  $n$  we use some of the experiments in  $\mathcal{K}$  to compute the next iterate, and the rest to validate the update. Thus, the cross validation [15] is performed based on a partition of  $\mathcal{K}$  into two sets  $\mathcal{S}$  and  $\mathcal{C}$ , normally of size  $s_n$  each, where  $\mathcal{S}$  is used to compute the model update and  $\mathcal{C}$  to validate the update. If the validation fails then the size of  $\mathcal{K}$  is doubled (i.e., we use additional experiments drawn at random).

After each outer iteration we estimate the actual misfit based on the experiments in  $\mathcal{S}$  and  $\mathcal{C}$ . If this misfit estimate satisfies the discrepancy principle then we validate it based on all experiments and terminate the algorithm when appropriate. The most obvious validation is a computation of the misfit based on the complete set of experiments. Alternatively, various estimates of the misfit and their confidence intervals based on random sampling of the total set of experiments could be performed. The resulting method is outlined in Algorithm 2.

---

**Algorithm 2** Adaptive selection of number of used experiments for dynamical regularization.  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|_{\mathcal{X}}$  denotes the norm over the experiments in set  $\mathcal{X}$  only.

---

```

m = m0
ε = (absolute) noise level
S = 2
K = a set of S random experiments
partition K in two equal size sets S and C
μc0 =  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|_{\mathcal{C}}$ 
done = false
dostoch = true
repeat
  if dostoch then
    compute update m ← m + δm as in algorithm 1, using only the data in S
    μc =  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|_{\mathcal{C}}$ 
    μS =  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|_{\mathcal{S}}$ 
    if μc < μc0 then
      μest =  $\sqrt{s(\mu_c^2 + \mu_s^2)}/S$ 
      if μest < ε and m passes validation then
        done = true
      else
        S = C
        C = a randomly selected set of S/2 experiments not in S
        μc0 =  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|_{\mathcal{C}}$ 
      end if
    end if
  else
    S ← 2S
    if S > s then
      dostoch = false
    else
      K = a set of S random experiments
      partition K in two equal size sets S and C
      μc0 =  $\|\mathbf{F}(\mathbf{m}) - \mathbf{b}\|_{\mathcal{C}}$ 
    end if
  end if
end if
until done

```

---

## 6 Numerical examples involving Algorithms 2 + 1

In Section 4 we have employed only a small number of  $s = 9$  right hand sides. In this situation we have found that using the stochastic method from Section 5 requires about the same amount of computational effort, as the majority of iterations utilize the full set of experiments. By contrast, in the present section we consider simulations with a larger number of experiments and show that substantial additional savings can be obtained with the stochastic method.

### 6.1 Example 4

We consider a true model consisting of two irregular shapes as depicted in Figure 4, with the same values as before for  $\sigma_I$  and  $\sigma_{II}$ . The number of experiments is  $s = 36$ , the noise level is 2.5%, and a  $65^2$  grid is used. Dynamical regularization (i.e.,  $\beta = 0$ ) is employed.

The L2 solution using Algorithm 1 is depicted in Figure 4(b), while Figure 4(c) shows the reconstruction obtained using the stochastic algorithm proposed in Section 5. The number of PDE solves required on 3 trials was 480, 352, and 370, an average of about 400.

Next, Figure 4 shows reconstructions using the level set method, starting either at the initial guess depicted in Figure 2(a) or at  $\mathbf{m} = \mathbf{0}$ , and using Algorithm 1 either normally or through the stochastic method of Algorithm 2. The results starting from the circle differ in details of the shape, whereas the results starting at  $\mathbf{m} = \mathbf{0}$  are visually almost indistinguishable amongst different random choices of the experiments during the outer iterations. The work comparison counts are summarized in Table 4. We observe that substantial savings can occasionally be made with the stochastic method.

regularization	experiments	all	stochastic
L2	36	2880	401
levelset(0)	36	2952	686
levelset	36	2880	1528
L2	625	162500	3196
levelset(0)	625	23125	706

Table 4: Examples 4 and 5. Listing work  $W$  required with the stochastic method compared to using all experiments. Levelset(0) indicates the level set method formulation starting at  $\mathbf{m} = \mathbf{0}$ .

