

RECOVERING MULTIPLE SMALL INCLUSIONS INSIDE AN ARBITRARY DOMAIN IN THREE DIMENSIONS USING A SINGLE MEASUREMENT

T. KOLOKOLNIKOV †AND A.E. LINDSAY ‡

ABSTRACT. We present and test an effective algorithm to recover multiple small heterogeneities inside arbitrary three dimensional bounded regions. In applications of electrical impedance tomography (EIT), such heterogeneities might represent tumours inside the body, weaknesses or cracks in a solid material or subterranean mineral deposits. The algorithm determines the location and size of all K heterogeneities using a single boundary measurement. The only nonlinear step in the algorithm is to find K roots of a single polynomial of degree K . The efficacy of this method is demonstrated by full numerical computation of the three dimensional forward problem. In addition, the methods employed can give information regarding the shape of irregular heterogeneities.

1. INTRODUCTION

The problem of inverse electrical impedance tomography (EIT) is to reconstruct the conductivity inside a cavity by applying a current to its periphery and measuring the response voltage. It is a classical problem which has been extensively studied since at least 1950's [1]; we refer the reader to reviews [2, 3] for an extensive list of applications and more recent works. The mathematical formulation of this problem is as follows. Let u be the voltage and σ be the conductivity inside a given bounded region $\Omega \subset \mathbb{R}^n$ for $n = 2, 3$. An input current $g(x)$ is applied along the boundary of the domain $\partial\Omega$ so that

$$(1) \quad \begin{cases} \nabla \cdot (\sigma \nabla u) = 0, & x \in \Omega; \\ \sigma \partial_n u = g(x), & x \in \partial\Omega; \end{cases}$$

The problem of EIT is to reconstruct the conductivity σ from measurements of the voltage on the boundary $\partial\Omega$, *i.e.* from $f(x) = u|_{\partial\Omega}$.

An important special case is when the conductivity is nearly uniform except for several small defects. In applications, these defects may represent a cancer [4, 5], a buried land-mine [6], or material weaknesses in a solid structure [7]. That is, consider a collection of K small non-overlapping inclusions $\{D_j\}_{j=1}^K \subset \Omega$, and a conductivity profile σ of form

$$(2) \quad \sigma(x) = \begin{cases} \kappa_j, & x \in D_j, \\ 1, & x \in \Omega \setminus \cup_{j=1}^K D_j. \end{cases}$$

The values $\{\kappa_j\}_{j=1}^K$ represent variable conductivities in each of the inclusions. A well-studied problem, especially in two dimensions, is to determine the location and shape of these discrete inclusions [8, 9, 10, 11, 7, 12, 13, 14, 15].

An effective algorithm for determining the location of a *single* defect is described in [9]. It is based on a layer potential reformulation of the problem and is applicable to arbitrary domains

in two or three dimensions. This algorithm was improved in [11] where the author introduced a more robust variation of this method, called “the effective dipole method”. These, and many other methods, use the “dipole approximation”; the fact that small inclusions effectively act as dipole forces.

For multiple inclusions, the problem was considered in [10] where an algorithm to recover K inclusions using only two boundary measurements was given. This was improved in more recent work by [8], where the authors give an algorithm which recovers multiple inclusions using a single measurement.

In the present work, we develop and test an algorithm for recovery of multiple inclusions inside an arbitrary three dimensional domain using a single measurement. The algorithm is an extension of the two-dimensional algorithm in [8] to three dimensions, using ideas developed in [16, 17, 18]. In two dimensions, we also show that the algorithm developed in [11] is a special case of the method presented in [8]. The method proposed herein generalize the very recent treatment of (c.f. [19]) to the problem of finding a single dipole inside a sphere.

In our derivation of the algorithm, the well known (c.f. [12]) asymptotic formulation of solutions to (1) in the limit of small inclusion plays a crucial role. More specifically, if each inclusion has a common scaling $\bar{D}_j = \varepsilon^{-1}D_j$ where $\bar{D}_j = \mathcal{O}(1)$, then as $\varepsilon \rightarrow 0$,

$$(3) \quad u(x) - u_0(x) = -\varepsilon^n \sum_{j=1}^K \nabla u_0(\xi_j) \mathcal{M}_j \nabla_y N(x, \xi_j) + \mathcal{O}(\varepsilon^{n+1}).$$

where the $\{\xi_j\}_{j=1}^K$ represent the centres of the inclusions. In the above formulation, $u_0(x)$ is the solution in the absence of heterogeneities and solves the *calibration problem*:

$$(4) \quad \Delta u_0 = 0, \quad x \in \Omega; \quad \partial_n u_0 = g, \quad x \in \partial\Omega; \quad \int_{\partial\Omega} g = 0.$$

The condition $\int_{\partial\Omega} g = 0$ is required for equation (4) to admit a consistent solution. The other key constituents of (3) are the *polarization matrices* $\{\mathcal{M}_j\}_{j=1}^K$ which are determined by the shape of each \bar{D}_j and $N(x, y)$, the *Neumann function* for Ω . As a consequence of (3), the following problem becomes central to our considerations;

Problem 1: *Suppose that $u(x)$ satisfies:*

$$(5) \quad \begin{cases} \Delta u = \sum_{j=1}^K w_j \cdot \nabla \delta(x - \xi_j), & x \in \Omega; \\ u(x) = f(x), \quad \partial_n u(x) = g(x), & x \in \partial\Omega. \end{cases}$$

Given $f(x)$ and $g(x)$, determine ξ_j and w_j , $j = 1 \dots K$.

