Teknillinen korkeakoulu
Teknillisen fysiikan ja matematiikan osasto

Sampsa Pursiainen

Numerical Methods in Statistical EIT

Diplomi-insinöörin tutkintoa varten tarkastettavaksi jätetty diplomityö

Espoo 13.10.2003

Työn valvoja: professori Erkki Somersalo
Työn ohjaaja: professori Erkki Somersalo

Teknillinen korkeakoulu   Diplomityön tiivistelmä
Teknillisen fysiikan ja matematiikan osasto

Tekijä:	Sampsa Pursiainen
Osasto:	Teknillisen fysiikan ja matematiikan osasto
Pääaine:	Matematiikka
Sivuaine:	Mekaniikka

Työn nimi:	Tilastollisen EIT-ongelman numeeriset menetelmät
Title in English:	Numerical Methods in Statistical EIT
Professuurin koodi ja nimi:	Mat-1 Matematiikka
Työn valvoja:	Professori Erkki Somersalo
Työn ohjaaja:	Professori Erkki Somersalo

Tiivistelmä:

Impedanssitomografia (EIT) on kuvantamismenetelmä, jolla selvitetään kaksi tai kolmiulotteisen kappaleen sähkömagneettisia ominaisuuksia perustuen kappaleen reunalla tehtäviin mittauksiin. Tässä työssä tuntematon on skalaariarvoinen johtavuusjakauma, kappaleeseen syötetään virtoja sen reunalle kiinnitetyjen elektrodien avulla ja virtojen aiheuttamat potentiaalit mitataan.

Työn tarkoitus on esitellä menetelmiä, joiden voidaan melko yleisesti sanoa sopivan hyvin EIT-ongelman numeeriseen ratkaisemiseen. Lisäksi menetelmiä sovelletaan yksinkertaisen esimerkkitapauksen ratkaisemiseen. Numeerisen ratkaisemisen vaatima laskennallinen työmäärä on yleensä suuri ja riippuu sovellettujen menetelmien tehokkuudesta. Tavoitteena on löytää työmäärältään mahdollisimman halpoja menetelmiä.

Työssä keskitytään käänteisongelman Bayeslaiseen ratkaisemisemiseen, jossa tehtävän tuntematonta mallinnetaan tiettyä todennäköisyysjakaumaa, nk. *posteriorijakaumaa* noudattavana satunnaismuuttujana. Posteriorijakauman ominaisuuksia estimoidaan nk. MCMC-menetelmien (*Markov chain Monte Carlo*) avulla. MCMC menetelmät ovat tilastollisia algoritmeja, joilla voidaan tuotettaa otoksia mielivaltaisista todennäköisyysjakaumista. Tavoitteena kehittää algoritmi, joka konvergoisi mahdollisimman nopeasti, ts. vaatisi mahdollisimman pienen otoksen tuottamista. Posteriorijakauman ominaisuuksien arvioiminen vaatii diskreetin suoran ongelman toistuvaa ratkaisemista. Toinen tärkeä tavoite onkin löytää mahdollisimman nopea lineaarialgebrallinen menetelmä suoran ongelman ratkaisemiseen. Tilastollisen menetelmän antamia estimaatteja verrataan regularisoidun pienimmän neliösumman menetelmien antamiin estimaatteihin.

Simulaatioissa rajoitutaan yksinkertaiseen tapaukseen, jossa vakiojohtavuudesta etsitään anomaliaa, ts. pientä poikkeamaa. Tehokas menetelmä pienten poikkeamien löytämiseksi on tarpeellinen käytännön sovelluksissa. Esimerkkitapausta vastaava kasvaim pehmeässä kudoksessa.

Saatujen tulosten perusteella on selvää, että kunnollisen numeerisen ratkaisun löytäminen on usein mahdotonta, mikä johtuu ongelman erittäin häiriöalttiista ja epälineaarisesta luonteesta. Tilastollisen menetelmän antamat tulokset ovat selvästi parempia kuin pienimmän nelisumman menetelmän ratkaisut vain, jos johtavuusjakaumasta tiedetään etukäteen tarpeeksi paljon. Erityisesti tapauksessa, jossa johtavuus on luonteeltaan hyvin epäjatkuva tilastollinen menetelmä on edullinen.

Sivumäärä: 82	Avainsanat:	impedanssitomografia (EIT), Bayesialainen statistiikka, MCMC, lineaarialgebra

Täytetään osastolla

Hyväksytty:	Kirjasto:

Helsinki University of Technology

Department of engineering physics and mathematics

Abstract of master's thesis

Author:	Sampsa Pursiainen
Department:	Department of engineering physics and mathematics
Major subject:	Mathematics
Minor subject:	Mechanics

Title:	Numerical Methods in Statistical EIT
Title in Finnish:	Tilastollisen EIT-ongelman numeeriset menetelmät
Chair:	Mat-1 Mathematics
Supervisor:	Professor Erkki Somersalo
Instructor:	Professor Erkki Somersalo

Abstract:

Electrical Impedance Tomography (EIT) is an imaging method that provides information about the electromagnetic properties within a 2D- or 3D-body Ω based on voltage measurements on the boundary $\partial\Omega$. In this case, the sought quantity is a scalar-valued conductivity distribution σ within Ω. Voltage measurements refer to a finite set of potential values that are measured by an array of contact electrodes attached on $\partial\Omega$. The voltage data is generated by injecting currents into the domain through the electrodes.

The issue of this work is to discuss numerical methods that can be applied to the discretized mathematical model of the EIT problem and also to use them in connection with some demonstrative numerical simulations. The computational work that has to be performed before resulting in a proper solution is usually large and can often be diminished remarkably by optimizing the efficiency of the applied numerical methods. One of the central aims of this thesis is to introduce methods that can rather commonly be told to be suitable for solving the EIT problem.

The major interest is concentrated on solving the inverse problem in terms of *Bayesian* statistics by treating σ as a random variable with some posterior probability distribution and by employing *Markov chain Monte Carlo* (MCMC) sampling methods for estimating the properties of the posterior distribution. The purpose is to develop such a Monte Carlo algorithm that finding a proper approximative solution would necessitate as small sample enesembles as possible. Drawing a sample from the posterior distribution demands for solving one or more forward problems, i.e. linear systems. Consequently, another important issue is to discover an effective linear algebraic method of solving the forward problem. Statistical solutions are measured against regularized least-squares solutions which appear more frequently in literature.

In the simulations, we restrict ourselves to cases where σ known in most parts of Ω and only a relatively small anomaly is sought. The need for a method of locating small perturbations arises in connection with various real world applications of EIT such as detecting and classifying tumors from breast tissue.

Summarizing the findings, due to the strong ill-conditioned nature and non-linearity of the inverse problem it is often difficult to obtain any appropriate numerical solutions. The statistical model is preferable to the least-squares approach only if there is accurate enough a priori knowledge available. Especially in cases where the nature of the conductivity distribution is strongly discontinuous it is advantageous to use the statistical formulation.

Number of pages: 82 Keywords: electrical impedace tomography (EIT),
Bayesian statistics, Markov chain Monte Carlo (MCMC),
linear algebra

Department fills

Approved: Library code:

Contents

Chapter 1

Introduction

Electrical Impedance Tomography (EIT) is an imaging method that provides information about the electromagnetic properties within a 2D- or 3D-body Ω based on voltage measurements on the boundary $\partial\Omega$. In this case, the sought quantity is a scalar-valued conductivity distribution σ within Ω. Voltage measurements refer to a finite set of potential values that are measured by an array of contact electrodes attached on $\partial\Omega$. The voltage data is generated by injecting currents into the domain through the electrodes.

The focus of this work is to discuss numerical methods and computational techniques that can be applied to the discretized mathematical model of the EIT problem and also to use them in connection with some demonstrative numerical simulations. The computational work that has to be performed before resulting in a proper solution is usually large and can often be diminished remarkably by optimizing the efficiency of the applied numerical methods. One of the central aims of this thesis is to introduce methods that can rather commonly be told to be suitable for solving the EIT problem.

The EIT problem is divided into the *forward problem*, which is to solve the electrode voltages corresponding to a given σ, and the *inverse problem*, that is to find out σ on the basis of the measured electrode voltages. The forward problem can be formulated mathematically as an $H^1(\Omega)$-elliptic boundary value problem and discretized through the *finite element method* (FEM) yielding a solvable linear system of equations. In contrast, the inverse problem is non-linear and *ill-conditioned*, i.e. small errors in measured data can cause large errors to the solution. Therefore, there is no direct method providing the solution. In this thesis, the major interest is concentrated on solving the inverse problem in a statistical sense by treating σ as a random variable with some posterior probability distribution and by employing *Markov chain Monte Carlo* (MCMC) methods for estimating the properties of the posterior distribution. MCMC methods are relatively simple algorithms that enable creating random but statistically dependent samples from an arbitrary probability distribution. The purpose is to develop such a Monte Carlo algorithm that the convergence would necessitate as small sample size as possible. In the case of the EIT problem drawing a sample from the posterior distribution demands for solving one or more forward problems, i.e. linear systems. Consequently, another important issue is to discover an effective method of solving the forward problem.

The inverse problem cannot be solved without having a priori information about the structure of σ. This is due to the ill-conditioned nature of the problem, the incompleteness of the measured boundary data, the noise in the measurements as well as the limited computational power of the computers. Again, finding out the most workable numerical methods is a most case-specific task. We are particularly interested in discovering methods that are applicable to the numerical demonstrations. Therefore, we attempt to construct the setups in such a way that the demonstrations would be at the same time both simple and close to some imaginable real world application.

In the demonstrations, we restrict ourselves to cases where σ known in most parts of Ω and only a relatively small anomaly is sought. Loosely taken the conductivity distribution is *a priori* assumed to be of the form

$$\sigma = \sigma_{bg} + \delta$$

where σ_{bg} is some background conductivity distribution having some rather well-known structure and δ is a small deviation having a small-sized support. The problem is mainly concentrated on locating δ of right size and value of conductivity. The need for a method of locating small perturbations arises in connection with various real world applications of EIT such as detecting and classifying tumors from breast tissue.

1.1 Outline of the Thesis

In chapter 2, we introduce the mathematical model of the EIT problem. We apply an electrode model, where the voltages are measured by a finite number of contact electrodes lying on $\partial\Omega$. The forward problem is formulated as an $H^1(\Omega)$-elliptic boundary value problem and discretized by employing the finite element method. The inverse problem is formulated both as a regularized *least-squares* (LS) problem and in terms of Bayesian statistics.

Monte Carlo methods are discussed in the chapter 3. We introduce the idea of Monte Carlo sampling techniques, the basics of the Markov chain Monte Carlo and a number of potentially applicable MCMC algorithms.

In chapter 4, we introduce a number of linear algebraic methods. Both direct and iterative methods are discussed. The aim is to find out methods that provide a fast way to solve the discretized forward problem. Workability of a method depends on the dimension of the system, the applied sampling method as well as the a priori knowledge of the structure of the conductivity distribution. Since it is laborious to compare the computational efficiencies in practice, we give just some rough estimates of the computational work loads.

In chapter 5, a small anomaly is sought in some numerical experiments by employing the methods introduced in the previous chapters. We implement both regularized least-squares and statistical algorithms.

Chapter 2

EIT Problem

This section introduces the mathematical model of both the *forward problem* and the *inverse problem.* The representation adopts largely the format of [1]. The statistical formulation of the inverse problem discussed in the last section is based also on [2].

2.1 The Forward Problem

Let $\Omega \in \mathbb{R}^n$, $n = 2, 3$ be a bounded, simply connected domain with a connected complement. We assume that Ω has a smooth boundary. Here, Ω represents the body with known electromagnetic properties. We consider time-harmonic electromagnetic fields in Ω with low frequencies. In the quasi-static approximation, the fields can be described in terms of scalar voltage potential u satisfying the equation

$$\nabla \cdot \sigma \nabla u = 0 \tag{2.1}$$

in Ω. Within this approximation, the function σ is complex valued and describes the admittivity (i.e. the inverse of impeditivity) of the body. We restrict ourselves to cases where the admittivity is real and positive, describing the conductivity of the body, i.e. $\sigma : \Omega \to \mathbb{R}_+$. Physically, this corresponds to the static measurement.

The following definition fixes the admissible class of conductivities.

Definition 1 *A conductivity distribution $\sigma : \Omega \to \mathbb{R}_+$ is in the admissible class of conductivities, denoted by $\mathcal{A} = \mathcal{A}(\Omega)$, if the following conditions are satisfied:*

1. *For some $N \geq 1$, there is a family $\{\Omega_j\}_{j=1}^{N}$ of open disjoint sets, $\Omega_j \subset \Omega$, having piecewise smooth boundaries and for which*

   $$\overline{\Omega} = \bigcup_{j=1}^{N} \overline{\Omega}_j.$$

   Furthermore, we require that $\sigma|_{\Omega_j} \in C(\overline{\Omega}_j)$, $1 \leq j \leq N$, i.e., σ restricted to each subset Ω_j allows a continuous extension up to the boundary of the subset.

2. *For some constants c and C,*

$$0 < c \leq \sigma(x) \leq C < \infty \quad \forall x \in \Omega$$

In medical applications the subsets Ω_j in the forward problem may represent the organs. In the inverse problem, the set of admissible conductivities provides a natural discretion basis.

Due to the possible discontinuities of $\sigma \in \mathcal{A}$, the equation (2.1) must be interpreted in the weak sense, discussed in detail below.

To describe the current injection and voltage measurements on the surface of the body, we define a set of surface patches $e_\ell \subset \partial\Omega$, $1 \leq \ell \leq L$, as a mathematical model of the contact electrodes. The electrodes are strictly disjoint, i.e. $\overline{e}_\ell \cap \overline{e}_k = \emptyset$ for $\ell \neq k$. If $\Omega \in \mathbb{R}^2$, the electrodes are strictly disjoint intervals of the boundary, and in the case $\Omega \in \mathbb{R}^3$, they are sets with a piecewise smooth simple boundary curve on $\partial\Omega$. Let I_ℓ be the electric current injected through the electrode e_ℓ. We call the vector $I = (I_1, \ldots, I_L)^T \in \mathbb{R}^L$ a current pattern if it satisfies the charge conservation condition

$$\sum_{\ell=1}^{L} I_\ell = 0. \tag{2.2}$$

Let U_ℓ denote the voltage on the ℓth electrode, the ground voltage being chosen so that

$$\sum_{\ell=1}^{L} U_\ell = 0. \tag{2.3}$$

The vector $U = (U_1, \ldots, U_L)^T \in \mathbb{R}^L$ is called a voltage vector. In terms of the current patterns and voltages, the appropriate boundary condition for the electric potential is given as

$$\begin{aligned} \int_{e_\ell} \sigma \frac{\partial u}{\partial n}\, dS &= I_\ell, \quad 1 \leq \ell \leq L, & (2.4) \\ \sigma \frac{\partial u}{\partial n}\Big|_{\partial\Omega \setminus \cup e_\ell} &= 0, & (2.5) \\ u + z_\ell \sigma \frac{\partial u}{\partial n} &= U_\ell, \quad 1 \leq \ell \leq L & (2.6) \end{aligned}$$

Here, the numbers z_ℓ are presumably known contact impedances between the electrodes and the body. We use the notation $z = (z_1, \ldots, z_L)^T$ in what follows. For simplicity, we assume that the contact impedances are real. Note that in the forward problem, only the current patterns on the boundary are specified. However, conditions (2.2) and (2.3) alone are not sufficient to uniquely determine the potential u, but one needs to require $u + z_\ell \partial u / \partial n$ to be constant U_ℓ on e_ℓ. Finding these voltages is part of the forward problem.

The following proposition was proved in [11]. In the following, we use the notation

$$\mathbb{H} = H^1(\Omega) \oplus \mathbb{R}^L, \tag{2.7}$$

where $H^1(\Omega)$ is the L^2-based Sobolev-space. Further, we denote

$$\dot{\mathbb{H}} = \mathbb{H}/\mathbb{R} \tag{2.8}$$

equipped with the quotient norm,

$$||(u,U)||_{\dot{\mathbb{H}}} = \inf_{c\in\mathbb{R}} ||(u-c, U-c)||_{\mathbb{H}}. \tag{2.9}$$

Thus, $(u,U) \in \mathbb{H}$ and $(v,V) \in \mathbb{H}$ are equivalent in $\dot{\mathbb{H}}$ if

$$u - v = U_1 - V_1 = \cdots = U_L - V_L = \text{constant}. \tag{2.10}$$

With these notations, the following proposition fixes the notion of the weak solution of the electrode model.

Proposition 1 *Let $\sigma \in \mathcal{A}(\Omega)$. The problem (2.1), (2.4)-(2.6) has a unique weak solution $(u,U) \in \dot{\mathbb{H}}$ in the following sense. There is a unique $(u,U) \in \dot{\mathbb{H}}$ satisfying the equation*

$$\mathcal{B}_{\sigma,z}((u,U),(v,V)) = \sum_{\ell=1}^{L} I_\ell V_\ell \tag{2.11}$$

for all $(v,V) \in \dot{\mathbb{H}}$, where the quadratic form $\mathcal{B}_{\sigma,z}$ is given as

$$\mathcal{B}_{\sigma,z}((u,U),(v,V)) = \int_\Omega \sigma \nabla u \cdot \nabla v \, dx + \sum_{\ell=1}^{L} \frac{1}{z_\ell} \int_{e_\ell} (u - U_\ell)(v - V_\ell)\, dS. \tag{2.12}$$

Furthermore, the quadratic form is coercive in $\dot{\mathbb{H}}$, i.e., we have the inequalities

$$\alpha_0 ||(u,U)||^2_{\dot{\mathbb{H}}} \le \mathcal{B}_{\sigma,z}((u,U),(u,U)) \le \alpha_1 ||(u,U)||^2_{\dot{\mathbb{H}}} \tag{2.13}$$

for some constants $0 < \alpha_0 \le \alpha_1 < \infty$.

2.1.1 Numerical Implementation of the Forward Problem

We apply the *finite element method* (FEM) [3] for the forward problem.

In order to simplify the numerics, Ω is approximated with a polygonal domain $\widehat{\Omega}$, which is partitioned by generating triangulation $\mathcal{T}_h = \{T_1, \ldots, T_M\}$ such that $T_i \cap T_j = \emptyset$ for $i \neq j$ and $\overline{\widehat{\Omega}} = \bigcup_{m=1}^{M} \{\overline{T}_m\}$. The subindex h indicates the mesh size. Additionally, we suppose that $\sigma \in H_h \subset \mathcal{A}(\widehat{\Omega})$, where

$$H_h := \text{span}\{\chi_{T_m} \,|\, 1 \le m \le M\}, \tag{2.14}$$

i.e. the basis functions of the discrete subspace H_h coincide with the characteristic functions of the triangles. The triangles of the partition $\mathcal{T}_h$ are called pixels and H_h-functions pixelwise constant functions. We write $\sigma = \sum_{i=1}^{M} \sigma_i \eta_i$ and identify σ with a vector in $\mathbb{R}^M$.

The discrete potential field is represented by using a piecewise linear *nodal basis* $\{\varphi_1, \ldots, \varphi_{N_n}\}$ of the triangulation $\mathcal{T}_h$, i.e. a set of piecewise linear functions which take on a nonzero value at precisely one of the nodes of $\mathcal{T}_h$. We define

$$S_h = \text{span}\{\varphi_i \,|\, 1 \le i \le N_n\} \tag{2.15}$$

The finite element approximation $u_h \in S_h$ satisfying the equations (2.1), (2.4)-(2.6) in the sense of proposition 1 is written as

$$u^h = \sum_{i=1}^{N_n} \alpha_i \varphi_i \tag{2.16}$$

In order that the condition (2.3) is satisfied, the voltage vector is represented as

$$U^h = \sum_{j=1}^{L-1} \beta_j n_j, \tag{2.17}$$

where the vectors $n_j \in \mathbb{R}^L$ are chosen as

$$\begin{aligned} n_1 &= \begin{pmatrix}1 & -1 & 0 & \dots & 0\end{pmatrix}^T, \\ n_2 &= \begin{pmatrix}1 & 0 & -1 & \dots & 0\end{pmatrix}^T, \\ n_{L-1} &= \begin{pmatrix}1 & 0 & \dots & -1\end{pmatrix}^T. \end{aligned} \tag{2.18}$$

By applying the theory of finite elements [3], a substitution of the approximations (2.16) and (2.17) to the weak form (2.11) yields a matrix equation

$$Ax = f, \tag{2.19}$$

where $x = (\alpha, \beta)^T \in N_n + L - 1$ and the data vector f is

$$f = \begin{pmatrix} \mathbf{0} \\ \sum_{\ell=1}^{L} I_\ell (n_j)_\ell \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathcal{C}^T I \end{pmatrix}. \tag{2.20}$$

where $\mathbf{0} = (0, \dots, 0)^T \in \mathbb{R}^{N_n}$ and $\mathcal{C} \in \mathbb{R}^{L \times (L-1)}$ is a sparse matrix given as

$$\mathcal{C} = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ -1 & 0 & \dots & & 0 \\ 0 & -1 & 0 & \dots & \vdots \\ \vdots & & \ddots & & \\ & & & \ddots & \\ 0 & \ddots & & & -1 \end{pmatrix} \tag{2.21}$$

The stiffness matrix $A \in \mathbb{R}^{(N_n+L-1)\times(N_n+L-1)}$ is the sparse block matrix of the form

$$A = \begin{pmatrix} B & C \\ C^T & G \end{pmatrix} \tag{2.22}$$

with

$$B_{i,j} = \int_\Omega \sigma \nabla \varphi_i \cdot \varphi_j \, dx + \sum_{\ell=1}^{L} \frac{1}{z_\ell} \int_{e_\ell} \varphi_i \varphi_j \, dS, \quad 1 \le i, \quad j \le N_n, \tag{2.23}$$

$$C_{i,j} = -\Big(\frac{1}{z_1} \int_{e_1} \varphi_i \, dS - \frac{1}{z_{j+1}} \int_{e_{j+1}} \varphi_i \, dS\Big), \; 1 \le i \le N_n, \;\; 1 \le j \le L-1 \tag{2.24}$$

$$G_{i,j} = \sum_{\ell=1}^{L} \frac{1}{z_\ell} \int_{e_\ell} (n_i)_\ell (n_j)_\ell \, dS \tag{2.25}$$

$$= \begin{cases} \frac{|e_1|}{z_1}, & i \neq j \\ \frac{|e_1|}{z_1} + \frac{e_{j+1}}{z_{j+1}}, & i = j, \end{cases} \tag{2.26}$$

By solving equation (2.19) an approximate solution for the forward problem is obtained. The N_n first coefficients in x give the solution u^h in the nodes and the last $L-1$ coefficients give the referenced voltages $\beta = (\beta_1, \ldots, \beta_{L-1})^T$ on the electrodes. The potentials U_ℓ on the electrodes are calculated with the aid of (2.17) yielding

$$U_h = \mathcal{C}\beta \tag{2.27}$$

2.2 The Inverse Problem

To solve the finite dimensional EIT inverse problem is to estimate the unknown conductivity distribution $\sigma \in \mathcal{A}(\Omega)$ on the basis of the voltage measurements on the boundary.

Since we want to get as much boundary data as possible, instead of injecting just one current pattern we inject a set of linearly independent current patterns $\{I^{(k)}\}_{k=1}^{K}$, $I^{(k)} \subset \mathbb{R}^L$, $K \leq L-1$ where L is the number of electrodes. Due to the condition (2.2) $L-1$ is the maximum number of linearly independent current patterns that can be generated.

The set of measured voltages corresponding to the set $\{I^{(k)}\}_{k=1}^{K}$ is denoted as $\{ V^{(k)} \}_{k=1}^{K}$, $V^{(k)} \subset \mathbb{R}^L$. The true measurements are noisy whereas the mathematical model of the forward problem excludes the noise. The set of electrode voltages corresponding to the current pattern $I^{(k)}$ and the conductivity distribution σ is denoted by $U^{(k)}(\sigma)$.

The solution is found iteratively based on the idea of seeking σ such that the set $\{U^{(k)}(\sigma)\}_{k=1}^{K}$ is in some sense the best possible estimate of the set $\{V^{(k)}\}_{k=1}^{K}$. Each set $\{U^{(k)}(\sigma)\}_{k=1}^{K}$ is computed by solving a column vector form of the finite dimensional forward problem (2.19)

$$A_\sigma X_\sigma = F \tag{2.28}$$

where $X_\sigma = (x_\sigma^{(1)}, \ldots, x_\sigma^{(K)})$ and $F = (f^{(1)}, \ldots, f^{(K)})$.

2.2.1 Current Patterns

The methods of injecting current patterns can be classified into *pair drive* methods and *multiple drive* methods. In pair drive methods, current is applied each time between a pair of electrodes. In multiple drive methods current is injected simultaneously to more than two electrodes.

Pair drive methods are advantageous over multiple drive methods in the sense that they are less sensitive to uncertainty in the values of the contact impedances, since in pair drive methods one does not usually measure voltages with the electrodes injecting currents. Multiple drive methods are better in terms of so called *distinguishability* which is defined as follows.

Two conductivity distributions σ_1 and σ_2 are distinguishable with measurement precision ϵ if there exist a current pattern $||I|| = 1$ such that

$$||U(\sigma_1) - U(\sigma_2)|| > \epsilon \tag{2.29}$$

An *optimal current pattern* to distinguish σ_1 from σ_2 is the current vector I which maximizes the distinguishability, i.e.

$$\max_I \frac{||U(\sigma_1) - U(\sigma_2)||}{||I||} \tag{2.30}$$

It can be shown that the trigonometric current patterns

$$I_\ell^{(k)} = \begin{cases} I_{\max}\cos(k\theta_\ell), & 1 \le \ell \le L,\ 1 \le k \le \frac{L}{2}, \\ I_{\text{max}}\sin((k-L/2)\theta_\ell), & 1 \le \ell \le L,\ \frac{L}{2} < k \le L-1 \end{cases} \tag{2.31}$$

where the constant I_{max} denotes the amplitude of the current, $\theta_\ell = 2\pi\ell/L$ is the angular location of the midpoint of electrode e_ℓ and k is the spatial frequency, are optimal current patterns to distinguish a centered rotation invariant annulus from a homogenous disc. As a general rule, low frequency current patterns of the form (2.31) yield the best sensitivity to the deeper regions of Ω and the high frequency patterns as (2.31) are mostly sensitive to the regions in vicinity of $\partial\Omega$.

2.3 Least-Squares Methods

The most commonly used method for solving the above described inverse problem is the *Least-squares approximation* (LS) where the idea is to minimize the error $||\mathbf{U}(\sigma) - \mathbf{V}||^2$.

