Rustam Latypov

Inverse problem for resistor networks

School of Science

Bachelor's thesis
Espoo September 18, 2018

Thesis supervisor and advisor:

Prof. Antti Hannukainen

Aalto University
School of Science

AALTO UNIVERSITY SCHOOL OF SCIENCE	ABSTRACT OF THE BACHELOR'S THESIS

Author: Rustam Latypov

Title: Inverse problem for resistor networks

Date: September 18, 2018 Language: English Number of pages: 3+26

Degree programme: Engineering physics and mathematics

Supervisor and instructor: Prof. Antti Hannukainen

This thesis presents the mathematics behind a grid-based temperature sensor. It also applies the computational theory to exemplary surfaces. This grid sensor could be utilized in many fields of industry. It would offer effortless and precise temperature measurements around any object. This could uncover previously undetected heat dissipation and increase the stability of an industrial process.

The sensor is a grid-like structure with nodes and edges. A resistance is assigned to each edge. Since temperature can be derived from resistance, the temperature distribution can be obtained when the resistance of the grid is known.

The resistance is obtained by formulating a non-linear ill-posed inverse problem using only boundary measurements and the topology of the grid. The problem is then solved using the Gauss-Newton algorithm paired with Tikhonov regularization. A Laplacian matrix with the Neumann boundary condition is used for regularization. The algorithm is found to converge on accurate solutions when the parameters are fitted for the application and the environment.

Keywords: Resistor network, Inverse problem, Tikhonov regularization, Gauss-Newton algorithm, Neumann boundary condition

Contents

Abstract **ii**

Contents **iii**

1 Introduction **1**

2 Resistor Networks **2**
2.1 Theory . . . 2
2.2 Temperature grid sensor problem . . . 3

3 Non-linear Least Square Problem **4**
3.1 The Kirchhoff Matrix . . . 4
3.2 The Response Matrix . . . 5
3.3 The Schur Complement . . . 5
3.4 Least Squares . . . 7
3.4.1 Linear Least Squares . . . 7
3.4.2 Non-linear Least Squares . . . 8

4 Gauss–Newton Algorithm **9**
4.1 Tikhonov regularization . . . 9
4.2 Algorithm . . . 9

5 Computing Gauss-Newton **11**
5.1 Forward problem . . . 11
5.2 Derivative of the Schur Complement . . . 11
5.3 Singularity of the Kirchhoff Matrix . . . 13
5.4 Regularization Term . . . 14

6 Boundary Conditions **16**
6.1 Dirichlet boundary condition . . . 16
6.2 Neumann boundary condition . . . 16

7 Results **18**
7.1 The Stability of the Problem . . . 18
7.2 Temperature Map . . . 20
7.3 Analysis . . . 21
7.4 Convergence . . . 24

8 Summary **25**

References **26**

1 Introduction

The ability to monitor temperature fields plays a crucial role in many fields of industry. It enables accurate surveillance and analysis of processes that include power generation and heat distribution. In many industrial processes, the temperature is monitored locally at predetermined points of interest. The drawback to this is the inability to monitor processes as a whole, possibly leading to undetected dissipation of heat. Temperature field measurements however, offer an effortless way to monitor the heat distribution of any surface or object. This approach is used to detect thermal shocks and to control the temperature endurance of even the most peculiarly shaped objects. With the appropriate software it is also possible to record the thermal history of any process environment. This could prolong the life cycle of components to further increase the reliability of a process.

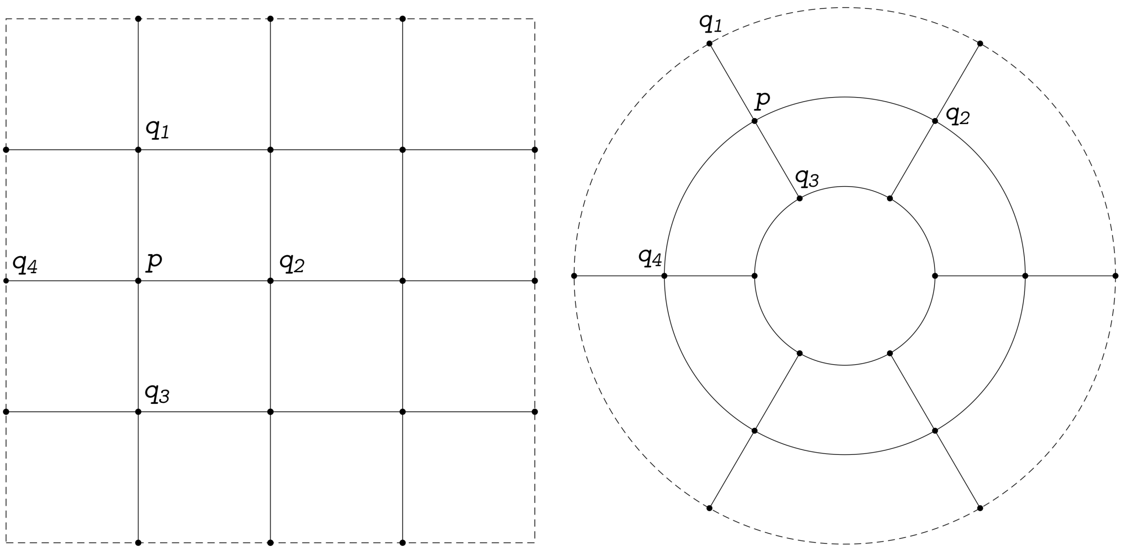
Currently there are several [1] methods to monitor temperature fields: temperature sensors, infrared cameras and temperature sensitive material coating. Temperature sensors provide fast and accurate measurements. However, they are often bulky and require individual cabling. The use of infrared cameras is suitable when the subject is optically accessible and is not thermally isolated. Unfortunately this is often not the case. Temperature sensitive material coating has similar drawbacks as infrared cameras, and in addition, it can also compromise the surface of the subject. The data retrieved from infrared cameras and temperature sensitive coatings is also not ideal, since it is limited to optical range.

This thesis examines the mathematics behind an alternative method to monitor temperature fields, a temperature grid sensor. Upon completion, this method would offer effortless and precise temperature measurements around any object. The grid sensor would be printed with conductive ink and consist of multiple electrical resistors assembled in a grid-like structure. Since temperature can be derived from resistance, the temperature distribution can be obtained when the resistance of the grid is known. The resistance can be computed from the data measured at the boundaries of the grid. This being a non-linear ill-posed inverse problem, the Gauss–Newton algorithm paired with Tikhonov regularization will be used for optimization. In addition to theory, some MATLAB-code will be presented to compute the results for some exemplary cases.

2 Resistor Networks

2.1 Theory

A graph G is a set of nodes d and edges e that are connected together. The connections between the nodes are called edges. A resistor network is composed of a graph G with a resistor at each edge and a conductivity function γ. Graph G can take on many forms as seen in Figure 2-1. The conductivity function γ assigns to each edge e a positive real number $\gamma(e)$ that is the conductance of its resistor. A resistor network is denoted $\Gamma = (G, \gamma)$ and its boundary nodes are denoted ∂G. The conductance of a resistor is a reciprocal of the resistance and it will be used in this approach for algebraic convenience [3].



(a) G_1

(b) G_2

Figure 2-1: Examples of graphs G with node p and its neighbors q_i.

A function u assigns to each node d a positive real number $u(d)$ that is its electric potential. If it is defined on all nodes of a resistor network $\Gamma = (G, \gamma)$, and an edge e has the endpoints p and q, the current flowing trough edge e is defined by Ohm's Law:

$$c(e) = \gamma(e)\big(u(p) - u(q)\big).$$

For each node p in G, there is a set of nodes q_i that are connected to node p by a joint edge as shown in Figure 2-1. This set of nodes is called the neighbors of p and is denoted $\mathcal{N}(p)$. The function u that is defined on the nodes of G is called γ-harmonic [3] if at each interior node $p \in G \backslash \partial G$ the sum of the currents from its neighboring nodes q is 0. This yields

$$\sum_{q \in \mathcal{N}(p)} \gamma(p,q)\big(u(p) - u(q)\big) = 0, \quad \forall p \in G \backslash \partial G. \tag{2.1}$$

For nodes $p \in \partial G$ Equation (2.1) does not necessarily hold, so the more general form is introduced:

$$\phi(p) = \sum_{q \in \mathcal{N}(p)} \gamma(p,q)\big(u(p) - u(q)\big), \quad \forall p \in G. \tag{2.2}$$

Imposing current on an electric circuit results in it exiting somewhere with an opposite sign. So summing $\phi(p)$ for all nodes in G yields

$$\sum_{p \in G} \phi(p) = 0. \tag{2.3}$$

Especially due to Equation (2.1) stating that at all interior nodes $\phi(p) = 0$,

$$\sum_{p \in \partial G} \phi(p) = 0. \tag{2.4}$$

The intuitive interpretation for $\phi(p)$ is the net current flowing in or out of a resistor network Γ at node p. For u to be non-zero and for current to flow in the network it is crucial that $\exists \phi(p) \neq 0,\ p \in G$.