In fact, we describe a constructive algorithm to solve Problem 1 for three dimensional domains Ω . The points ξ_j recover the locations of the inclusions whereas the vectors w_j are related to the shape of the inclusion via the polarization matrix. Specialising to the case where $D_j = B_{\varepsilon_j}(\xi_j)$ is a ball of a small radius ε_j centered at ξ_j , the explicit computation of the polarization tensor yields

[12],

$$(6) \quad w_j = \begin{cases} 2\pi \nabla u_0(\xi_j) \frac{1 - \kappa_j}{1 + \kappa_j} \varepsilon_j^2, & \text{in } \mathbb{R}^2; \\ 4\pi \nabla u_0(\xi_j) \frac{1 - \kappa_j}{2 + \kappa_j} \varepsilon_j^3, & \text{in } \mathbb{R}^3. \end{cases}$$

This formula will be used in numerical tests in §3 below.

As was done in two dimensions in [8], in three dimensions, Problem 1 is reduced to finding K roots of a certain complex polynomial of degree K , plus solving several linear systems. In this sense, the algorithm is deterministic (non-iterative except for finding the roots of a K -th degree polynomial, which is a standard numerical procedure). Note that in three dimensions, there are $6K$ unknowns (three components for each of ξ_j and w_j) that must be recovered. This makes a deterministic method especially valuable, since iterative methods often have convergence problems when faced with so many unknowns.

The efficacy of these algorithms is demonstrated on a number of sample problems comprised of multiple inclusions inside non-spherical domains. We employ the finite element solver FlexPDE to obtain a numerical solution to the forward problem (1) and the boundary data $f(x) = u|_{\partial\Omega}$ for arbitrary regions. Numerical experiments confirm the relative robustness of our algorithm (typical values we tried are $K = 4$ with 24 unknowns to solve for in \mathbb{R}^3). Interestingly, despite the fact that the algorithm is designed to find small ball-like inclusions, it can determine the shape of more general inclusions by approximating more general shapes using multiple inclusions. We now state our main results.

1.1. Description of the algorithm. We now state the algorithm in three dimensions (for completeness, the two-dimensional algorithm, which was derived in [8], is given in Appendix A).

Algorithm 1. (Solution to Problem 1 in \mathbb{R}^3).

(1) Compute the $2K$ complex integrals

$$(7) \quad A_n = \int_{\partial\Omega} f(y)(y_1 + iy_2)^{n-1}(\hat{n}_1 + i\hat{n}_2)dS(y) - \int_{\partial\Omega} \frac{1}{n}g(y)(y_1 + iy_2)^n dS(y), \quad n = 1 \dots 2K$$

where $\hat{n} = (\hat{n}_1, \hat{n}_2, \hat{n}_3)$ is the normal vector, and $i = \sqrt{-1}$.

(2) Solve the system

$$(8) \quad A_n = \sum_{j=1}^K \omega_j \eta_j^{n-1}, \quad n = 1 \dots 2K$$

using Lemma 1 to recover ω_j, η_j . The first two components of ξ_j, w_j are then given by $(\xi_{j1}, \xi_{j2}) = (\text{Re}(\eta_j), \text{Im}(\eta_j))$ and $(w_{j1}, w_{j2}) = (\text{Re}(\omega_j), \text{Im}(\omega_j))$.

(3) Compute the $2K$ complex integrals

$$(9) \quad B_n = \int_{\partial\Omega} f(y) \left\{ (n-1)(y_1 + iy_2)^{n-2} y_3 (\hat{n}_1 + i\hat{n}_2) + (y_1 + iy_2)^{n-1} \hat{n}_3 \right\} dS(y) \\ - \int_{\partial\Omega} g(y)(y_1 + iy_2)^{n-1} y_3 dS(y), \quad n = 1 \dots 2K.$$

(4) Solve the linear system of $2K$ variables

$$(10) \quad B_n = \sum_{j=1}^K (n-1)\eta_j^{n-2}\omega_j\xi_{j3} + \eta_j^{n-1}w_{j3}, \quad n = 1 \dots 2K$$

for the third components $\xi_{j3}, w_{j3}, j = 1 \dots K$.

We make several observations. There are three numerical tasks involved: computing $4K$ complex surface integrals (7,9); computing the roots of a K -th degree polynomial; and solving several linear systems of $O(K)$ variables. To compute the integrals, one must discretize the surface of the domain using N meshpoints with $N \gg K$. The only nonlinear step is solving for K roots of a K -degree polynomial, which is a standard numerical routine. The total running time is then of $O(KN)$ as computing the integrals is the most time-consuming of the three tasks, assuming that $K \ll N$. The number K itself can also be algorithmically determined, using the techniques described in [10].

