Virtual Resistive Network algorithm for inverse conductivity problems*

Yong-Jung Kim †, Min-Su Ko ‡, and Min Gi Lee §

Department of Mathematical Science, KAIST, Daejeon, Korea

(Dated: December 12, 2013)

Abstract

In this paper, we illustrated a numerical algorithm which we refer to as the Virtual Resistive Network algorithm, to reconstruct the interior conductivity from interior current densities. In the algorithm, the essence is to introduce a virtual resistive network onto the continuum domain. The algorithm is non-iterative and has certain degrees of flexibility in use. We reconstructed the isotropic conductivity, from one interior current density and known conductivity on partial boundary, and reconstructed the orthotropic conductivity, from two pieces of current densities and known conductivity on partial boundary. Numerically simulated results with noise are presented. Various features of the algorithm against the noise was reported.

1 Introduction

Problems of conductivity reconstruction have been an object of wide study. Among the various types of problems, we are concerned with the reconstruction problem from interior current densities. The subject of EIT(Electrical Impedance Tomography),

*This research was supported by the project of National Junior research fellowship of the National Research Foundation of Korea under grant number 2011-0013447.

†yongkim@kaist.edu;

‡minsu.ko@kaist.ac.kr;

§mgleemail@gmail.com; Corresponding author

which is more classical one, reconstructs conductivity from boundary information while our concerned problem reconstructs conductivity from interior information.

The authors are mainly motivated from MREIT(Magnetic Resonance Electrical Impedance Tomography). MREIT was firstly studied by Zhang in 1992 [1] and by Woo et. al. in 1994 [2] respectively. Let us deliver only a brief idea of MREIT. One runs an MRI machine to obtain internal magnetic field data. By collecting 3 spatial magnetic field data, one can, in principle, acquire internal current density data $\mathbf{J}$ using the Ampere's Law

$$\mathbf{J} = \frac{1}{\mu_0} \nabla \times \mathbf{B}.$$

Thus, the reconstruction problem using internal current density data can now be inquired. For a development of the subject, see the review papers of [3], [4], and [5], which are in chronological order. After the works of [1] and [2], many algorithms have been suggested, for instances, J-substitution [6], Equipotential Line [7], Algebraic Reconstuction [9], and other algorithms. We will demonstrate differences between our algorithm and a few previously developed algorithms either in numerical simulations or in inquiries on their structures.

We make clear here that it is not our present concern to reconstruct conductivity from the several z-directional components of magnetic field data, which is also a major topic of MREIT. However, if a certain algorithm for those problems assumes a procedure as an intermediate step that retrieves the projected currents from the B^z data, and the reconstruction is done using the projected current, then the discussion advanced here can be adapted for those problems after the intermediate step. One example of such an algorithm is suggested in [10].

Among the related subjects that also make use of interior information are the aquifer identification problem and the conductivity problem from power densities. For the former case, in our terminology, the interior data corresponding to potential u, instead of $\mathbf{J}$, is used for reconstruction. See [11], [12], and [13]. For the latter case, the interior power density $p = \mathbf{J} \cdot \nabla u$ is used for reconstruction. See [14], [15], [16], and [17].

Let us introduce the equation concerned and related quantities in the terminology of electromagnetism. We consider a following linear elliptic equation in a bounded domain $\Omega \subset \mathbb{R}^2$

$$-\nabla \cdot \big(\sigma(x) \nabla u\big) = 0, \quad \text{in } \Omega, \tag{1}$$

$$-\big(\sigma(x) \nabla u\big) \cdot \mathbf{n} = g, \quad \text{on } \partial\Omega. \tag{2}$$

Here, σ is the conductivity and u is the voltage. The current density is defined by $\mathbf{J} := -\sigma\nabla u$. When the coefficient σ is a symmetric matrix field, we say it is anisotropic, and when σ is in particular a diagonal matrix field, we say it is orthotropic, and lastly when σ is even a real-valued function, possibly is regarded as being multiplied by identity matrix, we say it is isotropic. In any cases we assume the conductivity is positive definite. Our algorithm we are advancing in this paper can reconstruct the isotropic and the orthotropic conductivity. We cannot say for certain yet whether the algorithm can be further generalized to take the full anisotropic case into the scope our algorithm applies.

In the algorithm, it is essential to introduce a virtual resistive network. This is, in a certain sense, an existing theory. In the approach, to approximate (1) numerically, one introduces a resistive network as in Figure 1. In that network, the voltage V is

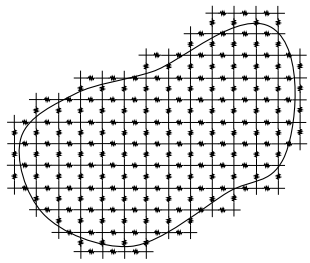


Figure 1: An example of a virtual resistive network on the domain.

a field defined on vertices of network, and the current I is a field defined on edges. The voltage difference between two vertices and the amount of current passing the edge joining the two vertices are determined by Ohm's Law

$$\Delta V = I\,R,$$

where R is the resistivity that is also defined on edges. The discrete field R approximates the inverse of conductivity $\sigma(x)^{-1}$ in this scheme.

The quantities V and I satisfy the Kirchoff's Voltage Law and the Kirchoff's Current Law respectively. In other words, the net voltage drop along any closed loop is zero, and the net outgoing current from any vertex is zero.

This kind of discrete level prescription is described in a book of Strang [18]. In fact, they are of a subject of the algebraic topology. In the subject, the fields V and I are called cochains, and the Kirchoff's Laws are written in terms of the dual-boundary and the dual-co-boundary operators. We will not need, however, any more of the subject here other than the prescription of Strang [18], and we will only use

networks, which is of rather simple geometry.

In this approach, the forward problem, solving for u provided with σ, can be approximated by solving for V field provided with R field with aid of two Kirchoff's Laws in the network.

What has to be noticed is that these considerations applies to the inverse problem as well as to the forward problem. For the inverse problem, we will soon see the reconstruction process of R when I is given also can be accomplished in the network. It will be a linear and non-iterative procedure. Also, we address that the resistive network we introduce is literally *virtual* as written, the algorithm has certain degrees of flexibility in arrangement and shape of network for a given continuum domain. We will make use of this flexibility in the algorithm.

2 Background and Preliminaries

In this Section, we illustrated two main ingredients of our algorithm. One is to introduce an equation whose approximation in the network will become the reconstruction process for resistivity in our algorithm. The other is on a consistence way to distribute the quantities, such as voltage, current, and resistivity onto the network. After those, we will be able to set up a reconstruction example in a simple square network.

2.1 Choice of governing equation for electrostatics

To solve the inverse problem of equation (1), one might consider a following minimization procedure. First, one guesses the conductivity σ_0 initially. Then, since the conductivity is now provided with, one can solve (1) for u and consequently for the current density, which we denote as $\mathbf{J}^0$. Nextly, one compares this $\mathbf{J}^0$ to the measured data $\mathbf{J}^{meas}$, and devises a certain algorithm to update the conductivity, which we denote σ_1, so that the difference between the updated current density, which we denote $\mathbf{J}^1$ obtained from the updated conductivity σ_1, and the measured data $\mathbf{J}^{meas}$ to be reduced. One iterates this procedure. The procedure basically is describing the J-substitution algorithm, which was introduced in early stage of development in MREIT.

Note that the procedure contains an inversion of partial differential operator as its non-linearity in each iteration. Thus the J-substitution algorithm is iterative and has a PDE solving step in each iteration. The paper [19] written by two of authors attributes the origin of such non-linear and iterative features to the inappropriate

choice of governing equation of (1). The work chose another choice of governing equation for the electrostatics, which fits more to the reconstruction problem we concern. They are

$$\nabla \times (r\mathbf{J}) = 0, \quad \text{in } \Omega, \tag{3}$$
$$\nabla \cdot \mathbf{J} = 0, \quad \text{in } \Omega, \tag{4}$$
$$\mathbf{J} \cdot \mathbf{n} = g, \quad \text{on } \partial\Omega,$$

where $r(x) := \sigma^{-1}(x)$ is the resistivity. Both of choices are from the first order system of three equations in electrostatics,

$$\nabla \times \mathbf{E} = 0, \tag{5}$$
$$\nabla \cdot \mathbf{J} = 0, \tag{6}$$
$$\sigma\mathbf{E} = \mathbf{J}, \quad \text{or} \quad \mathbf{E} = r\mathbf{J} \quad \text{(Ohm's Law)}. \tag{7}$$

Provided that the domain is simply connected, by introducing a potential u so that $\mathbf{E} = -\nabla u$, one can reduce above three equations (5)-(7) into the single equation (1) on the scalar u. It is clear also that the latter choice (3),(4) also is obtained from (5)-(7). See that the origin of the Kirchoff's Voltage Law and the Kirchoff's Current Law became clearer after observing (5) and (6).