2.3.1 Gauss-Newton Reconstruction

In *Gauss-Newton reconstruction* one minimizes the functional

$$\Phi_\alpha(\sigma) = ||\mathbf{U}(\sigma) - \mathbf{V}||_W^2 + \alpha A(\sigma), \tag{2.32}$$

where

$$\begin{aligned} ||\mathbf{U}(\sigma) - \mathbf{V}||_W^2 &= \sum_{k=1}^{K}\sum_{l=1}^{L} w_{k,l}(U_\ell^{(k)}(\sigma) - V_\ell^{(k)})^2, && (2.33) \\ \mathbf{U}(\sigma) &= (U^{(1)}(\sigma), \ldots, U^{(K)}(\sigma)), && (2.34) \\ \mathbf{V} &= (V^{(1)}, \ldots, V^{(K)}). && (2.35) \end{aligned}$$

$W = (w_{k,l})$ is a symmetric positive definite weight matrix, $A(\sigma)$ is a regularizing functional and $\alpha > 0$. The regularization method is known as *generalized Tikhonov regularization*. Usually $\Phi_\alpha(\sigma)$ is minimized by employing some iterative gradient-based optimization algorithms. The Gauss-Newton iteration is

$$\begin{aligned} \sigma^{(i+1)} &= \sigma^{(i)} - \lambda_s^{(i)}(H_\alpha^{(i)})^{-1}g^{(i)}, && (2.36) \\ H_\alpha^{(i)} &= (J^{(i)})^T W J^{(i)} + \frac{1}{2}\alpha D^2 A(\sigma^{(i)}) && (2.37) \\ g^{(i)} &= (J^{(i)})^T W(\mathbf{U}(\sigma^{(i)}) - \mathbf{V}) + \frac{1}{2}\alpha DA(\sigma^{(i)}). && (2.38) \end{aligned}$$

where $J^{(i)}$ is a differential and $H_\alpha^{(i)} \in \mathbb{R}^{M\times M}$ is a regularized Hessian matrix of the map $\sigma \to U(\sigma)$ evaluated at $\sigma^{(i)}$, $g^{(i)} = \nabla\Phi_\alpha(\sigma^{(i)})$, $DA(\sigma^{(i)})$ is a differential of the map $\sigma \to A(\sigma)$, $(D^2A(\sigma^{(i)}))_{k,l} = (\partial^2 A/\partial\sigma_k\partial\sigma_l)|_{\sigma^{(i)}}$ and $\lambda_s^{(i)} > 0$ is a relaxation parameter controlling the step size.

2.3.2 NOSER Algorithm

Another approach to least-squares approximation is the *Newton's one-step error reconstructor* (NOSER) which performs one Gauss-Newton iteration step starting from an optimally chosen background to minimize $||\mathbf{U}(\sigma) - \mathbf{V}||^2$. The reconstruction is computed as

$$\sigma = \sigma^{(0)} + (H + \alpha \operatorname{diag}(H))^{-1} g, \tag{2.39}$$

where $\alpha > 0$ is a regularization parameter and

$$H = (J^{(0)})^T J^{(0)}, \quad g = (J^{(0)})^T (V - U(\sigma^{(0)})), \tag{2.40}$$

Since the Jacobian is an ill-conditioned matrix, computing its inverse requires for regularization which, here, means adding a diagonal weight $\alpha \operatorname{diag}(H)$.

2.4 Statistical Model

Since the voltage measurements are assumed to be noisy, it seems reasonable to take a statistical approach to the inverse problem so as to get solutions as accurate as possible. Surely, no model can completely represent every detail of reality, but the aim is to abstract the key features of the problem into a workable mathematical form. The procedure of drawing conclusions concerning unobserved quantities on the basis of a probabilistic model is known as *statistical inference.*

2.4.1 Bayesian Methodology

We formulate the inverse problem in terms of Bayesian methodology. The idea of Bayesian statistics is to embed all sorts of problem related information and uncertainty, such as prior knowledge and physical randomness, in a joint probability distribution by treating all quantities involved in the model as random variables. The goal is to derive all inferential statements based purely on an appropriate conditional distribution of unknown variables.

Below, random variables are denoted by capital letters and their values are denoted by lower case letters.

Let $(S, \mathcal{B}, P)$ denote a probability space, $\mathcal{B}$ being the σ-algebra of measurable subsets of S and $P : \mathcal{B} \to [0, 1]$ a probability measure. Let

$$(X, N) \, : \, S \to \mathbb{R}^{n+k}, \quad V \, : \, S \to \mathbb{R}^m \tag{2.41}$$

Vector (X, N) represents all those quantities that cannot be directly measured while V represents a vector of observable quantities. $X \in \mathbb{R}^n$ represents those variables that we are primarily interested in while $N \in \mathbb{R}^k$ contains unknown but uninteresting variables.

In terms of Bayesian statistics (X, N) is a random vector following a *prior density*

$$\pi_{\text{pr}}(x, n),$$

which is typically regarded as known to the researcher independently of the data under analysis and contains the prior knowledge of the value of (X, N). The probability of observing V corresponding to a given realization of (X, N) follows a so called *likelihood density*

$$\pi(v \,|\, x, n). \tag{2.42}$$

More generally, we call a *likelihood function* any function that is proportional to the likelihood density. The realized value of (X, N) based the observations V is summarized in the *posterior density* , which is typically a conditional distribution obtained through an application of the well-known Bayes theorem:

$$\pi_{\text{post}}(x, n \,|\, v) = \frac{\pi(v, x, n)}{\pi(v)} = \frac{\pi(v \,|\, x, n)\pi_{\text{pr}}(x, n)}{p(y)} \propto \pi(v \,|\, x, n)\pi_{\text{pr}}(x, n). \tag{2.43}$$

The process of a typical Bayesian analysis can be described as consisting of three main steps:

1. Setting up a full probability model, the *joint distribution* $\pi(v, x, n)$ capturing the relationship among all variables in consideration. A standard procedure is to formulate the scientific question of interest through the use of a probabilistic model, based on which one can write down the likelihood density. The joint probability density can then be represented as

$$\pi(v, x, n) = \pi(v \,|\, x, n)\pi_{\text{pr}}(x, n) \tag{2.44}$$

2. Summarizing the findings for particular quantities of interest by appropriate posterior distributions. Usually, this means employing the formula (2.43). Moreover, since the realization of N is uninteresting, one often integrates n out from the density $\pi_{\text{post}}(x, n \,|\, v)$.

3. Evaluating the appropriateness of the model and suggesting improvements.

2.4.2 Setting Up the Probability Model

The observation is assumed to follow a deterministic law; that is, we assume that X and N determine the observable V uniquely,

$$V = F(X, N). \tag{2.45}$$

Here, X and N are assumed to take values $X = x \in \mathbb{R}^n$ and $N = n \in \mathbb{R}^k$ and $F : \mathbb{R}^{n+k} \to \mathbb{R}^m$ is assumed to be a known deterministic function. The probability distribution of the random variable V is formally given by

$$\pi(v \,|\, x, n) = \delta(v - F(x, n)) \tag{2.46}$$

where δ is the Dirac delta in $\mathbb{R}^m$. Let $\pi_{\text{pr}}(x, n)$ denote the prior probability density of the unknown vector (X, N). The joint probability density of (X, N) and V can be written as

$$\pi(x, n, v) = \pi(v|x, n)\pi_{\text{pr}}(x, n) = \delta(v - F(x, n))\pi_{\text{pr}}(x, n). \tag{2.47}$$

Since we have arranged the variables so that N represents all the variables whose values are not of primary interest, we integrate the variable n out and define the joint probability density of the variables X and V as a marginal distribution

$$\pi(x,v) = \int_{R^k} \pi(x,n,v)dn = \int_{\mathbb{R}^k} \delta(v - F(x,n))\pi_{\mathrm{pr}}(x,n)\, dn \tag{2.48}$$

For simplicity, we consider a simple model where the variables X and N are independent. This can be written as

$$\pi_{\mathrm{pr}}(x,n) = \pi_{\mathrm{pr}}(x)\pi_{\mathrm{noise}}(n) \tag{2.49}$$

where the variable N is identified as noise and is assumed to be additive quantity, i.e. the model equation (2.45) is of the form

$$V = f(X) + N \tag{2.50}$$

and the integral (2.48) is written as

$$\pi(x,v) = \int_R^k \delta(v - f(x) - n)\pi_{\mathrm{pr}}(x)\pi_{\mathrm{noise}}(n)\, dn = \pi_{\mathrm{pr}}(x)\pi_{\mathrm{noise}}(v - f(x)). \tag{2.51}$$

The posterior distribution of X is given by the Bayes formula

$$\pi_{\mathrm{post}}(x) = \pi(x|v) = \frac{\pi(x,v)}{\int \pi(x,v)dx} \tag{2.52}$$

Writing $\pi(v|x) = \pi_{\mathrm{noise}}(v - f(x))$ we have $\pi_{\mathrm{post}}(x) = \pi(x|v) \propto \pi_{\mathrm{pr}}(x)\pi(v|x)$, where $\pi(v|x)$ is the likelihood density.

2.4.3 Estimates from the Posterior Distribution

In the formal Bayesian procedure, solution of the inverse problem is the posterior distribution of X. However, to be able to draw representative images of the conductivity distribution within Ω one has somehow to estimate the realization of X. Therefore, the word solution is, as well, used to refer to some estimate of some property of the posterior distribution.

A commonly used estimate is the (possibly non-unique) *maximum a posteriori* (MAP) estimate

$$x_{MAP} = \arg\max_x \pi(x|v) \tag{2.53}$$

Computation of the MAP estimate leads to an optimization problem.

The *maximum likelihood* estimate amounts to determination of the maximum of the likelihood density; that is

$$X_{ML} = \arg\max_x \pi(v|x) \tag{2.54}$$

In highly non-linear and ill-conditioned problems ML estimates are often useless.

It is also common to estimate the conditional expectation

$$x|_v = \int_{\mathbb{R}^n} x\pi(x|v)dx. \tag{2.55}$$

2.4.4 Implementation of the Bayesian Model

In terms of the above described probability model, the sought posterior distribution of the EIT inverse problem is $\pi_{\text{post}}(\sigma) = \pi(\sigma|\mathbf{V})$, $\sigma \in H_h$ being the discrete approximation of the unknown conductivity distribution and $\mathbf{V}$ containing the voltage measurements as in (2.35).

We assume the random noise N of the measurements to be additive and independent of σ. Thus, similarly as in (2.50) we have

$$V = U(\sigma) + N \tag{2.56}$$

The contact impedances are assumed to be known. For convenience, we assume that the basis functions $\eta_k \in H_h$ are positive.

In this thesis, we employ prior distributions of the form

$$\pi_{\text{pr}}(\sigma) = \pi_+(\sigma)\tilde{\pi}_{\text{pr}}(\sigma), \tag{2.57}$$

where π_+ is the positivity prior of the form

$$\pi_+(\sigma) = \begin{cases} 1, & 0 < \sigma \leq \sigma_j \leq \sigma_{max} < \infty \\ 0, & \text{otherwise} \end{cases} \tag{2.58}$$

and $\tilde{\pi}_{\text{pr}}$ is a subspace constraint

$$\tilde{\pi}_{\text{pr}}(x) \propto \chi_{S_{\text{pr}}}(x), \tag{2.59}$$

where $\chi_{S_{\text{pr}}}$ is the characteristic function of S_{pr} denoting a subset of H_h chosen on the ground of the prior information. Often, it is not enough just to restrict the problem to some subspace, but more sophisticated prior distributions have to be applied (e.g. regularizing priors favoring anomalies of certain size).

In the computations, the noise vector N is a zero mean Gaussian random vector with positive definite covariance matrix C. With this choice, the posterior distribution given by formulae (2.51) and (2.52) is written as

$$\pi(\sigma \,|\, \mathbf{V}) \propto \pi_+(\sigma)\chi_{S_{\text{pr}}}(\sigma) \exp(-\frac{1}{2}(\mathbf{U}(\sigma) - \mathbf{V})^T C^{-1}(\mathbf{U}(\sigma) - \mathbf{V})). \tag{2.60}$$

The least squares solution discussed in section 2.3 corresponds to the MAP solution of (2.60) when $W = \frac{1}{2}C^{-1}$.

Chapter 3

MCMC Integration

Examining the posterior distribution numerically is usually quite problematic, since the dimension of the sample space is often large.

For instance, estimation of the conditional expectation requires for evaluation of the integral (2.55). Applying a standard numerical n-dimensional quadrature is often impossible, since the computational work load increases rapidly as a function of n.

In this work, the conditional expectation is estimated in a statistical sense through MCMC sampling methods, a class of relatively simple algorithms that by generating sample ensembles enable the exploration of an arbitrary probability distribution. MCMC methods offer a way to solve both integration and optimization problems. The use of MCMC is profitable in connection with high dimensional problems, since instead of the dimension the convergence rate depends on the size of the generated sample ensemble and the exactitude of a priori information.

In this section, we discuss the general idea of the MCMC methods and introduce some sampling strategies that appear frequently in the literature. We lay the emphasis on MCMC integration. The main references are [1], [7] and [2].

3.1 Motivation of Monte Carlo Techniques

The fundamental idea behind the Monte Carlo methodology is that the integral

$$I = \int_D f(x)\, dx, \tag{3.1}$$

over a compact $D \subset \mathbb{R}^n$ can be estimated in a statistical sense by drawing independent and uniformly ($\pi \sim \chi_D$) distributed random samples $x^{(1)}, \ldots, x^{(m)}$ from D. The law of large numbers states that the average of large number of independent random variables with common mean and finite variances tends to stabilize at their common mean. Therefore, we can approximate

$$I \approx \hat{I}_m = \frac{1}{m}\{f(x^{(1)}) + \cdots + f(x^{(m)})\}. \tag{3.2}$$

because $\lim_{m\to\infty} \hat{I}_m = I$, with probability 1. The convergence rate is assessed by the central limit theorem:

$$\sqrt{m}(\hat{I}_m - I) \to N(0, \gamma^2), \text{ in distribution,} \tag{3.3}$$

where

$$\gamma^2 = \operatorname{var}\{f(x)\} = \int_D (f - \overline{f})^2 \, dx \tag{3.4}$$

Thus, the error of the approximation (3.2) is $O(1/\sqrt{m})$, regardless of the dimensionality of x.

Deterministic methods of evaluating (3.1), such as the Riemann approximation and Simpson's rule, do not scale well as the dimension of D increases. For example, in n-dimensional space with $D = [0,1]^n$, one will have to evaluate $O(m^{10})$ grid points in order to achieve an accuracy of $O(m^{-1})$. Hence, due to the property (3.3) the Monte Carlo approach is especially advantageous when the dimension of D is large.

3.1.1 Example: Importance Sampling

In applications, achieving a feasible convergence rate can be problematic. The variance γ^2 can be formidably small indicating that only a small subset of the sample space D affects notably the value of (3.1), due to which the convergence rate of the estimate (3.2) would be slow. For similar reasons, an exceedingly large value of γ^2 causes slow convergence. It is also possible that one may not be able to produce uniform random samples in an arbitrary region D.

One way to overcome these difficulties is *importance sampling* in which the independent random samples $\{x^{(1)}, \dots, x^{(m)}\}$ are generated from a nonuniform easy-to-sample trial distribution $g(x)$ that puts more probability mass on "important" parts of the state space D and then correcting the bias by incorporating the importance weight $f(x^{(j)})/g(x^{(j)})$. The integral (3.1) is estimated as

$$\hat{I}_m = \frac{1}{m} \sum_{j=1}^{m} \frac{f(x^{(j)})}{g(x^{(j)})} \tag{3.5}$$

which has the variance

$$\gamma_g^2 = \operatorname{var}_g\{f/g\} = \int_D \left(\frac{f}{g} - \overline{\left(\frac{f}{g}\right)}\right)^2 g dx. \tag{3.6}$$

Thus, a good candidate for $g(\cdot)$ is the one that is close to $f(\cdot)$. By properly choosing $g(\cdot)$, one can reduce the variance of the estimate substantially. In the most fortunate case, we are able to choose $\pi(x) \sim g(x)$, but this is virtually never feasible in applications.

Because of the great potential of Monte Carlo methodology, various techniques have been developed by researchers in their respective fields. A fundamental step in all Monte Carlo methods is to generate random samples from a probability distribution function π, often known only up to a normalizing constant. As directly generating independent samples from the target distribution π is usually not feasible, it is often the case that either the distribution used to generate the samples is different from π, or the generated samples have to be dependent. Schemes that make use of samples generated from a a *trial distribution* g, which differs from, but should be similar to, the target distribution π, are the *rejection method*, importance sampling and *sampling-importance-resampling.*

3.1.2 The General Idea of Markov Chain Monte Carlo

The idea behind the MCMC methods is generating random but statistically dependent samples from an arbitrary probability distribution π. The advance of using MCMC is, that even the generated samples are not independent, one does not necessarily have to know much about the structure of π in order to draw a representative sample ensemble.

Below, we introduce some fundamental definitions concerning Markov chains in order to facilitate the closer inspection of the MCMC methods. We restrict ourselves to cases where the state space is $\mathbb{R}^n$.

Definition 2 *Let $\mathcal{B}$ denote the Borel sets over $\mathbb{R}^n$. A mapping $A : \mathbb{R}^n \times \mathcal{B} \to [0,1]$ is called a* transition function *(also* transition kernel*), if*

1. *For each $B \in \mathcal{B}$, the mapping $A : \mathbb{R}^n \to [0,1]$, $x \to A(x,B)$ is a measurable function;*
2. *For each $x \in \mathbb{R}^n$, the mapping $\mathcal{B} \to [0,1]$, $B \to A(x,B)$ is a probability measure.*

Definition 3 *A* time-homogenous *Markov chain with the transition function A is a* stochastic process $\{X^{(j)}\}_{j=1}^{\infty}$ *if the transition function satisfies*

$$\begin{aligned} P(X^{(j+1)} \in B \,|\, X^{(1)}, \ldots, X^{(j)}) &= P(X^{(j+1)} \in B \,|\, X^{(j)}), && (3.7)\\ A(x,B) &= P(X^{(j+1)} \in B \,|\, X^{(j)} = x) \quad \forall\, j. && (3.8) \end{aligned}$$

More generally, we define

$$\begin{aligned} A^{(k)}(x,B) &= P(X^{(j+k)} \in B \,|\, X^{(j)} = x)\\ &= \int_{\mathbb{R}^n} A(x,B)A^{(k-1)}(x,dy), \end{aligned}$$

where $A^{(1)}(x,B) = A(x,B)$.

Definition 4 *If π is a probability measure of $X^{(j)}$ and f is a scalar or vector-valued measurable function on $\mathbb{R}^n$, $f \in L^1(\pi(dx))$, then the distribution of $X^{(j+1)}$, is defined by*

$$(\pi A)(B) = \int_{\mathbb{R}^n} A(x,B)\pi(dx). \tag{3.9}$$

Af and πf are defined as

$$\begin{aligned} (Af)(x) &= \int_{\mathbb{R}^n} f(y)A(x,dy) && (3.10)\\ \pi f &= \int_{\mathbb{R}^n} f(y)\pi(dy) && (3.11) \end{aligned}$$

Definition 5 *The measure π is an* invariant measure *of $A(x,B)$ if $\pi A = \pi$, i.e., the distribution of the random variable after one transition step is the same as before the step.*

Definition 6 *Given a probability measure π. The transition kernel A is called π-*irreducibile *with respect to π if for each $x \in \mathbb{R}^n$ and $B \in \mathcal{B}$ with $\pi(B) > 0$ there exists an integer k such that $A^{(k)}(x,B) > 0$. Thus, regardless of the starting point, the Markov chain enters with a positive probability any set of positive measure.*

Definition 7 *A π-irreducible transition function A is* periodic *if for some integer $m \geq 2$ there is a set of disjoint non-empty sets $\{E_1, \ldots, E_m\} \subset \mathbb{R}^n$ such that for all $j = 1, \ldots, m$ and all $x \in E_j$, $A(x, E_{j+1\,(mod\, m)}) = 1$. Otherwise, A is* aperiodic.

Definition 8 *A π-irreducible chain $\{X^{(j)}\}_{j=1}^{\infty}$ with invariant distribution π is recurrent if, for each B with $\pi(B) > 0$,*

$$\begin{aligned} P\{X^{(n)} \in B\, i.o. \,|\, X^{(0)} = x\} &> 0 \quad for\ all\, x, & (3.12) \\ P\{X^{(n)} \in B\, i.o. \,|\, X^{(0)} = x\} &= 1 \quad for\ \pi\text{-}almost\ all\, x. & (3.13) \end{aligned}$$

The notation $\{X^{(n)} \in B\, i.o. \,|\, X^{(0)} = x\}$ meaning that the Markov chain starting from x visits B infinitely often i.e. $\sum_{X^{(n)} \in B} 1 = \infty$. The chain is Harris recurrent *if $P\{X^{(n)} \in B\, i.o. \,|\, X^{(0)} = x\} = 1$ for all x.*

Definition 9 *A π-irreducible recurrent Markov chain is* positive recurrent *if it has an invariant distribution, total mass of which is finite; otherwise it is* null recurrent.

Definition 10 *A Markov chain is called* ergodic *if it is positive Harris recurrent and aperiodic. If S_B^x denotes the hitting time for set B for a chain starting from x, then an ergodic chain with invariant distribution π is ergodic of degree 2 if*

$$\int_B \pi(dx) E[(S_B^x)^2] < \infty \tag{3.14}$$

In traditional Markov chain analysis, one is often given the transition function and is interested in knowing what the stationary distribution is, whereas in Markov chain Monte Carlo simulations, one knows the equilibrium distribution and is interested in prescribing an efficient transition rule so as to reach the equiblirium. The Monte Carlo approximation

$$\overline{f}_n = \frac{1}{n} \sum_{i=1}^{n} f(X_i) \approx \int_{\mathbb{R}^n} f(x)\pi(dx) = \pi f \tag{3.15}$$

converges, since the law of large numbers and the central limit theorem apply also to the Markov chains [7].

Theorem 1 (a law of large numbers) *Suppose $\{X^{(j)}\}_{j=1}^{\infty}$ is ergodic with equilibrium distribution π and suppose f is real and $\pi|f| < \infty$. Then for any initial distribution, $\overline{f}_n \to \pi f$ almost surely.*

Theorem 2 (the central limit theorem) *Suppose $\{X^{(j)}\}_{j=1}^{\infty}$ is ergodic of degree 2 with equilibrium distribution π and suppose f is real-valued and $\pi(f^2) < \infty$. Then there exists a real number $\gamma(f)$ such that the distribution of*

$$\sqrt{n}(\overline{f}_n - \pi f) \to N(0, \gamma(f)^2) \tag{3.16}$$

weakly (i.e., in distribution) for any initial distribution on $x^{(0)}$.

3.2 Metropolis-Hastings Algorithm

Metropolis-Hastings algorithm prescribes the transition rule based on a "trial - and - error" strategy. It uses a symmetric proposal function $T(x, y)$ to suggest a possible move from x to y and then via an acceptance-rejection rule ensures that the target distribution π is the equilibrium distribution of this chain.

Algorithm 3.2.1 (Metropolis-Hastings)

- *Given the current state $x^{(t)}$ and a proposal function $T(x, y)$ that satisfies $T(x, y) > 0$ if and only if $T(y, x) > 0$.*
- *Draw y from the proposal distribution $T(x^{(t)}, y)$.*
- *Draw $U \sim$ Uniform$[0, 1]$ and update*

$$x^{(t+1)} = \begin{cases} y, & \textit{if } U \leq r(x^{(t)}, y) \\ x^{(t)} & \textit{otherwise} \end{cases} \tag{3.17}$$

where

$$r(x, y) = \min\Big\{1, \frac{\pi(y)T(y, x)}{\pi(x)T(x, y)}\Big\}. \tag{3.18}$$

The algorithm is a generalization of the Metropolis algorithm, cornerstone of all MCMC techniques, which additionally sets a symmetry requirement $T(x, y) = T(y, x)$. Apparently, choice of the proposal function has a great effect on the convergence rate, which is why the Metropolis-Hastings algorithm is useful in many connections: it does not set serious restrictions on the proposal probability.

3.2.1 The Detailed Balance

To show that the Metropolis-Hastings algorithm prescribes a valid transition rule $A(x, y)$ with invariant distribution $\pi(x)$ we have to show that

$$\int \pi(x)A(x, y)\, dx = \pi(y). \tag{3.19}$$

$A(x, y)$ can be written down explicitly: For any $x \neq y$, the probability that we actually make the move from x to y is equal to $T(x, y)$ multiplied by the acceptance probability, i.e.

$$A(x, y) = T(x, y) \min\Big\{1, \frac{\pi(y)T(y, x)}{\pi(x)T(x, y)}\Big\}, \tag{3.20}$$

for $x \neq y$. Hence,

$$\begin{aligned} \pi(x)A(x,y) &= \pi(x)T(x,y)\min\left\{1, \frac{\pi(y)T(y,x)}{\pi(x)T(x,y)}\right\} \\ &= \min\{\pi(x)T(x,y), \pi(y)T(x,y)\} = \pi(y)A(y,x), \end{aligned} \tag{3.21}$$

which is the *detailed balance* condition implying that (3.19) holds, since

$$\int \pi(x)A(x,y)\,dx = \int \pi(y)A(y,x)\,dx = \pi(y)\int A(y,x)\,dx = \pi(y) \tag{3.22}$$

by symmetry. Thus, the samples $x^{(1)}, x^{(2)}, \ldots$ produced by the chain can be regarded as approximately following the target distribution π.

3.3 Algorithms Based on the Metropolis-Hastings Rule

We are especially interested in employing the Metropolis-Hastings transition rule for the EIT inverse problem as it allows adapting the proposal distribution to the structure of the posterior distribution (2.60). This section introduces some widely used algorithms based on this rule.

3.3.1 Metropolized Independence Sampler (MIS)

One of the most simple proposal transition functions is an independent trial density $T(x,y) = g(y)$, which generates the proposed move y independently from the from the previous state $x^{(t)}$. This method is an alternative to the importance sampling.

Algorithm 3.3.1 (MIS)

- *Given the current state* $x^{(t)}$.
- *Draw* $y \sim g(y)$.
- *Simulate* $U \sim$ *Uniform*$[0,1]$ *and let*

$$x^{(t+1)} = \begin{cases} y, & \textit{if } U \leq \min\left\{1, \frac{w(y)}{w(x^{(t)})}\right\} \\ x^{(t)}, & \textit{otherwise,} \end{cases} \tag{3.23}$$

  where $w(x) = \pi(x)/g(x)$ *is the usual importance sampling weight*

The efficiency of MIS depends on how close the trial density is to the target distribution $\pi(y)$.

Being a primitive sampling technique MIS can be applied in connection with more sophisticated algorithms. For instance, it is possible to insert a couple of MIS steps into Gibbs sampler iteration described in section 3.5 when correctly sampling from a conditional distribution is difficult. With low variances the conditional distribution differs virtually from zero only in a very close neighborhood of the point where it attains the maximum value. Therefore, drawing random numbers from the conditional distribution by employing regular grids can be very inefficient in terms of computing time and requires for treating numbers very unequal in magnitudes. In many Bayesian computations sampling from the conditional distribution can be performed reasonably well using MIS and a Gaussian approximation of the posterior distribution as $g(y)$. This might be worth trying also in connection with the EIT problem. Moreover, dealing with numbers of different magnitudes is not a problem when using MIS since (3.24) can be written as

$$x^{(t+1)} = \begin{cases} y, & \text{if } \log(U) \leq \min\left\{0, \log(w(y)) - \log(w(x^{(t)}))\right\} \\ x^{(t)}, & \text{otherwise,} \end{cases} \tag{3.24}$$

3.3.2 Random-walk Metropolis

The *random-walk Metropolis* algorithm is based on perturbing the current configuration $x^{(t)}$ by adding a random "error" so that the proposed candidate position is $y = x^{(t)} + \epsilon$ where $\epsilon \sim g_\gamma$ is identically distributed for all t. The parameter γ is the "range" of the exploration controlled by the user.