2.2 Temperature grid sensor problem

A temperature grid sensor is a grid-like conductive resistor network Γ, similar to G_1 in Figure 2-1a. The aim is to piece together the temperature distribution of a surface the grid sensor is placed upon by evaluating the resistance of each resistor. Direct resistance measurements would defeat the purpose of a temperature grid sensor. So resistance is obtained by measuring current flow into network Γ and electric potential at the boundary nodes of Γ.

This generates an ill-posed inverse problem because it violates the third condition of a well-posed problem suggested by Jacques Hadamard [2]. The condition of stability does not hold because the change in the initial conductance reflects poorly on the boundary measurement, the farther the change occurs from the boundary. This is shown in Section 7.1.

The larger the network, the more unknown variables there are compared to the known linearly independent equations. So the complexity of the problem increases rapidly with the size of the network. The observation sets needed for solving this inverse problem are obtained by permanently grounding one boundary node of the network and repeating steps 1-2 for each remaining boundary node:

1. Impose current on the node, resulting in two opposite entries in ϕ: the node in question and the grounded node.
2. Measure the resulting electric potential at each boundary node.

Note, that imposing current on multiple boundary nodes to gain additional observation sets is not feasible because it would result in linear dependency later on.

3 Non-linear Least Square Problem

3.1 The Kirchhoff Matrix

Suppose a resistor network $\Gamma = (G, \gamma)$ with a total of m nodes $d_1, .., d_m$. By extending the concept of current flow into the network at node p we construct the $m \times m$ Kirchhoff matrix K with the following interpretation. If functions u and ϕ are defined at all the nodes and edges of G, then the resulting current flow into Γ is given by

$$\phi = Ku. \tag{3.1}$$

This is a compact way to express Equation (2.2) for each node in G where the Kirchhoff's current law holds.

The Kirchhoff matrix K is of size $m \times m$ and is composed from G and γ in Γ by the following conditions:

1. $K_{i,j} = -\gamma_{i,j}$, when $i \neq j$
2. $K_{i,i} = \sum_{i \neq j} \gamma_{i,j}$

The notation $\gamma_{i,j}$ stands for the conductance $\gamma(e)$ at edge e joining d_i and d_j. If there is no edge joining d_i and d_j, $\gamma_{i,j} = 0$. The nature of G make $\gamma_{i,j}$ symmetric in i and j, and $\gamma_{i,j} > 0$ only when there exists an edge between d_i and d_j in Γ. By definition it is also singular since $K_{m,m} 1_m = 0$, where 1_m denotes a size m vector with ones at each entry.

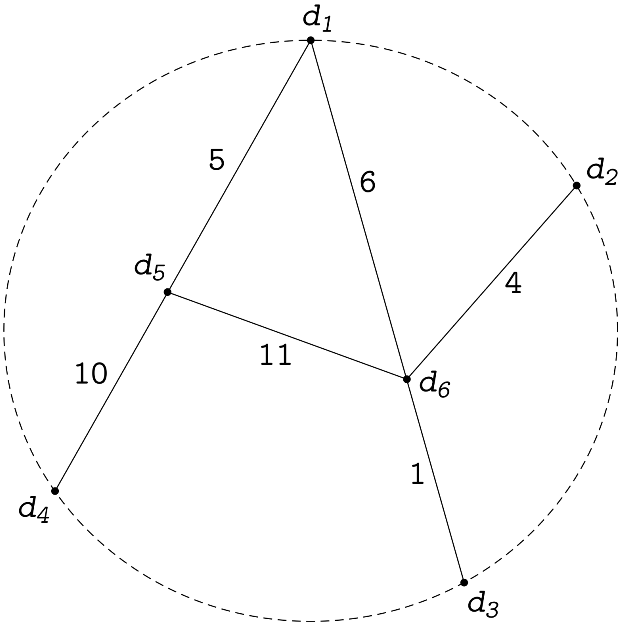


Figure 3-2: Graph G.

Example 3.1

Let $\Gamma = (G, \gamma)$ be a resistor network with G shown in Figure 3-2 with conductance indicated next to the edges. The corresponding Kirchhoff matrix K for Γ would be:

$$K = \begin{bmatrix} 11 & 0 & 0 & 0 & -5 & -6 \\ 0 & 4 & 0 & 0 & 0 & -4 \\ 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 10 & -10 & 0 \\ -5 & 0 & 0 & -10 & 26 & -11 \\ -6 & -4 & -1 & 0 & -11 & 22 \end{bmatrix}.$$

3.2 The Response Matrix

The Kirchhoff matrix enables us to express Equation (2.2) for each node in G in the form of $\phi = Ku$. This a useful form in general but in then case of the temperature grid sensor, the current flow into network Γ and the electric potential at the interior nodes of G are unknown, thus the expression $\phi = Ku$ is not that useful.

Suppose $\Gamma = (G, \gamma)$ is a resistor network with n boundary nodes and k interior nodes. The nodes are indexed so that the block of boundary nodes is $\{d_1, ..., d_n\}$ and the block of interior nodes is $\{d_{n+1}, ..., d_{n+k}\}$. Vectors u and ϕ can be also broken down into two blocks: $u = [g,\ f]$ and $\phi = [a,\ b]$, where vectors g and a contain n entries and correspond to the boundary nodes and vectors f and b contain k entries and correspond to the interior nodes. Since Equation (2.1) states that the current flow into the network Γ is zero at all interior nodes, $b = \vec{0}$. Thus, for the remainder of the thesis a will be referred to as ϕ. The response matrix $S \in \mathbb{R}^{n\times n}$ can be derived from the Kirchhoff matrix and used to modify Equation (3.1) into

$$\phi = Sg. \tag{3.2}$$

The response matrix should be interpreted as a linear map, which sends the imposed electric potential on the boundary nodes to the resulting current flow into the network at the boundary nodes. The response matrix can be obtained by block reducing K using the Schur complement.

3.3 The Schur Complement

Suppose M is a square matrix with D being a non-singular square sub-matrix of M. If M has the block structure

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix},$$

the Schur complement of D in M is defined to be the matrix

$$M/D = A - BD^{-1}C. \tag{3.3}$$

Suppose a resistor network $\Gamma = (G, \gamma)$ with D as a set of nodes and I as a subset of D denoting the interior nodes. If $K(I; I)$ denotes the sub-matrix of K with the index set I, the response matrix can be obtained by taking the Schur complement of $K(I; I)$ in K.

Lemma 3.1:

It is possible to derive $S \in \mathbb{R}^{n \times n}$ in $\phi = Sg$ when ϕ denotes the current flowing in or out of the boundary nodes and g the electric potential at the boundary nodes. S can be obtained by taking the Schur complement of $K(I; I)$ in K.

Proof:

If I is an empty set, then the response matrix S is defined to be $K/K(I; I) = K$. Otherwise I is non-empty. Let block D denote $K(I; I)$ and K have the block structure

$$K = \begin{bmatrix} A & B \\ B^T & D \end{bmatrix}.$$

Equation (2.1) states that the current flow into the network Γ is 0 at the enterior nodes. Thus, Equation (3.1) can be written in block form as

$$\begin{bmatrix} A & B \\ B^T & D \end{bmatrix} \begin{bmatrix} g \\ f \end{bmatrix} = \begin{bmatrix} \phi \\ 0 \end{bmatrix}. \tag{3.4}$$

This can be expanded to simple equations

$$Ag + Bf = \phi \tag{3.5}$$

$$B^T g + Df = 0 \tag{3.6}$$

and solved for ϕ by substituting f from Equation (3.5) for

$$f = -D^{-1} B^T f, \tag{3.7}$$

yielding

$$\phi = (A - BD^{-1} B^T) g. \tag{3.8}$$

Therefore, the response matrix is $S = A - BD^{-1}B^T$, which is the Schur complement of D in K and is in agreement with Equation (3.3). This operation leaves S non-linear with respect to γ. Note, that computing the Schur complement requires the non-singularity of block D. This is resolved by Curtis & Morrow, *Inverse Problems for Electrical Network* (p.40-43).

Example 3.2

Let $\Gamma = (G, \gamma)$ be a resistor network with G and K as in Example 3.1. There are four boundary nodes and two interior nodes. The response matrix is given by $S = K/K(5,6;5,6)$.

$$S = A - BD^{-1}B^T = \begin{bmatrix} 6.24 & -1.87 & -0.47 & -3.90 \\ -1.87 & 3.01 & -0.23 & -0.98 \\ -0.47 & -0.23 & 0.94 & -0.24 \\ -3.90 & -0.98 & -0.24 & 5.12 \end{bmatrix}.$$

where A, B and D are $K(1,4;1,4)$, $K(1,4;5,6)$ and $K(5,6;5,6)$ respectively.