The important thing for the latter choice is that one can directly solve the equation (3) for r with given coefficients $\mathbf{J}$. Indeed, analysis on the equation can be done, the well-posedness of this problem is proved in [19]. We will see soon that our algorithm is a way of direct solving (3) numerically.

Before we proceed to further, let us introduce one more quantity. In 2-dimensions, a divergence free vector field $\mathbf{J}$ admits a stream function ψ such that

$$\mathbf{J} = \nabla^{\perp}\psi, \quad \text{where } \nabla^{\perp} := \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \partial_x \\ \partial_y \end{pmatrix} = \begin{pmatrix} \partial_y \\ -\partial_x \end{pmatrix}.$$

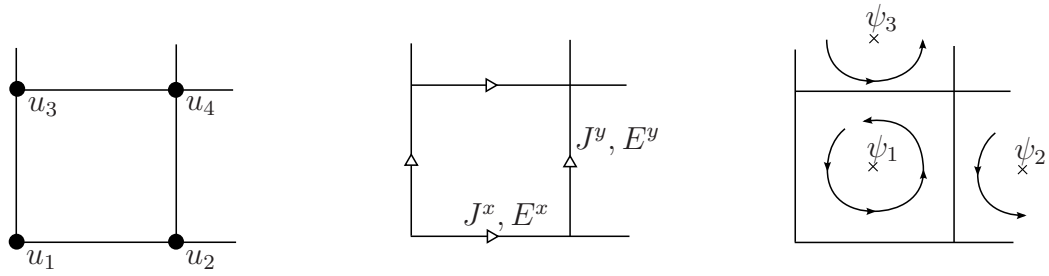
The expression $\nabla^{\perp}\psi$ is automatically divergence free for any ψ. One may prefer re-writting the equation (3) in terms of a single ψ instead of $\mathbf{J}$.

2.2 Placement of scalar, vector and density fields on network

We have seen several types of quantities. First, the voltage u is a scalar field, and the electric field $\mathbf{E}$ and the current density $\mathbf{J}$ are vector fields. The stream function

ψ can be regarded as density. Denoting the discrete field with same symbol would make little confusion, thus from now, let us abuse the same symbols for the discrete fields.

The locations of corresponding discrete fields on the network are as in Figure 2. The scalar field u is located on each vertex, and the vector fields $\mathbf{J}$ and $\mathbf{E}$ are on

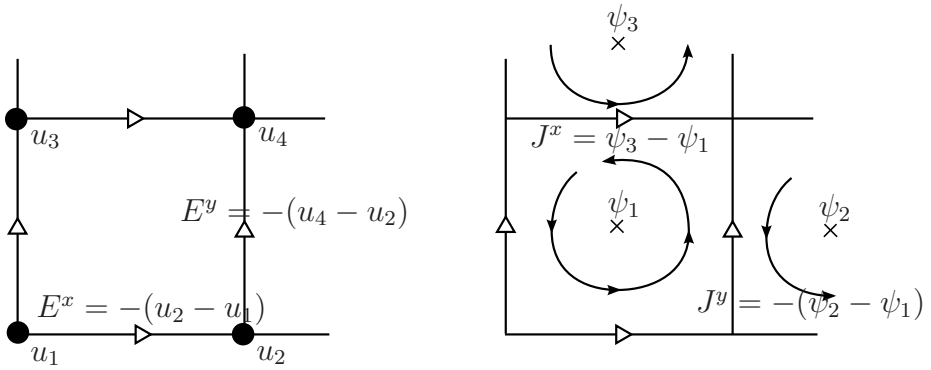


(a) scalar (b) vector (c) density

Figure 2: Location of discrete fields.

edges. Note that the x and y component of the vector are not assigned on a same point of network but they are staggered. x component is on the horizontal edge, and y component is on the vertical edge. The arrows designate the positive directions of the vector fields. Lastly, the density ψ is located on the midpoint of a cell enclosed by edges. The arrows do similar role, which we will see soon.

One can see the consistency of this prescription. First, as in Figure 3(a), the



(a) $\mathbf{E} = -\nabla u$ (b) $\mathbf{J} = \nabla^\perp \psi$

Figure 3: Use of potentials in the network and Kirchoff's Laws.

difference of u between any two vertices will be matched with the component of $-\mathbf{E}$ on that edge, which approximates

$$E^x = -\partial_x u, \quad E^y = -\partial_y u.$$

Secondly, the difference of ψ between any two cells is matched with the component of $\mathbf{J}$ on the shared edge. See Figure 3(b). This approximates

$$J^x = \partial_y \psi, \quad J^y = -\partial_x \psi.$$

Observe that the arrows of Figure 3(b) in cells transitively assign the correct positive directions on the shared edge. Thirdly the use of these potential concepts in this discrete level are of desirable ones. If $\mathbf{E}$ and $\mathbf{J}$ are given from a scalar u and a density ψ respectively, then the sum of edge values of $\mathbf{E}$ along any closed loop in the network automatically vanishes, and the net out going $\mathbf{J}$ on any vertex automatically vanishes.

The Ohm's Law $\mathbf{E} = r\mathbf{J}$ is now a family of equations defined on edges, since they are equations between vector fields. They are approximating two equations componentwisely

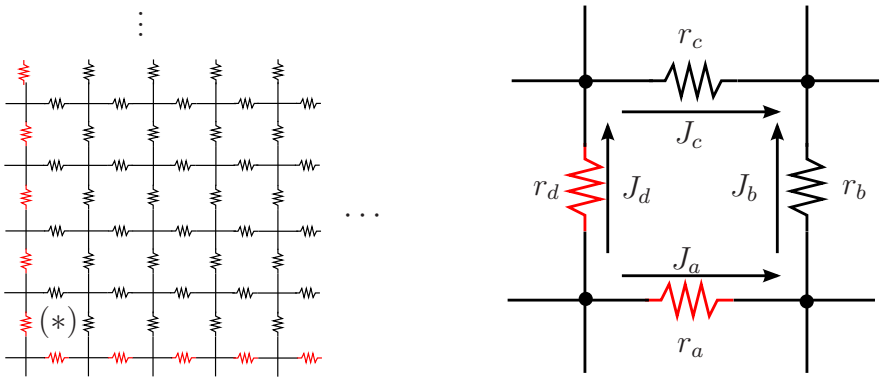
$$E^x = rJ^x, \quad E^y = rJ^y.$$

3 Virtual Resistive Network with Square Network

Having discussed on the governing equation and on the discrete fields in network, we are now able to devise our algorithm, which is nothing but an approximation of the curl-based equation (3) on the network. More precisely, the algorithm solves for r field defined on each edge using the known currents on edges and the Kirchoff's Voltage Law. The voltage field is obtained after the procedure incidently.

This Section, in particular, is devoted to implement the algorithm on the square network, the simplest one, as a preliminary model example. By studying the model example, we reveals the several characteristics of our algorithm. The algorithm used in numerical simulations differs a little from the one described here, and will be specified in the Section 4.

Let us look at the square network in Figure 4(a). First thing to demonstrate is that the square network reflects the orthotropic resistivity. We have a freedom to set different resistivity values on a horizontal and a vertical edge nearby. This is a certain way of approximation of anisotropy. This does not represent a full anisotropy, however. Note that the resistivity values assigned on edges are real number. These values can be understood as eigenvalues of resistivity diagonalized as to the arrangement of the network. Since the network is square lattice here, the corresponding resistivity must be the diagonal matrix field, whose eigenvectors are the horizontal and vertical vectors. In a book of Strang [18], he referred this to as just anisotropic one though.



(a) A simple lattice network. (b) The corner cell at left bottom.

Figure 4: Cell by cell reconstruction in a simple lattice network.

Now we specify the reconstruction procedure in this prescription. Let us firstly examine the orthotropic resistivity case. The current passing each edge is assumed to be given in the reconstruction problem. We also assume the resistivity on the left and the bottom boundary edges are given, which are red-colored ones in the Figure 4(a). One can equivalently assume the voltage u on the boundary vertices since once we know the voltage values on boundary vertices then the known current on the boundary edges together with Ohm's Law will give us resistivity on the boundary edges.