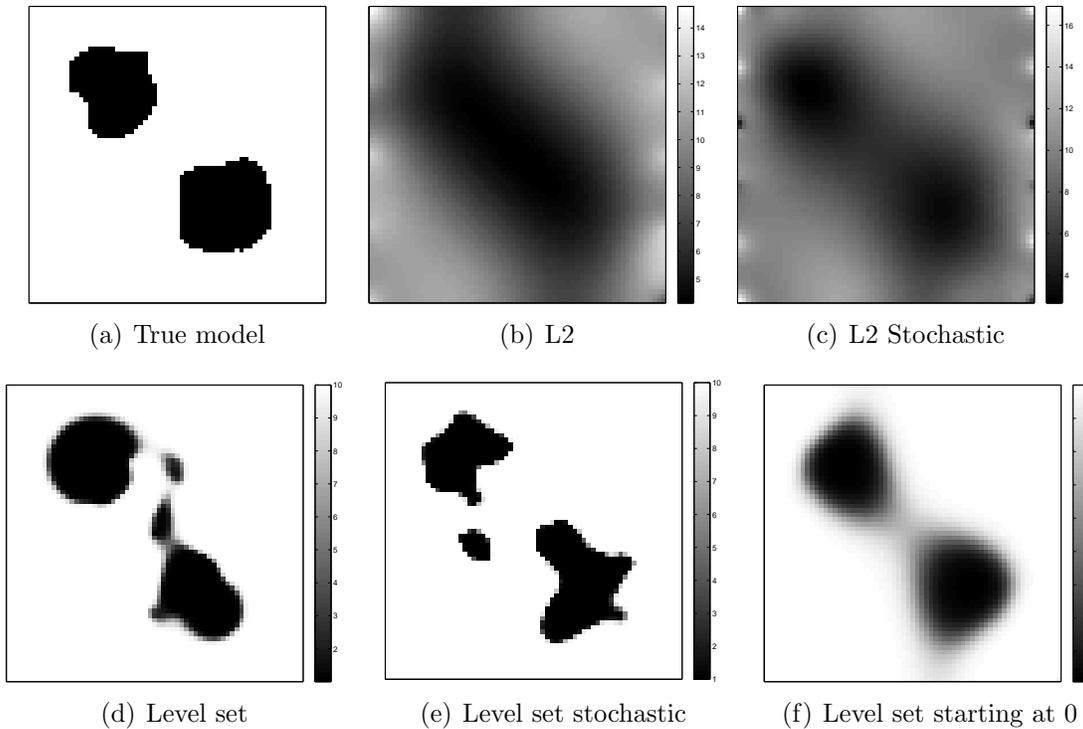


Figure 4: Example 4. Reconstructions in 2D, comparing usage of all experiments vs. Algorithm 2 for both L2 and level set methods.

## 6.2 Example 5

If even more experiments are available, the advantage of the stochastic method becomes more dramatic. Figure 5 depicts the L2 and level set (starting at 0) reconstructions with the same setup but with 625 experiments. As the corresponding last two rows of Table 4 show, about 30 – 50 times fewer PDE solves are required by the stochastic method.

Note that the quality of the reconstruction is noticeably better with this many sources (compare Figure 5 to Figure 4(b,c,f)), so it is not true that we could have just tossed out a large number of these experiments up front.

Incidentally, upon setting the noise level to 0 and synthesizing the data on the same grid (thus committing an “inverse crime”), the level set method using this many experiments reconstructs the “true solution” precisely at the given resolution.