3.4 Least Squares

3.4.1 Linear Least Squares

The objective of a linear least square (LS) problem is to adjust the parameters of a model function to best fit an observed data set. A data set consists of a independent variable a_j and a dependent variable b_i, where $j = 1, ..., n$ and $i = 1, ..., m$. LS-method is needed when a system is overdetermined, meaning $m \geq n$. If a system is underdetermined, the solution can not be solved uniquely. The model function has the form $f(a_j, x_j)$ with n adjustable parameters in vector x. The fit of a model is determined by its residual r, defined as

$$r_i = f_i(a_j, x_j) - b_i.$$

The LS-method finds the optimal value for x, that minimizes the sum

$$\sum_{i=1}^{m} r_i^2.$$

Since the model function $f_i(a_j, x_j)$ has the same form through out the observation set, it can be assembled into matrix Ax. Thereby, the LS-method solves

$$\min \|Ax - b\|_2^2 . \tag{3.9}$$

Assuming that A has linearly independent columns, the following solution is obtained. By defining

$$M(x) = \|Ax - b\|_2^2 = \sum_{i=1}^{m} \Big(\sum_{j=1}^{n} A_{ij} x_j - b_i \Big)^2, \tag{3.10}$$

the solution $\hat{x}$ satisfies

$$\frac{\partial M}{\partial x_j}(\hat{x}) = \nabla_j M(\hat{x}) = 0, \quad \forall j = 1, ..., n. \tag{3.11}$$

Taking partial derivatives yields

$$\nabla_j M(\hat{x}) = (2A^T(A\hat{x} - b))_j = 0 \tag{3.12}$$

$$\nabla M(\hat{x}) = 2A^T(A\hat{x} - b) = 0. \tag{3.13}$$

Solving this gives the LS-solution

$$\hat{x} = (A^T A)^{-1} A^T b = A^{\dagger} b. \tag{3.14}$$

Lemma 3.2:

Let $A \in \mathbb{R}^{n \times m}$. The solution to $\min \|Ax - b\|_2^2$ is given by $(A^T A)^{-1} A^T b$.

Proof:

Let $\hat{x} = (A^T A)^{-1} A^T b$, resulting in $A^T(A\hat{x} - b) = 0$.

For $\forall x$ applies

$$\begin{aligned}\|Ax - b\|^2 &= \|(Ax - A\hat{x}) + (A\hat{x} - b)\|^2 \\ &= \|A(x - \hat{x})\|^2 + \|A\hat{x} - b\|^2 + 2(A(x - \hat{x}))^T(A\hat{x} - b) \\ &= \|A(x - \hat{x})\|^2 + \|A\hat{x} - b\|^2 + 2(x - \hat{x})^T A^T (A\hat{x} - b) \\ &= \|A(x - \hat{x})\|^2 + \|A\hat{x} - b\|^2.\end{aligned}$$

Thus, for $\forall x$ holds $\|Ax - b\|^2 \geq \|A\hat{x} - b\|^2$ and $\hat{x}$ is indeed the minimizer of a LS-problem. If equality holds, $\|A(x - \hat{x})\|^2 = 0$, which implies that $x = \hat{x}$ and the LS-solution is simply $x = A^{-1}b$.

3.4.2 Non-linear Least Squares

A non-linear system is denoted $A(x)$, since is not linear with respect to x. The partial derivatives of r_i consist of both the independent variable and the parameters, so these gradient equations do not have a closed solution. The solution can still be obtained by giving x an initial value and by closing in on a solution with an iterative algorithm. In the case of the temperature grid sensor, the minimizing problem is derived from Equation (3.2) to have the form

$$\min \|g - S^{-1}(\gamma)\phi\|^2 \tag{3.15}$$

with $\|\cdot\|$ denoting the Euclidean norm. The inverse of S is taken for computational reasons and to emphasize that by inputting current ϕ into the network, we observe electric potential g. For simplicity, Equation (3.15) is written as

$$\min \|g - F(\gamma)\|^2 \tag{3.16}$$

for the remainder of the thesis.

4 Gauss–Newton Algorithm

4.1 Tikhonov regularization

When solving $\|A(x) - b\|^2$ iteratively, with j denoting iterations, problems occur when the singular values of matrix A tend to zero [4]. The more undetermined the system is, the more frequently it occurs. This causes $x_k \to \infty$ when $j \to \infty$. The basic idea of regularization is to give preference to a particular solution with desirable properties. Tikhonov regularization is one of the oldest and most commonly used method of regularization of ill-posed problems. The aim is to control the norm of the residual and the norm of some regularization term simultaneously. Using some prior knowledge of the system, a regularization term L can be added to Equation (3.16) to obtain

$$\min \left\{ \|g - F(\gamma)\|^2 + \alpha \|L(\gamma)\|^2 \right\} \quad \alpha > 0, \alpha \in \mathbb{R}, \tag{4.1}$$

where α is free variable that gives weight to the regularization term.

4.2 Algorithm

The solution to a non-linear inverse problem with Tikhonov regularization is of form

$$\overline{\gamma} = \text{sol min} \left\{ \|g - F(\gamma)\|^2 + \alpha \|L(\gamma)\|^2 \right\}, \tag{4.2}$$

where F is a non-linear map. By adding the Tikhonov regularization term to Equation (3.16), the functional is no longer a quadratic one. Assuming operator F is Fréchet differentiable, allows us to write

$$F(\gamma) = F(\gamma_0) + F'(\gamma_0)(\gamma - \gamma_0) + \mathcal{O}(\|\gamma - \gamma_0\|^2)$$

and by denoting $\gamma = \gamma_0 + \delta\gamma$,

$$F(\gamma_0 + \delta\gamma) = F(\gamma_0) + F'(\gamma_0)(\delta\gamma) + \mathcal{O}(\|\delta\gamma\|^2).$$

The Gauss-Newton method solves the problem in Equation (4.2) iteratively by approximating the functional at point $\gamma^{(j)}$ with a quadratic functional, which yields a solution $\gamma^{(j+1)}$. The quadratic functional approximation is then repeated at $\gamma^{(j+1)}$ in the next iteration. If the linearized set of solutions converge, it converges to a critical point of the original functional [4].

The aim is to iterate on $\gamma^{(j)}$ to obtain

$$\gamma^{(j+1)} = \gamma^{(j)} + \delta\gamma^{(j)},$$

where $\delta\gamma^{(j)}$ is the wanted update, so that $\gamma^{(j+1)}$ is the solution to the linearized problem at point $\gamma^{(j)}$. The linearized functional $\Xi(\gamma)$ at point γ_0 is given by

$$\Xi(\gamma) = \|g - (F(\gamma_0) + F'(\gamma_0)(\gamma - \gamma_0))\|^2 + \alpha \|L(\gamma)\|^2, \tag{4.3}$$

where $\alpha\|L(\gamma)\|^2$ is the choice of regularization and will be expanded on in Section 5.4.

The notation for recursion is as follows

$$\begin{aligned}\gamma_0 &\leftarrow \gamma^{(j)}\\ \gamma &\leftarrow \gamma^{(j+1)}\\ \delta\gamma &\leftarrow \gamma^{(j+1)} - \gamma^{(j)}.\end{aligned}$$

At each step j, $F(\gamma^{(j)})$ is evaluated. So by denoting $g^{(j)}$ as $F(\gamma^{(j)})$:

$$\Xi(\gamma^{(j+1)}) = \|g - F(\gamma^{(j)}) - F'(\gamma^{(j)})(\gamma^{(j+1)} - \gamma^{(j)}))\|^2 + \alpha\|L(\gamma^{(j+1)})\|^2 \tag{4.4}$$

$$\Xi(\gamma^{(j+1)}) = \|g - g^{(j)} - F'(\gamma^{(j)})\delta\gamma\|^2 + \alpha\|L(\gamma^{(j)} + \delta\gamma)\|^2. \tag{4.5}$$

This can be written as $\Xi(\gamma^{(j+1)}) = (V^{(j)})^T V^{(j)}$, where

$$V^{(j)} = \begin{pmatrix} F'(\gamma^{(j)}) \\ \sqrt{\alpha}L \end{pmatrix} \delta\gamma - \begin{pmatrix} g - g^{(j)} \\ \sqrt{\alpha}L(-\gamma^{(j)}) \end{pmatrix}. \tag{4.6}$$

$\Xi(\gamma^{(j+1)})$ is now in a linear LS form as in Equation (3.9). Thus, the update to $\gamma^{(j)}$ at iteration step j is simply the LS-solution

$$\delta\gamma = \begin{pmatrix} F'(\gamma^{(j)}) \\ \sqrt{\alpha}L \end{pmatrix}^{\dagger} \begin{pmatrix} g - g^{(j)} \\ \sqrt{\alpha}L(-\gamma^{(j)}) \end{pmatrix}. \tag{4.7}$$

From a computational point of view, the idea is to vertically stack the matrices produced by the observation sets to obtain the new direction $\delta\gamma$. Now the challenge consists of calculating the forward problem of $F(\gamma^{(j)})$, the derivative of $F(\gamma^{(j)})$ and composing a reasonable regularization term L. These are discussed in the following chapter.