To begin with the reconstruction procedure, look at the Figure 4(b) that is the bottom left corner cell marked as $(*)$ in the Figure 4(a). Remember the resistivity on the left and the bottom edges are known values, and the only ones on the right and the upper edges are the two unknowns. In order to solve for the unknowns, we employ the Kirchoff's Voltage Law as a furnished equation such that

$$r_a J_a + r_b J_b - r_c J_c - r_d J_d = 0.$$

In order to close the system, we need one more equation that comes from another data $\mathbf{J}^2$, and let us thus denote the first data as $\mathbf{J}^1$. Then the twice use of Kirchoff's Voltage Law give us

$$\begin{pmatrix} J_c^1 & J_d^1 \\ J_c^2 & J_d^2 \end{pmatrix} \begin{pmatrix} r_c \\ r_d \end{pmatrix} = - \begin{pmatrix} J_a^1 & J_b^1 \\ J_a^2 & J_b^2 \end{pmatrix} \begin{pmatrix} r_a \\ r_b \end{pmatrix} =: \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \tag{8}$$

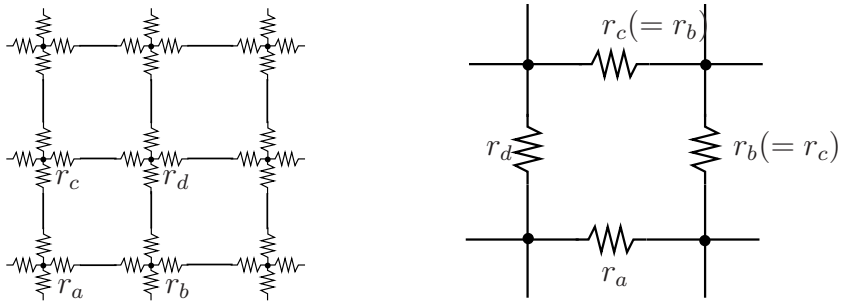
and if $\begin{pmatrix} J_c^1 & J_d^1 \\ J_c^2 & J_d^2 \end{pmatrix}$ is invertible,

$$\begin{pmatrix} r_c \\ r_d \end{pmatrix} = \begin{pmatrix} J_c^1 & J_d^1 \\ J_c^2 & J_d^2 \end{pmatrix}^{-1} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \frac{1}{J_c^1 J_d^2 - J_d^1 J_c^2} \begin{pmatrix} J_d^2 & -J_d^1 \\ -J_c^2 & J_c^1 \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}.$$

Thus we need the denominator $J_c^1 J_d^2 - J_d^1 J_c^2 \neq 0$. The expression $J_c^1 J_d^2 - J_d^1 J_c^2$ is an approximation of $\mathbf{J}_1 \times \mathbf{J}_2$, and it is well-known that the condition $\mathbf{J}_1 \times \mathbf{J}_2 \neq 0$ can be achieved by controlling boundary conditions. See for example [20].

Note that we are now able to apply the same for the right and the upper cell of the previous concerned corner cell. For those cells, the resistivity of their left and bottom edges are now known, and hence we apply the same to those cells. This procedure can be completed by applying the same repeatedly until we solve for all the resistivity of network.

Let us leave the orthotropic resistivity reconstruction procedure, and turn to the isotropic resistivity reconstruction procedure. Isotropic resistivity would be adapted to the network by identifying the resistivity values of a horizontal and a vertical edge near a certain point. Its consequence will be a reduction of the total degrees of freedom of resistivity values on edges to half. This process is ambiguous, however, there are many choices to accomplish that. For example, one might consider a following symmetric prescription as in Figure 5(a). In there we see the resistivity values near



(a) Symmetric choice. (b) Asymmetric choice.

Figure 5: Implementing an isotropic resistivity.

a vertex are set identical, and the resistivity on an edge can be set by the sum of the two resistivity values of two end vertices. On the while, one can think of an asymmetric prescription. Figure 5(b) illustrates one way of doing that, the upper and the right resistivity values are identified in this configuration.

We are going to take the asymmetric one as in Figure 5(b), and reject the sym-

metric one. This intentional choice will raise the dependency of the performance of our algorithm to the data, in particular on the directional tendency of a given vector field $\mathbf{J}$.

The reason why we choose an asymmetric one is accounted by following. Concerning only for an approximation of isotropy, the symmetric choice and the asymmetric choice are not much different, the former one merely has the symmetry. Also, while the symmetric choice shows a neutral performance against the noises regardless of the directional tendency of data, the asymmetric choice shows drastically varying performance dependently on the data. In a certain region that is correctly arranged with data, the reconstructed conductivity shows better result than ones from the neutral symmetric choice.

On the other hand, to change from one configuration of asymmetry to another one of asymmetry, and to solve for all resistivity values once more costs only a little more. This is because our reconstruction procedure only requires a little amount of computation. To be precise, there will be one divide operation for each cell. Therefore, we shall take a strategy to see the reconstruction results from as many different asymmetric configurations as possible, which are all non-trivially different from each other although they are obtained from same data. By the non-trivial difference I mean the case when the data contains noises, from which performance of algorithm possibly varies. This is of course an affirmative aspect of our network algorithm since the procedure might increase the chance to diagnose a certain region correctly. In the simulations in Section 5, we can observe this.

For the present, let us specify one possible way with one configuration of asymmetry as in Figure 5(b). If the bottom and the left resistivity values r_a and r_d in Figure 5(b) are known, by applying the Kirchoff's Voltage Law once,

$$\begin{aligned} &r_a J_a + r_b J_b - r_c J_c - r_d J_d = 0, \quad \text{and hence} \\ &r(= r_b = r_c) = -\frac{r_a J_a - r_d J_d}{J_b - J_c}, \end{aligned} \tag{9}$$

provided $J_b - J_c \neq 0$. It should be addressed that, in the contrary to the orthotropic case, this condition in general cannot be fulfilled by boundary control. In the following Section, I shall try to give more precise account for this issue in the connection to the choice of asymmetry. Again, one can obtains all the isotropic resistivity values by repeatedly applying the procedure.

3.1 Characteristics of the Square-VRN : Isotropic

We are now reveal a few characteristics of the algorithm with the square network specified in the preceding Section. After this, it will become clearer how we devise our main algorithm. This Section, in particular, investigates characteristics of the VRN algorithm for isotropic resistivity reconstruction.

3.1.1 Hyperbolic nature

First thing to look closely at is the hyperbolicity of the equation (3). For an isotropic problem, the equation (3) is a linear transport equation for r with the given coefficients $\mathbf{J}$.

$$\nabla \times (r\mathbf{J}) = -\mathbf{J}^{\perp} \cdot \nabla r + \left(\partial_x J^y - \partial_y J^x\right) = 0,$$

where $\mathbf{J}^{\perp} = \begin{pmatrix} -J^y \\ J^x \end{pmatrix}$. Certainly, the hyperbolicity will manifest itself in the algorithm in a certain way. The transitive reconstruction process described in the Section 3 reflects one of the hyperbolic nature, transporting the information from the boundary. We will examine further about this issue.

Before we inquire on VRN algorithm, let us look at two other approaches. In the consideration of the hyperbolicity, one natural approach is to consider the method of characterisitc. The characteristic lines of the equation (3) for isotropic resistivity are the integral curves of the vector field $\mathbf{J}^{\perp}$. With appropriate assumptions, all of the curves can be parameterized by two parameters s and t,

$$\mathbf{x}'(s,t) = \mathbf{J}^{\perp}(\mathbf{x}(s,t)), \tag{10}$$

$$\mathbf{x}(s,0) \in \partial\Omega. \tag{11}$$

Here s is a parameter labelling each characteristic line which can be identified with its initial position on the boundary, and t is a parameter to develop a curve for a fixed s. Then, one can solve for r by integrating from the boundary

$$r(\mathbf{x}_0) = r(\mathbf{x}(s_0,0)) \exp\left(\int_0^{t_0} \nabla \times \mathbf{J}(\mathbf{x}(s_0,\tau))\, d\tau\right), \tag{12}$$

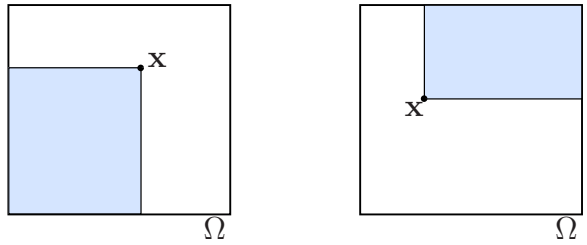
where $\mathbf{x}_0 = \mathbf{x}(s_0,t_0)$, and $\mathbf{x}(s_0,0) \in \partial\Omega$ so that $r(\mathbf{x}(s_0,0))$ is known. This is the formula (20) in the paper [19].