Algorithm 1 relies solving the system (8) for the $2K$ unknowns $\omega_j, \eta_j, j = 1 \dots K$. This procedure was used in many recent including papers [20, 16, 10, 8, 18] and reduces system (8) to finding roots of a K -th degree polynomial. We state it below as follows.

Lemma 1. (Lemma 1 from [10].) Consider the system of $2K$ equations

$$(11) \quad c_n = \sum_{j=1}^K \alpha_j \beta_j^n, \quad n = 0 \dots 2K - 1.$$

for the $2K$ unknowns $\{\alpha_j\}_{j=1}^K, \{\beta_j\}_{j=1}^K$. Then $\{\beta_j\}_{j=1}^K$ are the K roots of the polynomial

$$(12) \quad \beta^K + l_1\beta^{K-1} + \dots + l_{K-1}\beta + l_K = 0$$

where the coefficients $l_1 \dots l_K$ satisfy the linear system:

$$(13) \quad \begin{bmatrix} c_0 & c_1 & \dots & c_{K-1} \\ c_1 & c_2 & \dots & c_K \\ \vdots & & & \vdots \\ c_{K-1} & c_2 & \dots & c_{2K-2} \end{bmatrix} \begin{bmatrix} l_K \\ l_{K-1} \\ \vdots \\ l_1 \end{bmatrix} = \begin{bmatrix} -c_K \\ -c_{K+1} \\ \vdots \\ -c_{2K-1} \end{bmatrix}.$$

Once $\{\beta_j\}_{j=1}^K$ are computed, $\{\alpha_j\}_{j=1}^K$ are obtained by solving the linear system given by the first K equations of (11).

2. DERIVATION OF ALGORITHM 1.

The algorithm 1 combines the ideas of [8, 16, 18]. Given any harmonic function $v(y)$ which satisfies $\Delta v = 0$ inside Ω , multiply (5) by v and integrate by parts. This yields the identity

$$(14) \quad \int_{\partial\Omega} f(y)\hat{n} \cdot \nabla v(y)dS(y) - \int_{\partial\Omega} g(y)v(y)dS(y) = \sum_{j=1}^K w_j \cdot \nabla v(\xi_j).$$

Next we take $v(y) = |y|^n Y_n^m(\hat{y})$, where $\hat{y} = y/|y|$ and Y_n^m is the spherical harmonic, to obtain

$$(15) \quad \int_{\partial\Omega} f(y) \hat{n} \cdot \nabla_y (|y|^n Y_n^m(\hat{y})) dS(y) - \int_{\partial\Omega} |y|^n Y_n^m(\hat{y}) g(y) dS(y) = \sum_{j=1}^K w_j \cdot \nabla_{\xi} \left(|\xi|^n Y_n^m(\hat{\xi}) \right).$$

In principle, the expression (15) yields an arbitrarily large number of identities relating the unknowns w_j, ξ_j to the (known) integrals. Following [16, 18], we first consider the particular sequence of these identities corresponding to $m = n$. Note that

$$|y|^n Y_n^n(\hat{y}) = C(y_1 + iy_2)^n$$

for a known constant C independent of y and that

$$\hat{n} \cdot \nabla_y (|y|^n Y_n^n(\hat{y})) = Cn(y_1 + iy_2)^{n-1}(\hat{n}_1 + i\hat{n}_2).$$

Then, applying this expression to (15) with $m = n$ yields

$$(16) \quad \int_{\partial\Omega} [f(y)n(y_1 + iy_2)^{n-1}(\hat{n}_1 + i\hat{n}_2) - g(y)(y_1 + iy_2)^n] dS(y) = \sum_{j=1}^K n\eta_j^{n-1}\omega_j.$$

where we used the notation $\eta_j = \xi_{1j} + i\xi_{2j}$; $\omega_j = w_{1j} + iw_{2j}$. This identity is precisely the sequence given by (7-8), which determines the first two coordinates of ξ_j and w_j . To determine the third coordinate, we choose $m = n - 1$ in (15). Then we obtain

$$|y|^n Y_n^{n-1}(\hat{y}) = C(y_1 + iy_2)^{n-1}y_3$$

so that

$$\hat{n} \cdot \nabla_y (|y|^n Y_n^{n-1}(\hat{y})) = C \{ (n-1)(y_1 + iy_2)^{n-2}y_3(\hat{n}_1 + i\hat{n}_2) + (y_1 + iy_2)^{n-1}\hat{n}_3 \}.$$

This yields the second sequence of identities,

$$(17) \quad \int_{\partial\Omega} f(y) \{ (n-1)(y_1 + iy_2)^{n-2}y_3(\hat{n}_1 + i\hat{n}_2) + (y_1 + iy_2)^{n-1}\hat{n}_3 \} - g(y)(y_1 + iy_2)^{n-1}y_3 dS(y) \\ = \sum_{j=1}^K (n-1)\eta_j^{n-2}\xi_{3j}\omega_j + \eta_j^{n-1}w_{j3}.$$

These identities correspond to those presented in (7-10). The derivation now complete. \blacksquare

3. NUMERICAL EXPERIMENTS

We now perform several experiments to test the robustness of Algorithms 1 and 2. For all experiments, we have used the finite element solver FlexPDE to obtain the solution of the forward problem (1). This package uses adaptive mesh generation and local and global error controls. The user-specified error tolerance was chosen to be 10^{-5} ; we verified that changing this tolerance did not change the results. For all of the examples below, we have used $\partial_n u = g(x) = x_2$ as the boundary condition with internal conductivity $\kappa = 19$ in (2). Once the constants w_j and ξ_j were recovered using Algorithm 1 or 2, the connection formulae (6) were then used to recover the inclusion radius.