## 6.3 Example 6

Injection of tracers is often performed by hydrologists in an effort to obtain information about subsurface hydrologic properties in the earth [28]. The present example in three space dimensions employs surface electrodes to image the injection of a saline

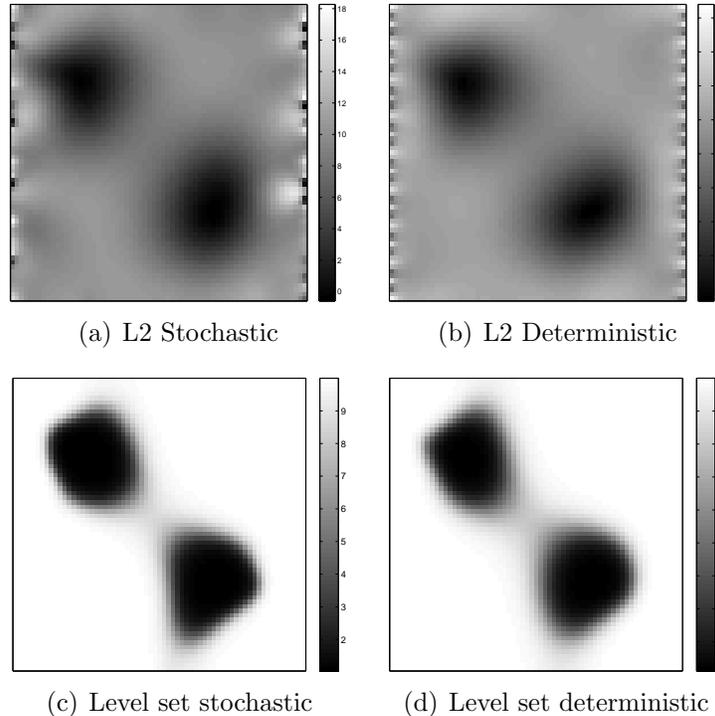


Figure 5: Example 5. Setting as in Example 4 but with 625 right hand sides. The level set reconstruction was started at  $\mathbf{m} = \mathbf{0}$ .

tracer into the subsurface. Here we simulate the tracer-saturated soil as a  $10 \Omega \cdot m$  resistivity material and the soil as a  $200 \Omega \cdot m$  material. On the surface we deploy 96 electrodes; 38 of these electrodes function as potential electrodes only, while the remaining 58 serve as both potential and current electrodes. The example involves  $s = 41$  independent current pairs. For each of these pairs, data are recorded at the remaining 94 electrodes. Data are recorded using dipoles, so for each current pair we record 93 independent data. This results in a total of 3813 data for the entire survey. Note that the measurement locations  $Q^i$  vary with the experiment counter  $i$  in this example. Artificial data is computed on a  $64^3$  grid, then polluted with 3% noise. The reconstructions are done on a  $16^3$  grid. The discretized PDEs (5) were solved as described in [10] with an error tolerance that is sufficiently small to consider these solutions as “exact”.

Figure 6 displays the true model resistivity and various reconstructions. Also displayed are horizontal slices at varying depth starting from the top (ground level) at top-left. The grey levels range from black (0) to white (200). The reconstructions all use dynamical regularization. Notice that in Figure 6(c) the blurring increases with depth, whereas in Figure 6(d) there is a shape distortion in that the bottom shape extends too far down. The L2 result resolves the two shapes a little bit in the top row, but they hopelessly merge deeper down.

Table 5 records work estimates. Compared are Algorithms 2+1 against a “good plain vanilla” method, by which we mean that (i) data from all experiments are used at every iteration (i.e.,  $s_n = s$ ), and (ii) the number of inner iterations  $M_n$  is fixed at  $M_n = 10$  for the L2 regularization and  $M_n = 5$  for the level set method. These values of  $M_n$  were determined to be good ones by trial and error.

regularization	vanilla	stochastic
L2	4920	2160
Levelset(0)	3690	1577
Levelset	4100	1785

Table 5: Example 6. Work  $W$  required by Algorithm 2+1 (“stochastic”) compared to using all experiments and a fixed number of PCG iterations (“vanilla”). The work level in the rightmost column represents an average over three trials. Levelset(0) indicates the level set method starting at  $\mathbf{m} = \mathbf{0}$ .

While the sophisticated method is more efficient, the factor of improvement is only a bit over 2 in this example, which must be considered a merely moderate gain in the present setting.