One other possible way is to introduce an intermediate step constructing voltage u first. This is from a following observation. In case of the isotropic resistivity,

$\nabla u \cdot \mathbf{J}^{\perp} = 0$, since ∇u and $\mathbf{J}$ are parallel. In other words, the characteristic lines are also the equipotential lines, that is to say, u is constant on each line. After determining u, r can be obtained from differential ∇u together with the given data $\mathbf{J}$ and Ohm's Law.

One may implement this procedure numerically. For the former, the procedure is implemented by constucting characteristic lines numerically, and approximating the integral in (12). For the latter, the procedure is implemented by constructing characteristic lines numerically, and approximating the differentials ∇u by differences of u. The latter is called the Equipotential Line algorithm, see [7]. Let us then refer to the former integration approach as the Direct Integration algorithm.

The described two algorithms make an emphasis on the hyperbolicity of the equation (3), while VRN algorithm is faithful to realize consistently the physical Laws in discrete level. One difference from that come to clear in the Figure 6. In the Figure 6(a), the colored region is where reconstructed resistivity at a given point $\mathbf{x} \in \Omega$ depends on of the current data and resistivity values reconstructed earlier. The region is called the domain of dependence. Similarly, the data at a point $\mathbf{x} \in \Omega$ will be used in deciding the resistivity in the the colored region in Figure 6(b), where the region is called the domain of influence. On the while, it is clear that in the two characteristic



(a) Domain of dependence

(b) Domain of influence

Figure 6: The domain of dependence of a conductivity value and the domain of influence of the data at a point $\mathbf{x} \in \Omega$ are in the figures.

line based algorithms, the domain of dependence and the domain of influence of a certain point $\mathbf{x}$ is restricted in a characteristic line crossing the point.

Therefore, if there arose a wrong value at a certain point $\mathbf{x}$ due to the noise or the discretizing process, for the two characteristic line based algorithms, it will be transported along the characteristic line that passes the point $\mathbf{x}$ without damping, which is the nature of the hyperbolic problem. For the VRN, however, the domain of influence, the colored region in the Figure 6(b) suggests that the noise at a single point have a chance to be diffused away, which can be observed in numerical simulations.

The hyperbolic nature is weakened in the algorithm and the reconstruction process is intrinsically regularized.

Consider such a case the virtual network is aligned along the characteristic lines as in Figure 7, then VRN gives a result which is similar to the Direct Integration

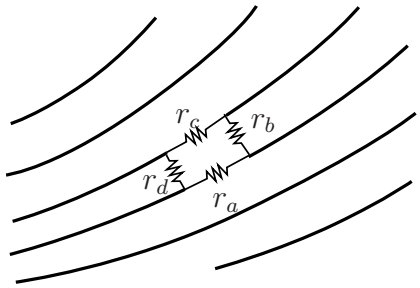


Figure 7: Equipotential Lines. If VRN is aligned along equipotential lines, then the conductivity reconstruction process becomes more sensitive to noise.

algorithm. Since no electrical current passes along the equipotential lines, we have $J_a = J_c = 0$. Therefore,

$$r(= r_b = r_c) = r_d J_d / J_b$$

and hence only the information of r_d is used and the domains of dependence and influence are restricted in the characteristic line. Therefore, the problem is not anymore regularized, and VRN becomes like the Direct Integration algorithm. In other words, it is important not to align the virtual network system in a direction which is parallel to the equipotential lines as in Figure 7.

3.1.2 Directional dependency

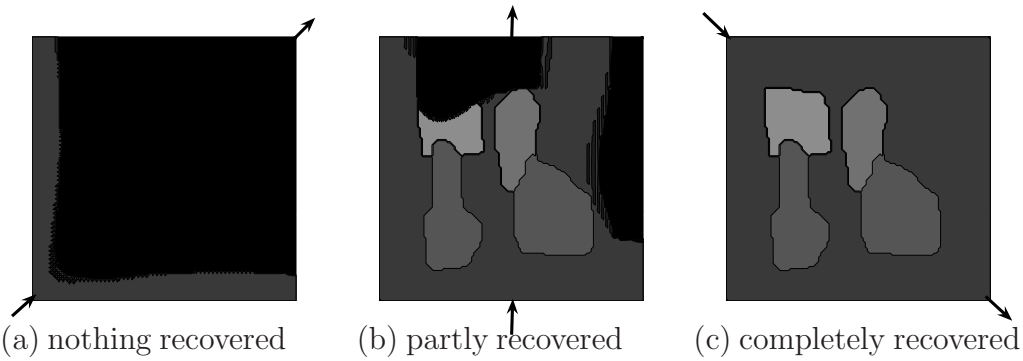
We shall now discuss the second property of the VRN in association with the choice of an asymmetry. For a configuration of Figure 5(b), the formula (9) has $J_b - J_c$ in the denominator. If another choice of asymmetry is taken, for instance, a configuration the upper and the left resistivity values are identified, and the right and the bottom resistivity values are a-priorily known, one will have the formula

$$r(= r_c = r_d) = \frac{r_a J_a + r_b J_b}{J_c + J_d},$$

and the denominator will be $J_c + J_d$.

If the denominator vanishes, the algorithm cannot be continued on. In case of Figure 5(b), the denominator $J_b - J_c$ of formula (9) suggests that the algorithm will show bad behavior in the case that the vertical component J_b and the horizontal

component J_c have same signs, which possibly reduces the size of denominator significantly. In other words, this choice of asymmetry, explicitly prefer the **J** data whose tendency of direction is the south eastern or the north western. We did not put the detailed setting of the simulation for Figure 8 here, but it illustrated this feature explicitly. For other configuration of asymmetry, the preferred direction is altered.



(a) nothing recovered (b) partly recovered (c) completely recovered

Figure 8: Two dimensional isotropic conductivity has been recovered under three different injection currents. Injection currents are denoted by arrows. The best result is (c). In this example noise levels are all zero.

One might give the tendency of the direction of current data that will be a trade-off between the strength of injecting current compared to the variation and the size of resistivity, and the safety of this measurement. However, it is certain that one cannot control the direction of the current pointwisely.

3.2 Characteristics of Square-VRN : Orthotropic

Now we investigate characteristics of the VRN algorithm for orthotropic resistivity reconstruction.

3.2.1 Hyperbolic nature

The system of two equations for the orthotropic problem also is hyperbolic. They are

$$\nabla \times (r\mathbf{J}_1) = 0,$$
$$\nabla \times (r\mathbf{J}_2) = 0,$$

where r is diagonal matrix field. If $r = \begin{pmatrix} a & 0 \\ 0 & b \end{pmatrix}$, then

$$\begin{pmatrix} \partial_y a \\ \partial_x b \end{pmatrix} = \begin{pmatrix} -J_1^x & J_1^y \\ -J_2^x & J_2^y \end{pmatrix}^{-1} \begin{pmatrix} \partial_y J_1^x & -\partial_x J_1^y \\ \partial_y J_2^x & -\partial_x J_2^y \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} =: \mathbf{A} \begin{pmatrix} a \\ b \end{pmatrix}.$$

If wants, by introducing $\tau = x + y$ and $\xi = x - y$, the system can be in the standard form of hyperbolic system of first order equations,

$$\begin{pmatrix} a \\ b \end{pmatrix}_\tau = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix}_\xi + \mathbf{A} \begin{pmatrix} a \\ b \end{pmatrix}.$$

The two families of characteristic lines for this system are the all horizontal lines and the all vertical lines. In the contrary to the isotropic problem, the characteristic lines are not dependent on the data but are always the horizontal and the vertical ones.

For this system, horizontal components of resistivity and vertical components of resistivity couple to each other, and their shared domain of dependence are an area enclosed by two characteristic lines and the domain of influence are similar as in 6(a) and 6(b). Hence, although the edges are arranged to be parallel to the characteristic lines in our square network, it can be said that the noise still have a chance to diffuse out.

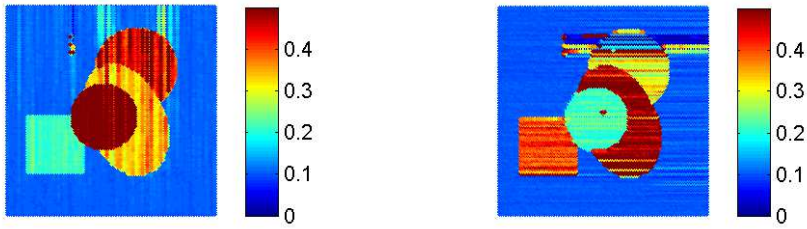
However, it becomes problematic if the data $\mathbf{J}_1$ and $\mathbf{J}_2$ also are parallel respectively to each family of characteristic lines, or equivalently $\mathbf{J}_1^\perp$ and $\mathbf{J}_2^\perp$ are. Indeed, this was the case we had a danger in the isotropic problem that the three of the network arrangement, characteristic lines, and $\mathbf{J}^\perp$ are aligned.