The integrals such as (7) were computed as follows. First, the output values generated by FlexPDE on the boundary of the surface were exported to Matlab. These points were then used to

generate a surface mesh using Matlab's `convhulln` function. The resulting surface mesh was used to generate the normals and surface elements and compute the required integrals.

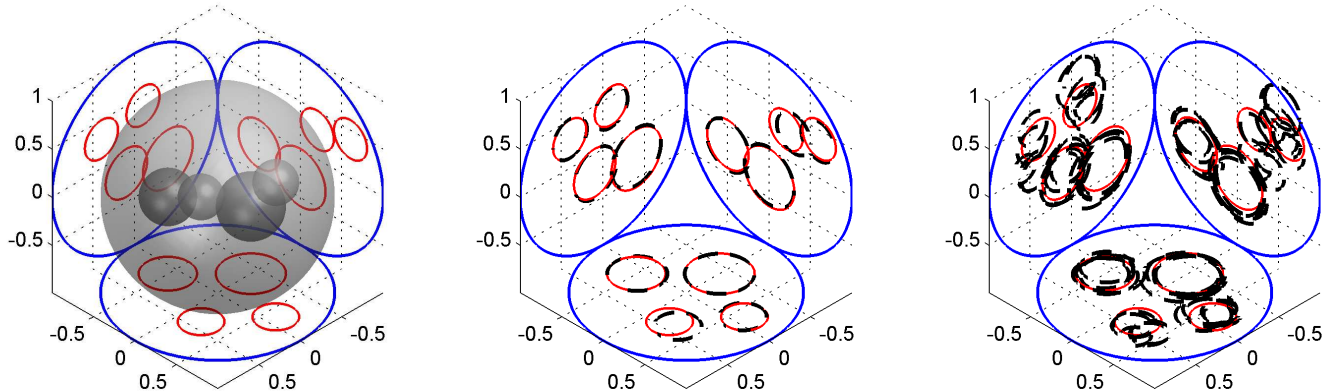


FIGURE 1. Recovery of four spherical inclusions inside a sphere (experiment 1). Left: the test configuration. Middle: no noise. Right: 2% noise.

Experiment 1. We take the domain to be unit ball with $K = 4$ spherical inclusions of varying size inside. The configuration and its recovery is shown in Figure 1; the sizes of four holes are $\varepsilon = 0.2, 0.2, 0.25, 0.3$. Without any noise, excellent recovery is observed. To test the resilience of the method against noise, we added 2% random perturbations to the output data of the forward solver. The right pane shows five distinct realizations of the recovery with noise. Note that the biggest inclusion exhibits the best recovery. Overall, very good agreement is observed.

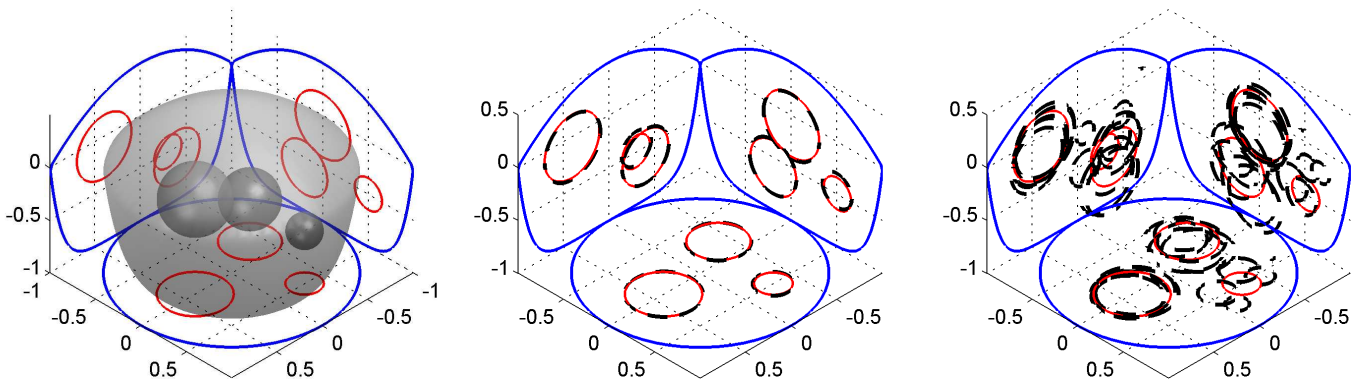


FIGURE 2. Muffin-shaped domain (experiment 2). Middle: zero noise. Right: 2% noise.