## 6.4 Example 7

For this example we use the same geometry and resistivity values as in the previous one. Data is collected at a grid of 81 receivers located at the surface. Four vertical boreholes near the corners are used to place virtual sources and sinks at various depths at a total of 17 locations. We assume the boreholes to be narrow enough to not disturb the conductivity [28, 17]. For each diagonally opposed pair of boreholes,  $17 \times 17 = 289$  pairs of sources and sinks are placed, forming a total of  $s = 578$  experiments. A slanted cylindrical object depicted in Figure 7(a,b) was used to generate  $u$ -values which were subsequently polluted with 1% Gaussian noise to yield the “observed data”. The termination criterion, using the discrepancy principle, was set at 1.4%. A comparison was made between using Algorithms 1 and 2 (i.e., using all experiments at each outer iteration vs. the stochastic way).

Note that the L2 reconstruction in Figure 7(c) becomes blurry about one quarter of the way down and does not seem to suggest a cylinder. The level set based reconstruction starting at  $\mathbf{m} = \mathbf{0}$  (Figure 7(g,h)) does seem to indicate a cylinder, but the object is placed too high and there are “ghost” objects at the bottom. Since the ghost objects are far from the sensors, it is (hopefully) unlikely that they would be trusted in practice. However, this poor result does indicate the usefulness of having several reconstruction methods at hand. Finally, the level set method starting at the shape depicted in Figure 7(e,f) works very well! It is the only decent reconstruction in this example.

Results regarding the computational effort are summarized in Table 6. Here, unlike in the previous example, the stochastic method provides impressive savings factors of up to 50, with the highest factor occurring for the good reconstruction! Indeed, the stochastic approach is useful when there are *many* right hand sides.

regularization	Algorithm 1	Algorithm 2
L2	102350	24369
Levelset(0)	55488	14116
Levelset	21964	400

Table 6: Example 7. Work  $W$  required with Algorithm 2 compared to using all experiments and Algorithm 1.

## 7 Conclusions

The adaptive Algorithm 1 for selecting the number of inner PCG iterations  $M_n$  performs remarkably well, both for the L2 regularization and for the level set method. The value of  $M_n$  is adjusted automatically, and more iterations than needed to make significant progress at each outer iteration are avoided.

For the problems considered in [10],  $M_n$  stays for all  $n$  close to being between 3 and 5, which agrees with what was experimentally determined there as optimal. However, for problems with lower noise level the best  $M_n$  gradually increases, and we obtain using our adaptive algorithm solutions with the desired accuracy that cannot be efficiently obtained while keeping  $M_n$  small throughout.

The stochastic method implemented in Algorithm 2 further improves efficiency of the L2 and level set methods, for both Tikhonov and dynamical regularizations. The extent of such improvement, however, appears to vary greatly from modest to highly impressive. Note that all our computations are performed for examples where there is no embarrassing redundancy. Private consultations indicate that such is also the fate of other stochastic algorithms proposed in the literature for problems of the sort considered here. The claims appearing in the literature in this regard should therefore be scaled in such light. An advantage of our cross validation approach is that it enables to claim great success when that is possible while avoiding failure in other situations.

At the innermost loop of any relevant method is the solution of (5), which is a large linear algebra system. If an iterative method is employed for this purpose then one can play with its error tolerance. We have ensured that such solutions are performed accurately enough so as not to form a factor while comparing PDE solves required by the different methods considered in this paper.

The L2 regularization approach, along with its well-known deficiencies, is a remarkable work-horse. However, if a priori information is available that enables using

a level set method then this may greatly improve both stability and accuracy. Even if the solution is not piecewise constant, if sufficiently many data sets are available then using the L2 approach reveals locations of larger solution gradients and a switch to total variation regularization can be gradually made [19].

While the 3D experiments are significantly larger and more time consuming than those in 2D (the limitation in Example 6 to a  $16^3$  grid results in roughly overnight-long runs on our laptop, thus corresponding well to our sleeping habits), our comparative conclusions do not vary much. Indeed, in terms of reconstruction quality the smaller problems are, if anything, harder. This is not really surprising when considering the fact that the ratio of “boundary to domain size” decreases as the number of dimensions increases.

While revising this article we became aware of the report [14] that had been submitted for publication roughly at the same time as our work. In it the authors prove in a general setting linear convergence rate for a gradient descent method with a constant step size, provided  $s_n$  increases exponentially with  $n$ . While the considered method is far slower than ours and the regularity assumptions don’t necessarily hold in our case, this theory generally agrees with our practical observations regarding the fast growth of  $s_n$ .