5 Computing Gauss-Newton

5.1 Forward problem

The forward problem assumes that the conductance $\gamma(e)$ for each edge e in a resistor network Γ is known. The aim is to compute $g^{(j)}$ at point $\gamma^{(j)}$. The first time this is invoked in the algorithm, is when an initial guess is given.

Since

$$g^{(j)} = F(\gamma^{(j)}) = S^{-1}(\gamma)\phi, \tag{5.1}$$

where ϕ is known and $S^{-1}(\gamma)$ can be composed using Equation (3.3), calculating the forward problem is trivial.

5.2 Derivative of the Schur Complement

The derivative of $F(\gamma^{(j)})$ is with respect to $\gamma^{(j)}$. This is to be interpreted as the slope of a multidimensional surface in the direction η. Let $h \in \mathbb{R}$, $h > 0$. The precise definition is

$$D(F(\gamma^{(j)}))(\eta) = \lim_{h \to 0} \frac{F(\gamma^{(j)}) - F(\gamma^{(j)} + h\eta)}{h}. \tag{5.2}$$

Obtaining $F'(\gamma^{(j)})(\eta)$ is not as straight forward as one could assume. Since

$$D(F(\gamma))(\eta) = D(S^{-1}(\gamma)\phi)(\eta) = D(S^{-1}(\gamma))(\eta)\phi, \tag{5.3}$$

the derivative of the inverse of the Schur complement with respect to γ has to be calculated.

Calculating the derivative of the inverse of the Schur complement that is composed with Equation (3.3) is a tedious task. Thus, another way to compose the Schur complement is introduced.

Section 3.3 states, that

$$Ku = \phi$$
$$K \begin{bmatrix} g \\ f \end{bmatrix} = \begin{bmatrix} \phi \\ 0 \end{bmatrix}.$$

Manipulating the equation yields another expression for the Schur complement.

$$\begin{bmatrix} g \\ f \end{bmatrix} = K^{-1} \begin{bmatrix} \phi \\ 0 \end{bmatrix}$$

$$\begin{aligned}
\begin{bmatrix} g \\ f \end{bmatrix} &= K^{-1} \begin{bmatrix} I \\ 0 \end{bmatrix} \phi \\
g &= \begin{bmatrix} I \\ 0 \end{bmatrix}^T K^{-1} \begin{bmatrix} I \\ 0 \end{bmatrix} \phi \\
g &= S^{-1}\phi
\end{aligned}$$

With this, Equation (5.3) takes the form of

$$D(S^{-1}(\gamma))(\eta)\phi = \begin{bmatrix} I \\ 0 \end{bmatrix}^T D(K^{-1}(\gamma))(\eta) \begin{bmatrix} I \\ 0 \end{bmatrix} \phi. \tag{5.4}$$

The derivative of an inverse matrix can be calculated with some manipulation. Since, the defining relationship between a matrix and its inverse is

$$K(\gamma)K^{-1}(\gamma) = I,$$

taking the derivative of both sides yields

$$D(K(\gamma))(\eta)K^{-1}(\gamma) + K(\gamma)D(K^{-1}(\gamma))(\eta) = 0.$$

Straightforward manipulation gives an expression for the derivative of an inverse matrix

$$D(K^{-1}(\gamma))(\eta) = K^{-1}(\gamma)D(K(\gamma))(\eta)K^{-1}(\gamma). \tag{5.5}$$

Substituting 5.5 into Equation (5.4) gives the most simple expression for $F'(\gamma)(\eta)$,

$$F'(\gamma)(\eta) = D(S^{-1}(\gamma))(\eta)\phi = \begin{bmatrix} I \\ 0 \end{bmatrix}^T K^{-1}(\gamma)D(K(\gamma))(\eta)K^{-1}(\gamma) \begin{bmatrix} I \\ 0 \end{bmatrix} \phi. \tag{5.6}$$

The direction η is of great value, since it gives direction to the Gauss-Newton algorithm and is denoted $\delta\gamma$ in Equation (4.7). The goal is to extract η from Equation (5.6). This is achieved by first denoting

$$K^{-1}(\gamma) \begin{bmatrix} I \\ 0 \end{bmatrix} \phi = \hat{c}. \tag{5.7}$$

and then manipulating

$$D(K(\gamma))(\eta)\hat{c} \to C(\hat{c})\eta. \tag{5.8}$$

This is best demonstrated by an example.

Example 5.1

Let $\Gamma = (G, \gamma)$ be a resistor network with

$$K = \begin{bmatrix} \gamma_3 & 0 & 0 & 0 & -\gamma_3 \\ 0 & \gamma_2 & 0 & 0 & -\gamma_2 \\ 0 & 0 & \gamma_4 & 0 & -\gamma_4 \\ 0 & 0 & 0 & \gamma_1 & -\gamma_1 \\ -\gamma_3 & -\gamma_2 & -\gamma_4 & -\gamma_1 & \gamma_3+\gamma_2+\gamma_4+\gamma_1 \end{bmatrix} \text{ and } \hat{c} = \begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \\ \hat{c}_3 \\ \hat{c}_4 \\ \hat{c}_5 \end{bmatrix}.$$

This translates into

$$\begin{aligned} D(K(\gamma))(\eta)\hat{c} &= \begin{bmatrix} \eta_3 & 0 & 0 & 0 & -\eta_3 \\ 0 & \eta_2 & 0 & 0 & -\eta_2 \\ 0 & 0 & \eta_4 & 0 & -\eta_4 \\ 0 & 0 & 0 & \eta_1 & -\eta_1 \\ -\eta_3 & -\eta_2 & -\eta_4 & -\eta_1 & \eta_3+\eta_2+\eta_4+\eta_1 \end{bmatrix} \begin{bmatrix} \hat{c}_1 \\ \hat{c}_2 \\ \hat{c}_3 \\ \hat{c}_4 \\ \hat{c}_5 \end{bmatrix} \\ &= \begin{bmatrix} \eta_3(\hat{c}_1 - \hat{c}_5) \\ \eta_2(\hat{c}_2 - \hat{c}_5) \\ \eta_4(\hat{c}_3 - \hat{c}_5) \\ \eta_1(\hat{c}_4 - \hat{c}_5) \\ \eta_3(\hat{c}_5 - \hat{c}_1) + \eta_2(\hat{c}_5 - \hat{c}_2) + \eta_4(\hat{c}_5 - \hat{c}_3) + \eta_1(\hat{c}_5 - \hat{c}_4) \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & \hat{c}_1 - \hat{c}_5 & 0 \\ 0 & \hat{c}_2 - \hat{c}_5 & 0 & 0 \\ 0 & 0 & 0 & \hat{c}_3 - \hat{c}_5 \\ \hat{c}_4 - \hat{c}_5 & 0 & 0 & 0 \\ \hat{c}_5 - \hat{c}_4 & \hat{c}_5 - \hat{c}_2 & \hat{c}_5 - \hat{c}_1 & \hat{c}_5 - \hat{c}_3 \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \end{bmatrix} \\ &= C(\hat{c})\eta. \end{aligned}$$

Using this kind of manipulation it is possible to express

$$F'(\gamma)(\eta) = \begin{bmatrix} I \\ 0 \end{bmatrix}^T K^{-1}(\gamma) D(K(\gamma))(\eta) K^{-1}(\gamma) \begin{bmatrix} I \\ 0 \end{bmatrix} \phi = F'(\gamma)\eta = F'(\gamma)\delta\gamma. \tag{5.9}$$

The only issue left to address is the fact that K is singular by definition, which makes computing Equation (5.9) impossible.

5.3 Singularity of the Kirchhoff Matrix

The Kirchhoff matrix is singular by definition, due to the way it is constructed in Section 3.1. This problem is resolved by further examining the way the observation sets are obtained in Section 2.2. In the very beginning, one boundary node is permanently grounded. This means that the all of the net current in the network is flowing though it and its electric potential is zero. This is true for all of the observation sets. Thus, the grounded boundary node holds no useful information. By systematic indexing, the grounded node can be agreed to be in the same place regardless of the size of G. It can be agreed to be the first element in γ. Thus, the first row and column of K can be omitted without any loss of information. This makes K non-singular as is needed in Equation (5.9).

5.4 Regularization Term

Regularization is needed to give preference to to a particular solution using some prior knowledge. It is intuitive that the temperature distribution of a surface can not have rapid oscillations but instead is rather smooth. So some regularization term is needed that gives preference to smooth solutions.