Experiment 2. To demonstrate that the recovery works well for an arbitrary domain and not just a sphere, consider a domain in the shape of a “muffin” shown in Figure 2. It consists of the intersection of the top surface $z = \frac{1}{2}(1 - (x^2 + y^2))$ and the bottom surface $z = -\left(1 - (x^2 + y^2)^4\right)$. Three spherical “raisins” of radii 0.15, 0.25, 0.3 are placed inside the “muffin”. As in Experiment 1, middle pane shown the recovery with no noise and the right pane shows five simulations of recovery with 2% random noise. Without noise, excellent recovery is observed. With noise, the biggest inclusion is still recovered relatively well.

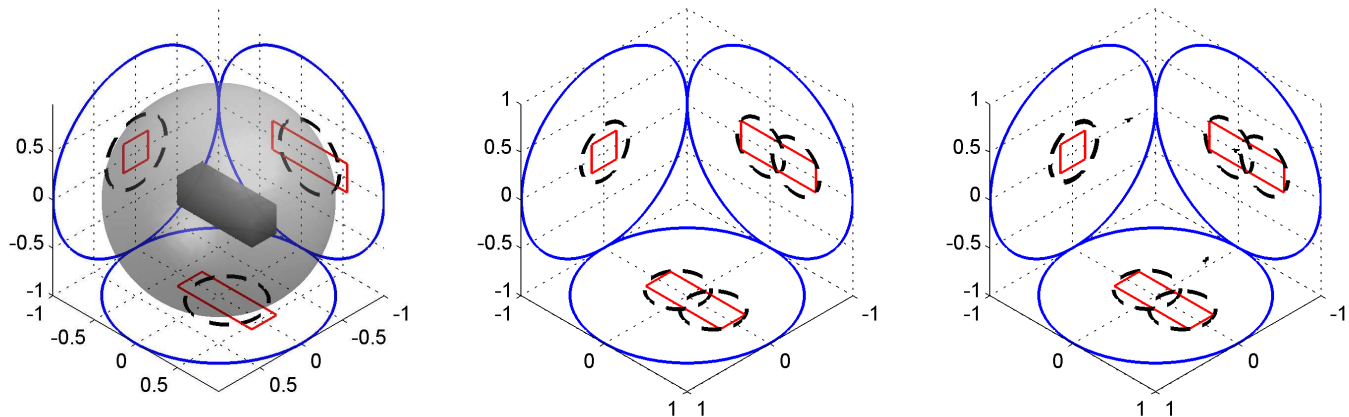


FIGURE 3. Recovery of box-shaped inclusion inside a sphere using $K = 1, 2, 3$ disks (experiment 3). The original box is parallel to the axes and has corners at $(0.15, 0.55, 0.15)$ and $(-0.15, -0.35, -0.15)$.

Experiment 3. Consider a box-shaped anomaly inside a sphere as in Figure 3. We attempt its recovery by applying Algorithm with $K = 1, 2, 3$. When $K = 1$, the recovered inclusion location captures well the center of the box. Running the algorithm with $K = 2$, the two resulting inclusions approximate the box even better. Finally, $K = 3$ yields two inclusions which are identical to $K = 2$, plus a spurious inclusion of a very small radius [which appears as a smudge in the right pane]. This indicates that the matrix in (13) is nearly singular when $K = 3$, and that $K = 2$ is a good place to stop.

Experiments 4. Figure 4 shows the recovery in two dimensions. For a circular domain as well as for a clover-shape domain. The recovery is performed with no noise, 2% and 4% noise. For the noisy data, five different random realisations are shown. In the presence of noise, bigger inclusions are generally recovered better than the smaller ones. This is not surprising, since the smaller inclusions have less of an “imprint” on the boundary and become swamped as noise is increased.

Experiment 5: L-shaped inclusion. In the final experiment, we attempt to recover an L-shaped inclusion in Figure 5. The output of Algorithm 2 with $K = 1 \dots 8$ is shown. Note that the corners are recovered surprisingly well when $K = 8$.