Our cross validation algorithm can also be applied to the method of simultaneous random sources, and preliminary experiments indicate that this combines into a very effective method in case that  $Q^i = Q, \forall i$ . We leave this to future work.

**Acknowledgment:** We are grateful to Drs. Eldad Haber and Felix Herrmann for introducing us to stochastic methods in the present context and for several fruitful discussions.

## References

- [1] A. Alessandrini and S. Vessella. Lipschitz stability for the inverse conductivity problem. *Adv. Appl. Math.*, 35:207–241, 2005.
- [2] U. Ascher and E. Haber. A multigrid method for distributed parameter estimation problems. *J. ETNA*, 18:1–18, 2003.
- [3] U. Ascher, E. Haber, and H. Huang. On effective methods for implicit piecewise smooth surface recovery. *SIAM J. Sci. Comput.*, 28:339–358, 2006.
- [4] K. Astala and L. Paivarinta. Calderon inverse conductivity problem in the plane. *Annals of Math.*, 163:265–299, 2006.
- [5] J. Barcelo, T. Barcelo, and A. Ruiz. Stability of the inverse conductivity problem in the plane for less regular conductivities. *J. Differential Equations*, 173:231–270, 2001.

- [6] D. Bertsekas and J. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1996.
- [7] L. Borcea, J. G. Berryman, and G. C. Papanicolaou. High-contrast impedance tomography. *Inverse Problems*, 12:835–858, 1996.
- [8] M. Cheney, D. Isaacson, and J. C. Newell. Electrical impedance tomography. *SIAM Review*, 41:85–101, 1999.
- [9] K. van den Doel and U. Ascher. On level set regularization for highly ill-posed distributed parameter estimation problems. *J. Comp. Phys.*, 216:707–723, 2006.
- [10] K. van den Doel and U. Ascher. Dynamic level set regularization for large distributed parameter estimation problems. *Inverse Problems*, 23:1271–1288, 2007.
- [11] K. van den Doel, U. Ascher, and A. Leitao. Multiple level sets for piecewise constant surface reconstruction in highly ill-posed problems. *Journal of Scientific Computation*, 43(1):44–66, 2010.
- [12] K. van den Doel, U. Ascher, and D. Pai. Source localization in electromyography using the inverse potential problem. *Inverse Problems*, 27(2):025008, 2011.
- [13] H. W. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Kluwer, Dordrecht, 1996.
- [14] M. Friedlander and M. Schmidt. Hybrid deterministic-stochastic methods for data fitting. 2011. Submitted.
- [15] S. Geisser. *Predictive Inference*. New York: Chapman and Hall, 1993.
- [16] E. Haber and U. Ascher. Preconditioned all-at-one methods for large, sparse parameter estimation problems. *Inverse Problems*, 17:1847–1864, 2001.
- [17] E. Haber, U. Ascher, and D. Oldenburg. Inversion of 3D electromagnetic data in frequency and time domain using an inexact all-at-once approach. *Geophysics*, 69:1216–1228, 2004.
- [18] E. Haber, M. Chung, and F. Herrmann. An effective method for parameter estimation with pde constraints with multiple right hand sides. 2010.
- [19] E. Haber, S. Heldmann, and U. Ascher. Adaptive finite volume method for distributed non-smooth parameter identification. *Inverse Problems*, 23:1659–1676, 2007.
- [20] E. Haber and D. W. Oldenburg. A GCV based method for nonlinear ill-posed problems. *Computational Geophysics*, 4:41–63, 2000.