When one does not have much information about the structure of the target distribution, g_γ is often chosen to be a spherically symmetric distribution. Typically, g_γ is the spherical Gaussian distribution $N(0, \gamma^2 I)$. The algorithm is,

Algorithm 3.3.2 (Random-walk Metropolis)

- *Given the current state* $x^{(t)}$
- *Draw* $\epsilon \sim g_\gamma$ *and set* $y = x^{(t)} + \epsilon$, *where* $g_\gamma \sim N(0, \gamma^2 I)$. *The variances* γ *is chosen by the user.*
- *Simulate* $U \sim$ *Uniform*$[0, 1]$ *and update*

$$x^{(t+1)} = \begin{cases} y, & \textit{if } U \leq \frac{\pi(y)}{\pi(x^{(t)})} \\ x^{(t)} & \textit{otherwise} \end{cases} \tag{3.25}$$

It has been suggested that γ should be chosen so that a 25% to 35% acceptance rate is maintained. Despite of the fact that the Metropolis-Hastings algorithm (3.2.1) allows one to use asymmetric proposal functions, a simple random-walk proposal is still most frequently seen in practice, since finding a good proposal transition kernel is rather difficult. However, in order to achieve an adequate acceptance rate, one is often bound to use very small step-size in the proposal transition, which will easily result in exceedingly slow movement of the corresponding Markov chain. In such case, convergence rate of the algorithm would arguably be very slow.

3.3.3 Multiple-Try Metropolis (MTM)

Multiple-Try Metropolis (MTM) is a generalization of the Metropolis-Hastings' transition rule allowing the sampler to take larger jumps without lowering the acceptance rate. The idea is to generate weighted samples by defining a weight function

$$w(x, y) = \pi(x)T(x, y)\lambda(x, y), \tag{3.26}$$

where $T(x, y)$ is an arbitrary proposal transition function and $\lambda(x, y)$ is a non-negative symmetric function that can be chosen by the user. A modest requirement is that both $T(y, x) > 0$ and $\lambda(x, y) > 0$ whenever $T(x, y) > 0$.

Algorithm 3.3.3 (MTM)

- *Given the current state $x^{(t)}$*
- *Draw k independent trial proposals $y_1, \ldots, y_k$, from $T(x^{(t)}, \cdot)$. Compute*

  $$w(y_j, x^{(t)}) = \pi(x^{(t)})T(x^{(t)}, y_j)\lambda(x^{(t)}, y_j) \tag{3.27}$$

- *Select y among the trial set $\{y_1, \ldots, y_k\}$ with probability proportional to $w(y_j, x^{(t)})$, $j = 1, \ldots, k$. Then, produce a "reference set" by drawing $x_1^*, \ldots, x_{k-1}^*$ from the distribution $T(y, \cdot)$. Let $x_k^* = x^{(t)}$.*
- *Accept y with probability*

  $$r_g = \min\left\{1, \frac{w(y_1, x^{(t)}) + \cdots + w(y_k, x^{(t)})}{w(x_1^*, y) + \cdots + w(x_k^*, y)}\right\} \tag{3.28}$$

  and reject it with probability $1 - r_g$. The quantity r_g is called the generalized M-H ratio.

A straightforward (but boring) calculation shows that the method fulfils the detailed balance condition. λ_j is often chosen to be a constant function, but it is also usual to give larger weights to larger j's in order to increase the step-size. For symmetric $T(x, y)$, one can choose $\lambda(x, y) = T^{-1}(x, y)$. Then, $w(x, y) = \pi(x)$. The resulting algorithm is known as *orientational bias Monte Carlo* (OBMC).

Algorithm 3.3.4 (OBMC)

- *Given the current state $x^{(t)}$*
- *Draw k trials $y_1, \ldots, y_k$ from a symmetric proposal function $T(x^{(t)}, \cdot)$.*
- *Select $y = y_l$ among the y's with probability proportional to $\pi(y_j)$, $j = 1, \ldots, k$; then, draw the reference points $x_1^*, \ldots, x_{k-1}^*$ from the distribution $T(y, \cdot)$. Let $x_k^* = x^{(t)}$.*
- *Accept y_l with probability*

  $$\min\left\{1, \frac{\pi(y_1) + \cdots + \pi(y_k)}{\pi(x_1^*) + \cdot + \pi(x_k^*)}\right\} \tag{3.29}$$

  and reject with the remaining probability.

Combining MTM algorithm with Metropolized independence sampler (3.3.1) results in *multiple-trial Metropolized independence sampler* (MTMIS) algorithm. Since the trial samples are generated independently one does not need to generate another "reference set".

Algorithm 3.3.5 (MTMIS)

- *Given the current state* $x^{(t)}$
- *Generate a trial set of i.i.d samples by drawing* $y_j \sim g$, $j = 1, \ldots, k$, *independently, where* g *is a trial distribution chosen by the user. Compute* $w(y_j) = \pi(y_j)/g(y_j)$ *and* $W = \sum_{j=1}^{k} w(y_j)$.
- *Draw* y *from the trial set* $\{y_1, \ldots, y_k\}$ *with probability proportional to* $w(y)$.
- *Let* $x^{(t+1)} = y$ *with probability*

$$\min\left\{1, \frac{W}{W - w(y) + w(x)}\right\} \tag{3.30}$$

and let $x^{(t+1)} = x$ *with the remaining probability.*

3.3.4 Correlated Multipoint Proposals

A more general scheme is provided by the multipoint method. For simplicity We use the notation

$$\begin{aligned} y_{[1:j]} &= (y_1, \ldots, y_j) \\ y_{[j:1]} &= (y_j, \ldots, y_1). \end{aligned}$$

in the following. Multipoint method chooses the proposed move from multiple correlated proposals at each iteration step. Let $y_1 \sim P_1(\cdot|x)$ and let

$$\begin{aligned} y_j &\sim P_j(\cdot|x, y_1, \ldots, y_{j-1}), \quad j = 2, \ldots, k \\ P_j(y_{[1:j]}|x) &= P_1(y_1|x) \cdots P_j(y_j|x, y_{[1:j-1]}), \end{aligned}$$

In this case, the weight function is defined as

$$w_j(x, y_{[1:j]}) = \pi(x) P_j(y_{[1:j]}|x) \lambda_j(x, y_{[1:j]}), \tag{3.31}$$

where λ_j is a sequentially symmetric function, i.e.

$$\lambda_j(a, b, \ldots, z) = \lambda_j(z, \ldots, b, a)$$

The algorithm is as follows:

Algorithm 3.3.6 (Multipoint method)

- *Given the current state* $x^{(t)}$.
- *Sample* y *from the trial set* $\{y_1, \ldots, y_k\}$ *with probability of* y_l *proportional to* $w(y_{[l:1]}, x^{(t)})$. *Suppose* y_j *is chosen.*

- *Create a reference set by letting $x_l^* = y_{j-l}$ for $l = 1, \ldots, j-1$, $x_j^* = x^{(t)}$, and drawing*

$$x_m^* \sim P_m(\cdot|y, x^*_{[1:m-1]}), \tag{3.32}$$

  for $m = j+1, \ldots, k$.

- *Let $x^{(t+1)} = y$ with probability*

$$r_{mp} = \min\Big\{1, \frac{\sum_{l=1}^k w(y_{[l:1]}, x)}{\sum_{l=1}^k w(x^*_{[l:1]}, y)}\Big\}, \tag{3.33}$$

  and let $x^{(t+1)} = x^{(t)}$ with the remaining probability.

A particular case of the multipoint method is the *random grid Monte Carlo* algorithm.

Algorithm 3.3.7 (Random grid method)

- *Given the current state $x^{(t)}$.*
- *Randomly generate a direction $e \in \mathbb{R}^n$ and a grid size $r \in \mathbb{R}$.*
- *Construct the candidate set as*

$$y_l = x + l \cdot r \cdot e, \tag{3.34}$$

  for $l = 1, \ldots, k$.

- *Draw $y = y_j$ from $\{y_1, \ldots, y_k\}$ with probability proportional to $w(y_j) = u_j \pi(y_j)$, where u_j is a constant chosen by the user (e.g. $u_j = \sqrt{j}$).*
- *Construct the reference set by letting $x_l^* = y - l \cdot r \cdot e$ for $l = 1, \ldots, k$. Therefore, $x_l = y_{j-1}$ for $l < j$ and $x_l^* = x^{(t)} - (l-j) \cdot r \cdot e$ for $l \geq j$.*
- *Accept the candidate y with probability*

$$p = \min\Big\{1, \sum_{l=1}^k \pi(y_l) / \sum_{l=1}^k \pi(x_l^*)\Big\} \tag{3.35}$$

  and reject with the remaining probability

Above, we have been able to choose $\lambda_j = u_j/P_j$, which is a sequentially symmetric function, since the trial set $\{y_1, \ldots, y_i\}$ determines y_{i+1} uniquely and the resulting trial proposal P_j is sequentially symmetric, i.e.

$$P_j(y_{[1:j]}|x) = P_j(x|y_{[j:1]}). \tag{3.36}$$

3.4 Dynamic Weighting

The use of *dynamic weighted* samples $(x^{(t)}, w^{(t)}) \in (\mathbb{R}^n, \mathbb{R}^+)$ for controlling Markov chain simulation was introduced in [13]. The dynamic weighting scheme has proved to be exceptionally useful as a tool for solving high dimensional integration problems. Below, we introduce two algorithms so as to illustrate the idea of employing dynamic weights.

Algorithm 3.4.1 (R-type move)

- *Given the current state* $(x^{(t)}, w^{(t)})$.
- *Draw* y *from* $T(x^{(t)}, y)$ *and compute the Metropolis-Hastings ratio*

$$r(x^{(t)}, y) = \frac{\pi(y)T(y, x^{(t)})}{\pi(x^{(t)})T(x^{(t)}, y)}. \tag{3.37}$$

- *Choose* $\theta = \theta(w^{(t)}, x^{(t)}) > 0$, *and draw* U *from Uniform*$(0, 1)$. *Then let*

$$(x^{(t+1)}, w^{(t+1)}) = \begin{cases} (y, w^{(t)}r(x^{(t)}, y) + \theta), & \textit{if } U \leq \frac{w^{(t)}r(x^{(t)},y)}{w^{(t)}r(x^{(t)},y)+\theta} \\ \left(x^{(t)}, \frac{w^{(t)}(w^{(t)}r(x^{(t)},y)+\theta)}{\theta}\right), & \textit{otherwise} \end{cases} \tag{3.38}$$

Algorithm 3.4.2 (Q-type move)

- *Given the current state* $(x^{(t)}, w^{(t)})$.
- *Propose the next state* y *from the proposal* $T(x^{(t)}, \cdot)$ *and compute the Metropolis ratio*

$$r(x^{t)}, y) = \frac{\pi(y)T(y, x^{(t)})}{\pi(x^{(t)})T(x^{(t)}, y)} \tag{3.39}$$

- *Choose* $\theta = \theta(w^{(t)}, x^{(t)}) > 0$, *and draw* $U \sim$ *Uniform*$(0, 1)$. *Update* $(x^{(t)}, w^{(t)})$ *to* $(x^{(t+1)}, w^{(t+1)})$ *as*

$$(x^{(t+1)}, w^{(t+1)}) = \begin{cases} (y, \max\{\theta, w^{(t)}r(x^{(t)}, y)\}), & \textit{if } U \leq \min\{1, w^{(t)}r(x^{(t)}, y)/\theta\} \\ (x^{(t)}, aw^{(t)}), & \textit{otherwise} \end{cases} \tag{3.40}$$

where $a > 1$ *can either be a constant or a random variable independent of all other variables.*

Instead of the Monte Carlo approximation (3.2) πf is estimated by employing the standard importance sampling estimate

$$\hat{f}_n = \frac{w^{(1)}f(x^{(1)}) + \cdots + w^{(m)}f(x^{(m)})}{w^{(1)} + \cdots + w^{(m)}} \approx \pi f, \tag{3.41}$$

The purpose of bringing importance weights into the dynamic Monte Carlo process is to provide enable large transitions not allowable by the standard Metropolis transition rules. Firstly, $w^{(t)}$ tends to increase in value as far as $x^{(t+1)} = x^{(t)}$, i.e. the proposed y is not accepted. Secondly, the probability of accepting the proposed move increases as the value of $w^{(t)}$ increases. As a result, the algorithm is not as likely stuck into local modes as the standard Metropolis would be.

Although the way of updating the weight variable allows an adjustment of the bias induced by the non-Metropolis moves, neither R-type move nor Q-type move does not have π as the equilibrium distribution. The above scheme is justified by employing the following two definitions.

Definition 11 *Random variable x is called correctly weighted by w with respect to π if $\sum_w wf(x,w) \propto \pi(x)$, where $f(x,w)$ is the joint distribution of (x,w).*

Definition 12 (IWIW) *A transition rule is said to be* invariant with respect to importance weighting *(IWIW) if it maintains the correctly weightedness of (x,w).*

R-type move satisfies the IWIW property, whereas Q-type move does not. In both cases, it is still possible that the weight variable does not have a stable distribution or it may have an infinite expectation, which is the case with R-type move with $\theta = 1$. By a general weak law of large numbers [14] the approximation (3.41) converges even if the expectation would be infinite, but the convergence rate in such a case would apparently be too slow for implementing the algorithm in applications.

Generally, the theory of dynamic weighting is still rather subtle, although it has been applied successfully in many difficult problems.

3.5 The Gibbs Sampler

The above described algorithms are based on "trial-and-error" strategy. In contrast, the *Gibbs sampler* is a conditional sampling technique no rejection being incurred at any of its sampling steps. The Markov transition rules of the algorithm are built upon conditional distributions derived from the target distribution.

Suppose that the random variable can be decomposed into d components, i.e. $x = (x_1, \ldots, x_d)$. In Gibbs sampler, one randomly or systematically chooses a coordinate, say x_k, and then updates it with a new sample x'_k drawn from the conditional distribution $\pi(\cdot|x_{[-1]})$, where $x_{[-A]}$ refers to $\{x_j, j \in A^c\}$ for any subset A of the coordinate indices. Algorithmically, we describe two types of Gibbs sampling strategy.

Algorithm 3.5.1 (Random-scan Gibbs sampler) *Let $x^{(t)} = (x_1^{(t)}, \ldots, x_d^{(t)})$ for iteration t. Then, at iteration $t+1$, we conduct the following steps:*

- *Randomly select a coordinate i from $\{1, \ldots, d\}$ according to a given probability vector $(\alpha_1, \ldots, \alpha_d)$ [e.g. (1/d, ..., 1/d)].*
- *Draw $x_i^{(t+1)}$ from the conditional distribution $\pi(\cdot|x_{[-i]}^{(t)})$ and leave the remaining components unchanged; that is, let*

$$x_{[-i]}^{(t+1)} = x_{[-i]}^{(t)} \tag{3.42}$$

Algorithm 3.5.2 (Systematic-scan Gibbs sampler) *Let* $x^{(t)} = (x_1^{(t)}, \ldots, x_d^{(t)})$. *At the* $t+1$ *iteration:*

- *We draw* $x_i^{(t+1)}$ *from the conditional distribution*

$$\pi(x_i \,|\, x_1^{(t+1)}, \ldots, x_{i-1}^{(t+1)}, x_{i+1}^{(t)}, \ldots, x_d^{(t)}) \tag{3.43}$$

for $i = 1, \ldots, d$

It is easy to check that every conditional update step in the Gibbs sampler leaves π invariant. To see this, suppose $x^{(t)} \sim \pi$. Then, $x_{[-i]}^{(t)}$ follows its marginal distribution under π. Thus,

$$\pi(x_i^{(t+1)} \,|\, x_{[-i]}^{(t)}) \times \pi(x_{[-i]}^{(t)}) = \pi(x_{[-i]}^{(t)}, x_i^{(t+1)}), \tag{3.44}$$

which means that after one conditional update, the new configuration still follows distribution π.

More detailed discussion of applying the Gibbs sampler to the EIT problem can be found in [1]. The Gibbs sampler is not implemented in this thesis, which is largely due to the difficulties that occurred in connection with drawing random samples from conditional distributions.

3.6 Surrogate Transitions

It is typical in Monte Carlo simulations that evaluation of $\pi(x)$ involves expensive computation, although it is cheap to obtain a relatively good approximation $\pi^*(x)$. In the EIT inverse problem, evaluation of the posterior density $\pi_{post}(\sigma)$ requires for solving the discretized forward problem, i.e. linear system of equations, the size of which depends on the resolution of the discretion. It is, however, often sufficient to speed up calculations simply by linearizing the mapping $\sigma \rightarrow U(\sigma)$ around the initial guess σ_0 and by approximating

$$\begin{aligned} U(\sigma) &\approx U^*(\sigma) = U(\sigma) + DU(\sigma_0)(\sigma - \sigma_0), & (3.45) \\ \pi_{post}(\sigma) &\approx \pi^*_{post}(\sigma) & (3.46) \end{aligned}$$

where $DU(\sigma)$ is the Jacobian matrix and $\pi^*_{post}(\sigma)$ is the value of the posterior density corresponding to the approximation $U^*(\sigma)$. The method simplifies considerably the numerics and has been successfully applied in [1].

The idea of the *surrogate transition method* is to draw samples from the target distribution π with aid of the approximation π^*. We assume that one can conduct a *reversible* Markov transition $S(x, y)$ leaving π^* invariant, i.e. the detailed balance

$$\pi^*(x)S(x, y) = \pi^*(y)S(y, x) \tag{3.47}$$

is satisfied. A valid surrogate transition can be devised by making use of the Metropolis principle on $\pi^*(x)$

Algorithm 3.6.1 (Surrogate transition method)

- *Given a current sample $x^{(t)}$.*
- *Let $y_0 = x^{(t)}$ and recursively*

$$y_i \sim S(y_{i-1}, \cdot), \tag{3.48}$$

  for $i = 1, \dots, k$.
- *Update $x^{(t+1)} = y_k$ with probability*

$$\min\Big\{1, \frac{\pi(y_k)/\pi^*(y_k)}{\pi(x^{(t)})/\pi^*(x^{(t)})}\Big\} \tag{3.49}$$

  and let $x^{(t+1)} = x^{(t)}$ with the remaining probability.

The proposal transition from y_0 to y_k can be written as

$$S^{(k)}(y_0, y_k) = \int \cdots \int S(y_0, y_1) \cdots S(y_{k-1}, y_k) dy_1 \cdots dy_{k-1}$$

In words, $S^{(k)}(\cdot, \cdot)$ is the k-step transition function for the surrogate Markov chain defined by S. It is easy to see that $\pi^*(x)S^{(k)}(x, y) = \pi^*(y)S^{(k)}(y, x)$, Thus, the actual transition function from $x^{(t)}$ to $x^{(t+1)} = y \neq x$ has the form

$$A(x, y) = S^{(k)}(x, y) \min\Big\{1, \frac{\pi(y)/\pi^*(y)}{\pi(x)/\pi^*(x)}\Big\}. \tag{3.50}$$

Hence,

$$\begin{aligned} \pi(x)A(x, y) &= \pi^*(x)S^{(k)}(x, y) \min\Big\{\frac{\pi(x)}{\pi^*(x)}, \frac{\pi(y)}{\pi^*(y)}\Big\} \\ &= \pi^*(y)S^{(k)}(y, x) \min\Big\{\frac{\pi(x)}{\pi^*(x)}, \frac{\pi(y)}{\pi^*(y)}\Big\} \\ &= \pi(y)A(y, x), \end{aligned} \tag{3.51}$$

which is the detailed balance.

3.7 Simulated Annealing

Besides integration, MCMC methods can also be applied to solve high dimensional optimization problems. In connection with EIT inverse problem an optimization method is needed if one wants to evaluate MAP or ML estimate. As an example of an optimization algorithm we introduce the *simulated annealing* (SA) method.

Suppose our task is to find the minimum of a target function $h(x)$. This is equivalent to finding the maximum of

$$\pi(x) \propto \exp\{-h(x)/T\} \tag{3.52}$$

at any given temperature T. Let $T_1 > T_2 > \cdots > T_k > \cdots$ be a sequence of monotone decreasing temperatures in which T_1 is reasonably large and $\lim_{k\to\infty} T_k = 0$. At each temperature T_k we run N_k steps of the Metropolis-Hastings (M-H) or

Gibbs sampling scheme with $\pi_k(x) \propto \exp\{-h(x)/T_k\}$ as the equilibrium distribution. An important mathematical observation is that for any system in which $\int \exp\{-h(x)/T\}dx < \infty$ for all $T > 0$, distribution π_k, as k increases, puts more and more of its probability mass into a vicinity of the global minimum of $h(x)$ when T_k is close to zero. Theoretically, at least, we should be able to obtain good samples from π_k if we let the number of M-H iterations N_k be sufficiently large. Decreasing the temperature during computation can be termed as variance reduction and is commonly used in the field of MCMC computing.

Algorithm 3.7.1 (SA)

- *Initialize at an arbitrary configuration $x^{(0)}$ and temperature level T_1.*
- *For each k, we run N_k iterations of an MCMC scheme with $\pi_k(x)$ as its target distribution. Pass the final configuration of x to the next iteration.*
- *Increase k to $k+1$.*

It can be shown that the global minimum of $h(x)$ can be reached by SA with probability 1 if the temperature variable T_k decreases sufficiently slowly.

3.8 Implementation Issues

Although MCMC algorithms are simple, there are several practical implementation issues that need to be considered before MCMC can be applied to examine a posterior distribution. It is commonly agreed that finding an ideal proposal chain is an art. In practice, one always tends to feel unsatisfied in settling down on any proposal chain.

3.8.1 Burn-in Phase

Usually, after starting a chain a number of iteration steps have to be taken before the chain has reached the important parts of the target distribution and starts to produce appropriate samples. The beginning of the chain is often called a burn-in phase. The length varies largely depending on the implemented sampling technique. A long burn-in phase is a problem occurring especially in connection with the random-step Metropolis algorithm and small step sizes.

3.8.2 Choosing a Sampling Plan

There are several ways to draw extesive sample enesembles through Markov chains. At one extreme, it is possible to generate n independent realizations from the posterior distribution by using n separate runs, each of length m, and retaining the final states from each chain. The run length m is to be chosen large enough to ensure that the chain has passed the burn-in phase. The other extreme is to use a single long run, or perhaps a small number of long runs. Experience appear to favor the use of long runs. The major drawback of using short runs is that it is virtually impossible to tell when a run is long enough based on such runs. Even using long runs, determining how much the initial series is affected by the starting state is difficult.

A complication that arises from the statistical dependence when using a single series is that variances of estimates are more difficult to obtain. One way to increase the level of independence is to retain every rth point of a sample path. Often, behavioral characteristics of a chain is analyzed in terms of autocorrelation curves. Below, the level of independence is studied in terms of autocorrelation.

3.8.3 Determining the Run Length

According to the traditional form of the central limit theorem the variance of the Monte Carlo estimate $\overline{f}_m$ decays as $\gamma^2(f)/\sqrt{m}$ supposing that the samples are independent and identically distributed. The Markov chain based sampling scheme was introduced, since drawing correlated samples eases generation of sample ensembles. However, correlation between samples usually decreases the statistical reliability of the estimate. Thus, it seems justified to argue, that the less correlated are the consecutive samples the faster is the convergence rate of the Monte Carlo estimate $\overline{f}_m$ and the shorter run lengths are needed. The concept of autocorrelation provides us an advantageous way to study the algorithm efficiency.

Let the Markov chain be such that the assumptions of the theorems 1 and 2 are satisfied. Suppose we have drawn samples $x^{(1)}, \ldots, x^{(m)}$ via an MCMC sampler with $\pi(x)$ as its equilibrium distribution. Let us further assume the process has run long enough needed for the equilibration of the chain. Then, the variance of the estimate can be approximated as

$$\begin{aligned} m\mathrm{var}\{\overline{f}_m\} = m\,\mathrm{var}\Big\{\frac{f(x^{(1)}) + \cdots + f(x^{(m)})}{m}\Big\} &= \gamma^2(f)\Big[1 + 2\sum_{j=1}^{m-1}\Big(1 - \frac{j}{m}\Big)\rho_j\Big] \\ &\approx \gamma^2(f)\Big[1 + 2\sum_{j=1}^{\infty}\rho_j\Big] \qquad (3.53) \end{aligned}$$

where $\gamma^2(f) = \mathrm{var}\{f(x)\}$ and $\rho_j = \mathrm{corr}\{f(x^{(1)}), f(x^{(j+1)})\}$. We define the *integrated autocorrelation time* of f as

$$\tau_{int}(f) = \frac{1}{2} + \sum_{j=1}^{\infty}\rho_j, \qquad (3.54)$$

using which

$$m\mathrm{var}\{\overline{f}_m\} = 2\tau_{int}(f)\gamma^2(f). \qquad (3.55)$$

In effect, this variance is equal to that of an estimator with $m/[2\tau_{int}(f)]$ independent random samples. Thus, we call $m/2\tau_{int}(f)$ the *effective sample size.*

Often, ρ_j decays exponentially. Therefore, we can model the autocorrelation curve as

$$|\rho_j| \propto \exp\Big\{-\frac{j}{\tau_{exp}(f)}\Big\}, \qquad (3.56)$$

where

$$\tau_{exp}(f) = \lim\sup_{j\to\infty} \frac{j}{-\log|\rho_j|} \tag{3.57}$$

is known as *exponential autocorrelation time.* When $\tau_{exp}(f)$ is large, the integrated autocorrelation time can be expressed as

$$\tau_{int}(f) \approx \sum_{j=0}^{\infty} e^{-j/\tau_{exp}(f)} - \frac{1}{2} = \frac{1}{1-e-1/\tau_{exp}(f)} - \frac{1}{2} \approx \tau_{exp}(f) \tag{3.58}$$

The relaxation time of the system is defined as

$$\tau_{exp} = \sup_{h\in L^2(\pi)} \tau_{exp}(f) \tag{3.59}$$

As an example showing that the concepts of autocorrelation and relaxation time are closely related to the convergence rate, suppose the state space of the Markov chain is finite and let f be an eigenfunction corresponding to an eigenvalue λ of the transition matrix, then it can be shown that $\rho_j(f) = \lambda^j$. Thus,

$$\tau_{int}(f) = \frac{1+\lambda}{2(1-\lambda)}, \quad \tau_{exp}(f) = -\frac{1}{\log|\lambda|}, \tag{3.60}$$

and the relaxation time is

$$\tau_{exp} = -\frac{1}{\log|\lambda_2|}, \tag{3.61}$$

where λ_2 is the second largest eigenvalue of the transition matrix, which reflects the convergence rate of the Markov chain also on the basis of the theory of Markov Chains [2].

Chapter 4

Linear Algebra

The computational work load required for evaluation of the posterior density $\pi_{post}(\sigma)$ is mainly concentrated in solving the discretized forward problem (2.19). Thus, the convergence rate of the implemented MCMC algorithm depends highly on the efficiency of applied linear algebra in terms of CPU time.

In this section, we introduce some linear algebra methods that can be applied to (2.19). Since the appropriateness of a method depends on both the prior information and the implemented sampling scheme, we give only fairly rough trendsetting estimates of the computational efficiency. Generally, the goal in discovering an effective method is to derive benefit from the property that the sampler typically perturbs σ only in a relatively small-dimensional subspace of H_h.

Determining the minimal computational effort needed for solving a linear system is an interesting issue. For instance, suppose a $n \times n$ system is such that the set of multiplying constants in each equation can be obtained by permuting the multipliers of an other equation and suppose the right hand sides of the equations are equals. Then, due to the symmetry argument we can deduce that the set of unknowns satisfy $x_1 = x_2 = \ldots = x_n$. Provided that the symmetry property is known solving the system requires for dividing the right hand side by the sum of the multipliers. This takes $O(cn)$ floating point operations. Apparently, the knowledge of symmetry diminishes drastically the computational effort which otherwise would be of magnitude $O(cn^3)$. Still, it is also apparent that there is no sense in checking the symmetry condition if there is no reason to believe that symmetry exists.