When this happens, for instance $\mathbf{J}_1$ mainly consists of horizontal vectors and $\mathbf{J}_2$ of vertical vectors, then for such a region, the formula (8) becomes

$$r_c = -J_a^1 / J_c^1 \, r_a,$$
$$r_d = -J_b^2 / J_d^2 \, r_b.$$

In other words, the domain of dependence and influence of a certain horizontal resistivity are the one line of horizontal edges including the concerned edge along the vertical axis. The ones of a vertical resistivity are one line of vertical edges along the horizontal axis. Then a noise in a horizontal edge will just be transported vertically, and a noise in a vertical edge will be transported horizontally. These phenomena are indeed observed in simulations as illustrated in Figure 9(a) and 9(b). We did not

put the detailed configuration of this simulation, but we can see from figures vertical stripes and horizontal stripes. The current is injected only through the bottom, and extracted only from the top for the $\mathbf{J}_1$ so that it has the vertical directional tendency, and the current is injected only through the left, and extracted only from the right for the $\mathbf{J}_2$ so that it has the horizontal directional tendency.



(a) Horizontal component (b) Vertical component

Figure 9: The reconstructed image of orthotropic conductivity with $\mathbf{J}_1$ and $\mathbf{J}_2$, where $\mathbf{J}_1$ has a vertical directional tendency, and $\mathbf{J}_2$ has a horizontal directional tendency.

4 Algorithm Specification and Preparations for Simulation

We are now in a position to specify our main algorithm that is used in our simulations. It is the one that we take those characteristics investigated in the previous Sections with the square network into considerations.

4.1 Algorithm Specification : Isotropic

4.1.1 Rotation of network

As analyzed in Section 3, the reconstruction of isotropic resistivity in the square network on which the isotropy is implemented asymmetrically as in Figure 5(b), shows a data dependent performance. In order for each region to have a chance to be in a right configuration, one may think of altering a choice of asymmetry, and see the new reconstruction result. To take an example, one may alter an asymmetry identifying the resistivity on the upper and the left edges.

Note that identifying the resistivity on the upper and the left edges is equivalent to rotating the previous virtual resistive network counter-clock-wisely by 90 degrees.

With this observation, we also can consider to rotate the virtual network by any angle.

For a given data, which can have any tendency of direction that is also can vary by region to region, we place virtual networks rotated by an angle from 0 to 360 degrees from the fixed one, and see the reconstruction result for each angle. It is an advantage of the algorithm that this procedure is affordable.

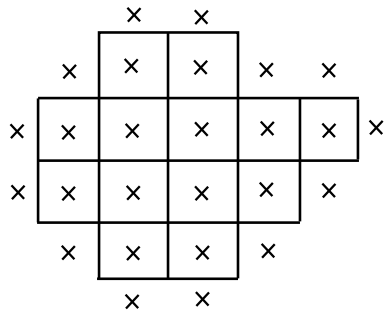
It has to be noticed, however, that we are not saying that this prescription can resolve the problem of the possibility to have vanishing denominator in formula (9). Even if the network in any local region has at least one chance to be aligned correctly to the data, the values in the region are influenced by ones in the outside of the region that are possibly already contaminated. Hence it is desirable for the region not to be far from the boundary, and thus one may consider an option to divide the domain into a few patches and reconstruct the resistivity on each patch using its nearest boundary information. We will not incorporate this procedure in this paper, however.

4.1.2 Stream function generation

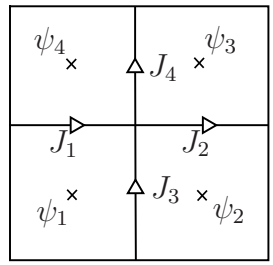
It is an important and non-trivial issue to assign current on tilted edges consistently. When a certain algorithm requires to use a current value at a different point other than ones provided by data, we need to interpolate the vector quantity. It has to be a combination of x and y components of $\mathbf{J}$ at nearby points with correct weights. Furthermore, it has a constraint that the current has to be divergence-free, and thus mere interpolation of the vector data is not sufficient. As a resolution of this problem, we offer a consistent way to accomplish the procedure.

Let us begin with a certain configuration of network. It is assumed that Kirchoff's Current Law is satisfied on all vertices. Boundary vertices can have fewer emanating edges in applying the law. Then it is possible to well-define the discrete stream function ψ up to an addition of a constant on each midpoints of cell and a few more points that are marked by $\times$ in Figure 10(a). It is easy to verify that we can define ψ such that for examples in Figure 10(b),

$$\begin{aligned}
\psi_4 - \psi_1 &= J_1, \\
\psi_1 - \psi_2 &= J_3, \\
\psi_2 - \psi_3 &= -J_2, \\
\psi_3 - \psi_4 &= -J_4.
\end{aligned}$$



(a) × points



(b) Stream function and current

Figure 10: The points marked by × are where ψ is defined.

Although it seems that there are more equations than unknowns, Kirchoff's Current Law guarantees that they are not over-determined. One here sees an use of concept of potential in a discrete geometry.

Now, it is easy to obtain interpolated value of ψ for other points than the midpoints since it is a real-valued function, but is not a vector field anymore. Having made this point, we may go on from this to conclusion that we can access $\psi(\mathbf{x})$ at any point $\mathbf{x} \in \Omega$ with an required accuracy if provided that ψ is sufficiently smooth function.

As a consequence, now we are free to place virtual resistive network consistently in any shape and in any arrangement. Suppose we have another virtually placed network that is tilted in a certain angle. Then one realizes that once values of ψ at midpoints of cells for the tilted network are known, it is easy to assign consistent current values on edges. We assign the current on each edge by subtracting the midpoint values of ψ of cells sharing the edge with an appropriate sign according to the designated arrows as in Figure 2(c). As discussed in Section 2, a discrete level divergence-free condition, the Kirchoff's Current Law is automatically satisfied.

Let us mention one other advantage of using stream function, a fact that ψ is the raw data of the MREIT measurement. This indicates that the data can be acquired with minimum additional process. The two dimensional model we are concerned with can be regarded as three dimensional model which is symmetric in the z-direction. It is treated as one slice of infinitely long z-directional cylinder, which is an ideal situation compared to a practical situation. For this model, one assumes the z-directional derivatives of involved quantities vanish due to the symmetry. Then,

from the Ampere's Law,

$$\begin{pmatrix} J^x \\ J^y \end{pmatrix} = \frac{1}{\mu_0} \begin{pmatrix} \partial_y B^z \\ -\partial_x B^z \end{pmatrix}$$

on any slice. It is clear from above that ψ is a constant multiple of the B^z, which is a raw data measured in MREIT.

4.1.3 Algorithm

We are now ready to write down a pseudo-code for our algorithm for isotropic resistivity reconstruction as follows.

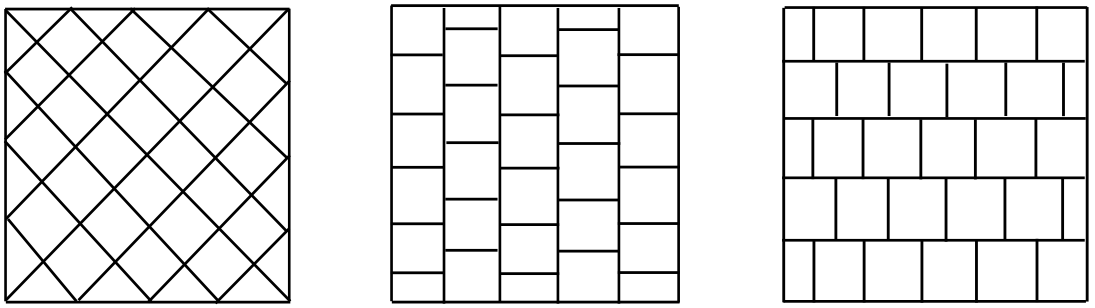
```
For θ = 0 to 360 with step size Δθ
    1. Place VRN with tilted angle θ.
    2. Interpolate ψ at midpoints of cells of placed VRN.
    3. Compute the currents on all edges of placed VRN.
    4. Reconstruct all the resistivity using formula (9).
    5. Display result.
End
```

4.2 Algorithm Specification : Orthotropic

4.2.1 Mimicking the diagonal network

Now, we turn to the orthotropic resistivity reconstruction problem. We inquired in Section 3 to conclude we have a danger when the data are aligned to the network. One possibility in order to avoid a situation that have the domain of dependence and the domain of influence be restricted on a line, is to consider a network tilted by 45 degree as in Figure 11(a). Then, it cannot happen that the characteristic lines that are parallel to either horizontal lines or vertical lines, align to the network because the network is now tilted. However, this is to go to another story because each real value assigned on each edge is not anymore an eigenvalue of the resistivity field.