- [21] M. Hanke. Regularizing properties of a truncated newton-cg algorithm for non-linear inverse problems. *Numer. Funct. Anal. Optim.*, 18:971–993, 1997.
- [22] M. Hanke and P. C. Hansen. Regularization methods for large scale problems. *Survey on Mathematics for Industry*, 3:253–315, 1993.
- [23] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems*. SIAM, 1998.
- [24] F. Herrmann, Y. Erlangga, and T. Lin. Compressive simultaneous full-waveform simulation. *Geophysics*, 74:A35, 2009.
- [25] A. Juditsky, G. Lan, A. Nemirovski, and A. Shapiro. Stochastic approximation approach to stochastic programming. *SIAM J. Optimization*, 19(4):1574–1609, 2009.
- [26] J. Nocedal and S. Wright. *Numerical Optimization*. New York: Springer, 1999.
- [27] L. Paivarinta, A. Panchenko, and G. Uhlmann. Complex geometrical optics solutions for Lipschitz conductivities. *Rev. Mat. Iberoamericana*, 19:57–72, 2003.
- [28] A. Pidlisecky, E. Haber, and R. Knight. RESINVM3D: A MATLAB 3D Resistivity Inversion Package. *Geophysics*, 72(2):H1–H10, 2007.
- [29] A. Rieder. Inexact newton regularization using conjugate gradients as inner iteration. *SIAM J. Numer. Anal.*, 43:604–622, 2005.
- [30] A. Rieder and A. Lechleiter. Towards a general convergence theory for inexact newton regularizations. *Numer. Math.*, 114(3):521–548, 2010.
- [31] J. Rohmberg, R. Neelamani, C. Krohn, J. Krebs, M. Deffenbaugh, and J. Anderson. Efficient seismic forward modeling and acquisition using simultaneous random sources and sparsity. *Geophysics*, 75(6):WB15–WB27, 2010.
- [32] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.
- [33] A. Shapiro, D. Dentcheva, and D. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. Philadelphia: SIAM, 2009.
- [34] N. C. Smith and K. Vozoff. Two dimensional DC resistivity inversion for dipole dipole data. *IEEE Trans. on geoscience and remote sensing*, GE 22:21–28, 1984.
- [35] A. N. Tikhonov and V. Ya. Arsenin. *Methods for Solving Ill-posed Problems*. John Wiley and Sons, Inc., 1977.