4.1 Updating the System Matrix

In order to simplify the notation, we denote the total number of the degrees of freedom as $N_s = N_n + L - 1$ in further discussion. Let $t \in \mathbb{R}$ and $d \in \mathbb{R}^M$ such that

$$d_i = \begin{cases} 1, & \text{for } i \in I \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

where the number of entries in I is equal to k, which is the dimension of the finite element mesh underlying the support of the H_h function corresponding to d. Suppose the conductivity distribution is updated as

$$\sigma \to \sigma + td. \tag{4.2}$$

The corresponding update to the system matrix $A_\sigma \in \mathbb{R}^{N_s \times N_s}$ can be written as

$$A_{\sigma+td} = A_\sigma + tV_d \Lambda_d V_d^T, \tag{4.3}$$

where $V_d \in \mathbb{R}^{N_s \times k}$, $V_d = \begin{pmatrix} e_1 & e_2 & \dots & e_k, \end{pmatrix}$, where ith column is the I_ith standard basis vector, i.e.

$$(e_i)_j = \begin{cases} 1, & \text{for}\, j = I_i \\ 0, & \text{otherwise} \end{cases}, \tag{4.4}$$

and $\Lambda_d \in \mathbb{R}^{k \times k}$ is symmetric and positive definite matrix of the form

$$(\Lambda_d)_{ij} = \begin{cases} \int_\Omega \nabla \varphi_i \cdot \nabla \varphi_j \, dx, & i, j \in I \\ 0, & \text{otherwise} \end{cases}. \tag{4.5}$$

On the basis of (4.3) it is clear that the rank of the update is k. Often, k is a relatively small number. For instance, $k = 3$ in a two dimensional case, where the update is $d = (0, \dots, 1, \dots, 0)$, the local-basis functions of H_h are the characteristic functions of FEM elements and the basis of FEM mesh is piecewise linear. In the following sections, we show that the system (2.19) is especially easily solved in the case of low-rank updates.

4.2 The Residual Form

Suppose the X_σ is known and the task is to solve $X_{\sigma+td}$. Thus, we may write $X_{\sigma+td} = X_\sigma + R_{\sigma+td}$, where $R_{\sigma+td}$ is the solution of a residual form of the equation $A_{\sigma+td} X_{\sigma+td} = F$ and can be written as

$$(A_\sigma + tV_d \Lambda V_d^T) R_{\sigma+td} = -tV_d \Lambda_d V_d^T X_\sigma \tag{4.6}$$

Additionally, it is convenient to define

$$R_{\sigma+td} = -tZ_\sigma^{d,t} \Lambda_d V_d^T X_\sigma, \tag{4.7}$$

where $Z_\sigma^{d,t}$ is found as a solution of the equation

$$(A_\sigma + tV_d \Lambda_d V_d^T) Z_\sigma^{d,t} = V_d. \tag{4.8}$$

$Z_\sigma^{d,t} \in \mathbb{R}^{N_s \times k}$, since k being the dimension of the update is independent of the number of current patterns $L-1$. When k is small enough the number of columns in $Z_\sigma^{d,t}$ is smaller than the number of columns in $X_{\sigma+td}$. Thus, it is favorable to apply (4.8). The residual form is especially important when using an iterative solver, since in that case the computational workload depends usually linearly on the number of load vectors.

In the section (4.6) we will show that the residual form is useful if the sampler operates only in a small subset of Ω.

4.3 Choleski factorization

The linear system (2.19) is symmetric and positive definite. Thus, the system matrix can be factorized as $A = LL^T$, which is the well-known Choleski factorization. The lower triangular matrix L can be calculated as

```
for $j = 1 : n$
    $L_{jj} = \sqrt{A_{ii} - \sum_{j=1}^{j-1} L_{ij}^2}$
    for $i = j + 1 : n$
        $L_{ij} = (A_{ij} - \sum_{k=1}^{j-1} L_{jk} L_{ik})/L_{jj}$
    end
end.
```

which can be easily verified in a straightforward manner. The standard process of solving a symmetric and positive linear system $Ax = b$ employs the Choleski factorization as follows:

Factorization Factor the matrix A into LL^T.

Triangular solution Solve $Ly = b$ and $L^T x = y$.

The process of solving $Ly = b$ is known as *backward substitution* and is written as

```
for i = 1 : n
    $z_i = (b_i - \sum_{j=1}^{i-1} L_{ij} z_j)/L_{ii}$
end
```

4.3.1 Computational Work Load

We can approximate the number of floating-point operations needed for computing L as

$$\begin{aligned} \propto \sum_{j=1}^{n} \sum_{i=j+1}^{n} 2j &= \sum_{j=1}^{n} (n-j)j = 2\Big(n \sum_{j=1}^{n} j - \sum_{j=1}^{n} j^2\Big) \\ &= 2\Big(\frac{n^2(n+1)}{2} - \frac{n(n+1)(2n+1)}{6}\Big) \\ &\propto 2\Big(\frac{n^3}{2} - \frac{n^3}{3}\Big) = \frac{n^3}{3}, \end{aligned} \tag{4.9}$$

since both the jth outer and the jth inner loop perform both j multiplications and j summations. With similar arguments we conclude that the backward substitution takes approximately

$$\propto \sum_{i=1}^{n} 2i = \frac{2n(n+1)}{2} \propto n^2 \tag{4.10}$$

floating-point operations. Thus, solving an arbitrary symmetric and positive definite linear system takes $O(n^3/3)$ operations as a whole.

4.4 Choleski Factorization of a Sparse Matrix

The system (2.19) is sparse meaning that the matrix consists largely of zeros. Sparseness can be taken into account in the computational process the entire manoeuvre being

Ordering Find a "good" ordering (permutation P) for the given matrix A.

Storage allocation Determine the necessary information about the Choleski factor L of PAP^T to ensure correct data structures. This is known as *symbolic factorization.*

Factorization Factor the permuted matrix PAP^T into LL^T.

Triangular solution Solve $Ly = b$ and $L^T z = y$. Then set $x = P^T z$.

The first two phases are so called *symbolic part.* The second phase, symbolic factorization, is discussed more closely in [8]. In this thesis, we discuss only the first phase.

The purpose of permuting the system matrix before factoring is to decrease the number of nonzero elements in L. Generally, the number of nonzeros is largely predetermined by the bandwith of the system matrix, which we define as

Definition 13 *We call $d \geq 0$ the bandwidth of the matrix $A \in \mathbb{R}^{n\times n}$ provided that*

$$A_{k,\ell} = 0, \quad \forall\, |k-\ell| > d,\ 1 \leq k, \ell \leq n \tag{4.11}$$

Suppose that $A \in \mathbb{R}^{n\times n}$ is a symmetric and positive definite matrix having a bandwidth d. Then, the first column of its Choleski factor L is

$$\begin{aligned} L_{11} &= \sqrt{A_{11}} \\ L_{i1} &= A_{ij}/L_{11}, \quad \text{for} \quad i = 2, \ldots n, \end{aligned}$$

since $A_{k,1} = 0$ for all $|1-k| > d$. Thus, $L_{1+d+\ell,1} = 0$ for all $1 \leq \ell \leq n-d-1$. Assuming that

$$L_{k+d+\ell,k} = 0, \quad \forall 1 \leq k < j,\ 1 \leq \ell \leq n-d-k, \tag{4.12}$$

we have

$$\begin{aligned} L_{j+d+\ell,j} &= (0 + \sum_{k=1}^{j-1} L_{j,k} L_{j+d+\ell,k})/L_{j,j} = \frac{1}{L_{j,j}} \sum_{k=1}^{j-1} L_{j,k} L_{(j-1)+d+(\ell+1),k} \\ &= \frac{1}{L_{j,j}} \sum_{k=1}^{j-1} L_{j,k} \cdot 0 = 0, \quad \forall\, 1 \leq \ell \leq n-d-j. \end{aligned} \tag{4.13}$$

and we can inductively deduce that

$$L_{k,\ell} = 0, \text{for all} \quad |\ell - k| > d \tag{4.14}$$

Thus, the bandwidth of L is $\leq d$.

In practice, it is reasonable to assume that the bandwidth of A is equal to the bandwidth of L, since it is justified to assume that the element of L is nonzero as far as it cannot be shown to be zero. However, on the basis of (4.14) we see that the Choleski factor L can be obtained as

```
for $j = 1 : n$
$L_{jj} = \sqrt{A_{ii} - \sum_{j=\max\{1,j-d\}}^{j-1} L_{ij}^2}$
for $i = j + 1 : \min\{j + d, n\}$
$L_{ij} = (A_{ij} - \sum_{k=\max\{1,j-d\}}^{j-1} L_{jk} L_{ik})/L_{jj}$
end
end,
```

and the corresponding backward substitution can be written as

```
for i = 1 : n
$z_i = (b_i - \sum_{j=\max\{1,i-d\}}^{i-1} L_{ij} z_j)/L_{ii}.$
end
```

4.4.1 Computational Work Load

In this case, the inner loop depends on d. Thus, the number of floating-point operations is approximated as

$$\propto \sum_{j=1}^{n} \sum_{i=j+1}^{j+d} 2(d+1) = \sum_{j=1}^{n} d(d+1) \propto nd^2 \tag{4.15}$$

and the number of floating-point operations taken by the backward substitution is simply

$$\propto \sum_{i=1}^{n} d = nd \propto O(dn). \tag{4.16}$$

indicating that the number of operation needed for solving a sparse system is predetermined by the factorization. Thus, the total number of operations is of magnitude $O(nd^2)$.

Bandwidth of the System Matrix

Excluding the boundary conditions we can approximate the bandwidth of the matrix (2.22) supposing that

$$A_{ij} = \int_\Omega \sigma \nabla \varphi_i \cdot \nabla \varphi_j \, dx, \tag{4.17}$$

which is the standard FEM stiffness matrix. Since we use piecewise linear basis functions φ, A_{ij} differs from zero, if and only if both φ_i and φ_j are connected to the same triangle. Hence, if we want to constrict the bandwidth as small as possible, the system has to be ordered (permuted) so that all the degrees of freedom connected to an arbitrarily chosen triangle are numbered close to each other.

It can be shown that the bandwidth corresponding to an optimal numbering is in one dimensional case of magnitude $d \propto 1$, in two dimensional case $d \propto \sqrt{n}$ and in three dimensional case $d \propto n^{2/3}$. This is motivated by the figure (4.1). Hence, we can estimate the number of floating-point operations required for sparse Choleski factorization as

$$\begin{array}{rcl} 1D & : & O(cn) \\ 2D & : & O(cn^2) \\ 3D & : & O(cn^{7/3}) \end{array}$$

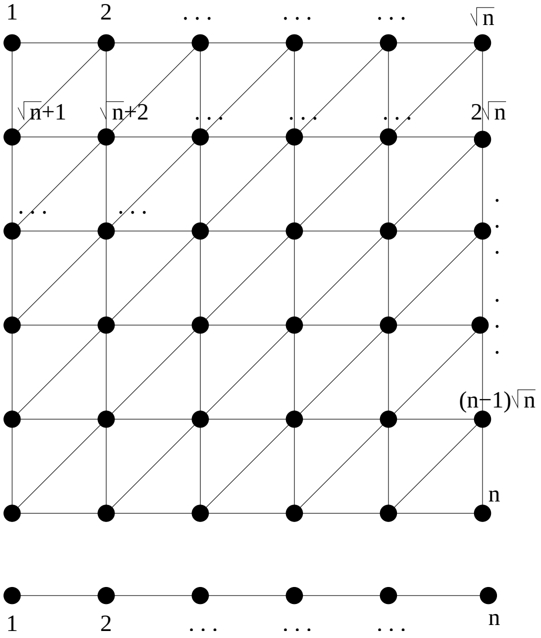

Figure 4.1: The square shaped lattice is numbered so that the distance between any two degrees of freedom connected to an arbitrarily chosen triangle is of magnitude $\sqrt{n}$. The corresponding distance in the one dimensional lattice does not depend on n.

4.5 Sherman-Morrison-Woodbury –formula

Let $A \in \mathbb{R}^{n \times n}$ be invertible, $U_1, U_2 \in \mathbb{R}^{n \times k}$ and let $I + U_2^T A^{-1} U_1$ be invertible. Then, [4]

$$(A + U_1 U_2^T)^{-1} = A^{-1} - A^{-1} U_1 (I + U_2^T A^{-1} U_1)^{-1} U_2^T A^{-1}. \qquad (4.18)$$

Thus, rank k correction to the matrix A causes a rank k correction to its inverse.

The equation (4.18) is known as the Sherman-Morrison-Woodbury –formula. The validity can be proven in a straightforward manner simply by expanding the product

$$\begin{gathered}(A + U_1U_2^T)^{-1}(A + U_1U_2^T) \\ = \\ (A^{-1} - A^{-1}U_1(I + U_2^T A^{-1}U_1)^{-1}U_2^T A^{-1})(A + U_1U_2^T) \quad . \end{gathered} \tag{4.19}$$

Suppose that $d, \sigma \in H_h$ and $t \in \mathbb{R}$ is chosen so that also $\sigma+td$ is admissible. We apply (4.18) to updating the inverse matrix of (2.22) by choosing $A = A_\sigma$, $U_1 = tV_d\Lambda_d$ and $U_2 = V_d$. Then, $A + U_1U_2^T = A_\sigma + tV_d\Lambda_dV_d^T$ and

$$I + U_2^T A^{-1}U_1 = I + tV_d^T A_\sigma^{-1}V_d\Lambda_d, \tag{4.20}$$

is invertible, which can be shown as follows.

Since $\sigma + td$ is admissible, the matrix $A_{\sigma+td}$ is positive definite. Moreover, we can choose $s > \max\{-t, 0\}$ so that the condition

$$0 < c \leq (\sigma - sd)(x) \leq C < \infty \quad \forall x \in \Omega.$$

is satisfied. Thus, $\sigma - sd$ is also admissible and we may write

$$A_{\sigma+td} = A_{\sigma-sd} + (s + t)V_d\Lambda_dV_d^T, \tag{4.21}$$

where $A_{\sigma+td}$, $A_{\sigma-sd}$ and Λ_d are positive definite matrixes. As a consequence, Λ_d^{-1} and $A_{\sigma-se_l}$ exist and are positive definite. Hence, we have

$$\begin{aligned} x^T(\Lambda_d^{-1} + tV_d^T A_\sigma^{-1}V_d)x &= x^T(\Lambda_d^{-1} + (t + s)V_dA_{\sigma-sd}^{-1}V_d)x \\ &= x^T\Lambda_d^{-1}x + (t + s)(V_dx)^T A_{\sigma-sd}^{-1}(V_dx) > 0 \end{aligned} \tag{4.22}$$

for all $x \neq 0$, since s was chosen so that $s + t > 0$, from which we see that $\Lambda_d^{-1} + tV_d^T A_\sigma^{-1}V_d$ is positive definite and invertible and we may write

$$(I + tV_d^T A_\sigma^{-1}V_d\Lambda_d)^{-1} = \Lambda_d^{-1}(\Lambda_d^{-1} + tV_d^T A_\sigma^{-1}V_d)^{-1}. \tag{4.23}$$

As a result, we conclude that the inverse of $A_{\sigma+td}$ can be written as

$$\begin{gathered} A_{\sigma+td}^{-1} = (A_\sigma + tV_d\Lambda_dV_d^T)^{-1} \\ = \\ A_\sigma^{-1} - tA_\sigma^{-1}V_d\Lambda_d(I + tV_d^T A_\sigma^{-1}V_d\Lambda_d)^{-1}V_d^T A_\sigma^{-1} \quad . \end{gathered} \tag{4.24}$$

4.5.1 Computational Work Load

Supposing that the dimension of the update k is much smaller than the dimension of the whole system, i.e. $k << N_s$. Then, it is preferable to compute first the inverse of the full $k\times k$ -matrix $\Lambda_d(I+tV_d^T A^{-1}V_d)$, which is known to take not more than $O(k^3)$ floating point operations. Taking into account that $A_\sigma^{-1}V_d$ represent just picking the columns I from A_σ^{-1} we conclude that the product

$$\underset{k \times N_s}{K_1} = \underset{k \times k}{[\Lambda_d(I + tV_d^T A_\sigma^{-1}V_d\Lambda_d)^{-1}]} \; \underset{k \times N_s}{[V_d^T A_\sigma^{-1}]} \tag{4.25}$$

requires for evaluating k^2N_s separate multiplications of floating-point values and $k(k-1)N_s$ additions, thus, $O(k(2k-1)N_s)$ flops as a whole. Similarly, computing the product

$$\underset{N_s\times N_s}{K_2} = t\,\underset{N_s\times k}{[A_\sigma^{-1}V_d]}\;\;\underset{k\times N_s}{[K_1]} \tag{4.26}$$

requires for $O((2k-1)N_s^2)$ operations. In the summation

$$\underset{N_s\times N_s}{(A_\sigma + tV_d\Lambda_d V_d^T)^{-1}} = \underset{N_s\times N_s}{A_\sigma^{-1}} + \underset{N_s\times N_s}{K_2} \tag{4.27}$$

N_s^2 elements are added together taking $O(N_s^2)$ operations.

Hence, computational work load of updating the inverse matrix by employing the Sherman-Morrison-Woodbury –formula (4.24) is of magnitude

$$O(k^3) + O(k(2k-1)N_s) + O((2k-1)N_s^2) + O(N_s^2) = O(2kN_s^2) \tag{4.28}$$

Comparing this to the work load required for the Choleski factorization, we see that there is no sense in updating the whole inverse of (2.22). However, supposing that we do updates only in a small subspace of $\mathbb{R}^{N_s}$, the Sherman-Morrison-Woodbury –formula provides a fast solution method, which is shown in the following section.

4.6 Restriction to a Submatrix

Multiplying the equation (4.24) from right by V_d gives

$$A_{\sigma+td}^{-1}V_d = A_\sigma^{-1}V_d - tA_\sigma^{-1}V_d\Lambda_d(I + tV_d^TA_\sigma^{-1}V_d\Lambda_d)^{-1}V_d^TA_\sigma^{-1}V_d. \tag{4.29}$$

Employing the notation introduced in section (4.2) and denoting $Z_\sigma^d = Z_\sigma^{d,0}$ this is

$$Z_{\sigma+td} = Z_\sigma^d - tZ_\sigma^d(I + tV_d^TZ_\sigma^d\Lambda_d)^{-1}V_d^TZ_\sigma^d. \tag{4.30}$$

Thus, supposing that Z_σ^d is known the solution of (4.8) can be obtained very easily. Again, multiplying (4.24) from right gives

$$X_{\sigma+td} = X_\sigma - tZ_\sigma^d(I + tV_d^TZ_\sigma^d\Lambda_d)^{-1}V_d^TX_\sigma. \tag{4.31}$$

and we see that by knowing Z_σ^d one is able to directly correct the solution X_σ.

More generally, suppose that we do contiguous updates; that is,

$$\sigma_1 \rightarrow \sigma_1 + t_1d_1 = \sigma_2 \rightarrow \sigma_2 + t_2d_2 = \sigma_3 \rightarrow \ldots, \tag{4.32}$$

such that $d_i \in \mathcal{W} \subset \mathbb{R}^{N_s}$ for all d_i, where the dimension of $\mathcal{W}$ is $N_\mathcal{W}$. Let W be a matrix consisting of the standard basis vectors of $\mathcal{W}$

$$W = \begin{pmatrix} e_{\mathcal{W}_1} & e_{\mathcal{W}_2} & \ldots & e_{\mathcal{W}_{N_\mathcal{W}}} \end{pmatrix} \tag{4.33}$$

Multiplying the equation (4.24) corresponding to the update $\sigma_i \rightarrow \sigma_i + t_id_i$ from right and by denoting $Z_{\sigma_{i+1}}^W = Z_{\sigma_i+t_id_i}^W = A_{\sigma_i+t_id_i}^{-1}W$ we have

$$Z_{\sigma_{i+1}}^W = Z_{\sigma_i}^W - tZ_{\sigma_i}^d(I + tV_{d_i}^TZ_{\sigma_i}^d\Lambda_{d_i})^{-1}V_{d_i}^TZ_{\sigma_i}^W. \tag{4.34}$$

By noting that W is a submatrix of identity matrix we see that $Z_{\sigma_i}^W$ is a submatrix of $A_{\sigma_i}^{-1}$. Again, V_{d_i} is a submatrix of W indicating that $Z_{\sigma_i}^d$ is submatrix of $Z_{\sigma_i}^W$. Hence, we obtain $Z_{\sigma_{i+1}}^W$, if we know $Z_{\sigma_i}^W$.

To summarize, if the updates are restricted to the subspace $\mathcal{W}$, it is enough to correct only $N_{\mathcal{W}}$ columns of A_σ^{-1}; that is, the matrix $A_\sigma^{-1}W$.

To give an example, the MCMC sampling strategies are based either on acceptance-rejection strategy or sampling from conditional distributions, due to which it is, apparently, possible to construct an algorithm in which a number of $t-$ or $d-$ values have to be tested before one is accepted. In such a case, we could proceed as

Algorithm 4.6.1

- *Start with σ and a proposed update $\sigma \rightarrow \sigma + td \in H_h$, $d \in \mathcal{W}$.*
- *Solve $X_{\sigma+td}$ employing (4.31) and, then, evaluate the posterior density.*
- *If the proposed update is accepted, solve $Z_{\sigma+td}^W$ using (4.34) and set $\sigma = \sigma + td$.*

Above, the idea is each time to update as small number of columns as possible. Generally, (4.31) is cheaper operation than (4.34). However, $Z_{\sigma+td}^W$ has to be solved before setting $\sigma = \sigma + td$, since Z_σ is needed in (4.31).

4.6.1 Computational Work Load

Evidently, updating each column of A_σ^{-1} with Sherman-Morrison-Woodbury –formula requires for a similar computational effort, since the columns can be updated independently from each other. Thus, supposing that

$$\alpha = \frac{N_{\mathcal{W}}}{N_s}. \tag{4.35}$$

it seems justified to estimate the computational work load of (4.34) as

$$O(2\alpha k N_s^2) \tag{4.36}$$

on the basis of (4.28). Hence, the efficiency seems to depend linearly on the ratio α.

4.7 Domain Decomposition

When the updates $\sigma \rightarrow \sigma + td$ are not restricted to a small dimensional subset the Sherman-Morrison-Woodbury is not a workable method. Generally, in the case of global updates there is no faster direct method of solving the forward problem than the sparse Choleski factorization. However, it is possible that the structure of the global updates can be taken into account when determining a method of solution. To give an example of such a case we introduce the idea of *domain decomposition* methods.

Let the polygonal domain $\hat{\Omega}$ be decomposed so that $\hat{\Omega} = \cup_{i=1}^{N} \{\hat{\Omega}_i\}$, where the subdomains $\hat{\Omega}_i$ satisfy $\hat{\Omega}_i \cap \hat{\Omega}_j = \emptyset$, for all $i \neq j$ and $\hat{\Omega}_i = \{T_k^{(i)}\}_{k=1}^{N_i}$, $T_k^{(i)} \in \mathcal{T}_h$, i.e. each subdomain $\hat{\Omega}$ is polygonal and coincides with some subset of the triangulation $\mathcal{T}_h$.

Suppose first that conductivity distribution has a cylindrical structure illustrated in figure **??** the decomposition being $\hat{\Omega} = \partial\hat{\Omega}_1 \cup \partial\hat{\Omega}_2$. The value of conductivity is constant in both $\hat{\Omega}_1$ and $\hat{\Omega}_2$. Then, the matrix (2.22) can be partitioned as

$$A = \begin{pmatrix} (1+t_1)A_{II}^{(1)} & 0 & (1+t_1)A_{IB}^{(1)} \\ 0 & (1+t_2)A_{II}^{(2)} & (1+t_2)A_{IB}^{(2)} \\ (1+t_1)A_{BI}^{(1)} & (1+t_2)A_{BI}^{(2)} & (1+t_1+t_2)A_{BB}^{(1)} + (1+t_1+t_2)A_{BB}^{(2)} \end{pmatrix} \tag{4.37}$$

where $t_1, t_2 \in \mathbb{R}$, B refers to the set of nodes lying on $\partial\hat{\Omega}_1 \cap \partial\hat{\Omega}_2$ and I refers to other degrees of freedom related to the subdomain indicated by the superior index. To update the system matrix is to update either t_1 or t_2. A straightforward multiplication shows that A can be written as

$$A = LDL^T = LD_1D_2L^T \tag{4.38}$$

where L and D are sparse block matrices of the form

$$L = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ A_{BI}^{(1)}(A_{II}^{(1)})^{-1} & A_{BI}^{(2)}(A_{II}^{(2)})^{-1} & I \end{pmatrix}, \tag{4.39}$$

$$D_1 = \begin{pmatrix} (1+t_1)A_{II}^{(1)} & & \\ & (1+t_2)A_{II}^{(2)} & \\ & & I \end{pmatrix}, \tag{4.40}$$

$$D_2 = \begin{pmatrix} I & & \\ & I & \\ & & S^{(1)} + S^{(2)} \end{pmatrix}. \tag{4.41}$$

$S^{(1)}$ and $S^{(2)}$ are *Schur complements* of the form

$$S^{(1)} = (1+t_1+t_2)A_{BB}^{(1)} - (1+t_1)A_{BI}^{(1)}(A_{II}^{(1)})^{-1}A_{IB}^{(1)}, \tag{4.42}$$

$$S^{(2)} = (1+t_1+t_2)A_{BB}^{(2)} - (1+t_2)A_{BI}^{(2)}(A_{II}^{(2)})^{-1}A_{IB}^{(2)}. \tag{4.43}$$

Denoting

$$x = \begin{pmatrix} x_I^{(1)} \\ x_I^{(2)} \\ x_B \end{pmatrix}, \quad f = \begin{pmatrix} f_I^{(1)} \\ f_I^{(2)} \\ f_B \end{pmatrix} \tag{4.44}$$

In terms of (4.38) the equation $Ax = f$ can be equivalently written as $D_2L^Tx = D_1^{-1}L^{-1}f$, i.e.

$$\begin{pmatrix} I & 0 & (A_{II}^{(1)})^{-1}A_{IB}^{(1)} \\ 0 & I & (A_{II}^{(2)})^{-1}A_{IB}^{(2)} \\ 0 & 0 & S^{(1)} + S^{(2)} \end{pmatrix} \begin{pmatrix} x_I^{(1)} \\ x_I^{(2)} \\ x_B \end{pmatrix} = \begin{pmatrix} ((1+t_1)A_{II}^{(1)})^{-1} & 0 & 0 \\ 0 & ((1+t_2)A_{II}^{(2)})^{-1} & 0 \\ 0 & 0 & I \end{pmatrix} \times \tag{4.45}$$

$$\begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ -A_{BI}^{(1)}(A_{II}^{(1)})^{-1} & -A_{BI}^{(2)}(A_{II}^{(1)})^{-1} & I \end{pmatrix} \begin{pmatrix} f_I^{(1)} \\ f_I^{(2)} \\ f_B \end{pmatrix},$$

a block wise *backward substitution* yields the solution. Since matrices $A_{II}^{(i)}$ and $A_{IB}^{(i)}$ are now independent of σ, the computational effort of solving (4.46) after updating the conductivity distribution is largely determined by the undermost equation

$$(S^{(1)} + S^{(2)})x_B = f_B - A_{BI}^{(1)}(A_{II}^{(1)})^{-1}f_I^{(1)} - A_{BI}^{(2)}(A_{II}^{(2)})^{-1}f_I^{(2)} \tag{4.46}$$

A similar treatment applies to the general case. Suppose that the number of subdomains is N and let $\tilde{R}_i$ be a restriction matrix such that $\tilde{R}_i u$ corresponds to the nodes lying on the boundary $\partial\Omega_i$. Denoting $u_B = \sum_{i=1}^{N} \tilde{R}_i^T u_B^{(i)}$ the generalization of (4.46) can be written as

$$\Big(\sum_{i=1}^{N} \tilde{R}_i^T S^{(i)} \tilde{R}_i\Big)u_B = f_B - \sum_{i=1}^{N} \tilde{R}_i^T A_{BI}^{(i)}(A_{II}^{(i)})^{-1}f_I^{(i)} \tag{4.47}$$

4.7.1 Computational Work Load

Apparently, the computational work load is a function of the length of $\bigcup_{i=1}^{N} \partial\hat{\Omega}_i$. The Schur complement $S^{(i)}$ is practically always a full matrix, since $(A_{II}^{(i)})^{-1}$ is full as an inverse matrix. Thus, the workability of the method decreases rapidly as the boundaries $\partial\hat{\Omega}_i$ lengthen. Therefore, it is not evident whether the method is applicable in practice.