The smoothness prior should penalizes changes in the second derivative, thus encouraging regions of constant first derivative. This is achieved with the finite difference method [6].

The finite difference method computes the second derivative by a difference quotient in the classic formulation. It offers a more direct and economic approach to the numeric solution of a partial differential equation, since it does not require the evaluation of second derivatives. It also calculates the difference using point-wise values, which is essential, since the temperature grid sensor is discrete. Its main drawbacks are the lack of flexibility in the structuring of the problem and imposing boundary conditions.

Consider the Poisson's equation in a two dimensional (x, y)-plane:

$$-\Delta u = f \text{ in } \Omega \backslash \partial\Omega, \tag{5.10}$$

where Δ is the Laplace operator,

$$\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}, \tag{5.11}$$

and $\Omega \in \mathbb{R}^2$ is a bounded open set. The finite difference method expresses the Laplace operator using point-wise values of u. Let $h \in \mathbb{R},\ h > 0$. Using the Taylor expansion with respect to x component gives

$$u(x+h, y) = u(x, y) + \frac{\partial u}{\partial x}(x, y)h + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}(x, y)h^2 + \frac{1}{6}\frac{\partial^3 u}{\partial x^3}(x, y)h^3 + \mathcal{O}(h^4)$$
$$u(x-h, y) = u(x, y) - \frac{\partial u}{\partial x}(x, y)h + \frac{1}{2}\frac{\partial^2 u}{\partial x^2}(x, y)h^2 - \frac{1}{6}\frac{\partial^3 u}{\partial x^3}(x, y)h^3 + \mathcal{O}(h^4).$$

Summing the two equations and dividing by h^2 yields

$$\frac{\partial^2 u}{\partial x^2}(x, y) = \frac{u(x+h, y) - 2u(x, y) + u(x-h, y)}{h^2} \tag{5.12}$$

and by symmetry, for the y component,

$$\frac{\partial^2 u}{\partial y^2}(x, y) = \frac{u(x, y+h) - 2u(x, y) + u(x, y-h)}{h^2}. \tag{5.13}$$

The formulas above can be used to discretize Equation (5.10). By introducing a uniform grid of points of size $m \times m$ with indices $1 \leq i, j \leq m$ and by using these difference formulations, the Laplace operator at point (x_i, y_j) gives

$$(\Delta u)_{i,j} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \quad i, j = 2, ..., n-1 \tag{5.14}$$

$$= \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h_x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{h_y^2}. \tag{5.15}$$

In the case of a uniform grid, $h_x = h_y$. So this simplifies to

$$-(\Delta_h u)_{i,j} = \frac{4u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1}}{h^2}, \tag{5.16}$$

In the case of the temperature grid sensor, u and Ω correspond to γ and G respectively. The challenge is index handling, since γ is defined on the edges of the network rather than the nodes. With this computational issue resolved, minimizing the operation in Equation (5.16) for all elements in γ gives preference to a smooth temperature distribution.

Let $\Gamma = (G, \gamma)$. Considering the problem

$$\min \left\{ \|g - F(\gamma)\|^2 + \alpha \|L(\gamma)\|^2 \right\}, \tag{5.17}$$

the regularization term $\alpha\|L(\gamma)\|^2$ is to be interpreted as follows. L is the Laplace operator that operates on γ, giving $\Delta\gamma$ at all interior edges of G. α is a free variable that is to be modified to give weight to the regularization. The larger the graph G, the more observation sets there are and the larger γ becomes. This all leads to larger matrices in Equation (4.7). However, it makes sense to maintain a constant weight of regularization regardless of the dimension of G, resulting in

$$\alpha \not\propto \dim(G). \tag{5.18}$$

Since Equation (5.16) is defined with indices $i, j = 2, ..., n-1$, it utilizes the boundary points but can not be evaluated on them. This would require indices outside the current lattice. Computing Δu on $\partial\Omega$ requires boundary conditions.

6 Boundary Conditions

6.1 Dirichlet boundary condition

The Dirichlet boundary condition imposes an additional condition

$$u = g \text{ on } \partial\Omega \tag{6.1}$$

on the Poisson's equation. The physical interpretation of this is as follows. Using prior knowledge of the temperature distribution, a specific function g is defined on the boundary of Ω. Namely $u_{i,j} = g(x_i, y_j)$ for $(x_i, y_j) \in \partial\Omega$, thus solving the boundary issue.

This approach works in the case of the grid sensor but it is not ideal. Firstly, considering the applications of a grid sensor it is unreasonable to assume that the boundary temperature is known or stays constant. Secondly, the Dirichlet boundary condition leads to weaker regularization, since it omits the boundary all together. The Neumann boundary condition is an improvement on the previous, but is more difficult to derive and compute.

6.2 Neumann boundary condition

The Neumann boundary condition imposes an additional condition

$$\frac{\partial u}{\partial n} = g \text{ on } \partial\Omega \tag{6.2}$$

on the Poisson's equation. Instead of specifying u on the boundary of Ω, it specifies the normal derivative of u. The normal is defined as a vector n, with $n \perp x$, $n \perp y$. This condition requires the introduction of ghost points and the extension of the lattice by allowing the indices $0 \leq i, j \leq n+1$. Ghost points are imagined values of u next to the boundary. Since γ is only defined on the edges, using point-wise thinking forces every real point to neighbour two ghost points and two real points as illustrated in Figure 6-3.

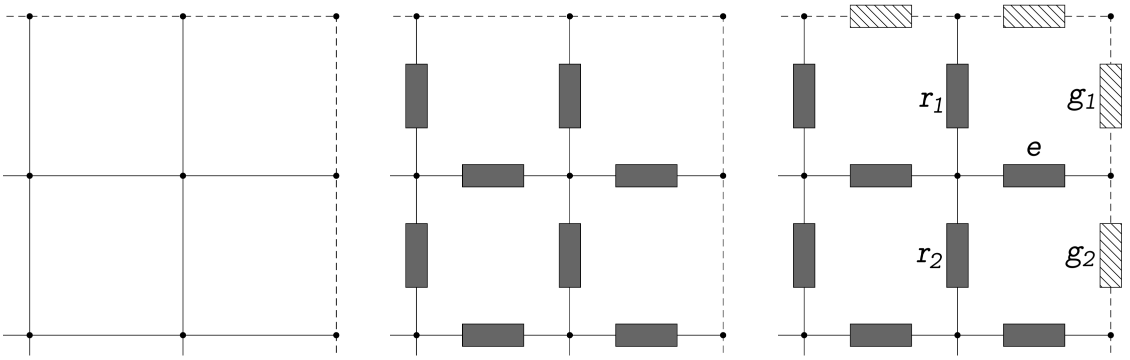


Figure 6-3: Boundary edge e neighbouring real points r_1, r_2 and ghost points g_1, g_2.

Taking a point (x_1, y_1) as an example, the discrete central difference formulation gives

$$\frac{\partial u}{\partial x}(x_1, y_1) = \frac{u_{0,1} - u_{2,1}}{2h} \tag{6.3}$$

$$\frac{\partial u}{\partial y}(x_1, y_1) = \frac{u_{1,0} - u_{1,2}}{2h}. \tag{6.4}$$

Also, Equation (5.16) states that

$$-\left(\Delta_h u\right)_{1,1} = \frac{4u_{1,1} - u_{2,1} - u_{0,1} - u_{1,2} - u_{1,0}}{h^2}. \tag{6.5}$$

These equations compose a system of equations:

$$u_{0,1} - u_{2,1} = 2hg_{1,1} \tag{6.6}$$

$$u_{1,0} - u_{1,2} = 2hg_{1,1} \tag{6.7}$$

$$4u_{1,1} - u_{2,1} - u_{0,1} - u_{1,2} - u_{1,0} = h^2 f_{1,1}. \tag{6.8}$$

Solving 6.6 and 6.7 for the ghost points and substituting into 6.8 yields

$$4u_{1,1} - 2u_{2,1} - 2u_{1,2} = h^2 f_{1,1} + 4hg_{1,1} \tag{6.9}$$

$$f_{1,1} = \frac{4u_{1,1} - 2u_{2,1} - 2u_{1,2} - 4hg_{1,1}}{h^2}. \tag{6.10}$$

For a smooth boundary the normal derivative of u should be zero, giving $g = 0$. Thus, the definition of f can be extended to $\overline{\Omega}$ as

$$-\Delta u = f \text{ on } \overline{\Omega}$$

with

$$f_{i,j} = \frac{4u_{i,j} - u_{i+1,j} - u_{i-1,j} - u_{i,j+1} - u_{i,j-1}}{h^2} \text{ on } \Omega$$

$$f_{i,j} = \frac{4u_{i,j} - 2u_{i+1,1} - 2u_{1,j+1}}{h^2} \text{ on } \partial\Omega.$$

Again, with careful indexing, this reasoning holds for the grid sensor by denoting u as γ, f as $L(\gamma)$ and Ω as G. In this thesis, the Neumann boundary condition is imposed.