One another possibility is still available. Without forgiving the easy implementation of orthotropic resistivity with squares, one may consider virtual networks of Figure 11(b) and 11(c). Both of virtual networks are mimicking the tilted network but with cells not tilted. Consider a network of Figure 11(b) with two data $\mathbf{J}_1$ and $\mathbf{J}_2$ that are parallel to horizontal lines and vertical lines respectively, which was the



(a) Diagonal disposition of network

(b) Alternate disposition of Horizontal Edges

(c) Alternate disposition of Vertical Edges

Figure 11: The two different virtual networks that mimic a diagonal network.

case we encountered a problem. For this configuration, one verifies that the horizontal resistivity is still depends on and influenced by the vertical line, but the vertical resistivity is now depends on the colored region in the Figure 11(b). It is opposite with a configuration of Figure 11(c).

Therefore, we are going to reconstruct twice, one with a network of Figure 11(b), and one with a network of Figure 11(c), and then pick the vertical values from the former, and the horizontal ones from the latter. Of course, to place the two networks, stream functions of $\mathbf{J}_1$, and $\mathbf{J}_2$ are used.

4.2.2 Algorithm

We are now ready to write down a pseudo-code for our algorithm for orthotropic resistivity reconstruction as follows.

1. Place VRN as in Figure 11(b).
2. Interpolate stream functions ψ_1 of $\mathbf{J}_1$, and ψ_2 of $\mathbf{J}_2$ at mid points of cells of placed VRN.
3. Compute currents $\mathbf{J}_1$ and $\mathbf{J}_2$ on all edges respectively.
4. Reconstruct all the resistivity using formula (8).
5. Take values on horizontal edges from the reconstructed result.

6. Place VRN as in Figure 11(c).
7. Interpolate stream functions ψ_1 of $\mathbf{J}_1$, and ψ_2 of $\mathbf{J}_2$ at mid points of cells of placed VRN.
8. Compute currents $\mathbf{J}_1$ and $\mathbf{J}_2$ on all edges respectively.
9. Reconstruct all the resistivity using formula (8).
10. Take values on vertical edges from the reconstructed result.

4.3 Noise model

Now, we specify noise model used in the simulation. We have added *multiplicative* and *additive noises* to the current data.

By a $p\,\%$ multiplicative noise for a quantity f we mean

$$\left(\frac{p}{100}|f(\mathbf{x})|\right) X,$$

where $-1 \leq X \leq 1$ is a random number with a uniform distribution.

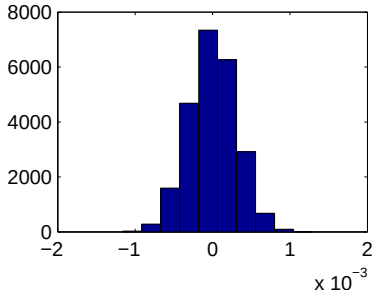
Additive noise, denoted by a random number N, is modeled by a normal distribution of average 0, and the strength of noise is parameterized by its standard deviation.

We have added the one with the standard deviation 0.0003 for a physical domain of a circle with diameter $50cm$ with $10mA$ current injection. We will specify the setup in detain in the next Section. In [9], a background noise with a scale of 10^{-9} standard deviation has been used for a three dimensional domain of size $32cm \times 32cm \times 64cm$ and the $1500mA$ injecting current. In our simulation, the diameter $50cm$ was discretized to have 128 vertices, and we had totally 23868 edges for a circular domain for example. If we set up a 3 dimensional resistive network with a rectangular parallelepiped $32cm \times 32cm \times 64cm$ with the same edge length, then there will be about $1,600,000$ edges with injecting current $1500mA$. In average, the size of current would be double to our simulation. In this consideration, the standard deviation of our simulation is quite large.

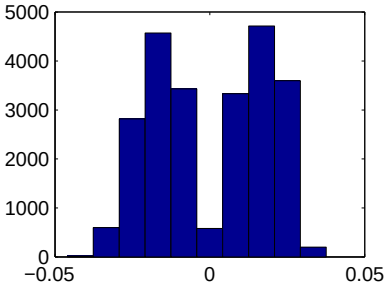
It would have only a restricted meaning, however, to compare the strength of noises between two simulations in this way, and the only meaningful discussion on the issue would be illustrated by comparison of size of noises to the one of data, and by total amount of contribution of noises. In Figure 12 we displayed the histogram of additive noises and the one of a sample data that is used in simulations. Total number of random number and the values of data were both 23868, the number of edges. One may feel the size of noises in L^∞ or L^1 sense from the histograms.

In summary, we have used in simulations a discrete field J defined on edges such that

$$J = J^{exact} + \left(\frac{p}{100}|J^{exact}|\right) X + N.$$



(a) Noise



(b) J

Figure 12: Two histograms of absolute values of noise and J respectively. Total number of sample is 23868 for both of histograms.

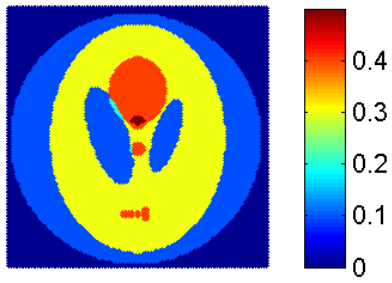


Figure 13: True conductivity image used in simulations of isotropic resistivity reconstruction. The domain is from an outer blue-colored circle and outside of the circle is excluded for our domain so color there is a redundant one.

4.4 Simulation setup : Isotropic

The true conductivity σ used in simulations is from a Matlab function, which is given in Figure 13. The true conductivity has been scaled to be

$$0.1 \leq \sigma \leq 0.5, \quad \text{or equivalently,} \quad 2 \leq r \leq 10$$

for resistivity. The challenging part of this target conductivity is its sharp discontinuity between different regions, which causes a lot more trouble than smooth ones. However, the simulation results are still okay, which we will see in next Section, and the method seems more robust than the theory says.

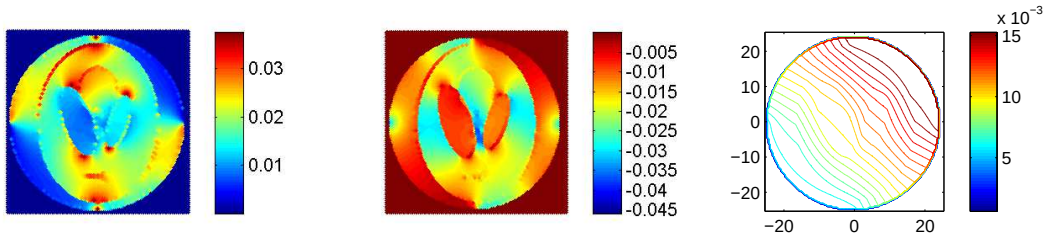
The physical domain of conductivity body used in the simulation was a circle with a diameter $d = 50cm$. An electrical current of total amount $10mA$ was uniformly injected through the portion of boundary that is lied in a range of angle $\frac{\pi}{2} \leq \theta \leq \pi$

assuming the center of the circle is at origin. The same amount of current is uniformly extracted from the portion of boundary that is lied in a range of angle $-\frac{\pi}{2} \leq \theta \leq 0$.

The boundary condition we have assigned guarantees that the whole set of characteristic lines are parameterized by two variables t and s as in (10) and (11). See the Theorem 1 and discussions in [19]. This is for the Equipotential Line and the Direct Integration algorithms, which we are going to compare with.

To generate an electrical current, we also have used the network, prescribed for forward problem. One might use any 2 dimensional poisson solver, for examples, a Matlab FEM solver or a Matlab FDM solver. One can veryfy, for example from [18], that the network prescription for forward problem is identical to FDM.

For the network solver, we have currents naturally on edges. This is same for any FDM solver, if one regards edges between grid points as resistors. For a case that a solver provides currents as vectors on grid points, one needs an intermediate step to distribute the vectors into the network edges to use network for backward problem.