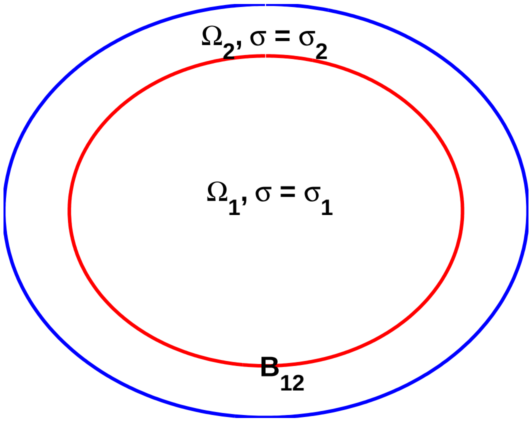

Figure 4.2: A circular domain Ω that has been decomposed in Ω_1 and Ω_2. σ is assumed to be constant in each subdomain. The red circle indicates the boundary B_{12}.

4.8 Conjugate Gradients (CG)

According to the well-known discrepancy principle, due to the noise in the measurements one cannot expect an approximate solution $U^*(\sigma) \approx U(\sigma)$ yield a smaller residual error than what is the measurement error; that is, an approximation $U^*(\sigma)$ can be considered to be accurate enough provided that

$$||U^*(\sigma) - U(\sigma)|| \leq \epsilon, \tag{4.48}$$

where ϵ is an overall estimate of the noise level. Thus, in some cases it might be sufficient just to find an approximate solution X_σ^* of (2.19). Therefore, it seems possible that some iterative linear algebraic method could turn out to be a preferable way to solve the forward problem.

To give an example of an iterative method, we introduce the *conjugate gradient* (CG) method, which is one of the most often used iterative methods applied for solving symmetric and positive definite linear systems.

4.8.1 Quadratic Function J

To be able to motivate the idea behind conjugate gradients, we introduce first some concepts of crucial importance.

Let $J : \mathbb{R}^n \to \mathbb{R}$ be a quadratic function,

$$J(y) = \frac{1}{2} y^T A y - y^T b. \tag{4.49}$$

Let $A \in \mathbb{R}^{n \times n}$ be a symmetric and positive definite matrix and let $Ax = b$. Then, J attains its minimum at $x = A^{-1}b$.

This is easily verified by writing

$$\begin{aligned} J(x+y) &= \frac{1}{2}(x+y)^T A(x+y) - (x+y)^T b \\ &= \frac{1}{2}(x^T A x + 2x^T A y + y^T A y) - (x+y)^T b \\ &= \frac{1}{2}(b^T A^{-1} b + 2b^T y + y^T A y) - b^T A^{-1} b - y^T b \\ &= \frac{1}{2} y^T A y - \frac{1}{2} b^T A^{-1} b \\ &\geq -\frac{1}{2} b^T A^{-1} b = J(x+0) = J(x). \end{aligned} \tag{4.50}$$

J is a convex function (i.e. $J(\lambda x + (1-\lambda)y) \leq J(\lambda x) + J((1-\lambda)y)$, for all $0 \leq \lambda \leq 1$), since it is convex in all directions $p \in \mathbb{R}^n$, which we see by writing

$$\frac{d^2}{dt^2} J(x_0 + tp) = \frac{d^2}{dt^2}[\frac{1}{2}(x_0 + tp)^T A(x_0 + tp) - (x_0 + tp)^T b] = \frac{1}{2} p^T A p. \tag{4.51}$$

Therefore, the minimum

$$\rho_k = \arg\min_{t \in \mathbb{R}} J(x_k + tp_k), \tag{4.52}$$

is attained provided that

$$\frac{d^2}{dt}[\frac{1}{2}(x_k + \alpha_k p_k)^T A(x_k + \alpha_k p_k) - (x_k + \alpha_k p_k)^T b] = 0 \tag{4.53}$$

implying that $\alpha_k p_k^T A(x_k + \alpha_k p_k) = p_k^T b$, which is equivalent to

$$\alpha_k = \frac{p_k^T(b - Ax_k)}{p_k^T A p_k} \tag{4.54}$$

Please note that unlike solution of the complete linear system, the minimum of $J(x_k + tp_k)$ with respect to t, since computing α_k requires only for evaluating the matrix-vector product Ap_k, summation of two vectors plus evaluation of two inner products.

The gradient of $J(x_k)$ is

$$\nabla J(x_k) = Ax_k - b = -r_k. \tag{4.55}$$

Thus, the residual $r_k = b - Ax_k$ is the direction in which J decreases most rapidly.

4.8.2 CG Algorithm

In the conjugate gradient method, the solution of a symmetric and positive definite system $Ax = b$, $A \in \mathbb{R}^{n \times n}$ is found based on the idea of updating the candidate solution x_k as

$$x_{k+1} = x_k + \alpha_k p_k, \tag{4.56}$$

where α_k is chosen as in (4.54) and $p_0, \ldots, p_k$ are so-called *conjugate directions*; that, is

$$p_k A p_j = 0, \quad \text{for all} \quad j = 1, \ldots, k-1. \tag{4.57}$$

Consequently, kth step minimizes J in the direction p_k and due to the effective way of choosing the consecutive directions the exact solution of the complete system is reached in not more than n steps. Moreover, evaluation of α_k is computationally cheap operation and each conjugate direction p_k can easily be computed from the gradient r_{k-1}.

Algorithm 4.8.1 (Conjugate Gradient Method)

```
$x_0 = 0$, $r_{-1} = r_0 = b$, $p_{-1} = 0$
for $k = 0, 1, 2, \ldots$
$\mu_k = ||r_k||^2/||r_{k-1}||$
$p_k = \mu_k p_{k-1} + r_k$
$\alpha_k = ||r_k||^2/p_k^T A p_k$
$x_{k+1} = x_k + \alpha_k p_k$
$r_{k+1} = r_k - \alpha_k A p_k$
end
```

The following result states the validity of the algorithm.

Theorem 3 *Let $A \in \mathbb{R}^{n \times n}$ be symmetric and positive definite and let $Ax = b$. Then, the conjugate gradient method satisfies*

1. *x_{k+1}, p_k, r_k lie in the* Krylov-space *$\mathcal{K}_n$, i.e.*

$$x_{k+1}, p_k, r_k \in \mathcal{K}_{k+1} := span\{b, Ab, \ldots, A^k b\}, \tag{4.58}$$

2. *The residuals are orthogonal, i.e.* $r_k^T r_j = 0$, *for all* $k \neq j$ *and the directions* p_j *are A-orthogonal, i.e.* $p_k^T A p_j = 0$ *for all* $k \neq j$. *Moreover, provided that* $r_k \neq 0$,
$$span\{r_1, \ldots, r_k\} = span\{p_1, \ldots, p_k\} = \mathcal{K}_k \tag{4.59}$$

3. *Let*
$$|| \cdot ||_A : \mathbb{R}^n \to \mathbb{R}, \quad ||y||_A = \sqrt{y^T A y}. \tag{4.60}$$
The function $g(y) = ||x - y||_A^2$ *satisfies* $g(y) \geq g(x_k)$ *for all* $y \in \mathcal{K}_k$.

4. *The algorithm converges to the exact solution in not more than n steps; that is,* $r_k = 0$ *for some* $k \leq n$. *Consequently,*
$$r_{dim(\mathcal{K}_n)} = 0. \tag{4.61}$$

Proof.

1. Since $p_0 = \mu_0 p_{-1} + r_0 = r_0$ and $x_1 = x_0 + \alpha_0 p_0 = \alpha_0 r_0$, we can assume that (4.58) holds for some k. Then, $r_{k+1} = r_k - \alpha_k A p_k$. Thus, $x_{k+1} \in \mathcal{K}_{k+1}$. Because $p_{k+1} = \mu_{k+1} p_k + r_{k+1}$ and $x_{k+2} = x_{k+1} + \alpha_{k+1} p_{k+1}$, we have also $x_{k+2}, p_{k+1} \in \mathcal{K}_{k+1}$.

2. Since $r_0 = p_0$, a straightforward calculation gives
$$\begin{aligned} r_0^T r_1 &= r_0^T (r_0 - \alpha_0 A p_0) = p_0^T (p_0 - \frac{||p_0||^2}{p_0^T A p_0} A p_0) \\ &= ||p_0||^2 - ||p_0||^2 = 0 \end{aligned} \tag{4.62}$$
Additionally, $r_0 - r_1 = A p_0 / \alpha_0$ implying that
$$\begin{aligned} p_1^T A p_0 &= \frac{1}{\alpha_0} p_1^T (r_0 - r_1) = \frac{1}{\alpha_0} (\mu_1 r_0 + r_1)^T (r_0 - r_1) \\ &= \frac{1}{\alpha_0} (\mu_1 ||r_0||^2 - ||r_1||^2) = 0 \end{aligned} \tag{4.63}$$
Suppose both $r_i^T r_j = 0$ and $p_i^T A p_j = 0$ for all $j < i \leq k$. Apparently, this is true with $k = 1$. Moreover,
$$\begin{aligned} r_{k+1}^T r_k &= (r_k - \alpha_k A p_k)^T r_k = ||r_k||^2 - \frac{||r_k||^2}{p_k^T A p_k} (p_k - \mu_k p_{k-2})^T \\ &= ||r_k||^2 - ||r_k||^2 = 0 \qquad (4.64) \\ r_{k+1}^T r_j &= (r_k - \alpha_k A p_k)^T r_j = \alpha_k p_k^T A r_j = \alpha_k p_k^T A (p_j - \mu_j p_{j-1}) \\ &= 0, \quad j < k. \qquad (4.65) \end{aligned}$$
Thus, $r_i^T r_j = 0$ for all $j < i \leq k + 1$ implying that $r_k^T r_j = 0$ for all $k \neq j$. Particularly, the residuals are linearly independent.

Since $r_j \in \mathcal{K}_j$ for all $j \leq k$, we have $\text{span}\{r_0, r_1, \ldots, r_k\} \subset \mathcal{K}_k$. Again, $\dim(\mathcal{K}_k) \leq k$ and due to the linear independence $\dim(\text{span}\{r_1, \ldots, r_k\}) = k$. Thus, we can deduce
$$\text{span}\{r_1, \ldots, r_k\} = \mathcal{K}_k. \tag{4.66}$$

Consequently, $p_j \in \mathcal{K}_j \perp r_k$, i.e. $r_k^T p_j = 0$ for all $j < k$. Hence,

$$\begin{aligned}
p_{k+1}^T A p_k &= \frac{1}{\alpha_k}(\mu_{k+1} p_k + r_{k+1})^T (r_k - r_{k+1}) \\
&= \frac{1}{\alpha_k}(\mu_{k+1}(\mu_k p_{k-1} + r_k) + r_{k+1})^T (r_k - r_{k+1}) \\
&= \frac{1}{\alpha_k}(\mu_{k+1}||r_k||^2 - ||r_{k+1}||^2) = 0 \qquad (4.67) \\
p_{k+1}^T A p_j &= p_j^T A p_{k+1} = \frac{1}{\alpha_k}(\mu_j p_{j-1} + r_j)^T (r_{k+1} - r_{k+2}) = 0 \quad j \leq k (4.68)
\end{aligned}$$

which is $p_i^T A p_j = 0$ for all $j < i \leq k+1$. Thus, $p_k^T A p_j = 0$ for all $k \neq j$. The positive definiteness of A implies the linear independence:

$$\sum_{j=1}^{k} \beta_j p_j = 0 \quad \Rightarrow \quad p_i^T A \sum_{j=1}^{k} \beta_j p_j = \beta_i ||p_i||_A^2 = 0 \quad \Rightarrow \quad \beta_i = 0 \tag{4.69}$$

Again, we have

$$\text{span}\{p_1, \ldots, p_k\} = \mathcal{K}_k \tag{4.70}$$

3. On the basis of (4.59), we can write

$$\begin{aligned}
y_0 &:= \text{argmin}_{y \in \mathcal{K}_1} ||x - y||_A^2 = \text{argmin}_{y \in \text{span}\{p_0\}} ||x - y||_A^2 \\
&= ||A^{-1}b - \beta_0 p_0||_A^2 = \beta_0^2 p_0^T A p_0 - 2\beta_0 b^T p_0 + b^T A^{-1} b, \qquad (4.71)
\end{aligned}$$

which is a polynomial of order two with respect ro β_0. Thus,

$$\beta_0 = \frac{b^T p_0}{p_0^T A p_0} = \frac{||r_0||^2}{p_0^T A p_0} = \alpha_k \tag{4.72}$$

where we have used the identity $b = p_0 = r_0$. Hence, we have $y_0 = \alpha_0 p_0 = x_0 + \alpha_0 p_0 = x_1$.

Now, let $y_k = x_k$ for some k, i.e. we suppose that x_k minimizes the function $g(y) = ||x - y||_A$ in $\mathcal{K}_k$. On the basis of (4.59) and A-orthogonalization $\text{span}\{p_{k+1}\} = \mathcal{K}_{k+1} \setminus \mathcal{K}_k$ and $p_{k+1}^T y = 0$ for all $y \in \mathcal{K}_{k+1}$. Hence,

$$\begin{aligned}
y_{k+1} &= \text{argmin}_{y \in \mathcal{K}_{k+1}} ||x - y||_A = z_k + \beta_{k+1} p_{k+1} \\
||x - y_{k+1}||_A &= ||x - z_k||_A^2 + 2\beta_{k+1} p_{k+1}^T A (A^{-1}b - z_k) + \beta_{k+1}^2 ||p_{k+1}||_A^2 \\
&= ||x - z_k||_A^2 + 2\beta_{k+1} p_{k+1}^T b + \beta_{k+1}^2 ||p_{k+1}||_A^2. \qquad (4.73)
\end{aligned}$$

Since $g(y) \geq g(x_k)$ for all $y \in \mathcal{K}_{k+1}$, we have $||x - x_k||_A \leq ||x - z_k||$. Thus, $z_k = x_k$. Again, β_{k+1} is obtained as

$$\begin{aligned}
\beta_{k+1} &= \frac{b^T p_{k+1}}{p_{k+1} A p_{k+1}} = \frac{(b - Ax_k)^T p_{k+1}}{p_{k+1} A p_{k+1}} = \frac{r_k^T(\mu_{k+1}(\mu_k p_{k-1} + r_k) + r_{k+1})}{p_{k+1} A p_{k+1}} \\
&= \frac{\mu_{k+1}||r_k||^2}{p_{k+1} A p_{k+1}} = \frac{||r_{k+1}||^2}{p_{k+1} A p_{k+1}} = \alpha_{k+1}. \qquad (4.74)
\end{aligned}$$

As a result,

$$y_{k+1} = x_k + \alpha_{k+1} p_{k+1} = x_{k+1} \tag{4.75}$$

Now, an inductive deduction implies that $y_k = x_k$ for all k.

4. Provided that $r_0 \neq 0, \ldots, r_{n-1} \neq 0$, we have $\dim(\text{span}\{r_0, \ldots, r_n\}) = n$. Thus, $x \in \text{span}\{r_0, \ldots, r_n\} = \mathcal{K}_n$. Consequently,

$$x_n = \text{argmin}_{y \in \mathcal{K}_n} ||x - y||_A^2 = ||x - x||_A = 0 \tag{4.76}$$

from which we see that the finding the exact solution takes not more than $\leq n$ steps. The equation (4.59) indicates that

$$\dim(\text{span}\{r_0, \ldots, r_k\}) \leq \dim(\mathcal{K}_n) \tag{4.77}$$

for all $k \leq n$. Due to the orthogonality we conclude that

$$r_{\dim(\mathcal{K}_n)} = 0. \tag{4.78}$$

4.8.3 Convergence Rate

Since (3) states that $x_k \in \mathcal{K}_k$, we have $x_k = p_{k-1}(A)b$, where $p_{k-1} \in \mathbb{P}_{k-1}$ is a polynomial of order $k-1$ at the utmost. Again, (3) implies that

$$\begin{aligned} ||x - x_k||_A &= \min_{y \in \mathcal{K}_k} ||x - y||_A = \min_{p \in \mathbb{P}_{k-1}} ||x - p(A)b||_A \\ &= \min_{p \in \mathbb{P}_{k-1}} ||x - Ap(A)A^{-1}b||_A = \min_{p \in \mathbb{P}_{k-1}} ||x - Ap(A)x||_A \\ &= \min_{p \in \mathbb{P}_k, p(0)=1} ||p(A)x||_A \end{aligned} \tag{4.79}$$

Since A being a symmetric and positive definite $n \times n$-matrix has n orthonormal eigenvectors, we can write $A = Z^T \Lambda Z$, where Z is an orthogonal matrix. Thus,

$$\begin{aligned} ||p(A)x||_A^2 &= x^T p(A)Ap(A)x = x^T Z p(\Lambda)\Lambda p(\Lambda) Z^T x \\ &= \sum_{j=1}^{n} |p(\lambda_j)|^2 \lambda_j |(Z^T x)_j|^2 \leq \max_{1 \leq i \leq n} |p(\lambda_i)|^2 \sum_{j=1}^{n} \lambda_j |(Z^T x)_j|^2 \\ &= \max_{1 \leq i \leq n} |p(\lambda_i)|^2 ||x||_A^2 \leq \max_{\lambda \in [\lambda_{min}, \lambda_{max}]} |p(\lambda)|^2 ||x||_A^2 \end{aligned} \tag{4.80}$$

Additionally, it not very difficult to show that

$$\begin{aligned} \min_{p \in \mathbb{P}_k, p(0)=1} \max_{\lambda \in [\lambda_1, \lambda_n]} |p(\lambda)| &= t_k\Big(\frac{\lambda_{max} + \lambda_{min}}{\lambda_{max} - \lambda_{min}}\Big)^{-1} \\ &= t_k\Big(\frac{\kappa + 1}{\kappa - 1}\Big)^{-1} \leq 2\Big(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\Big)^k, \end{aligned} \tag{4.81}$$

where t_k is a Tshebytsev polynomial of order k and $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of A. Hence,

$$||x - x_k||_A \leq 2\Big(\frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1}\Big)^k ||x - x_0||_A, \tag{4.82}$$

where $\kappa(A) = \lambda_{max}/\lambda_{min} = ||A^{-1}||||A||$ is the so-called *condition number* of A.

4.8.4 Computational Work Load

Let Ω be a two dimensional domain. It can be shown [3] that there are constants c_1 and c_2 independent from the mesh size h such that

$$c_1 h^{-2} \leq \kappa(A) \leq c_2 h^{-2}, \tag{4.83}$$

for a system matrix corresponding to any $H^1(\Omega)$-elliptic, problem discretized by using FEM. Assuming that Ω is a two dimensional domain and the triangulation is regular, we can, apparently, estimate the mesh size as $h \propto 1/n^2$. Thus, we have

$$K \leq c\sqrt{\kappa(A)} \leq ch^{-1} \leq c\sqrt{n} \tag{4.84}$$

Taking into account that the condition number is likely to be $>> 1$ we obtain the following approximation

$$\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1} = \frac{1-1/\sqrt{\kappa}}{1+1/\sqrt{\kappa}} = (1-\frac{1}{\sqrt{\kappa}})\frac{1}{1+1/\sqrt{\kappa}} \approx (1-\frac{1}{\sqrt{\kappa}})^2 \approx 1-\frac{2}{\sqrt{\kappa}}, \tag{4.85}$$

since $1/(1+t) \approx 1-t$ and $t^2 \approx 0$ for small t. Again, $\log(1+t) \approx t$ for small t. Hence,

$$\log\frac{||x-x_k||_A}{||x-x_k||_A} \geq ck\log\Big(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\Big) \geq ck\log\Big(1-\frac{2}{\sqrt{\kappa}}\Big) \geq k\frac{c}{\sqrt{\kappa}}. \tag{4.86}$$

Thus, supposing that the convergence criterium is of the form $||x-x_N||/||x-x_0|| < \epsilon$, i.e. we qualify x_N for a solution provided that the relative residual is $< e$, we have

$$N \leq c\sqrt{\kappa} \leq c\sqrt{n} \tag{4.87}$$

In conjugate gradient method, the work load required for taking one step is largely predetermined by the product Ap_k. In a regular triangulation the number of neighboring triangles is proportional to some constant implying that every column of A has a standard number of non-zeros. Thus, it seems justified to estimate the number of floating-point operations required for Ap_k to be of magnitude $O(cn)$. This together with (4.87) indicates that the exact solution is found in not more than

$$O(cn\sqrt{n}) \tag{4.88}$$

floating-point operations.

4.8.5 Diffusion of Information

So as to give an mental picture of the structure of the Krylov-space

$$\mathcal{K}_n = \text{span}\{b, Ab, \ldots, A^{n-1}b\},$$

we assume that the system matrix is of the simple form

$$\underset{n\times n}{A} = \begin{pmatrix} 2 & -1 & & & & & & \\ -1 & 2 & -1 & & & & & \\ & -1 & 2 & -1 & & & & \\ & & -1 & 2 & -1 & & & \\ & & & \ddots & \ddots & \ddots & & \\ & & & & -1 & 2 & -1 & \\ & & & & & -1 & 2 & -1 \\ & & & & & & -2 & 2 \end{pmatrix}. \tag{4.89}$$

and the load vector is $b = e_1 = \begin{pmatrix} 1 & 0 & \dots & 0 \end{pmatrix}^T$. Then, the exact solution $x = A^{-1}b = (1, \dots, 1)^T$ and the basis of the Krylov-space is constructed as

$$\begin{aligned}
Ab &= \begin{pmatrix}
2 & -1 & & & & & & \\
-1 & 2 & -1 & & & & & \\
 & -1 & 2 & -1 & & & & \\
 & & -1 & 2 & -1 & & & \\
 & & & \ddots & \ddots & \ddots & & \\
 & & & & -1 & 2 & -1 & \\
 & & & & & -1 & 2 & -1 \\
 & & & & & & -2 & 2
\end{pmatrix}
\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix}
= \begin{pmatrix} 2 \\ -1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} \\
A^2b = A(Ab) &= \begin{pmatrix}
2 & -1 & & & & & & \\
-1 & 2 & -1 & & & & & \\
 & -1 & 2 & -1 & & & & \\
 & & -1 & 2 & -1 & & & \\
 & & & \ddots & \ddots & \ddots & & \\
 & & & & & -1 & 2 & \\
 & & & & & -1 & 2 & -1 \\
 & & & & & & -2 & 2
\end{pmatrix}
\begin{pmatrix} 2 \\ -1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix}
= \begin{pmatrix} 5 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} \\
A^3b = A(A^2b) &= \dots \\
&\vdots
\end{aligned} \tag{4.90}$$

indicating that

$$(A^k b)_i = 0, \quad \text{for all} \quad i > k. \tag{4.91}$$

Thus, $(1, \dots, 1)^T = x \notin \mathcal{K}_k$ for $k < n$. In this case, the conjugate gradient iteration cannot reach the exact solution in less than n steps, since $x_k \in \mathcal{K}_k$ according to (3).

The above described phenomenon occurs when the conjugate gradient method is applied to the problem (2.19). In order to demonstrate this, suppose that we want to solve the residual $Z_{\sigma+td}$.

Since CG algorithm is based on computing products Ap_k, each column of $Z_{\sigma+td}$ has to be solved separately. Restricting ourselves to one single column we can write the system (4.8) as

$$Ax = e. \tag{4.92}$$

where $e = (0, \dots, 1, \dots, 0)^T$. Thus the problem related Krylov-space is

$$\mathcal{K}_n = \mathrm{span}\{e, Ae, A^2e, \dots, A^{n-1}e\}.$$

Since A_{ij} is nonzero, if and only if the measure of the set $\mathrm{supp}\{\varphi_i\} \cap \mathrm{supp}\{\varphi_j\}$ is strictly positive, i.e. the degrees of freedom i and j are close to each other, we have to take a few iteration steps before the every element of the candidate solution x_k differs from zero. This is illustrated by the figure 1. Generalizing, provided that $A_{ij} = 0$ updating the value of $(x_k)_i$ does not affect the value of $(x_{k+l})_j$ for $l < l_d$, where l_d depends on the distance between i and j. In terms of nodes the diameter of Ω can be approximated as $\propto 1/h \propto \sqrt{n}$. Thus, supposing that the convergence rate is predetermined by the diameter the number of floating-point operations is again

$$\propto n\sqrt{n} \tag{4.93}$$

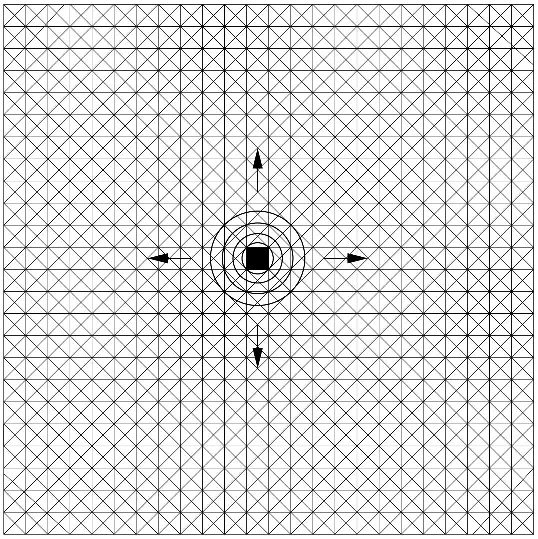

Figure 4.3: The Q^h-counterpart of e differs from zero in the dark region. The nested circles represent the regions, where Q^h counterparts of the residuals r_0, r_1, r_2 and r_3 differ from zero. The arrows show the direction of the diffusion.

The phenomenon can be interpreted as diffusion of information. The system matrix A in (2.22) can be considered as a discretized diffusion operator. Operating with A causes diffusion, which we see by writing

$$\partial_t u - \nabla \cdot (\sigma \nabla u) = 0 \tag{4.94}$$

and by approximating

$$\frac{x_{t+h} - x_t}{h} - Ax_t = 0 \quad \Rightarrow \quad x_{t+h} = x_t + hAx_t \tag{4.95}$$

Thus, it seems that we can equate $\{x_0, x_1, \ldots, x_K\}$ with a diffusion process equilibrium of which is the exact solution $x = A^{-1}e$ and the convergence rate of the iteration is predetermined by the rate of the diffusion.

Practically all the iterative linear algebraic methods are one way or other based on computing matrix-vector products. Therefore, the problem with diffusion is likely to occur also in connection with other methods.