7 Results

As a proof of principle, around 350 lines of MATLAB-code was written to implement the Gauss-Newton optimization for a grid temperature sensor of scalable size. Since no real-life measurements were conducted, they had to be simulated. This was accomplished with the following steps.

1. Decide on size of the network and the test current ϕ.

2. Decide on the value of γ, which is the solution to the problem.

3. Construct S with K to obtain g.

4. Add noise to g with `k*rand(dim,dim)` and use the result as measurements.

This creates a controlled environment to test out the algorithm. This environment enables control over the solution γ and the precision of the measurements. In a real-life setting this is not the case. Of course the values for α, h, ϕ and the number of iterations are also free to manipulation.

The code fragment that performs the Gauss-Newton iteration is

```
y = y_0
for i = 1:itr
    g_i = F(y);
    dy = [dF(y);sqrt(alpha)*L] \ [g-g_i;-sqrt(alpha)*L*y];
    y = y + dy;
end
```

where an initial guess is given for γ, `F.m` computes the forward problem and `dF.m` computes $F'(\gamma)$ from Section 5.2. The properties of the algorithm can be monitored by calculating the relative difference of any vector f with

$$f_D = \frac{\|f - f_m\|_2}{\|f\|_2}.$$

The convergence of the algorithm can be monitored with the relative difference of g and the accuracy of the solution with the relative difference of γ since in this case the correct solution is known. The terminating condition of the iteration can be set to depend on the relative difference of g.

7.1 The Stability of the Problem

To show that the problem of a grid sensor violates the condition of stability as stated in Section 2.2, a resistor networks with $\dim(\gamma) \approx 1000$ is simulated. To test the stability of the problem, two values of g, from different γ, are compared. g and g_m are obtained from γ and γ_m respectively. γ is uniform and γ_m is uniform except for one differently manipulated element at different distances from the boundary. For precision, no noise is added to g. The results are presented in Figure 7-4.

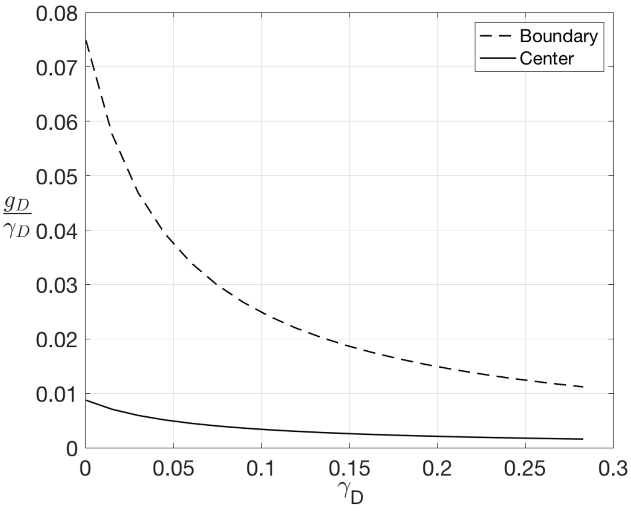


Figure 7-4: $\dfrac{g_D}{\gamma_D}$ against γ_D.

Depending on the distance of the manipulated element from the boundary, g_D is only $1 - 8\%$ of γ_D with a descending tendency. This proves that relatively large changes in γ, the initial conditions, map to to relatively small changes in g, the observations. This behaviour is in violation of the stability condition of a well-posed problem, making this problem ill-posed.

Changes in γ that occur in the center of the network reflect more poorly on g than changes occurring near the boundary of the network. This is problematic, since the center of the network is just as important as the boundary for the temperature distribution.

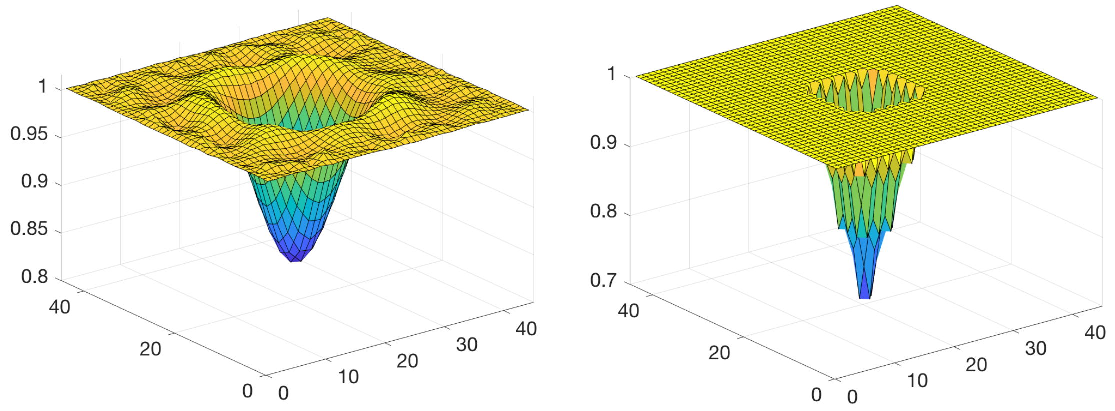
Suppose the manipulated element value in γ_m is significantly smaller than the otherwise uniform γ and it is located in the center of the network. This corresponds to one heated resistor in the center of the network. The current flowing through the network "avoids" edge e_m, resulting in considerable change in potential at surrounding nodes. This effect gets damped by the time the current reaches the boundary, since the current gets split at each node in the network. The farther γ_m is from the boundary, the more significant the damping. The same logic can be applied for any manipulation to any element of γ.

Considering the material used for the resistor in a grid sensor, preference should be given to metals which have a fixed temperature sensitivity range that suits the application. This is justified, since large fluctuations in conductance are harder to detect than small ones according to Figure 7-4. However, noise is always present in real-life measurements dictating the observational threshold for γ_D.

7.2 Temperature Map

This thesis also visualizes the heat distribution of a simulated surface using the Gauss-Newton optimization algorithm. Due to the environment this is simulated in, the solution is simple to validate. The algorithm is tested on two kinds of surfaces with different temperature distributions. The surface is monitored with a resistor network with $\dim(\gamma) \approx 1000$.

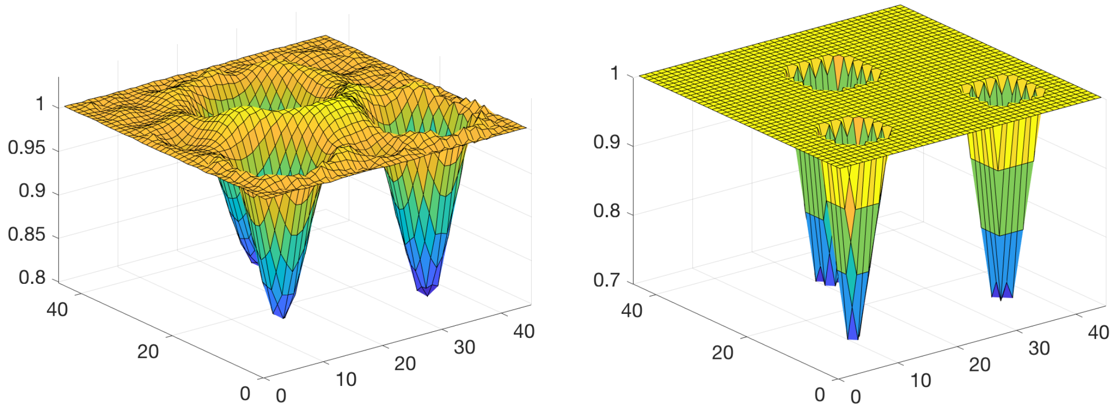
The first surface S_1 in Figure 7-5 has a steeply heated subject in the center which maps to a heated center with gradual cooling towards the boundary in the temperature distribution produced by the algorithm. The second surface S_2 in Figure 7-6 has three heated subjects. Since resistance rises with temperature and conductance is the reciprocal of the resistance, the lower the conductance, the higher the temperature in the area.



(a) Distribution produced by the algorithm.

(b) Actual distribution.

Figure 7-5: Temperature distributions of S_1.



(a) Distribution produced by the algorithm.

(b) Actual distribution.

Figure 7-6: Temperature distributions of S_2.