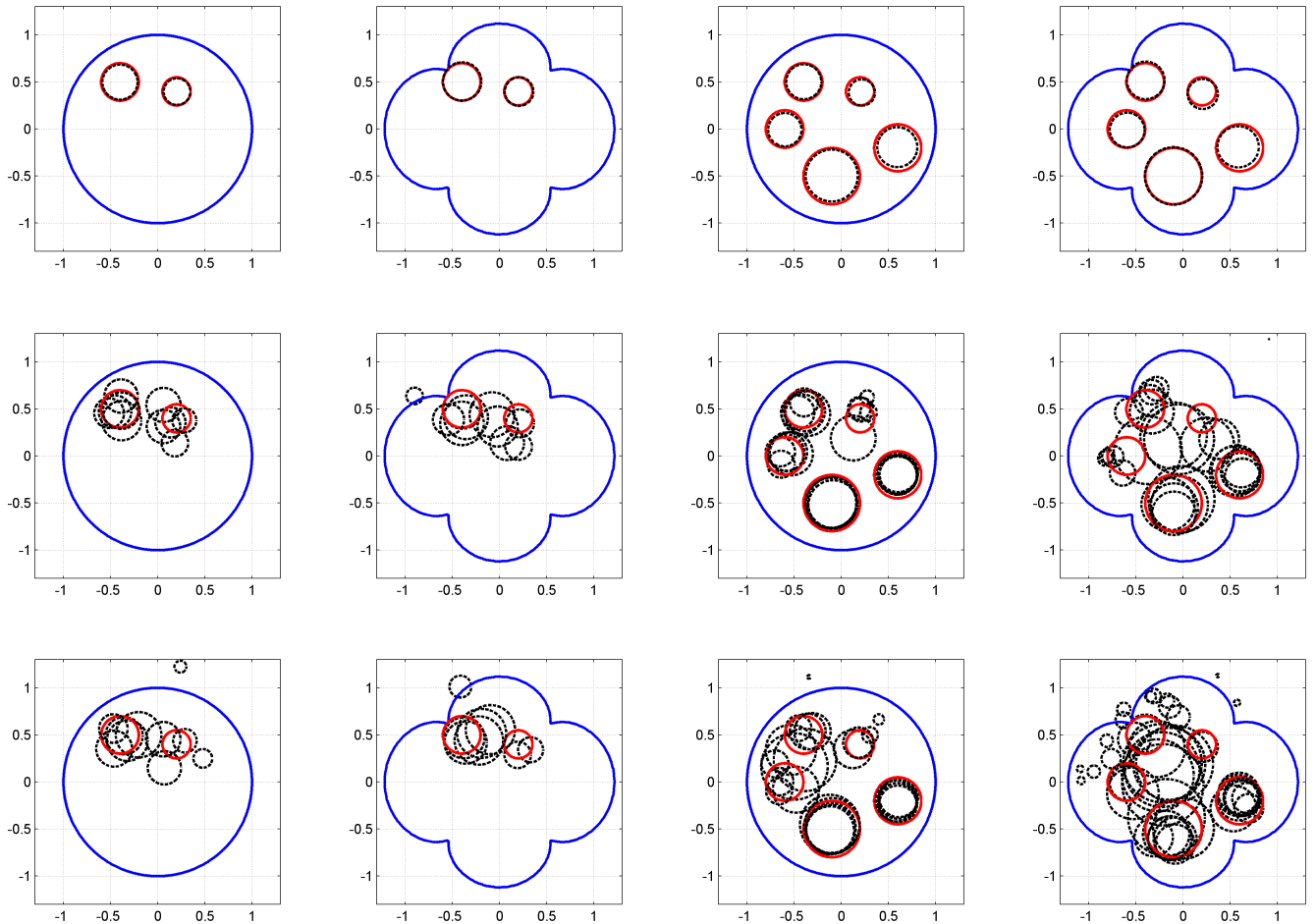


FIGURE 4. Recovery of two and five circular defects for a disk domain and for a “clover” domain (experiment 4). Dashed lines represent recovered defects using Algorithm 2. Solid lines (red online) give the original defect locations. First row: No noise. Second row: 2% noise in the output function $f(x)$. Third row: 4% noise. Five different random simulations are shown for in each figure.

4. DISCUSSION

In this paper we tested an algorithm to recover K small inclusions inside a three-dimensional domain. It is based on the methods introduced in [8, 16, 18]. It generalizes the algorithm discussed in [19] in two ways: it is applicable for an arbitrary domains, and it is able to systematically handle multiple inclusions.