4.8.6 Preconditioned Conjugate Gradients (PCG)

The preconditioned conjugate gradient method is simply the conjugate gradient method applied to the trinity $\tilde{x} = Sx$, $\tilde{x}_k = Sx_k$, $\tilde{b} = S^{-1}b$ and $\tilde{A} = S^{-1}AS^{-1}$, where S is a symmetric and positive definite matrix. We denote $M = S^2$ and call M a preconditioner. The algorithm can be written as

Algorithm 4.8.2 (Preconditioned Conjugate Gradients)

```
$x_0 = 0,\ r_{-1} = r_0 = b,\ p_{-1} = 0$
for $k = 0, 1, 2, \dots$
$Mz_k = r_k$
$\mu_k = r_k^T z_k / r_{k-1}^T z_{k-1}$
$p_k = \mu_k p_{k-1} + z_k$
$\alpha_k = r_k^T z_k / p_k^T A p_k$
$x_{k+1} = x_k + \alpha_k p_k$
$r_{k+1} = r_k - \alpha_k A p_k$
end
```

By writing

$$(S^{-1}AS^{-1})Z = \Lambda Z \quad \Leftrightarrow \quad \Lambda S^{-1}Z = S^{-1}(S^{-1}AS^{-1})Z = (M^{-1}A)S^{-1}Z \tag{4.96}$$

we see that $M^{-1}A$ has same eigenvalues as $\tilde{A}$. Thus, $\kappa(\tilde{A}) = \kappa(M^{-1}A)$. A straightforward calculation shows that equation (3) applied to PCG states that

$$\begin{aligned} x_{k+1} &\in \mathrm{span} S^{-1}\{\tilde{b}, \tilde{A}\tilde{b}, \dots, \tilde{A}^k\tilde{b}\} \\ &= \{M^{-1}b, (M^{-1}A)M^{-1}b, \dots, (M^{-1}A)^k M^{-1}b\} \end{aligned} \tag{4.97}$$

Apparently, the convergence rate can be estimated as

$$||x - \tilde{x}_k||_A \leq 2\Big(\frac{\sqrt{\kappa(M^{-1}A)} - 1}{\sqrt{\kappa(M^{-1}A)} + 1}\Big)^k ||x - \tilde{x}_0||_A \tag{4.98}$$

Thus, the number of iterations can is approximated as

$$K \leq c\sqrt{\kappa(M^{-1}A)} \tag{4.99}$$

It is easy to show that the better M^{-1} estimates A^{-1} the smaller is the value of $\kappa(M^{-1}A)$. Again, M^{-1} can be interpreted as a diffusion operator and the preconditioned iteration as a diffusion process the convergence rate of which is the faster the closer M^{-1} is to A.

In applications, PCG is usually preferred to CG.

4.8.7 Complete Preconditioning

The matrix $M = A_\sigma$ a complete preconditioner. By denoting $C_d := A_\sigma^{-1} V_d \Lambda_d V_d^T$, we have

$$M^{-1}A_{\sigma+td} = A_\sigma^{-1}(A_\sigma + tV_d\Lambda_d V_d^T) = I + tA_\sigma^{-1}V_d\Lambda_d V_d^T = I + tC_d, \tag{4.100}$$

using which gives

$$\begin{aligned} (Z_{\sigma+td})_k &\in \mathrm{span}\{M^{-1}V_d, (M^{-1}A_{\sigma+te_d})M^{-1}V_d, \dots, (M^{-1}A_{\sigma+td})^{k-1}M^{-1}V_d\} \\ &= \mathrm{span}\{C_d, C_d^2, \dots, C_d^k\}V_d\Lambda_d^{-1} \end{aligned} \tag{4.101}$$

Since $\mathrm{rank}(\Lambda_d) = m$, we see that $\mathrm{rank}(C) = m$. Thus,

$$\dim(\mathrm{span}\{C_l, C_l^2, \dots, C_l^k\}) \leq m, \quad \forall k = 0, 1, \dots, n. \tag{4.102}$$

This together with (3) implies, that using A_σ as a preconditioner causes PCG to converge to the exact solution in not more than $\leq m$ steps.

Even though m is often a relatively small number, complete preconditioning does not ensure a fast solution in terms of CPU time, since computing

$$Mz_k = A_\sigma z_k = r_k \tag{4.103}$$

requires for relatively heavy computation. Clearly, it is possible to factorize A_σ as $A_\sigma = L_\sigma L_\sigma^T$ in advance, in which case computing $z_k = L^{-T}L^{-1}r_k$ is the only required operation. Thus, on the basis of (4.10) we can approximate the computational work load of complete preconditioning to be of magnitude $O(cN_s\sqrt{N_s})$. Employing A_σ as a preconditioner is studied more closely in [15].

4.8.8 SSOR preconditioning

One of the most often used ways of preconditioning is SSOR, where

$$M = [\omega(2-\omega)]^{-1}(D_\sigma - \omega L_\sigma)D_\sigma^{-1}(D_\sigma - \omega L_\sigma^T), \tag{4.104}$$

where $0 < \omega < 2$ is the relaxation parameter, D_σ is the diagonal of A_σ and L_σ is a lower triangular part of A_σ. It can be shown [3] that with optimal value of ω, the condition number is $\kappa(M^{-1}A) \propto 1/h \propto \sqrt{n}$. Thus, on the basis of (4.99), it seems that

$$K \leq cn^{1/4} \tag{4.105}$$

indicating that the SSOR preconditioned PCG demands for

$$\leq cn^{5/4} \tag{4.106}$$

floating-point operations. Again, due to the problem with diffusion we cannot expect much faster convergence.

Chapter 5

Numerical Experiments

In this chapter, we demonstrate how the introduced methods are applied to the EIT problem. We are not interested in simulating any realistic application but just describing some general phenomena. For simplicity, the computations are performed in the unit disc

$$\Omega = B(\mathbf{0}, 1) \subset \mathbb{R}^2. \tag{5.1}$$

Voltages are measured with six electrodes (i. e. $L = 6$) evenly distributed along the boundary curve. The contact impedances are assumed to be equal to one, i.e. $z_1 = z_2 = \ldots = z_2 = 1$, and the noise in the measured electrode potentials to be independent. In this model, the impedances are known exactly. Thus, we use $L-1 = 5$ current patterns of the form (2.31) so as to get the best possible distinguishability in the sense of (2.30).

5.1 Small Perturbations

We seek a small circular perturbation (anomaly) from Ω. That is, the exact conductivity distribution $\sigma^{ex} \in \mathcal{A}(\Omega)$ is a priori assumed to be of the form,

$$\begin{aligned} \sigma^{ex}(x) &= \sigma^{ex}_{bg}(x) + \delta^{ex}(x), & (5.2) \\ \sigma^{ex}_{bg}(x) &= 1, \quad \text{for all } x \in \Omega, & (5.3) \\ \delta^{ex}(x) &= 1 + t, \quad \text{for all } x \in B(\mathbf{c}, r), & (5.4) \end{aligned}$$

where $\mathbf{c} = (c_1, c_2) \in \mathbb{R}^2, t, r \in \mathbb{R}$ are unknown constants. The task is to find out the quadruple r, t, c_1, c_2. We denote

$$\sigma^{ex} \widehat{=} \begin{pmatrix} r \\ t \\ c_1 \\ c_2 \end{pmatrix} \in \mathbb{R}^4. \tag{5.5}$$

Please note that in the computations the conductivity distribution is a pixelwise constant function. Thus, it is not feasible to exactly implement this model, but the circular objects are approximated in the framework of the triangulation.

An imaginable real life application analogous to this scheme could be detecting a tumor from breast tissue, where the background conductivity is close to a constant. We explore both the least-squares approximation and statistical modelling.

5.1.1 Setup

The triangulation $\mathcal{T}_h = \{T_m\}_{m=1}^M$ that is used in the following computations is illustrated in (figure 5.1). A uniform triangulation is used, i.e. all the triangles are nearly equal in size. The variation of the potential distribution is fastest near to the boundary, due to which the triangular mesh is often refined towards the boundary in order to make the discretion error small. However, the number of electrodes is in this case so small that no refinement is made.

In the computations, the true conductivity distribution is

$$\sigma^{ex} \widehat{=} \begin{pmatrix} 0.125 & -0.9 & 0.5 & 0.2 \end{pmatrix}^T \tag{5.6}$$

Since only discrete problems can be solved numerically, we generate the measurement data $\mathbf{V}$ by approximating σ^{ex} with a function that is pixelwise constant in a refined triangulation (figure 5.1) denoted as $\mathcal{T}_{h/2}$. This is obtained simply by dividing each triangle of $\mathcal{T}_h$ into four subtriangles. Depending on the connection, we use the super index *ex* to refer either to a $\mathcal{A}(\Omega)$ function defined by (5.2)-(5.4), to its $H_{h/2}$-approximation or to a vector in $\mathbb{R}^{4M}$. The $H_{h/2}$-approximation of the exact conductivity distribution is illustrated in figure 5.2. The data $\mathbf{V}$ is obtained by solving the forward problem (2.19) corresponding to the approximation. We add to the measurements gaussian zero mean random noise $N \sim N(0, 10^{-6}I)$ so that

$$\frac{||N||_{\text{fro}}}{\max_{i,j} V_{ij}} \approx 0.1\%, \quad \frac{||N||_{\text{fro}}}{||U(\sigma^{ex}) - U(\sigma^{ex}_{bg})||_{\text{fro}}} \approx 6.5\% \tag{5.7}$$

The data is generated with the aid of the refined mesh, since otherwise the solver would be likely to find 'too good' candidate solutions. Using the same mesh in both generating the data and solving the inverse problem is known as committing an *inverse crime*. The discretion error is measured as

$$\frac{||U(\sigma_{bg}) - U(\sigma^{ex}_{bg})||_{fro}}{||U(\sigma^{ex}) - U(\sigma^{ex}_{bg})||_{fro}} \approx 35\%, \tag{5.8}$$

5.2 Least-Squares Approximation

In order to get some idea how accurate solutions can obtained through the least-squares approximation, we explore the NOSER method

$$\sigma^{(1)} = \sigma^{(0)} + (H + \alpha \text{diag}(H))^{-1} g, \tag{5.9}$$

and a simple Gauss-Newton reconstruction, which is obtained by performing one step of Gauss-Newton iteration. Since the measurements are assumed to be independent

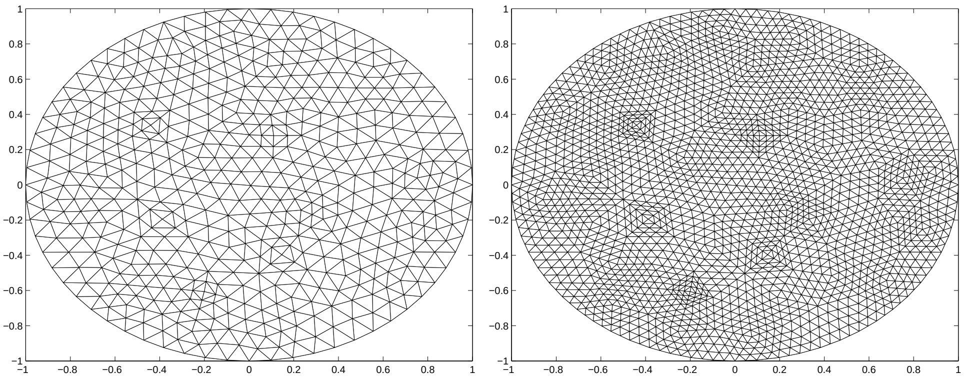

Figure 5.1: The triangulation $\mathcal{T}_h$ (left) and the refined mesh $\mathcal{T}_{h/2}$ (right), which is used when generating the data $\mathbf{V}$.

we choose $W = I$ and the algorithm is written as

$$\begin{aligned}
\sigma^{(1)} &= \sigma^{(0)} - \lambda(H_\alpha^{(0)})^{-1} g^{(0)}, \\
H_\alpha^{(0)} &= (J^{(0)})^T J^{(0)} + \frac{1}{2}\alpha D^2 A(\sigma^{(0)}), \\
g^{(0)} &= (J^{(0)})^T(\mathbf{U}(\sigma^{(0)}) - \mathbf{V}) + \frac{1}{2}\alpha D A(\sigma^{(0)}). \qquad (5.10)
\end{aligned}$$

In general, taking more than just one step seldom leads to considerably better estimates. In this connection, taking one step seems to be enough.

5.2.1 Computation of the Jacobian Matrix

Both algorithms are based on the use of the Jacobian matrix

$$J = \begin{pmatrix} \frac{U^{(1)}}{\partial \sigma_1} & \cdots & \frac{\partial U^{(1)}}{\partial \sigma_M} \\ \vdots & \ddots & \vdots \\ \frac{U^{(K)}}{\partial \sigma_1} & \cdots & \frac{\partial U^{(K)}}{\partial \sigma_M}, \end{pmatrix} \qquad (5.11)$$

which is obtained by writing $B_{i,j}$ of (2.23) as

$$\begin{aligned}
B_{i,j} &= \sum_{m=1}^{M} \sigma_m K_{i,j}^{(m)} + \sum_{\ell=1}^{L} \frac{1}{z_\ell} \int_{e_\ell} \varphi_i \varphi_j \, dS, \qquad (5.12) \\
K_{i,j}^{(m)} &= \int_\Omega \chi_m \nabla\varphi_i \cdot \nabla\varphi_j \, dxdy. \qquad (5.13)
\end{aligned}$$

Differentiation of equation (2.19) with respect to σ_m results in

$$\frac{\partial A}{\partial \sigma_m} F + A \frac{\partial F}{\partial \sigma_m} = 0. \qquad (5.14)$$

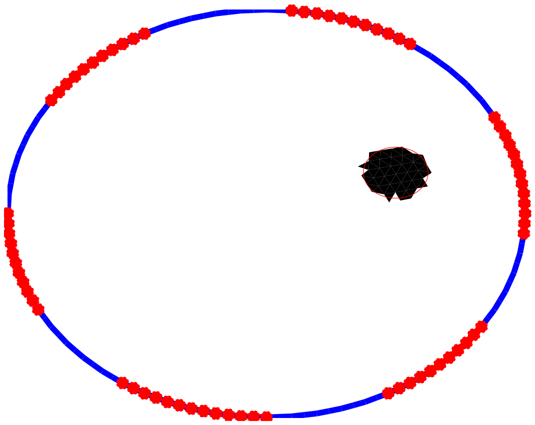

Figure 5.2: $H_{h/2}$-approximation of the exact conductivity distribution $\sigma^{ex} \widehat{=} (0.125, -0.9, 0.5, 0.2)^T$. The red circle denotes the right size and location.

Using the relation $X = A^{-1}F$, we obtain

$$\begin{aligned} \frac{\partial F}{\partial \sigma_m} &= -A^{-1}\frac{\partial A}{\partial \sigma_m}F, & (5.15)\\ \frac{\partial A}{\partial \sigma_m} &= \begin{pmatrix} -\sigma_m^2 K^{(m)} & 0 \\ 0 & 0 \end{pmatrix}. & (5.16) \end{aligned}$$

The electrode voltages $U^{(k)}$ are obtained as in (2.27);that is

$$U^{(k)} = \begin{pmatrix} 0 & \mathcal{C} \end{pmatrix} f^{(k)} \tag{5.17}$$

Differentiating this and using (5.15) yields

$$\frac{\partial U^{(k)}}{\partial \sigma_m} = -\begin{pmatrix} 0 & \mathcal{C} \end{pmatrix} A^{-1}\frac{\partial A}{\partial \sigma_m} f^{(k)}, \tag{5.18}$$

Since there is no sense in evaluating a whole inverse matrix the Jacobian is attained in practice by first solving $A^{-1}\begin{pmatrix} 0 & \mathcal{C}^T \end{pmatrix}^T$ and $A^{-1}F$ after which the partial derivatives are given by (5.18).

5.2.2 Smoothness Regularization

In the Gauss-Newton reconstruction we use $A(\sigma)$ of the form

$$A(\sigma) = |||\sigma|||^2_{\gamma,k} = \sigma^T B^k_\gamma \sigma \tag{5.19}$$

where $\gamma > 0$, $k \in \mathbb{N}$ and B^k_γ is the kth exponent of the matrix $B_\gamma \in \mathbb{R}^{M\times M}$ which is obtained as

$$(B_\gamma)_{ij} = \int_\Omega \nabla\varphi^d_i \cdot \nabla\varphi^d_j \, dxdy + \gamma\delta_{ij}, \tag{5.20}$$

where δ_{ij} is the Kronecker's delta and $\{\varphi^d_1, \ldots, \varphi^d_M\}$ is the piecewise linear nodal basis of a delaunay triangulation $\mathcal{T}^d_h$ (i.e. a set of triangles such that no data points are contained in any triangle's circumcircle) that is generated with respect to the nodal basis formed by the set of circumcenters of the triangulation $\mathcal{T}_h$ (figure 5.3). Additionally, we define

$$S_h^d = \text{span}\{\varphi_1^d, \dots, \varphi_M^d\} \tag{5.21}$$

Apparently, B_γ is symmetric and positive definite. Each $\sigma \in H_h$ has an uniquely determined piecewise linear counterpart $\sigma^d \in S_h^d$ such that $\sigma - \sigma^d$ vanishes in the set of circumcenters $\{P_1, \dots, P_M\}$ of the triangulation $\mathcal{T}_h$. Thus, by identifying σ and σ^d as vectors in $\mathbb{R}^M$ we have $\sigma = (\sigma_1, \dots, \sigma_M) = (\sigma_1^d, \dots, \sigma_M^d)$ and

$$\begin{aligned} |||\sigma|||_{\gamma,1} = \sigma^T B_\gamma \sigma &= \sum_{i,j=1}^{M} \sigma_i \sigma_j \nabla\varphi_i^d \cdot \nabla\varphi_j^d \, dxdy + \gamma||\sigma||^2 \\ &= \sum_{i,j=1}^{M} \sigma_i^d \sigma_j^d \nabla\varphi_i^d \cdot \nabla\varphi_j^d \, dxdy + \gamma||\sigma^d||^2 \\ &= ||\nabla\sigma^d||^2 + \gamma||\sigma^d||^2. \end{aligned} \tag{5.22}$$

Hence, we see that $A(\sigma)$ is small provided that the S_h^d counterpart σ^d is smooth in the sense of the norm $||\nabla\sigma^d||$. Moreover, decreasing the value of γ increases the importance of smoothness. We call 5.19 smoothness regularization. By noticing that $|||\sigma|||_{\gamma,1}^2 \geq \gamma||\sigma||$ and by writing

$$|||\sigma|||_{\gamma,k}^2 = \big(\sum_i c_i z_j\big)^T \big(\sum_j c_i \lambda_i^k z_i\big) = \sum_l \lambda_l^k c_l^2 = |||\sigma|||_{\gamma,k/2}^2, \tag{5.23}$$

where λ_l is an eigenvalue of B_γ, z_l is the corresponding eigenvector and c_l is some constant, we can deduce that

$$\gamma^{-t/2}|||\sigma|||_{\gamma,t} \geq \gamma^{-s/2}|||\sigma|||_{\gamma,s}, \quad \text{for } t \geq s. \tag{5.24}$$

Hence, we see that $|||\sigma|||_{\gamma,k}$ increases while k increases. The larger is the value of k the stronger is the smoothing effect of the regularization.

In order to demonstrate the structures generated by the regularization method, we draw so-called *white noise* random samples $W \in \mathbb{R}^M, W \propto \mathcal{N}(0, I)$ and set $X = B_\gamma^{-k/2}W$. Then,

$$||W||^2 = W^T W = X^T B_\gamma^{k/2} B_\gamma^{k/2} X = X^T B_\gamma^k X = |||X|||_{\gamma,k}^2, \tag{5.25}$$

i.e. the white noise regularization and the smoothness regularization are of equal order. The random draws are plotted in figures 5.4 and 5.5, first columns of which are identical. The random structures in the 4th column of 5.4 are of same size but much smoother that in the 2nd column of 5.5. Thus, it seems that by varying the value of γ we can control the size of the structures whereas varying k seems to have greater effect on the level of smoothness.

The measured voltages are much less sensitive to the values of σ in the central parts of Ω than to the values close to the boundary $\partial\Omega$. In other words, the sensitivity of the EIT reconstruction to the measurement noise increases when moving towards the center of Ω. That is why increasing noise level arises a need to decrease the order of regularization towards the center. Achieving feasible results with high noise

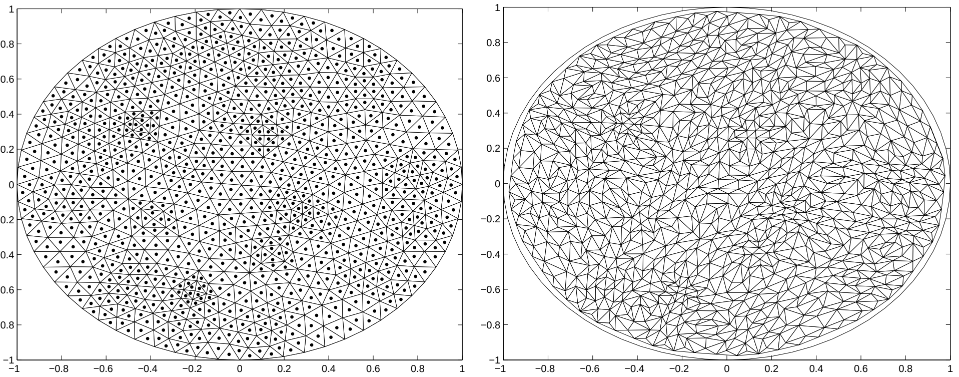

Figure 5.3: The set of circumcenters of the triangulation $\mathcal{T}_h$ (left) and the corresponding delaunay triangulation $\mathcal{T}_h^d$

levels seems to require for increasing 'stiffness' of the regularization in the vicinity of the boundary. Therefore, in the computations, we let the boundary block of B_γ be identity, i.e.

$$B_{ij} = \delta_{ij}, \quad \text{for } \{i, j \,|\, \mathrm{dist}(P_{ij}, \partial\widehat{\Omega}) < \epsilon\}, \tag{5.26}$$

where P_{ij} denotes a circumcenter of the triangle T_{ij}.

5.2.3 Results

Both NOSER and Gauss-Newton reconstruction yield very similar results. Both methods seem to give rather credible information about the location of the anomaly but the exact size and value of conductivity remain uncertain. The reconstructions are plotted in figure (5.6). On the basis of the figure the output of NOSER algorithm can be ranked as less regular. This is, apparently, due to its not so sophisticated regularization method, which does not assume anything of the structure of the triangulation.

Since least-squares reconstructions are easily computed and seem to localize the anomaly relatively confidentially, we use least-squares approximation as a method of determining a so-called *region of interest*, i.e. a subset $\mathcal{R}_{pr} \subset \widehat{\Omega}$ in which the anomaly lies with high reliability. The idea is to fasten the convergence of more complicated methods.

We determine the region of interest in a heuristic way.

$$\mathcal{R}_{pr} = \{x \in \widehat{\Omega} \,|\, |\sigma^{(1)}(x) - \sigma^{(0)}(x)| \geq \kappa\, \mathrm{std}\{\sigma^{(1)}\}\}, \tag{5.27}$$

where $\sigma^{(0)}$ is the initial guess, $\sigma^{(1)}$ the least squares solution, std$\{\sigma^{(1)}\}$ its standard deviation and $\kappa > 0$ some real-valued constant. This appeared to work pretty well in practice.

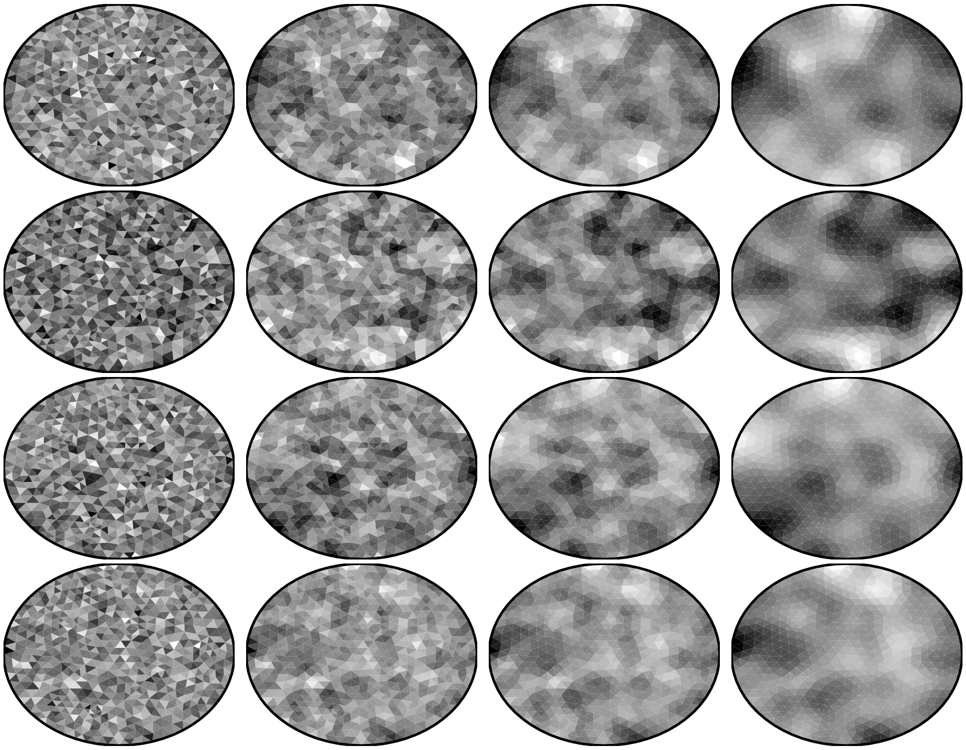

Figure 5.4: Four realizations of W (1th column) and the realizations of $X = B_1^{-k} W$ corresponding to $k = 2$ (2nd column), $k = 4$ (3rd column) and $k = 8$ (4th column).

5.3 Statistical Solution

In this section, the anomaly is sought by employing the Bayesian model and MCMC integration. Statistical solution refers in this case to a Monte Carlo approximation of the conditional expectation

$$\overline{\sigma}_m = \frac{\sigma^{(1)} + \cdots + \sigma^{(m)}}{m} \approx \int_{\mathbb{R}^M} \sigma \pi_{\text{post}}(\sigma)\, d\sigma \tag{5.28}$$

The idea is both to develop an efficient sampling technique and to compare statistical solutions with least-squares solutions.

In practice, achieving an reasonably accurate Monte Carlo estimate requires for heavy computation compared to least-squares approximation. Therefore, statistical solutions should be at least in some sense more precise than the corresponding least-squares solutions. Again, the applied sampling technique and linear algebra have a significant effect on the convergence rate and thereby the usability of the statistical approach.

The convergence of the statistical algorithms is studied both on the basis of autocorrelation series and by plotting estimates of the conditional expectation.

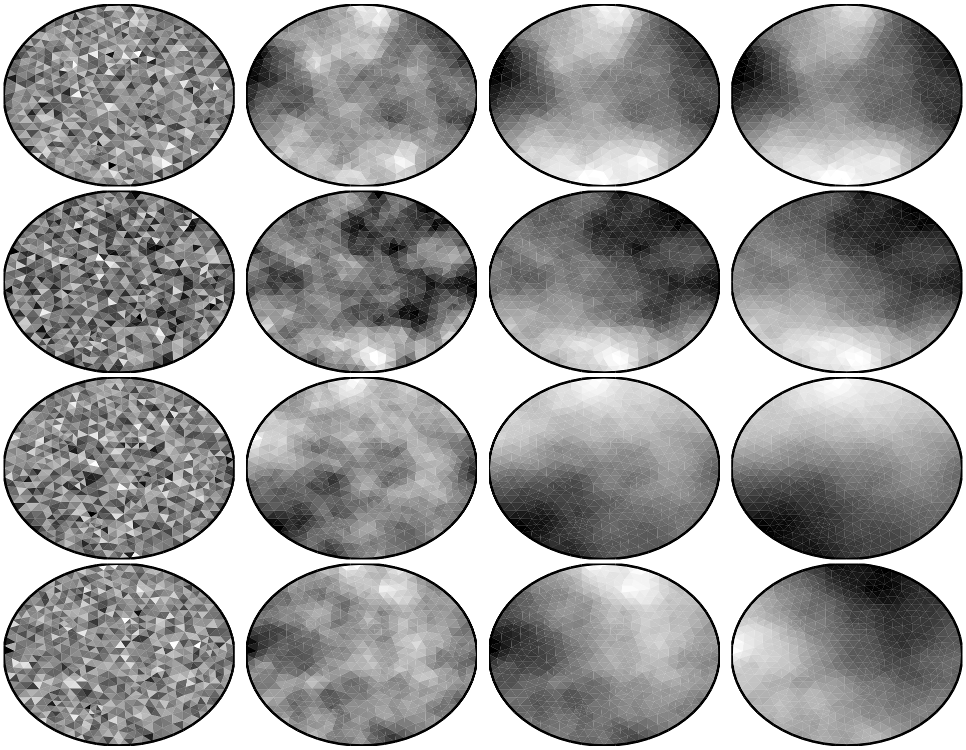

Figure 5.5: Four realizations of W (1th column) and the realizations of $X = B_\gamma^{-2}W$ corresponding to $\gamma = 0.1$ (2nd column), $\gamma = 0.01$ (3rd column) and $\gamma = 0.001$ (4th column).