The algorithm parameters for these temperature distributions are

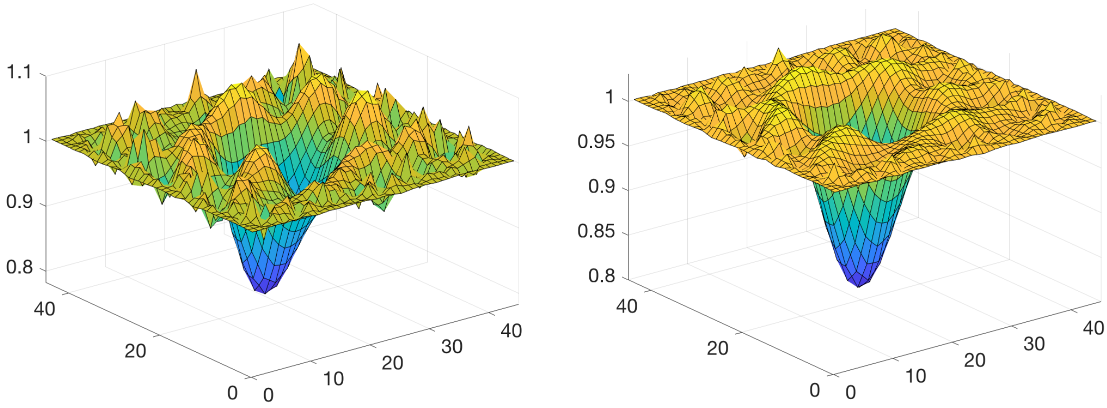
$$\phi = 100, \quad h = 1, \quad \alpha = 1, \quad k = 0.1, \quad \text{itr} = 10.$$

The base temperature is set to correspond to $\gamma = \{1\}$, with elements gradually decreasing to value 0.7 around the heated subjects. Since only some elements are affected by heated objects, $\gamma = \{1\}$ is a reasonable initial guess. Noise is added simply to visualize a distribution that could be obtained through real-life measurements. Since the algorithm converges on reasonable solutions, these parameters will be used for remainder of this chapter unless stated otherwise.

7.3 Analysis

It is worth noting the problem of oscillation at the edges of the heated objects in Figures 7-5 and 7-6. This behaviour comes across as Runge's or Gibbs phenomenon but is not explored further in this thesis.

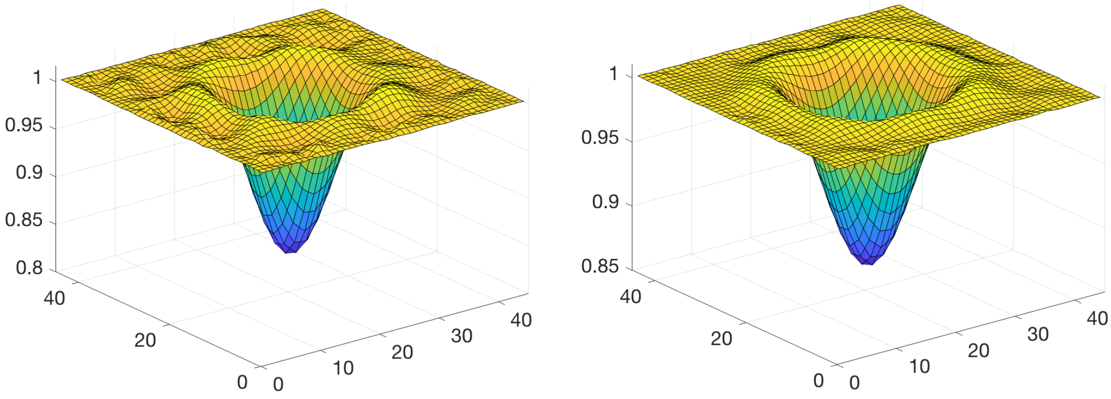
However, the main issue with the results is that, though the center of the heated object corresponds to values 0.7, the temperature map detects central heat corresponding to values 0.8. This occurs due to the instability of the problem and the chosen value for α, the weight of regularization. Figures 7-7 and 7-8 suggest that decreasing the value of α for S_1 helps the algorithm to find a more accurate solution in the region of the heated object. The drawback to this is noise in previously stable regions of the distribution, thus affecting γ_D and g_D.



(a) $\alpha = 0.01$

(b) $\alpha = 0.1$

Figure 7-7: Temperature distributions of S_1 with small α values.



(a) $\alpha = 1$

(b) $\alpha = 10$

Figure 7-8: Temperature distributions of S_1 with large α values.

Since the same surface is analyzed, noise added to g is kept constant. Iterations in Figures 7-7 and 7-8 yield

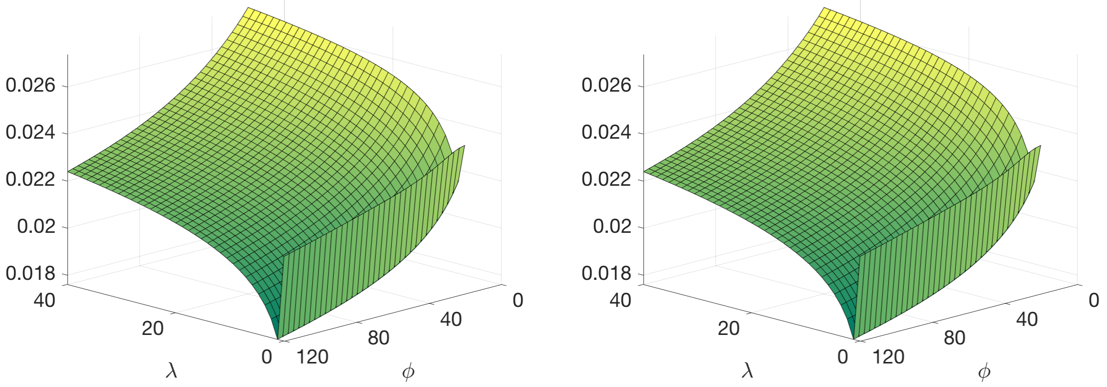
$$\begin{aligned} \alpha = 0.01: &\quad \gamma_D = 0.0286, \quad g_D = 9.9606\times10^{-5} \\ \alpha = 0.1: &\quad \gamma_D = 0.0176, \quad g_D = 1.0001\times10^{-4} \\ \alpha = 1: &\quad \gamma_D = 0.0180, \quad g_D = 1.0044\times10^{-4} \\ \alpha = 10: &\quad \gamma_D = 0.0213, \quad g_D = 1.0120\times10^{-4}. \end{aligned}$$

Now even though the algorithm converges and γ_D is of acceptable magnitude, the distribution is mainly inaccurate at low α values. However, the application of the grid sensor will dictate the threshold for accuracy and thus what value for α is suitable. Some applications may require the knowledge of the exact temperature of heated subjects, regardless of the slight fluctuation in the stable regions. These applications should use small values for α. If on the other hand smooth distributions are required, large values for α should be used.

It is of interest to find the parameter values that minimize γ_D for S_1, since γ_D represent the accuracy of the solution. Varying γ_D values in Figures 7-7 and 7-8 sugges that there is some value for α that minimizes γ_D. It is also interesting how the value of the test current ϕ affects γ_D. It is trivial that noise weakens the solution so there is no motive to examine its effect. The results are illustrated in Figure 7-9 and 7-10, where parameters ϕ and λ are tuned. Considering the interaction of α and h in 4.7, λ is defined as

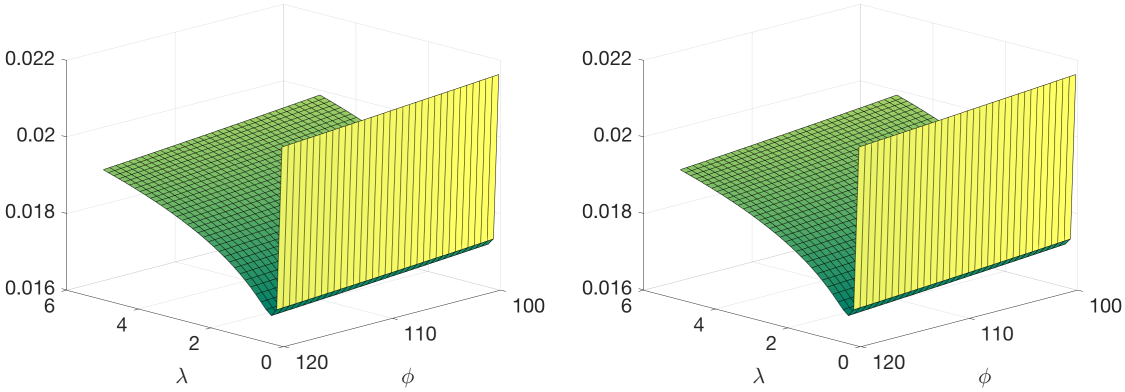
$$\lambda = \frac{\sqrt{\alpha}}{h^2}.$$

For accurate analysis, the noise added to g is kept constant. It is worth mentioning that in real life the optimum value for λ can only be guessed, since the correct temperature distribution is unknown and cannot be obtained.



(a) S_1 (b) S_2

Figure 7-9: γ_D against ϕ and λ, $[\lambda, \phi]$.