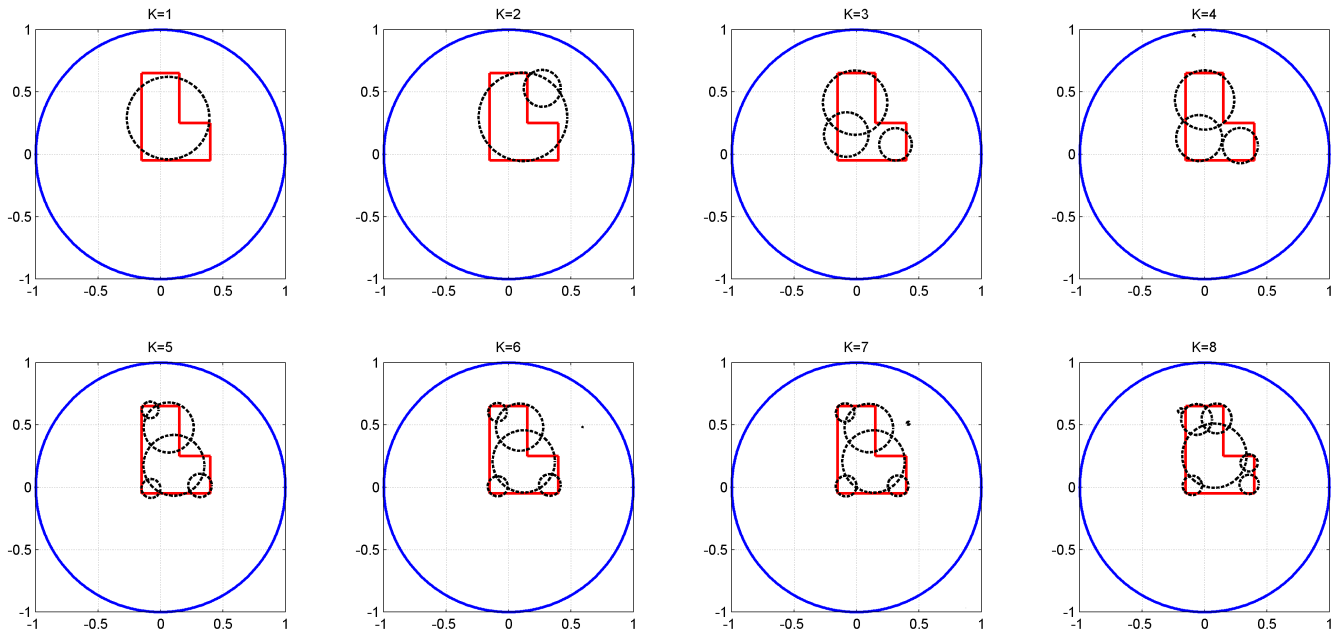


FIGURE 5. “Recovery” of irregular shape using K circles (experiment 5).

In [19], the authors write “*given that these multi-source inverse problems are finite-dimensional, it may then be better to abandon the analytical approach and resort to a numerical method*”. We show that the numerical problem of identifying K dipoles in three dimensions for an arbitrary domain (which involves $6K$ unknowns), can be reduced to solving a single K degree polynomial, plus several linear algebra problems.

There are many remaining interesting questions. In [19], a recovery of a single source for Helmholtz equation was studied. The question of recovering multiple sources for acoustic problems remains largely open.

In the presence of noise, higher spherical harmonics can easily get swamped. Since Algorithm 1 depends on spherical harmonics of degree $l = 2K$ for the recovery of K dipoles, this makes it particularly prone to noise for larger values of K . In principle, there are $l(l + 1)/2$ different spherical harmonics of degree at most l that can be used to recover the data; however Algorithm 1 only uses $2l$ of these. Therefore the algorithm is suboptimal as it does not use all of the available information. An optimal algorithm would only use harmonics of degree up to $O(\sqrt{K})$ and would therefore be much less susceptible to noise. The design of such an optimal algorithm is an open problem. Another approach which can be explored to combat noisy data is oversampling: it is possible to improve recovery by generating more than the minimal number of equations. This leads to an overdetermined system; nonlinear error minimization techniques could then be used to reduce the effect of the noise.

Numerical experiments with irregular-shaped inclusions suggest that this approach can be useful for recovery of larger contiguous patches (see Figure 5). These appear to be approximated surprisingly well by multiple inclusions, even though they are not separated. A related approach was used in [21] to recover inclusions consisting thin segments.

APPENDIX A: TWO DIMENSIONS

Here we recall the algorithm in two dimensions which was derived in [8].

Algorithm 2. ([8], *Solution to Problem 1 in \mathbb{R}^2*).

(1) *Compute the $2K$ complex integrals*

$$(18) \quad A_n = \int_{\partial\Omega} f(y)y^{n-1}\hat{n}dS(y) - \frac{1}{n} \int_{\partial\Omega} g(y)y^n dS(y), \quad n = 1 \dots 2K$$

where y and \hat{n} are to be interpreted as complex numbers with $y^{n-1}\hat{n}$ corresponding to complex multiplication.

(2) *Solve the system*

$$(19) \quad A_n = \sum_{j=1}^K w_j \xi_j^{n-1}, \quad n = 1 \dots 2K$$

using the Lemma 1 to recover $w_j, \xi_j, j = 1 \dots K$.

We now show that the method used in [11] for a single inclusion is equivalent to this algorithm; this also provides its alternative derivation. We summarize the method used in [11]. It is based on the function originally defined in [9],

$$(20) \quad H(x) = \int_{\partial\Omega} f(y)\hat{n} \cdot \nabla_y \Phi(x, y) dS(y) - \int_{\partial\Omega} g(y)\Phi(x, y) dS(y)$$

where

$$\Phi(x, y) = \frac{1}{2\pi} \ln |x - y|$$

is the Free-space Green's function, $\Delta\Phi = \delta(x - y)$. It then follows from (14) that for $x \notin \Omega$,

$$(21) \quad H(x) = \sum_{j=1}^K w_j \cdot \nabla_{\xi} \Phi(x, \xi_j), \quad x \notin \Omega.$$

The idea in [11] is to take a constant R such that $x \notin \Omega$ whenever $|x| = R$, and then look at the Fourier decomposition of $H(Re^{i\theta})$; then solve the resulting system for ξ_j, w_j . We now show that this is equivalent to Algorithm 2. Indeed the following identity holds,

$$(22) \quad \ln |x - y| = - \sum_{n=1}^{\infty} \frac{1}{2n} (x^{-n}y^n + \overline{x^{-n}y^n}) + \ln |x|, \quad |x| > |y|$$

where $x, y \in \mathbb{C}$ are complex numbers. Take $x = Re^{i\theta}$ and compute using (22),

$$(23) \quad \int_0^{2\pi} \Phi(Re^{i\theta}, y)e^{in\theta} d\theta = -\frac{1}{4n} R^{-n} y^n.$$