5.3.1 Prior and Posterior Densities

We suppose that the anomaly is *a priori* known to be of the form (5.2) and lies somewhere in the region of interest $\mathcal{R}_{pr}$ that is determined based on the Gauss-Newton reconstruction as in (5.27). In other words, we hypothesize that the conductivity distribution can be written as in (5.5), i.e.

$$\sigma \widehat{=} \begin{pmatrix} r & t & c_1 & c_2 \end{pmatrix}^T, \tag{5.29}$$

where $r, \mathbf{c}$ and t are realizations of random variables $R, \mathbf{C}$ and T. We do not assume anything particular of the shape of the prior distribution (e.g. Gaussian distribution), but let R, $\mathbf{C}$ and T be independent and uniformly distributed, i.e. $R \sim \mathrm{Uniform}[0, \mathrm{diam}(\mathcal{R}_{pr})]$, $\mathbf{C} \sim \mathrm{Uniform}\{\mathcal{R}_{pr}\}$, $T \sim \mathrm{Uniform}(-1, 1]$ and

$$\pi_{\mathrm{pr}}(\sigma) \propto \pi_+(\sigma)\chi_{[0,\mathrm{diam}(\mathcal{R}_{pr})]}(r)\chi_{\mathcal{R}_{pr}}(\mathbf{c})\chi_{(-1,1]}(t), \tag{5.30}$$

where $\pi_+(\sigma)$ is a positivity prior defined in (2.58). In the computations, each sampled conductivity distribution is a H_h-function. Since there are no exactly circular shaped objects in H_h, we cannot draw samples exactly distributed as π_{pr}, since circles have to be approximated in the sense of H_h. Two H_h-approximations of random draws from π_{pr} are plotted in figure (5.8).

The posterior density is of the form (2.60), where the covariance matrix $C = \gamma^2 I$,

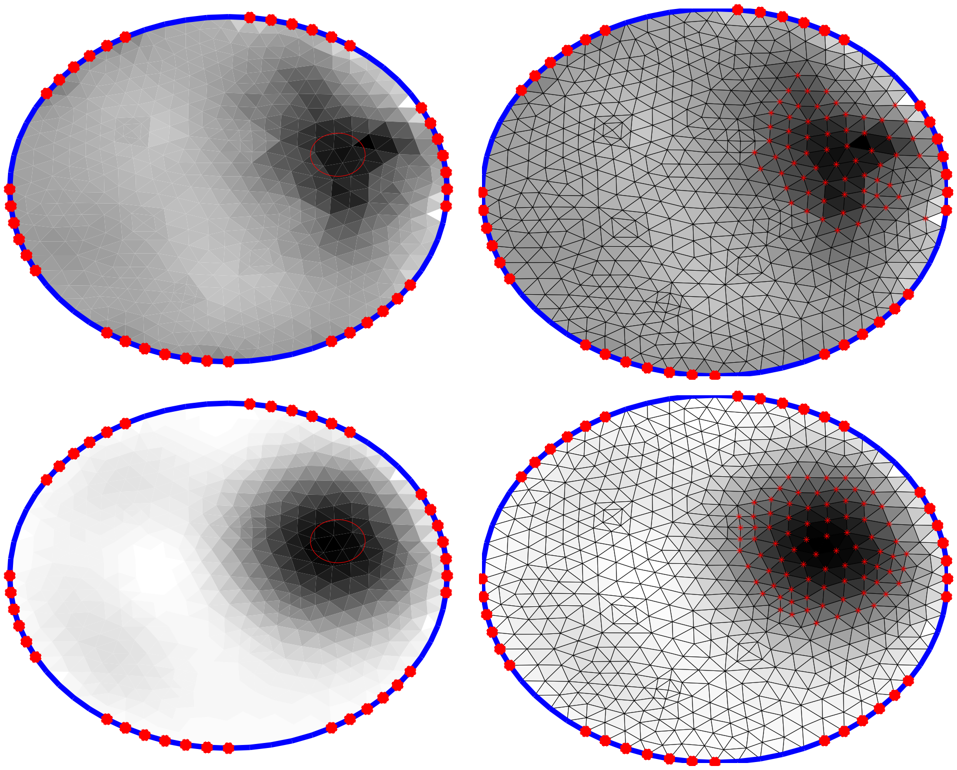

Figure 5.6: The NOSER approximation (top left), where $\alpha = 1.5 \cdot 10^{-6}$ *and the Gauss-Newton reconstruction (bottom left), where* $\alpha = 0.15$, $A(\cdot) = ||\cdot||^4_{B^6_{0.02}}$. *In both cases, the region of interest is determined as in (5.27) with* $\kappa = 2.2$. *(right column).*

since the electrode potentials are assumed to be independent. Thus, we have

$$\pi(\sigma \,|\, \mathbf{V}) \propto \pi_{\text{pr}} \exp\Big(-\frac{1}{2\gamma^2}(\mathbf{U}(\sigma) - \mathbf{V})^T(\mathbf{U}(\sigma) - \mathbf{V})\Big). \tag{5.31}$$

Since the problem is restricted as (5.30), the integration task (5.28) is only four dimensional. Therefore, MCMC integration is not necessarily needed in this particular case. However, a similar approach is workable also in more complex cases where the background conductivity σ_{bg} has to be included into the statistical model (e.g. some shell structured σ_{bg}).

5.3.2 Linear Algebra

As the computations are restricted to the region of interest we know a priori that the perturbations are small. Hence, each sampled conductivity can be represented as $\sigma = \sigma_{bg} + td$ so that the number of non-zeros in d is small. Thus, on the basis of discussion in sections 4.5 and 4.6 it is advantageous to solve the forward problem

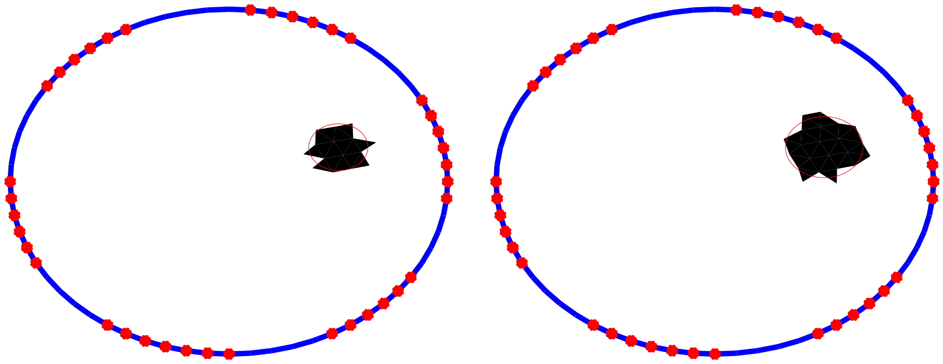

Figure 5.7: Two H_h-approximations of random draws from π_{pr}. The red circle shows the exact size and shape.

through the Sherman-Morrison-Woodbury -formula as

$$X_\sigma = X_{\sigma_{bg}} - tZ^d_{\sigma_{bg}}(I + tV_d^T Z^d_{\sigma_{bg}}\Lambda_d)^{-1}V_d^T X_{\sigma_{bg}}. \tag{5.32}$$

Since we correct each time the background solution $X_{\sigma_{bg}}$, the computational workload can be diminished by directly correcting the potential values as

$$U(\sigma) = U(\sigma_{bg}) - t\begin{pmatrix}0 & \mathcal{C}\end{pmatrix} Z^d_{\sigma_{bg}}(I + tV_d^T Z^d_{\sigma_{bg}}\Lambda_d)^{-1}V_d^T X_{\sigma_{bg}}, \tag{5.33}$$

where the product $\begin{pmatrix}0 & \mathcal{C}\end{pmatrix} Z^d_{\sigma_{bg}}$ can be calculated in advance. It is easy to see that in this case the computational effort is largely determined by the effort of evaluating $(I + tV_d^T Z^d_{\sigma_{bg}}\Lambda_d)^{-1} \in \mathbb{R}^{k\times k}$.

5.3.3 The Sampling Plan

The sampling plan is excessively straightforward. We choose

$$\sigma^{(0)} = \sigma_{bg} = 1. \tag{5.34}$$

Since σ_{bg} is known to be close to the exact distribution, we neglect the burn-in phase. The samples $\{\sigma^{(1)}, \dots, \sigma^{(m)}\}$ are generated in a single long run accepting all the generated samples.

The statistical efficiency is analyzed on the basis of values $\sigma^{(j)}(x^\spadesuit)$, $\sigma^{(j)}(x^\diamondsuit)$, $\sigma^{(j)}(x^*)$, $\sigma^{(j)}(x^\heartsuit)$ and $\sigma^{(j)}(x^\clubsuit)$, $j = 1, 2, \dots, m$, where m is the size of the sample. The points $x^\spadesuit$, $x^\diamondsuit$, x^*, $x^\heartsuit$ and $x^\clubsuit$ are plotted in figure (5.3.3).

Autocorrelation of the sample ensemble is estimated as

$$\rho_j(x) = \mathrm{corr}\{\sigma^{(1)}(x), \sigma^{(j+1)}(x)\} \approx \tilde{\rho}_j(x) = \tilde{\gamma}_j(x)/\tilde{\gamma}_0(x) \tag{5.35}$$

where $\tilde{\gamma}_j(x)$ is an estimator of the autocovariance calculated as

$$\tilde{\gamma}_j(x) = \frac{1}{m}\sum_{i=1}^{m-j}(\sigma^{(i)}(x) - \overline{\sigma}_m(x))(\sigma^{(i+j)}(x) - \overline{\sigma}_m(x)). \tag{5.36}$$

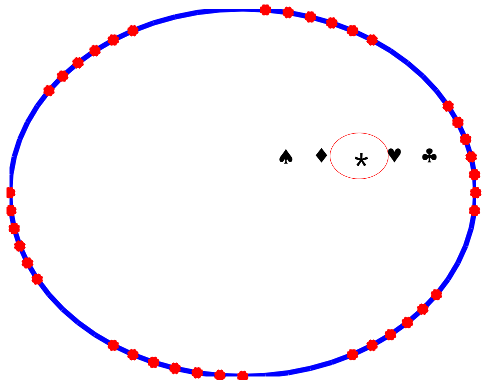

Figure 5.8: Locations of points $x^{\spadesuit}, x^{\diamondsuit}, x^*, x^{\heartsuit}$ *and* $x^{\clubsuit}$.

So as to be able to compare the algorithm efficiencies we approximate the integrated autocorrelation time as

$$\tau_{int}(\sigma) \approx \frac{1}{2} + \sum_{j=1}^{\infty} |\tilde{\rho}_j| \tag{5.37}$$

5.3.4 Metropolized Independence Sampler (MIS)

At first, we apply the MIS sampling method introduced in section 3.3.1, since it is easily implemented and all the proposed moves are independent. The proposal distribution is chosen to be

$$g(\cdot) = \pi_{\text{pr}}(\cdot), \tag{5.38}$$

where π_{pr} is as in (5.30). Performing one step of the algorithm is simply to draw σ from prior distribution and to check the acceptance-rejection condition.

5.3.5 Random-walk Metropolis

Another implemented algorithm is the random-walk Metropolis, one of the most often used MCMC algorithms. Since we do not have much information about the structure of the posterior density, the proposal is chosen to be spherically symmetric Gaussian distribution similarly as in algorithm 3.3.2.

$$g_{var} \sim N(0, \gamma^2 I), \tag{5.39}$$

where I is identity matrix in $\mathbb{R}^{4\times4}$. In the random-walk Metropolis, the proposed moves are not independent from each other. The advance of the method is that by varying the step size user can efficiently control the acceptance rate.

5.3.6 Correlated Multipoint Proposals

To give an example of slightly more complicated algorithm based on the Metropolis transition rule, we implement the multipoint method, i.e. algorithm 3.3.6, applied for two proposal functions. Again, we employ spherically symmetric Gaussian distributions as proposals; that is, we draw

$$\epsilon_1 \sim N(0, \gamma_1^2 I) \tag{5.40}$$

$$\epsilon_2 \sim N(0, \gamma_2^2 D), \tag{5.41}$$

where $D = \mathrm{diag}(1, 1, 0, 0)$ and set $y_1 = x + \epsilon_1$, $y_2 = x + \mathrm{sign}(\epsilon_2)||\epsilon_2||$. The weight functions λ_1 and λ_2 are constants. The idea is to propose two moves of different sizes on each step. Due to the shape of the posterior distribution the longer moves are restricted to the first and third quadrant of the rt-plane. The probability of accepting a large move is increased by choosing $\lambda_2 >> \lambda_1$.

5.3.7 Surrogate Transitions

The workability of surrogate transition method is also experimented by examining the accurateness of the approximation

$$U(\sigma) \approx U^*(\sigma) = U(\sigma) + DU(\sigma_0)(\sigma - \sigma_0) \tag{5.42}$$

In this demonstrative case, the Sherman-Morrison-Woodbury -formula functions so well, that applying (5.42) is not reasonable. In cases, where the sampler perturbs the conductivity distribution more globally, the surrogate transition method can, however, be of great importance.

5.3.8 Results

Although the proposed samples in MIS are independent, the acceptance ratio is exceedingly low. Apparently, this is because the variance of the posterior density is small compared to the variance of the prior density. Convergence of the method is plotted in figure (5.9).

The random-walk metropolis algorithm was implemented by choosing $\gamma^2 = 0.02$, that resulted in acceptance rate of 29%. Although the acceptance rate is feasible the overall level of movement remains slow, which is illustrated by the figure (5.10). The autocorrelation curves indicate that the convergence is exceptionally slow close to the most important parts of the distribution. This can be explained to be due to the awkward shape of π_{post}.

The posterior distribution is banana shaped in the rt-plane (i.e. the plane, where the center of the anomaly is fixed) and suffers from local minima, which is illustrated by the figure (5.17). According to [2], the random-walk metropolis commonly fails in banana shaped distributions. Finding an appropriate step size is often impossible, since small enough step sizes tend result in slow movement of the chain. Banana structure results from nonlinearity of the map $U(\cdot)$ and low variance of $||V - U(\cdot)||$. Local minima are due to the fact that the inverse problem is outstandingly ill-conditioned in the rt-plane; that is, various combinations of r- and t-values result in electrode voltages very close to the measured voltage values. Figure (5.18) shows how two concentric anomalies unequal in size perturb the potential distribution in a similar way.

Multipoint method was implemented in order to allow random-walk to take larger steps in rt-plane and thereby increase the level of general movement. The parameters were chosen as

$$\begin{array}{rclcrcl} \gamma_1^2 & = & 0.02 & & \lambda_1 & = & 1 \\ \gamma_2^2 & = & 0.1 & & \lambda_2 & = & 1000. \end{array} \tag{5.43}$$

In the implementation the acceptance rate was 30% 7% of all the accepted moves being longer moves. Figure (5.11) illustrates that the level of global movement is indeed increased compared to the simple random walk. Autocorrelation curves indicate more balanced behavior. It is also clearly seen that the algorithm does not converge to the exact solution. Monte Carlo estimates after 10000 and 50000 are plotted in the figure (5.3.8). The images are close to identical. Thus, it seems that 10000 is a large enough sample size.

Interestingly, the random walk metropolis yields fairly reasonable solutions and rapid convergence rates provided that either r or t is fixed to its true value. This is illustrated by figures (5.12), (5.13) and (5.3.8). Yet, accuracy of the obtained estimates diminishes remarkably if the forward problem is solved only approximately through the linearization (5.42).

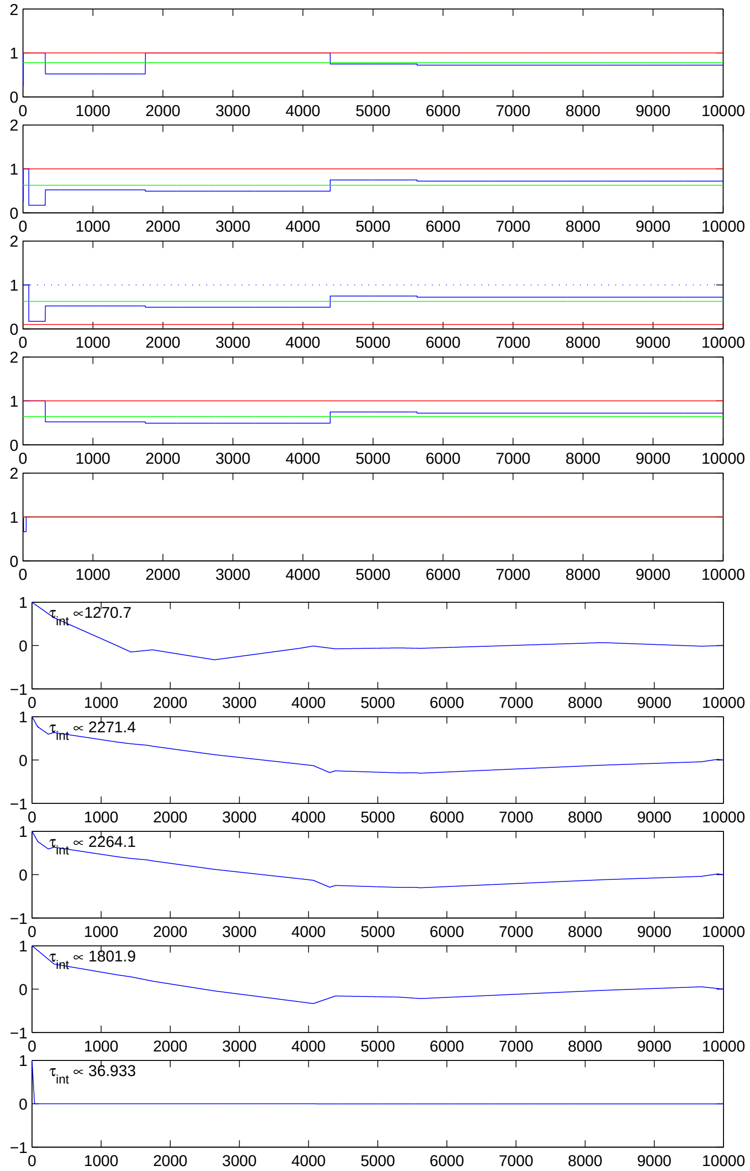

Figure 5.9: MIS. The acceptance rate is extremely low. From top to bottom, values $\sigma^{(j)}(x^{\spadesuit})$, $\sigma^{(j)}(x^{\diamondsuit})$, $\sigma^{(j)}(x^{*})$, $\sigma^{(j)}(x^{\heartsuit})$ and $\sigma(x^{\clubsuit})$, $j = 1, \ldots, 10000$ (left) and the corresponding autocorrelation curves (right).

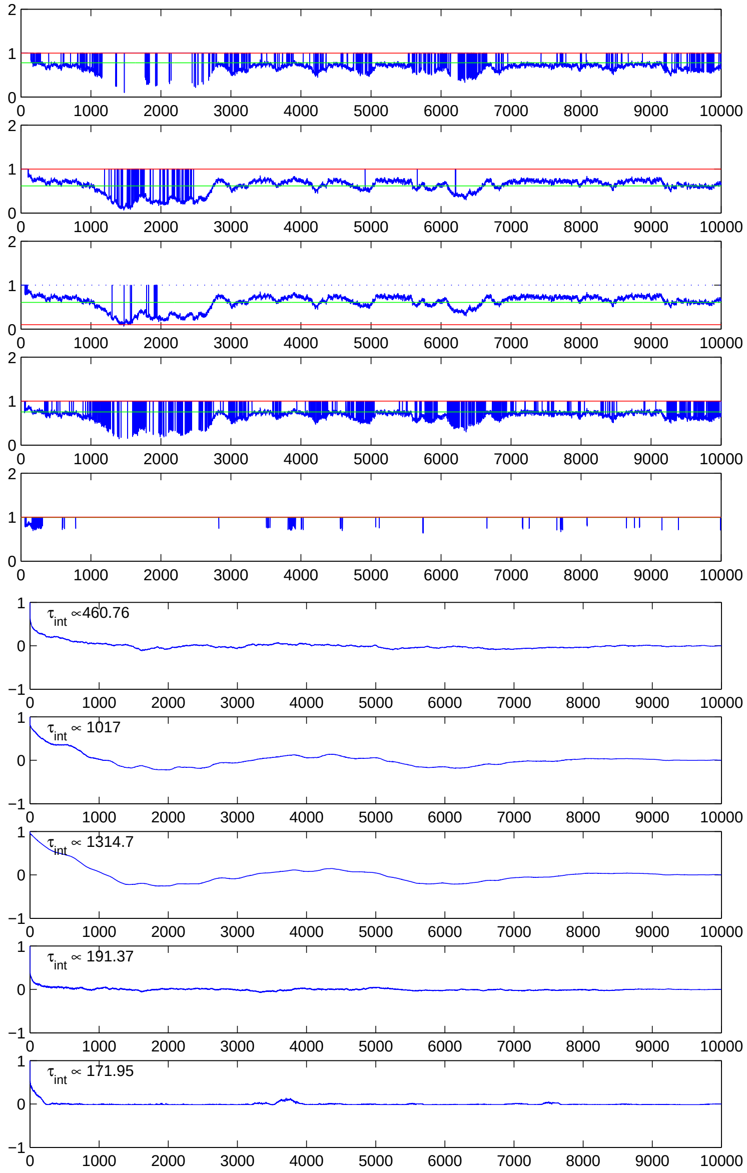

Figure 5.10: Random walk metropolis. Due to the shape of the posterior distribution choosing a feasible step size is difficult. In this case, the acceptance rate is sufficient but the overall movement of the chain is relatively slow. The autocorrelation curves indicate that the level of independency remains low. From top to bottom, values $\sigma^{(j)}(x^{\spadesuit})$, $\sigma^{(j)}(x^{\diamondsuit})$, $\sigma^{(j)}(x^{*})$, $\sigma^{(j)}(x^{\heartsuit})$ and $\sigma^{(j)}(x^{\clubsuit})$, $j = 1, \ldots, 10000$ (left) and the corresponding autocorrelation curves (right).

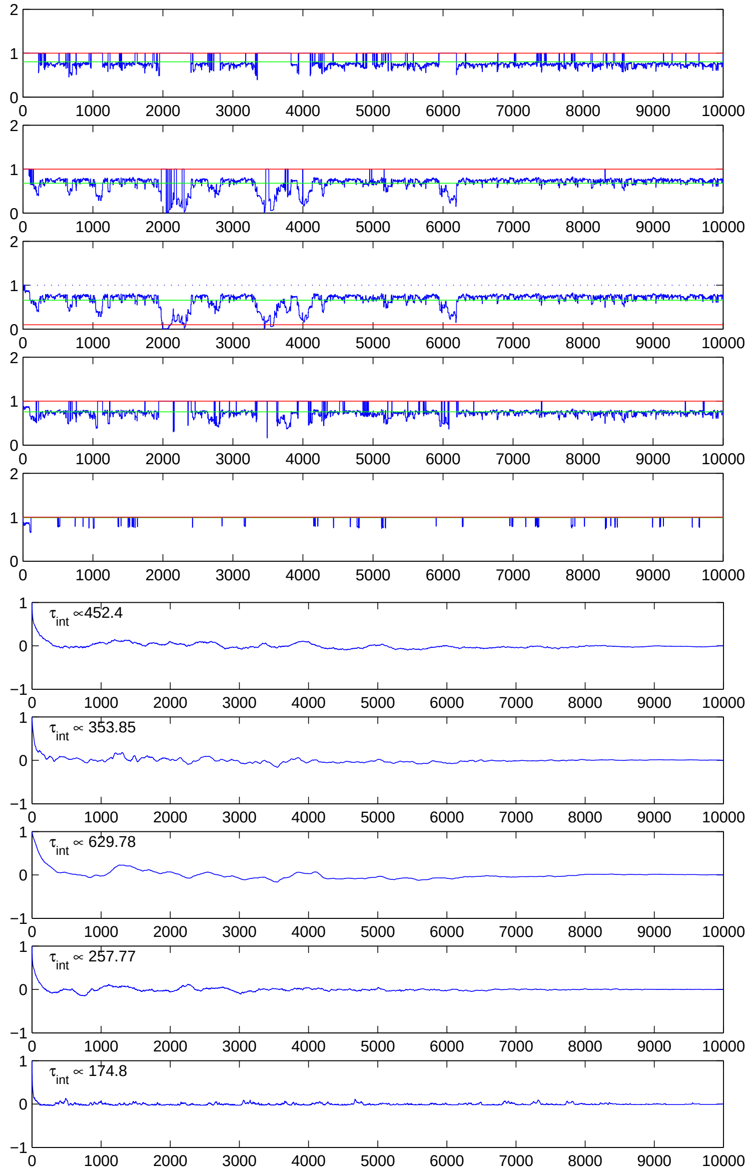

Figure 5.11: Multipoint method. The overall movement is faster than i random walk algorithm, which is indicated by the autocorrelation curves. It is clearly seen, that the estimate does not approach the exact solution due to the inadequate a priori information. From top to bottom, values $\sigma^{(j)}(x^{\spadesuit})$, $\sigma^{(j)}(x^{\diamondsuit})$, $\sigma^{(j)}(x^{*})$, $\sigma^{(j)}(x^{\heartsuit})$ *and* $\sigma^{(j)}(x^{\clubsuit})$, $j = 1, \ldots, 10000$ *(left) and the corresponding autocorrelation curves (right).*

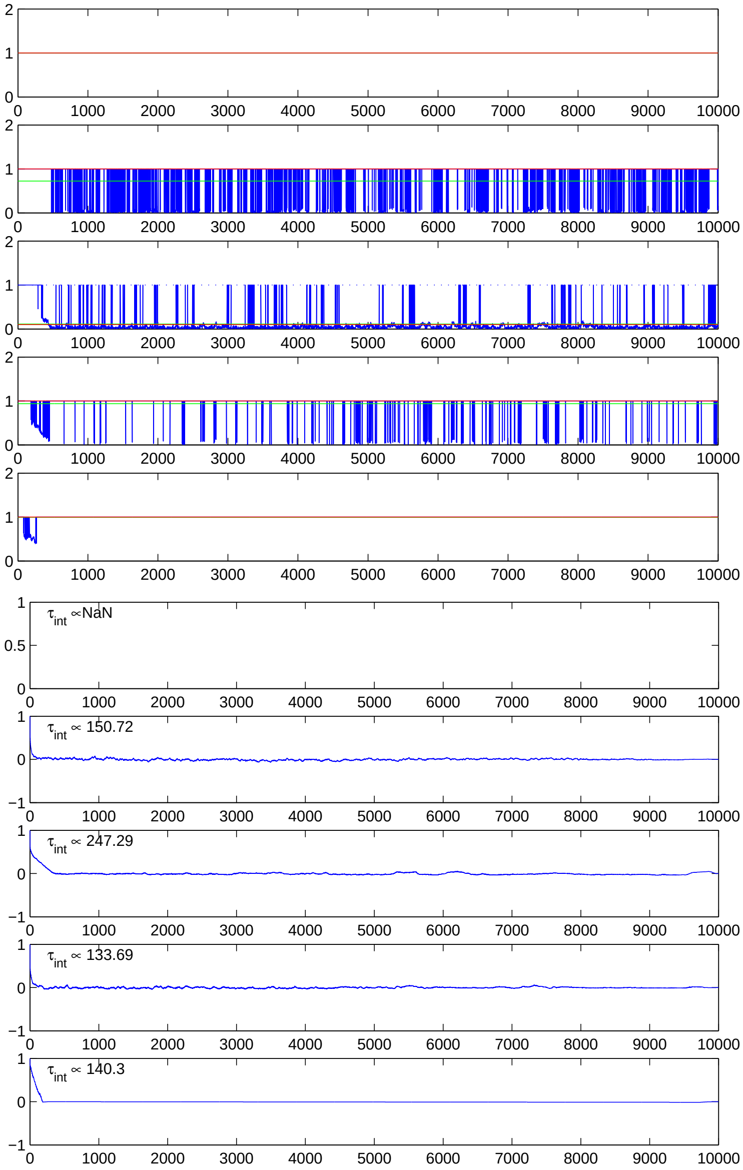

Figure 5.12: Random walk metropolis. The radius of the anomaly is fixed to the exact value $r = 0.125$ due to which the algorithm converges rapidly. From top to bottom, values $\sigma^{(j)}(x^{\spadesuit})$, $\sigma^{(j)}(x^{\diamondsuit})$, $\sigma^{(j)}(x^{*})$, $\sigma^{(j)}(x^{\heartsuit})$ and $\sigma^{(j)}(x^{\clubsuit})$, $j = 1, \ldots, 10000$ (left) and the corresponding autocorrelation curves (right).