(a) S_1 (b) S_2

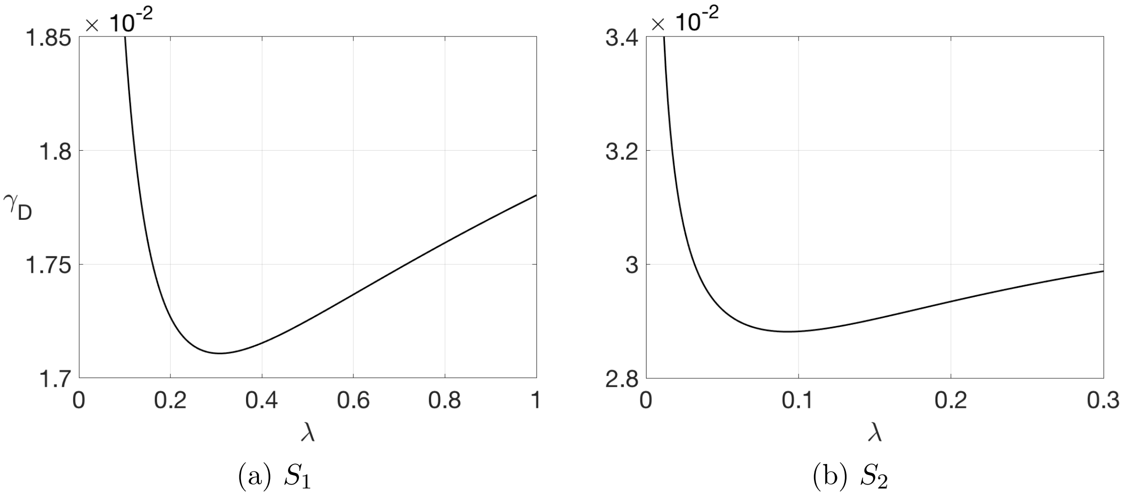
Figure 7-10: γ_D against ϕ and λ, $[\lambda^-, \phi^+]$.

Figure 7-9 suggests that the optimum parameters are located in the $[\lambda^-, \phi^+]$ corner which is examined more closely in Figure 7-10. With this it is clear that $\gamma_D \to 0$ when $\phi \to \infty$. This can be understood through stability in Section 7.1. The higher the current in the network, the larger the relative difference g_D when a element in γ is manipulated. Since γ_D stays constant,

$$\frac{g_D}{\gamma_D}$$

grows with ϕ. This is interpreted as large changes in γ_D inducing large changes in g_D. Thus making the problem well-posed and easily solvable. This statement is valid in theory. However, since noise is kept constant for this analysis, this conclusion would not hold in practise. Because increasing the current would also increase the noise. This effect can not be studied in this environment.

Since Figure 7-10 is similar for both S_1 and S_2, the effect of λ on γ_D is examined more closely in Figure 7-11.



(a) S_1 (b) S_2

Figure 7-11: γ_D against λ.

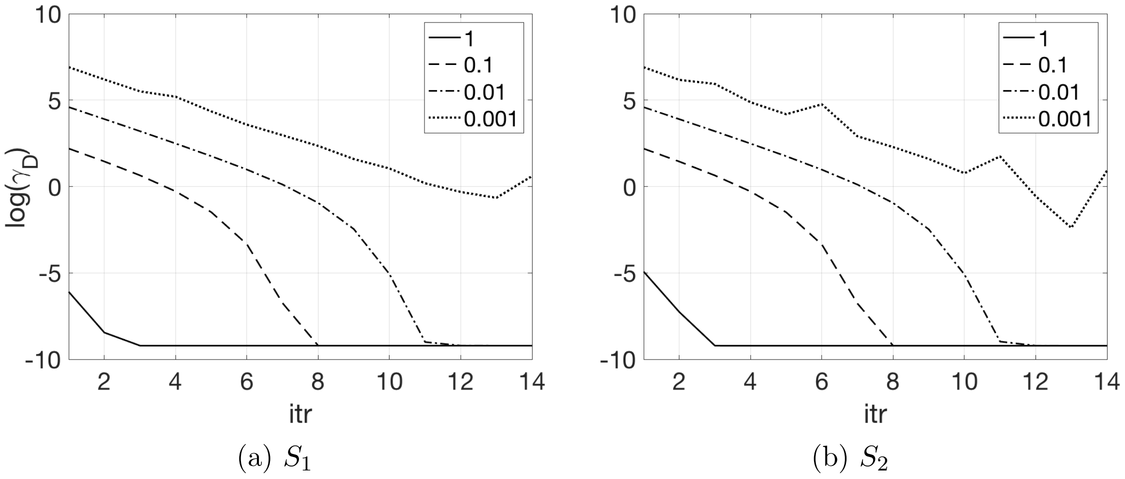
By LS-approximation the minimizer of γ_D is found to be

$$S_1 : \bar{\lambda} = 0.2221 \qquad S_2 : \bar{\lambda} = 0.0931$$

It is reasonable to assume that the minimizer $\bar{\lambda}$ is unique for the temperature distribution of a surface and the dimension of γ. Whereas greater test current always yields more accurate solutions. It also seems that the more oscillating the distribution, the less accurate the solution.

7.4 Convergence

The Gauss-Newton algorithm is known to work better on small-residual problems than on large-residual problems and converge if the initial guess is ‘close enough’ to the solution [7]. The convergence rates for S_1 and S_2 with different initial guesses are examined in Figure 7-12 with γ_D plotted on a logarithmic scale.



(a) S_1 (b) S_2

Figure 7-12: Convergence for S_1 and S_2 with varying initial guesses.

The range $[0, ..., 1]$ for initial guesses is used since the grid sensor anticipates heated object that lower conductance. For S_1 and S_2 the algorithm converges similarly and on the same solution even with inaccurate guesses. The base temperature seams to be the optimal guess that converges in only 4 iterations, while the other guesses take considerably longer. The algorithm still converges when the initial guess is 1% of the base temperature. However, it breaks down when the guess reaches 0.1%. Again, the conditions and the application of the grid sensor will dictate the base temperature and thus the optimal initial guess.

8 Summary

This thesis explored the mathematics behind an alternative method to monitor temperature fields, a temperature grid sensor. The grid sensor is composed of a resistor network and realized using the Gauss-Newton algorithm paired with Tikhonov regularization. A Laplacian matrix with the Neumann boundary condition was used for regularization. Obtaining he solution required careful formulation of the problem and tedious computational indexing. It involved matrix derivation, block reduction and non-linear least square manipulation. Imposing the Neumann boundary condition required the imagining of ghost points outside the resistor network.

MATLAB-code was written to implement the algorithm and tested on some exemplary cases. The algorithm yielded accurate temperature distributions despite the added noise to the measurements. It is concluded that more accurate results are obtained with resistor materials that have a suitable temperature sensitivity range, greater test current and smaller noise. The optimum weight for regularization is found to depend on the underlying temperature distribution and the dimensions of the grid sensor. The parameter values should be fitted for the application and the environment in which the grid sensor is used, to yield the most accurate results.

The Laplacian matrix with the Neumann boundary condition worked well, but some other regularization term or boundary condition might be even more suited for this problem. The algorithm was not tested on extremely large grid sensors, where it might have broken down. Also only specific distributions were tried out. Since no actual grid sensor was constructed, some mechanical issues were probable not yet uncovered. For example, problems with gathering measurements could occur. Considering computational time, it takes a desktop computer around 22 seconds to produce an accurate distribution for a sensor with around a thousand resistors, which can be unacceptably long for some applications. Altogether, the algorithm performed well but should be further tested in order for it to suit industrial use.

References

[1] Schäfer, T., Schubert, M. and Hampel, U. *Temperature Grid Sensor for the Measurement of Spatial Temperature Distributions at Object Surfaces.* Sensors vol. 13 p. 1593–1602, 2013.

[2] Jennifer, L. M. and Siltanen, S. *Linear and Nonlinear Inverse Problems with Practical Applications.* ISBN 1611972345 13 p. 3–6, 2012.

[3] Curtis, E. B. and Morrow, J. A. *Inverse Problems for Electrical Networks.* Mathematics Department, University of Washington, Seattle, USA. ISBN 9810241747 p. 27–58, 2000.

[4] Kaipio, J. P. *Käänteisongelmat* University of Kuopio, Department of Applied Physics, Kuopio, Finland. ISSN 0788-4672 p. 27–58, 1998.

[5] Kaipio, J. P. and Somersalo, E. *Statistical and Computational Inverse Problems* Springer Science. ISBN 0-387-22073-9 p. 24–26, 2005.

[6] Chen, L. *Finite Difference Methods* Department of Mathematics, University of California. 2017.

[7] Ferreira, O. P., Gonçalves, M. L. N., Oliveira, P. R. *Local convergence analysis of the Gauss–Newton method under a majorant condition.* Journal of Complexity vol. 27, Issue 1, 2011, p. 111-125.