Next, note that for any vector w we have

$$\begin{aligned} w \cdot \nabla y^n &= n(y_1 + iy_2)^{n-1} (\hat{i} + i\hat{j}) \cdot w \\ &= ny^{n-1}w \end{aligned}$$

so that, in combination with (22)

$$(24) \quad \int_0^{2\pi} w \cdot \nabla_y \Phi(Re^{i\theta}, y) e^{in\theta} d\theta = -\frac{1}{4} R^{-n} y^{n-1} w.$$

Next, choose R sufficiently large such that $x \notin \Omega$ for whenever $|x| = R$. Integrate both sides of (14) with respect to x on a circle of radius R . Using formulae (23) and (24), we then obtain

$$\int_{\partial\Omega} nf(y)y^{n-1}\hat{n}dS(y) - \int_{\partial\Omega} g(y)y^n dS(y) = \sum_{j=1}^K n\xi_j^{n-1}w_j$$

This corresponds precisely to (18, 19). The derivation is complete. ■

REFERENCES

- [1] F. Wilson, R. Bayley, The electric field of an eccentric dipole in a homogeneous spherical conducting medium, *Circulation* 1 (1950) 84–92.
- [2] L. Borcea, Electrical impedance tomography, *Inverse Problems* 18 (2002) R99–R136.
- [3] H. Amani, H. Kang, *An Introduction to Mathematics of Emerging Biomedical Imaging*, Vol. 62, Mathematics and Applications, Springer-Verlag, Berlin, 2008.
- [4] M. H. Choi, T.-J. Kao, G. J. Saulnier, J. C. Newell, A reconstruction algorithm for breast cancer imaging with electrical impedance tomography in mammography geometry, *IEEE Trans Biomed Eng* 54(4) (2007) 700–710.
- [5] H. Ammari, O. Kwon, J. K. Seo, E. J. Woo, T-scan electrical impedance imaging system for anomaly detection, *SIAM Journal on Applied Mathematics* 65 (1) (2004) 252–266.
- [6] M. Metwaly, G. El-Qady, J. Matsushima, S. Szalai, N. S. N. Al-Arifi, A. Taha, Contribution of 3-D electrical resistivity tomography for landmines detection, *Nonlin. Processes Geophys.* 15 (2008) 977–986.
- [7] D. J. Cedio-Fengya, S. Moskow, M. S. Vogelius, Identification of conductivity imperfections of small diameter by boundary measurements: Continuous dependence and computational reconstruction, *Inverse Problems* 14 (1998) 553–595.
- [8] Y.-S. Chung, S.-Y. Chung, Identification of the combination of monopolar and dipolar sources for elliptic equations, *Inverse Problems* 25 (8) (2009) 085006.
- [9] O. Kwon, J. K. Seo, J.-R. Yoon, A real-time algorithm for the location search of discontinuous conductivities with one measurement, *Comm. Pure Appl. Math.* 55 (2002) 1–29.
- [10] H. Kang, H. Lee, Identification of simple poles via boundary measurements and an application to EIT, *Inverse Problems* 20 (2004) 1853–1863.
- [11] M. Hanke, On real-time algorithms for the location search of discontinuous conductivities with one measurement, *Inverse Problems* 28.
- [12] H. Ammari, H. Kang, *Reconstruction of Small Inhomogeneities from Boundary Measurements*, Springer, 2004.
- [13] M. Hanke, Locating several small inclusions in impedance tomography from backscatter data, *SIAM Journal on Numerical Analysis* 45(5) (2011) 1991–2016.
- [14] H. Hakula, H. Hyvonen, Two noniterative algorithms for locating inclusions using one electrode measurement of electric impedance tomography, *Inverse Problems* 24.
- [15] A. E. Badia, Inverse source problem in an anisotropic medium by boundary measurements, *Inverse Problems* 21 (2005) 1487–1506.

- [16] T. Nara, S. Ando, A projective method for an inverse source problem of the poisson equation, *Inverse Problems* 19 (2) (2003) 355–369.
- [17] T. Nara, J. Oohama, M. Hashimoto, T. Takeda, S. Ando, Direct reconstruction algorithm of current dipoles for vector magnetoencephalography and electroencephalography, *Physics in Medicine and Biology* 52 (2007) 3859–3879.
- [18] T. Nara, An Explicit Method for Inverse Reconstruction of Equivalent Current Dipoles and Quadrupoles, *Magnetoencephalography*, InTech, 2011.
URL <http://cdn.intechweb.org/pdfs/24372.pdf>
- [19] N. L. Tsitsas, P. A. Martin, Finding a source inside a sphere, *Inverse Problems* 28 (1) (2012) 015003.
- [20] G. H. Golub, P. Milanfar, J. Varah, A stable numerical methods for inverting shape from moments, *SIAM Journal of scientific computing* 21 (4) (1999) 1222–1243.
- [21] H. Lee, W.-K. Park, Location search algorithm of thin conductivity inclusions via boundary measurements, *ESAIM. Proceedings* 26.

† MATHEMATICS DEPARTMENT, DALHOUSIE UNIVERSITY, HALIFAX, NOVA SCOTIA, CANADA, B3H 3J5.,

‡ DEPARTMENT OF MATHEMATICS AND MAXWELL INSTITUTE FOR MATHEMATICAL SCIENCES, HERIOT-WATT UNIVERSITY, EDINBURGH, EH14 4AS.