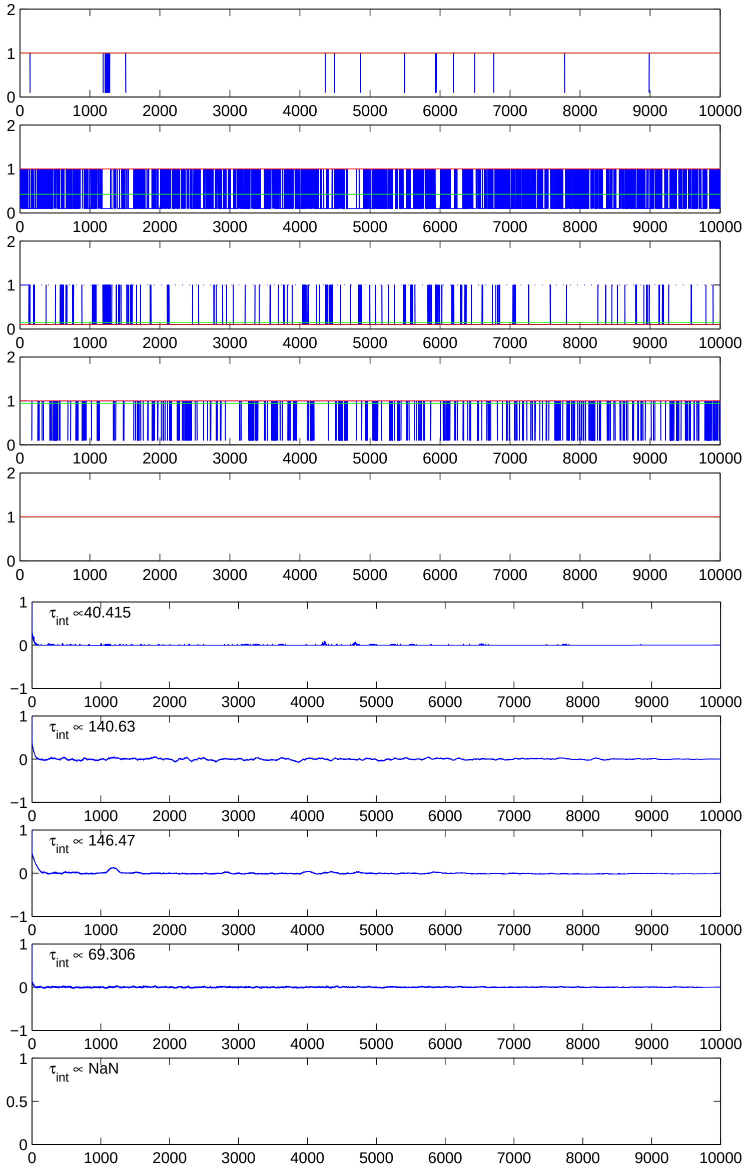

Figure 5.13: Random walk metropolis. t is fixed to the exact value $t = -0.9$. Again, the algorithm converges rapidly. From top to bottom, values $\sigma^{(j)}(x^{\spadesuit})$, $\sigma^{(j)}(x^{\diamondsuit})$, $\sigma^{(j)}(x^{*})$, $\sigma^{(j)}(x^{\heartsuit})$ and $\sigma^{(j)}(x^{\clubsuit})$, $j = 1, \ldots, 10000$ (left) and the corresponding autocorrelation curves (right).

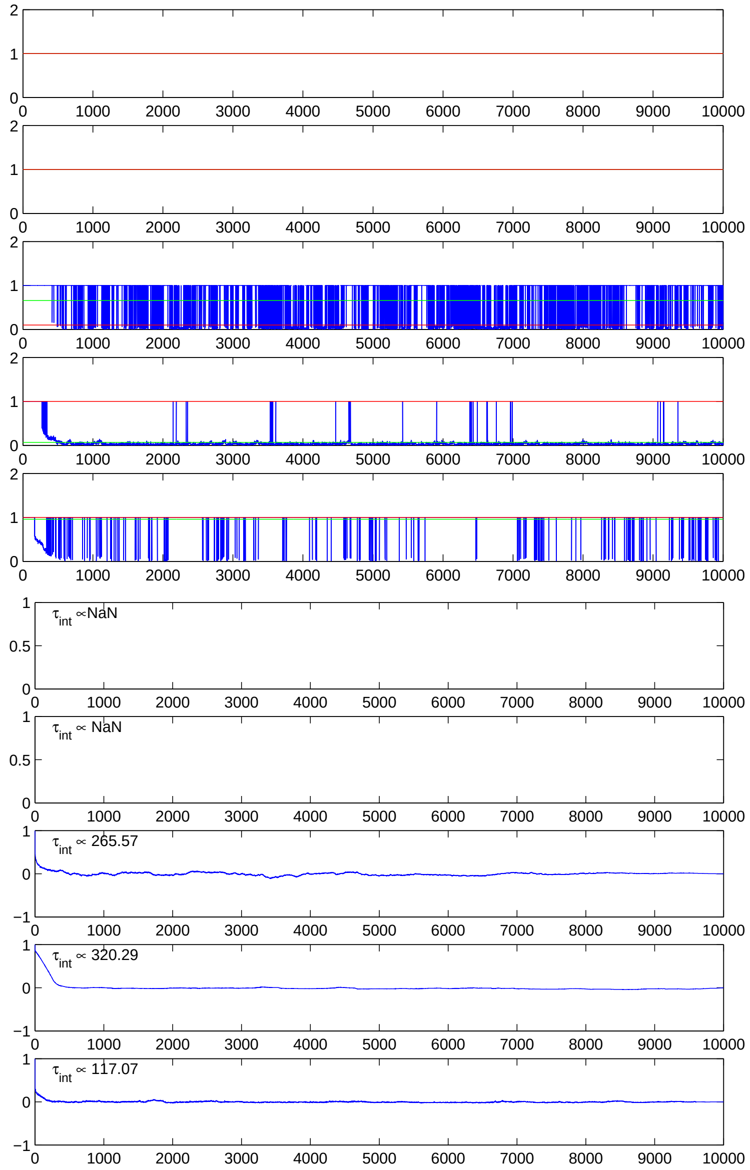

Figure 5.14: Surrogate Markov chain generated through the approximation (3.45). From top to bottom, values $\sigma^{(j)}(x^{\spadesuit})$, $\sigma^{(j)}(x^{\diamondsuit})$, $\sigma^{(j)}(x^{*})$, $\sigma^{(j)}(x^{\heartsuit})$ and $\sigma^{(j)}(x^{\clubsuit})$, $j = 1, \ldots, 10000$ (left) and the corresponding autocorrelation curves (right).

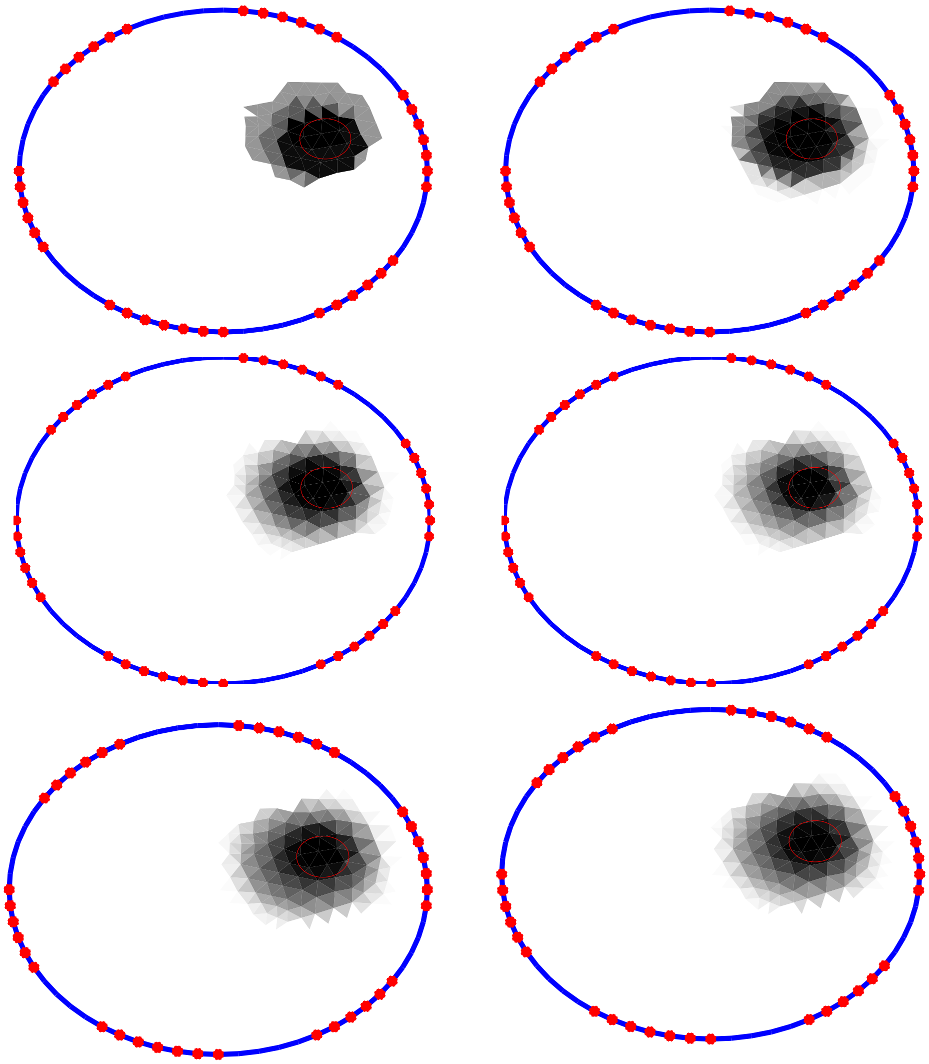

Figure 5.15: Monte Carlo estimates after 10000 *samples (left) and 50000 samples (right). The algorithms from top to bottom: MIS, random walk metropolis, multipoint method. MIS estimates (1st row) differ noticeably from each other, which indicates that the algorithm has not yet converged. In contrast, estimates attained through multipoint method (3rd row) are nearly indentical.*

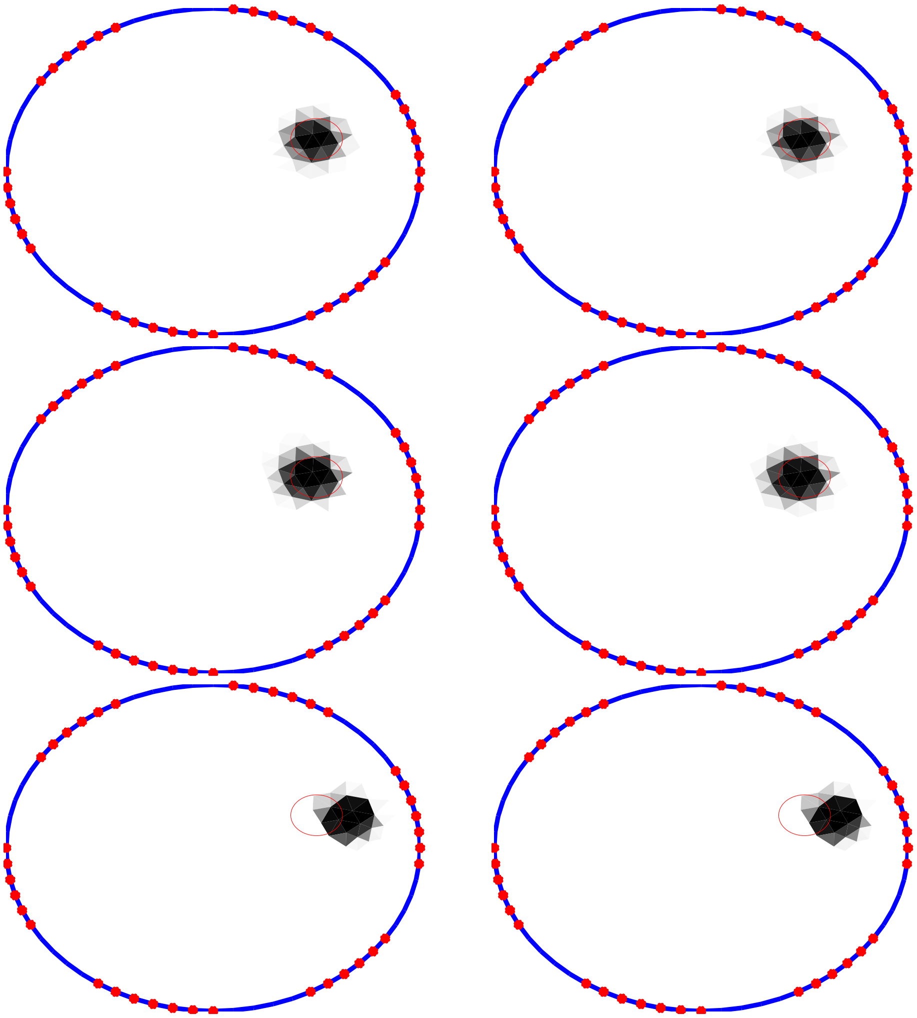

Figure 5.16: Monte Carlo estimates after 10000 *samples (left) and 50000 samples (right). From top to bottom: random walk metropolis with r fixed, random walk metropolis with t fixed, random walk conditional mean estimate based on a surrogate Markov chain produced by the linearization* (3.45)*. Upper two random walks produce rather substantial estimates. When the forward problem is solved using* (3.45) *(3rd row) the anomaly is mislocated.*

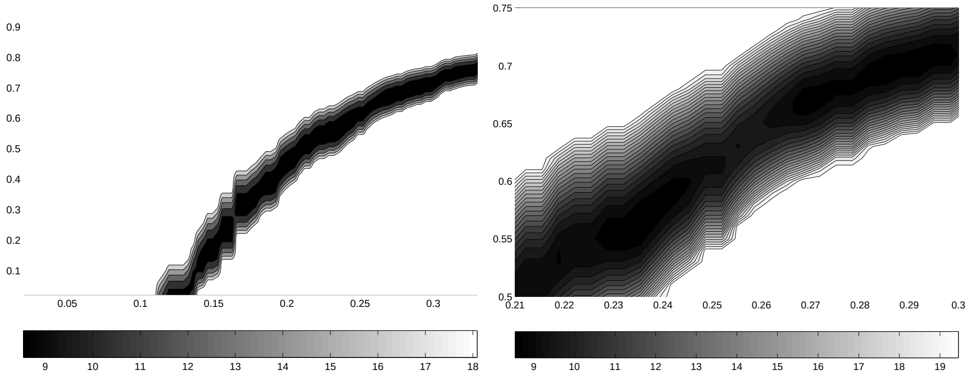

Figure 5.17: Contour plot of the posterior distribution $\pi_{post}(\sigma)$, $\sigma \widehat{=} \{r,(0.5;0.2),t\}$, *i.e. the center of the anomaly is fixed to its exact value. Radius (x-axis) and the value of conductivity (y-axis) are varied. The distribution is banana-shaped (left) and suffers from local minima (right) which explains the slow movement of the random walk sampler.*

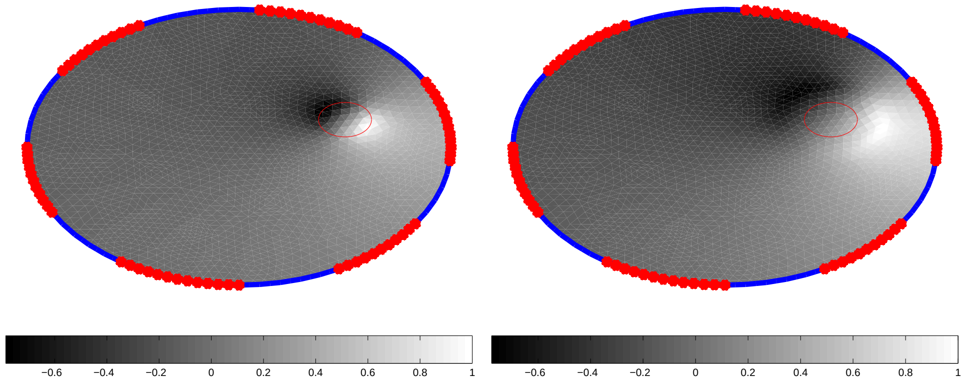

Figure 5.18: Two illustrations of how the potential distribution corresponding to $\sigma = \sigma_{bg}$ *is perturbed after adding a circular anomaly. Although the added anomalies are of different size both cases yield very similar electrode potentials.*

Chapter 6

Discussion

In the numerical examples, both least-squares approximation and Bayesian modeling succeed rather well in locating the anomaly, i.e. in finding $\mathbf{c}$, but the two other sought quantities r and t remain uncertain. In cases where either r or t is fixed to the true value the conditional mean estimate is fairly close to the exact solution in eyeball norm. Yet, modifying the statistical model by solving the discrete forward problem approximately through (3.45) diminishes the accuracy remarkably.

It seems that the more one has a priori knowledge of σ the more preferable is the statistical approach. A priori information can more easily be decoded into a prior distribution than into a regularizing functional, since implementing the Gauss-Newton algorithm by applying any regularization method favoring discontinuous conductivities, such as anomalies of certain size and shape, is problematic, since Gauss-Newton is based on differentiability of the map $\sigma \to \mathcal{A}(\sigma)$. Again, obtaining any confidence intervals of the Gauss-Newton reconstruction is difficult since there is no strict statistical interpretation of the method. Therefore, achieving a practicable least-squares estimate is more or less an art of fixing the free-floating parameters so that the outcome is close to the optimal. The result is often a compromise between smoothness and resolution. In the demonstrated cases, the numerical least-squares estimates are rather comparable to their counterparts obtained through the statistical modelling as far as both r and t are to be solved. Only assuming either of the quantities to be given caused a distinct difference between least-squares and conditional mean estimates.

Although the statistical problem is restricted to seeking the anomaly from a reasonably small region of interest $\mathcal{R}_{pr} \subset \Omega$, applied MCMC algorithm has a strong effect on the converge rate of the Monte Carlo estimate. The explanation is the awkward characteristics of the target distribution. In the demonstrations, the posterior distribution suffers from low variance, generic shape of a banana as well as a number of local minima. On the basis of the results it is clear that all of these properties have to be taken into account so as to construct an efficient proposal function; that is, the sampler has to be adapted to follow both the local and global dynamics of the target distribution. Due to the nonlinearity and strong ill-conditioned nature of the inverse problem it is apparent that the posterior distribution is more complex in cases where σ_{bg} is not a constant function. Thus, also the structure of the background conductivity sets requirements for the sampler.

Since the conductivity distribution is updated only in the region of interest, the exact solution of the forward problem is easily obtained by employing the Sherman-Morrison-Woodbury –formula. More global updates would require for more costly linear algebra. Since computational cost of the linear approximation (3.45) is independent from the structure of σ, it is interesting to study whether it is possible to use it as a substitute for the exact solution or in generating surrogate Markov chains introduced in section 3.6. The numerical experiments indicate that employing the linearization as a pure substitute diminishes the accuracy of the conditional mean estimate considerably. The anomaly is distinctly mislocated even though its size is given. Due to the clear mislocation applying the approximation to generate surrogate Markov chains does not seem a reasonable idea: each time the actual chain would come close to the true location, the surrogate chain would be likely to drift away from it. Therefore, the most efficient way to derive benefit from (3.45) appears to be using it as a pure substitute for the solution while running the burn-in phase, yet, performing the actual sampling through rigorous linear algebra.

In the experiments, conception of the anomaly is analogous to an electromagnetic dipole with unknown length and charge within a vacuum cavity. Writing the equation (2.1) as

$$\nabla \cdot (\sigma_{bg} + \delta)\nabla(u_{bg} + u_\delta) = 0 \tag{6.1}$$

and noting that $\nabla \cdot \sigma_{bg} \nabla u_{bg} = 0$ yields

$$\nabla \cdot (\sigma_{bg} + \delta)\nabla u_\delta = -\nabla \cdot \delta \nabla u_{bg}. \tag{6.2}$$

Since the support of the perturbation δ is small it seems reasonable to assume that adding δ to the background conductivity does not affect greatly the directions of the currents within Ω. Thus, we estimate $\nabla u_\delta \approx c\nabla u_{bg}$, where c is some scalar-valued function. Substitution to (6.2) plus a slight manipulation gives

$$\Delta u_\delta \approx -\frac{1+c}{\sigma_{bg}} \nabla \cdot \delta \nabla u_{bg}, \tag{6.3}$$

where σ_{bg} has been treated as a scalar-valued constant. Accordingly, the potential field u_δ is approximately induced by a small supported electromagnetic field $\delta \nabla u_\delta$. Therefore, the inverse problem can be considered to be parallel to finding an electromagnetic dipole with unknown length and charge within a vacuum cavity based on voltage measurements on the boundary. The problem is ill-conditioned, since both varying the length and varying the charge results in very similar changes far from the dipole. Contrary to the charge, the length has a slight effect on the shape of the potential distribution on the boundary. To be able to find out the solution, one has to be able to distinguish these changes. Certainly, this is not possible if the number of voltage measurements is too small. Again, the number of injected currents is closely related to the oscillation frequency of ∇u_{bg}, that on the basis of the right hand side of (6.3) reflects the resolution of the installation. Hence, despite the simplicity of the demonstrative problem the number of electrodes, apparently affects considerably the accuracy of the estimates.

The error due to discretion is relatively large and much less random than the error due to the noise in the measurements. Apparently, the statistical solutions would be more accurate if the discretion error was somehow taken into account in the a priori distribution.

Most of the Real life applications would undoubtedly utilize more than six electrodes. However, inserting electrodes to the model, would arise a need for refining the triangulation $\mathcal{T}_h$ so as to keep the discretization error at the same level, since injecting more currents to Ω would make ∇u to oscillate more frequently. The limited memory capacity of the available computer hardware did not allow significant refinements. Therefore, the influence of increasing the number of measurements is not studied in this thesis.

6.1 Summary and Conclusions

The findings and conclusions of this thesis can be formulated as follows.

- In addition to the traditional least-squares approach, the EIT problem can be formulated in terms of Bayesian statistics.
- The Bayesian statistics treat all sorts of uncertainty as random variables, which enables inclusion of the measurement noise into the mathematical model and effective utilization of all available a priori knowledge about the conductivity distribution.
- Numerical implementation of the Bayesian model results in need for effective high dimensional integration or optimization method.
- Monte Carlo sampling techniques offer a versatile collection of statistical integration and optimization methods, the convergence rate of which do not depend on the dimension but on how well the sampler is adapted to follow the posterior probability distribution.
- The numerical forward problem can be solved through various methods workability of which depends on the available information of the structure of the conductivity distribution, the applied sampling tehcnique and the level of measurement noise.
- Due to the strong ill-conditioned nature and non-linearity of the inverse problem it is often difficult to obtain any appropriate numerical solutions.
- Workability of the least-squares approach depends on the applied regularization method. It was found that a regularization method favoring smooth solutions can be produced effectively with the aid of the finite element method as described in secion (5.2.2). It is, yet, difficult to construct regularization method favoring arbitrary structures, e.g. strongly discontinuous conductivities.
- The statistical model is preferable to the least-squares approach only if there is accurate enough a priori knowledge available; In the numerical examples, statistical model was superior only if either the size or the value of conductivity of the anomaly was given.
- Since even the seemingly primitive demonstrative problem described in the numerical experiments chapter turned out to be difficult, further analysis of the EIT problem would be a natural continuation of the study.

Bibliography

[1] J. P. Kaipio, V. Kolehmainen, E. Somersalo, M. Vauhkonen: Statistical inversion and Monte Carlo sampling methods in electrical impedance tomography, *Inverse Problems*, 16: 1487-1522, 2000

[2] J. S. Liu, Monte Carlo Strategies in Scientific Computing, *Springer Verlag*, 2001

[3] D. Braess: Finite elements, *Cambridge University Press*, 2001

[4] G. Golub, C. van Loan: Matrix computations, *The John Hopkins University Press*, 1989

[5] A. F. M. Smith, G. O. Roberts: Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods, *J. R. Stat. Soc*, B 55 3-23, 1993

[6] E. Nummelin: General Irreducible MArkov Chains and Non-negative Operators *(Cambridge: Cambridge University Press)*, 1984

[7] L. Tierney: Markov chains for exploring posterior distributions *Ann. Stat.*, 22 1701-62, 1994

[8] H. Hakula: High-Order Finite Element Tools for Shell Problems, *Helsinki University of Technology, Institute of Mathematics, Research Reports* A376, 1997

[9] Cheng K-S, Isaacson D, Newell J and Gisser D: Electrode models for electric current computed tomography, *IEEE Trans. Biomed. Eng.* 36: 918-24, 1989

[10] Paulson K, Breckon W and Pidcock M: Electrode modelling in electrical impedance tomography *SIAM J. Appl. Math.* 52: 1012-1022, 1992

[11] Somersalo E, Cheney M and Isaacson D: Existence and uniqueness for electrode models for electric current computed tomography *SIAM J. Appl. Math.* 52: 1023-40, 1992

[12] Quin Z, Liu J: Multipoint Metropolis method with application to hybrid Monte Carlo, *Journal of Computational Physics 172: 827-840* , 2001

[13] Wong W and Liang F: Dynamic weighting in Monte Carlo and optimization, *Proceedings of the National Academy of Sciences of USA* 94(26): 14220-14224, 1997

[14] Chung K: A course in probability theory, *Academic Press, New York*, 1974

[15] S. Pursiainen: Erikoistyö 1, *TKK, Matematiikka*, 2000