(a) Horizontal component of J (b) Vertical component of J (c) Stream function of J

Figure 14: The current J used in simulations of isotropic resistivity reconstruction.

From the physical settings above, we obtained an electrical current density of scale

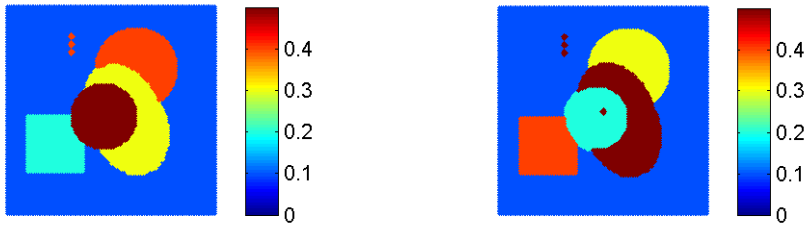
$$1.1 \times 10^{-3} \ \ [A/m^2] \leq \quad |J| \quad \leq 4.6 \times 10^{-2} \ \ [A/m^2].$$

The generated current is illustrated in Figure 14. In particular, the Figure 14(c) is the contour plot of the stream funtion. The level set curves are the streamlines of current vector field. Indeed, they flow from left upper to right bottom.

The dimension of a current density in our two dimensional model is in fact $[A/m]$, which is a current on an edge divided by the length of an edge, but we put an SI units $[A/m^2]$ in a sense that our two dimensional domain is one of identical slices of three dimensional infinitely long cylinder.

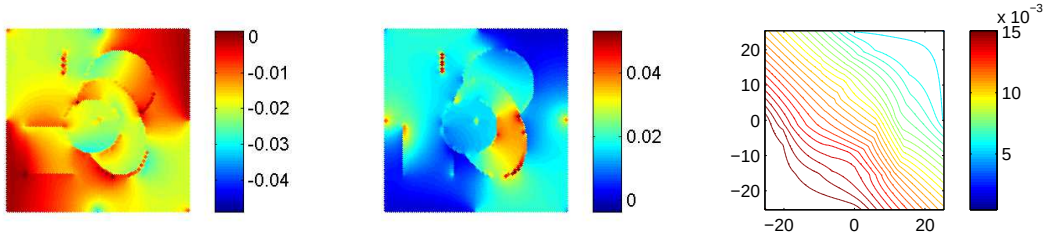
4.5 Simulation setup : Othotropic

The true conductivity for the orthotropic reconstruction problem is prepared as follows.

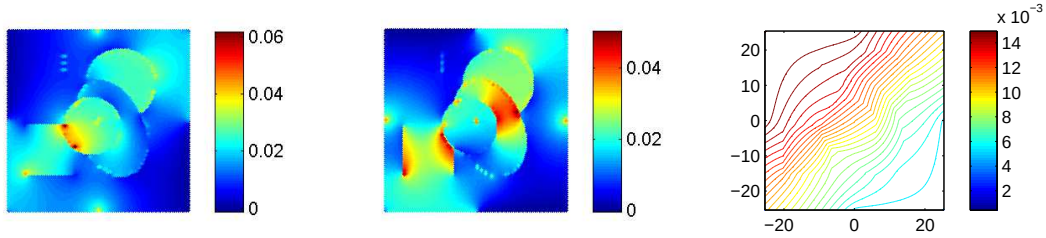


(a) Horizontal eigenvalue (b) Vertical eigenvaluen

Figure 15: True images of horizontal and vertical eigenvalues of conductivity that are used in simulations of orthotropic resistivity reconstruction and simulation setup.



(a) Horizontal component of J_1 (b) Vertical component of J_1 (c) Stream function of J_1



(d) Horizontal component of J_2 (e) Vertical component of J_2 (f) Stream function of J_2

Figure 16: Two current data J_1 and J_2 used in simulations of orthotropic resistivity reconstruction.

The eigenvalues also have been scaled to be

$$0.1 \leq \sigma_1, \sigma_2 \leq 0.5, \quad \text{or equivalently,} \quad 2 \leq r_1, r_2 \leq 10,$$

where σ_1 and σ_2 are two eigenvalues of conductivity, and r_1 and r_2 are the ones of resistivity.

The physical size of conductivity body used in the simulations was $50cm \times 50cm$. An electrical current of total amount $10mA$ was injected uniformly through the boundary lied in right bottom quadrant assuming the center of body is at origin, and extracted uniformly from the boundary lied in left upper quadrant for the first current data J_1. The same amount through left bottom quadrant and from right upper quadrant for the second data J_2. The locations of injections and extractions are chosen so that the currents generated do not have tendencies that are aligned with network edges. Also the combination of boundary conditions are the ones that guarantee $\mathbf{J}_1 \times \mathbf{J}_2 \neq 0$. They are illustrated in Figure 16.

The sizes of absolute value of data were

$$\begin{aligned} 3.7 \times 10^{-5} \ [A/m^2] \leq \ & |J_1| \ \leq 5.3 \times 10^{-2} \ [A/m^2], \\ 1.2 \times 10^{-4} \ [A/m^2] \leq \ & |J_2| \ \leq 6.2 \times 10^{-2} \ [A/m^2]. \end{aligned}$$

5 Simulation Results

5.1 Simulation results : Isotropic

In this section we presented the numerically reconstructed conductivity results using VRN algorithm in particular that is specified in Section 4. They are reconstructed from the simulated data that are prepared as described in Section 4. We also listed results using the Equipotential Line algorithm and the Direct Integration algorithm.

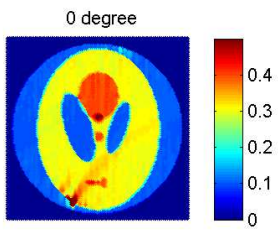
Figure 17 illustrates various reconstructed results. The figures in the first column are the ones that are reconstructed from the J generated as specified in the Section 4.4 with 1% multiplicative noise only. The ones in the second column are from the J with 5% multiplicative noise. The ones in the last column are from J with 5% multiplicative noise and with additive noise with standard deviation 0.0003.

One can see that the VRN shows a distinguished stability against noises. One can observe stripes in the direction of equipotential lines. In particular one can find more of them when equipotential lines cross discontinuities. These stripes are relatively weak in VRN simulations in compare with other methods, as was expected from the studies on the characteristic lines.

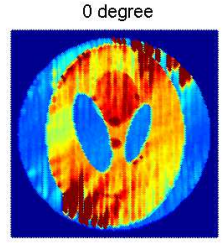
See the results in the third column that are of a case with 5% of multiplicative noises and additive noises with standard deviation 0.0003. The additive noises results

in more severe distortion. One can find wrinkles produced due to the noise. However, the simulations gives acceptable images and still VRN gives the most robust image among these three algorithm. The direct integration gives a little bit better result than the equipotential line method does. The wrinkles in figures of the Equipotential Line and the Direct Integration have been magnified and deteriorate the image. However, the image by VRN method looks more robust and noises are neutralized.

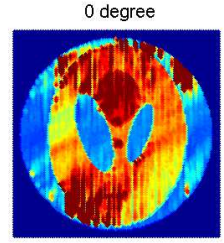
In the second row, we can see the figures from 10 degree tilted VRN. Many distorted values in the first row except the ones of brown region in the left bottom clearly get better in the figures in the second row. The three small dots in the bottom of the images are hardly seen in figures of the Equipotential Line and the Direct Integration, which are in the third and the fourth row respectively. However, the image of VRN one can find a thin spots for it in the second row. One indeed observes the advantage of VRN algorithm that we have a different chance to look at the data by different VRN.