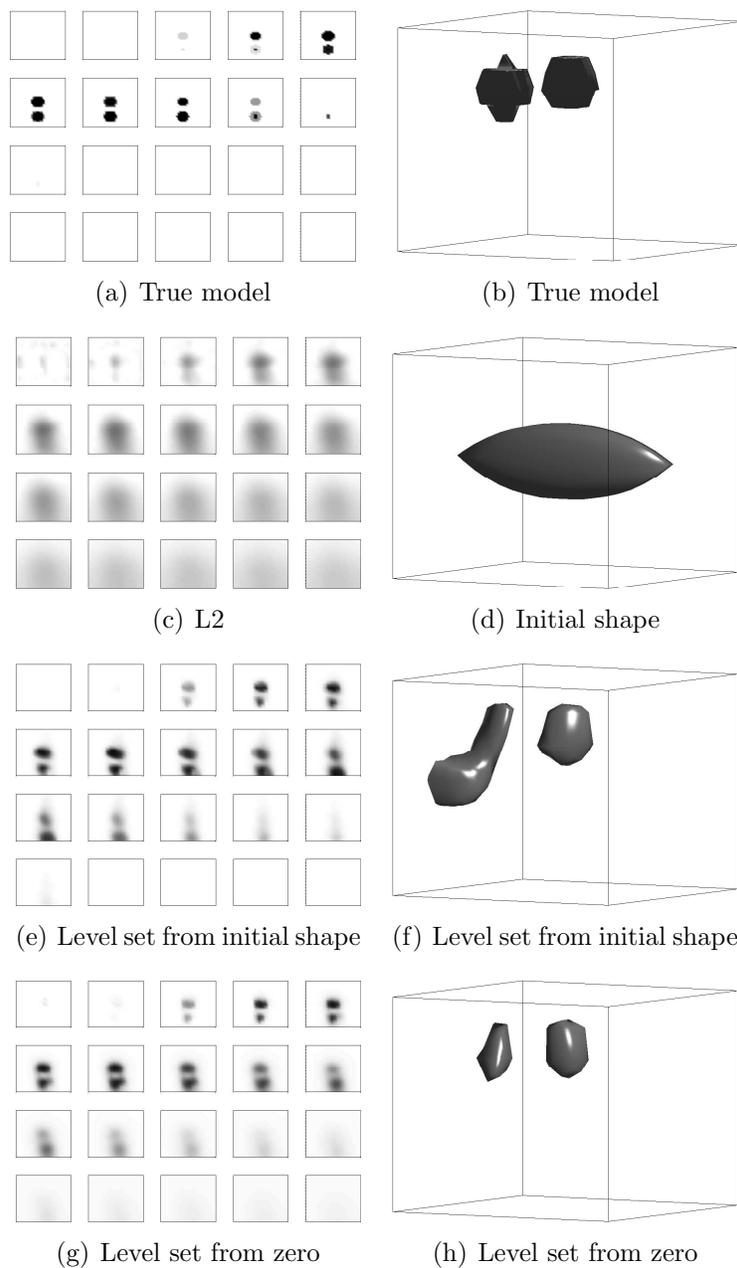


Figure 6: Example 6. Reconstruction of two submerged objects. Here,  $s = 41$  experiments injecting currents at the surface were used, with data collected at the surface.

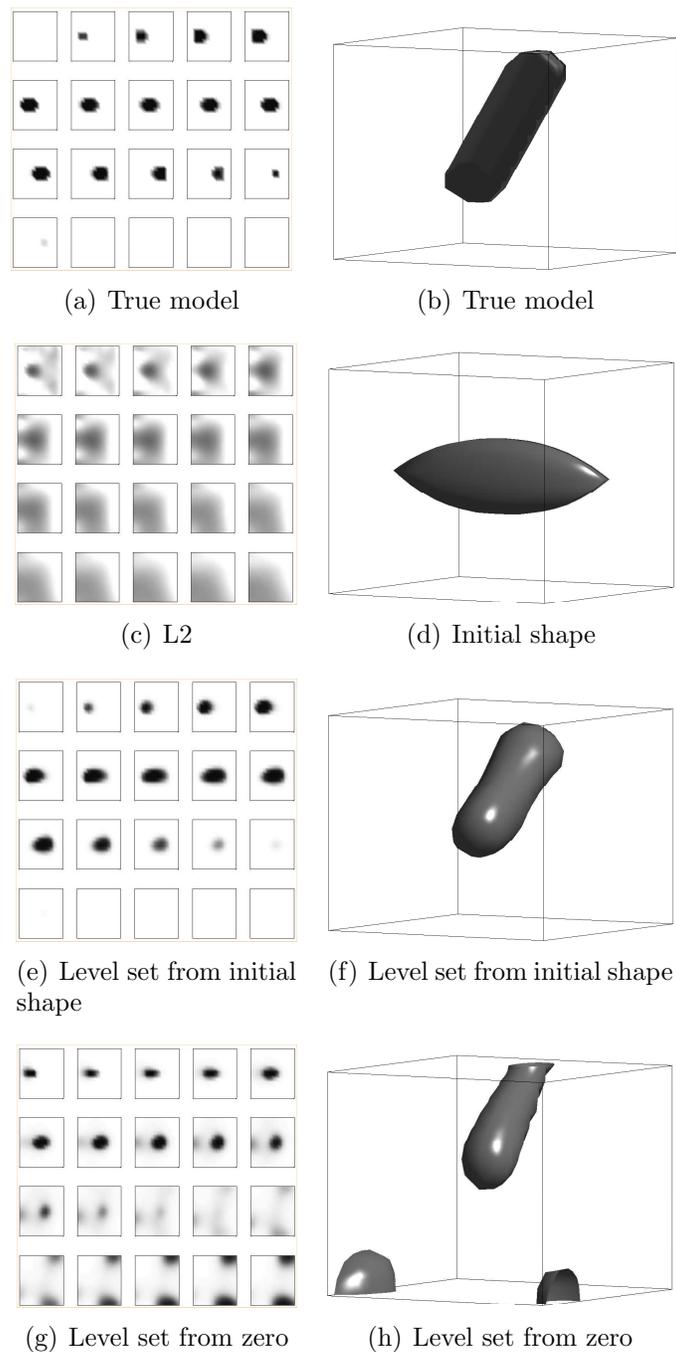


Figure 7: Example 7. Reconstruction of a slanted cylinder. Here,  $s = 578$  experiments injecting currents from boreholes were used, with data collected at the surface.