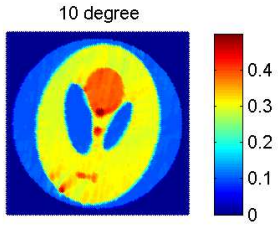
(a) VRN



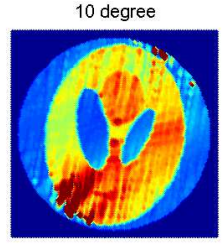
(b) VRN



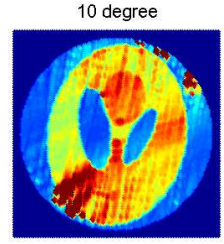
(c) VRN



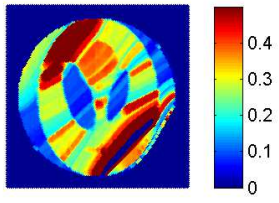
(d) 10° tilted VRN



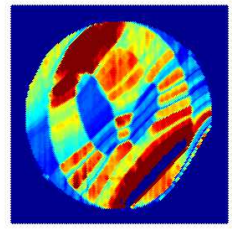
(e) 10° tilted VRN



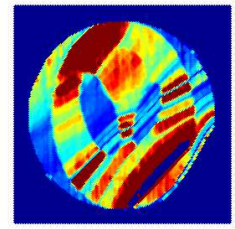
(f) $10^n\circ$ tilted VRN



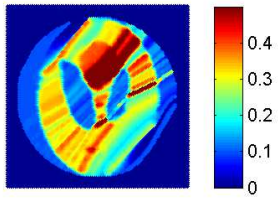
(g) Equipotential Line



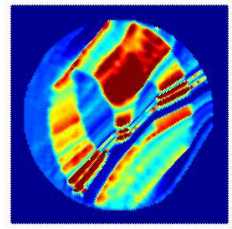
(h) Equipotential Line



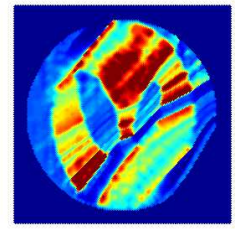
(i) Equipotential Line



(j) Direct Integration



(k) Direct Integration



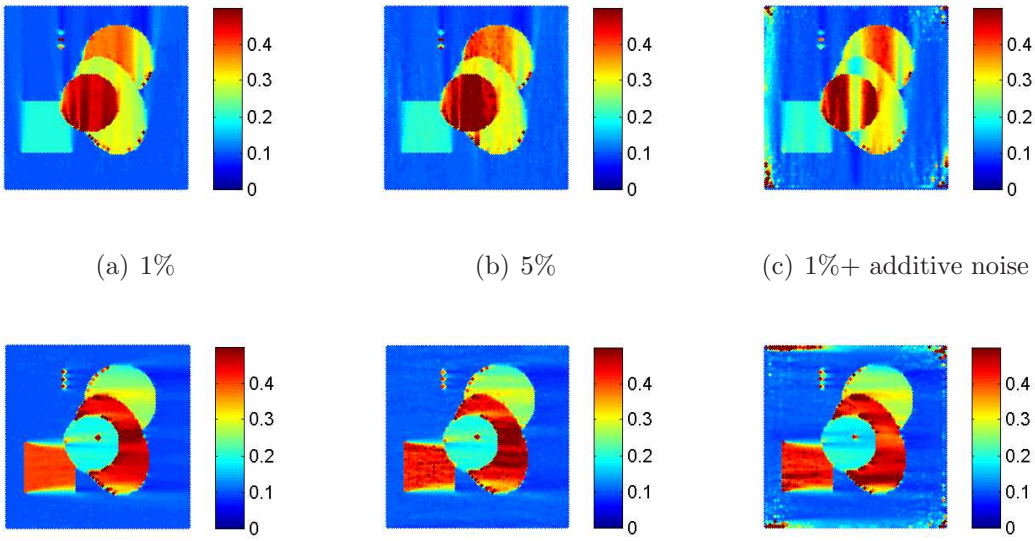
(l) Direct Integration

Figure 17: The various reconstructed isotropic conductivity images from one J by three different algorithms. The figures in the first column were reconstructed from the J with 1% multiplicative noise, and the ones in the second column with 5% multiplicative noise, and the ones in the third column with both of 5% multiplicative noise and additive noise with $s.d. = 0.0003$.

5.2 Simulation results : Orthotropic

Let us now discuss on the results of orthotropic resistivity cases. In Figure 18, figures in the first row are the horizontal eigenvalues reconstructed, and the ones in the second row are the vertical eigenvalues. They are reconstructed with 1% muliplicative noise, 5% multiplicative noise, and 1% multiplicative noise together with additive noise with standard deviation 0.0003 respectively. The data J_1 and J_2 are the ones generated as specified in the Section 4.5

Other than the images of Figure 9, the stripes are significantly reduced, although we still see them in particular in the figures in the third column. However, the edges of each regions can be clearly identified. It is very impressive that the small dot in the center of the images of the second row are clearly seen in all of them. The three small dots in the left upper region also are observed as well in all of the figures.



(a) 1% (b) 5% (c) 1%+ additive noise

(d) 1% (e) 5% (f) 1%+ additive noise

Figure 18: The orthotropic conductivity is reconstructed from J_1 and J_2 that has 1% multiplicative noise, and 5% multiplicative noise, and additive noise + multiplicative noise of 1%, respectively. The first row are images of horizontal eigenvalues, and the second row are the ones of vertical eigenvalues.

References

[1] Nanping Zhang, *Electrical impedance tomography based on current density imaging.* Master's Thesis, University of Toronto, 1992.

[2] Eung Je Woo, Soo Yeol Lee, and Chi Woong Mun, Impedance tomography using internal current density distribution measured by nuclear magnetic resonance. *Proc. SPIE*, **2299**, 377–85, 1994.

[3] Jin Keun Seo, Ohin Kwon, and Eung Je Woo. Magnetic resonance electrical impedance tomography (MREIT): conductivity and current density imaging. *Journal of Physics: Conference Series*, **12**, 140, 2005.

[4] Eung Je Woo, and Jin Keun Seo, Magnetic resonance electrical impedance tomography (MREIT) for high-resolution conductivity imaging. *Physiological Measurement*, **29**, no. 10, R1, 2008.

[5] Jin Keun Seo,n and Eung Je Woo, Magnetic resonance electrical impedance tomography (MREIT). *SIAM Review*, **53**, no. 1, 40–68, 2011.

[6] Ohin Kwon, Eung Je Woo, Jeong-Rock Yoon, and Jin Keun Seo, Magnetic resonance electrical impedance tomography (MREIT): simulation study of j-substitution algorithm. *Biomedical Engineering, IEEE Transactions on*, **49**, no. 2, 160–167, February 2002.

[7] Ohin Kwon, June-Yub Lee, and Jeong-Rock Yoon, Equipotential line method for magnetic resonance electrical impedance tomography. *Inverse Problems*, **18**, no. 4, 1089–1100, 2002.

[8] Adrian Nachman, Alexandru Tamasan, and Alexandre Timonov, Conductivity imaging with a single measurement of boundary and interior data. *Inverse Problems*, **23**, no. 6, 2551–2563, 2007.

[9] Y.Z. Ider, and S. Onart, Algebraic reconstruction for 3D magnetic resonance-electrical impedance tomography (MREIT) using one component of magnetic flux density. *Physiological measurement*, **25**, 281, 2004.

[10] Hyun Soo Nam, Chunjae Park, and OhIn Kwon, Non-iterative conductivity reconstruction algorithm using projected current density in MREIT. *Physics in medicine and biology*, **53**, no. 23, 6947, 2008.

[11] Giovanni Alessandrini, An identification problem for an elliptic equation in two variables. *Annali di matematica pura ed applicata*, **145**, no. 1, 265–295, 1986.

[12] Gerard R. Richter, An inverse problem for the steady state diffusion equation. *SIAM Journal on Applied Mathematics*, **41**, no. 2, 210–221, 1981.

[13] Gerard R. Richter, Numerical identification of a spatially varying diffusion coefficient. *mathematics of computation*, **36**, no. 154. 375–386, 1981.

[14] Guillaume Bal, Eric Bonnetier, Franois Monard, and Faouzi Triki, Inverse diffusion from knowledge of power densities. *Inverse Problems and Imaging*, **7**, no. 2, 353–375, 2013.

[15] Franois Monard, and Guillaume Bal, Inverse anisotropic conductivity from power densities in dimension $n \geq 3$. *Comm. Partial Differential Equations*, **38**, no. 7, 1183–1207, 2013.

[16] Franccois Monard, and Guillaume Bal, Inverse anisotropic diffusion from power density measurements in two dimensions. *Inverse Problems*, **28**, no. 8, 084001, 2012.

[17] Franccois Monard, and Guillaume Bal, Inverse diffusion problems with redundant internal information. *Inverse Problems and Imaging*, **6**, no. 2, 289–313, 2012.

[18] Gilbert Strang, *Introduction to applied mathematics*. Wellesley-Cambridge Press, Wellesley, MA, 1986.

[19] Yong-Jung Kim and Min-Gi Lee, Well-posedness of the conductivity reconstruction from an interior current density in terms of schauder theory. *To appear in Quart. Appl. Math.*

[20] Sungwhan Kim, Ohin Kwon, Jin Keun Seo, and Jeong-Rock Yoon, On a nonlinear partial differential equation arising in magnetic resonance electrical impedance tomography. *SIAM Journal on Mathematical Analysis*, **34**, no. 3, 511–526 (electronic), 2002.