Ray-Tracing Problems for Tomographic Reconstruction in Materials Science

Camilla H. Trinderup

Kongens Lyngby 2011
IMM-M.Sc.-2011-79

Technical University of Denmark
DTU Informatics
Building 305, DK-2800 Kongens Lyngby, Denmark
Phone +45 45253351, Fax +45 45882673
reception@imm.dtu.dk
www.imm.dtu.dk

IMM-M.Sc.: ISSN XXXX-XXXX

Summary

This master thesis deals with the formulation of a mathematical model for the *General Ray Tracing Problem* seen in the context of a diffraction problem. The experimental set-up that lies behind the model takes basis in the science of crystallography. The model is an inverse problem and a discrete setting of it will be considered throughout the thesis.

With the purpose of gaining a thorough understanding of the problem, is a simplified model considered at first. The analysis of this model revealed weaknesses about the model, which were factors that could then be encountered in the formulation of a more complex model. In order to make the model more complex and by this reaching a model describing the real experiment better is a blurring of the diffraction patterns is introduced. To gain an understanding of the model is the Singular Value Decomposition used as a tool for the analysis of the problem. For solving the inverse problem, different classes of iterative methods are considered.

As well as presenting the findings of the model will the thesis present simulation studies and evaluations of the performance of the iterative solvers in accordance with these simulated test problems.

Resumé

Dette kandidatspeciale omhandler formuleringen af en matematisk model for det, der kaldes *General Ray Tracing Problem*. Mere specifikt vil der blive taget udgangspunkt i et diffraktionsproblem. Den eksperimentelle opsætning bag modellen tager udgangspunkt i krystallografi. Problemet kan karakteriseres som et inverst problem. Ud fra en kontinuert formulering af problemet vil diskret model blive behandlet i opgaven.

Som udgangspunkt for at forstå problemstillingen bliver en simpel model betragtet først. Analysen af modellen afslørede både svagheder og styrker ved denne, som kan bruges i det videre arbejde med udviklingen af den mere realistiske form af modellen, hvor også sløring på diffraktionbillederne introduceres. Dette gøres for at komme et skridt nærmere beskrivelsen af den virkelige eksperimentelle situation. Gennem hele opgaven er *Singular Value Decomposition* et vigtigt værktøj i analysen og forståelsen af modellerne. Til at løse det opstillede inverse problem bliver flere forskellige iterative metoder taget i betragtning.

Udover en analyse af den opstillede model, præsenterer opgaven studier af opstillede testproblemer, de iterative metoders løsninger på disse problemer og en sammenligning af resultaterne.

Preface

This master thesis is prepared at the department of DTU Informatics at The Technical University of Denmark. The thesis marks the completion of the master degree in Mathematical Modelling and Computation and represent a workload of 35 ECTS points. The work has been prepared during the spring and summer of 2011, more specifically from February 14th till October 1st. The work has been conducted under the supervision of Professor Per Christian Hansen and Senior Scientist Søren Schmidt from DTU Risø.

Lyngby, October 1st 2011

Camilla Himmelstrup Trinderup

List of Symbols

The following table lists the commonly used symbols in the thesis.

Symbol	Quantity	Dimension
$\boldsymbol{A}$	System matrix	$m \times n$
$\boldsymbol{a}_i$	i'th row of $\boldsymbol{A}$	m
$\boldsymbol{b}$	right-hand side	m
$\boldsymbol{b}^{\text{exact}}$	exact right-hand side	m
b_j	j'th element of the vector $\boldsymbol{b}$	scalar
$\boldsymbol{e}$	noise vector	m
e_j	j'th element of the vector $\boldsymbol{e}$	scalar
k	iteration number	scalar
λ	relaxation parameter	scalar
m,n	matrix dimensions	scalars
N_d	No. of gridpoints of the detector	scalar
N_s	No. of gridpoints of the detector in 2D problem	scalar
N_w	No. of gridpoints of the source in 1D problem	scalar
N_ϕ	No. of gridpoints of the radial grid in 2D problem	scalar
N_θ	No. of gridpoints of the angular grid	scalar
ϕ	radial angle	
$\mathcal{P}$	Poisson distribution	
σ_i	i'th singular value	scalar
Σ	matrix with singular values in the diagonal	$m \times n$

$\boldsymbol{U}$	matrix with all left singular vectors	$m \times m$
$\boldsymbol{u}_i$	i'th left singular vector	m
$\boldsymbol{V}$	matrix with all right singular vectors	$n \times n$
$\boldsymbol{v}_i$	i'th right singular vector	n
φ_i	i'th regularization parameter	scalars
θ	angle of the cone in the 2D model	
$\boldsymbol{x}$	solution of $\boldsymbol{Ax} = \boldsymbol{b}$	n
$\boldsymbol{x}^{[k]}$	solution in the k'th iteration	n
$\boldsymbol{x}^{\text{exact}}$	exact solution	n
z, w, y, t	cartesian axes in the set-up of the problem	

Contents

Chapter 1

Introduction

A sample of a metal or an alloy consists of not only a big crystal lattice, but a lot of grains that each have a certain orientation of the lattice. In [2] it is described how it it possible to determine the placement of these grains using the method called three-dimensional X-ray diffraction microscopy. This method deals with the determination of the grain structure, inside a metal with polycrystal structure, from diffraction images. In this setting diffraction refers to the change of direction for an X-ray when it penetrates a sample of a polycrystal structure. The experiments have been conducted at the European Synchrotron Radiation Facility in Grenoble, France, and the experimental setup consists of the polycrystalline sample, an X-ray beam and a detector, in the form of a CCD. For several measurements for different positions of the sample, it is then possible to use the detections or projections of the X-rays, denoted the diffraction images, as a basis for reconstructing the grain structure within the sample. The information for two-dimensional layers of the sample is stacked and a special routine connects the layers such that a three-dimensional model of the grains in the polycrystal is obtained. The grains of the samples in this experiment are what one could regard as perfect grains. This means that when the X-rays are diffracted within the sample due to a specific grain, they are emitted in the same direction. Due to the parallel properties of the diffracted rays, is it only necessary to have one detector in order to determine the two-dimensional structure in the layers. By determining the grain structure before and after, e.g., an annealing of a certain material, it is possible to see what influence this process has had on the grain structure in the material. This can help us in the understanding of the material properties.

The above described results are what the material scientists have been able to obtain so far. What we wish to do now, is not only to be able at working with polycrystals that have perfect grains, but also to be able to reconstruct the grain structure of a polycrystal with deformed grains. The great difference going from perfect to deformed grains, is that when the X-rays are diffracted due to a specific grain, they will no longer be parallel. This means that it is no longer enough to have just one detector to figure out where a ray emits from. We know that the rays emit in a straight line and it is therefore necessary to have at least two points to find this line and the course of the ray. So the alternative experimental set-up is to instead have three detectors. Two detectors for determining the course of the rays and a third for verifying. This leads us to the underlying problem of this thesis work - the General Ray Tracing Problem.

Even though the problem setting of this thesis originates from the science of crystallography, it will not be the main focus. This would involve a thorough understanding of the science and it is not a prerequisite for understanding the ray tracing problem. So in a way one could say that in this thesis work will the model disengage from the crystallography, though it still originates from it.

The goal is to set up a mathematical model that will describe the process of diffraction and ray tracing in general terms. This model will be an inverse problem and it will be discretized in such a way that it will be possible to use deterministic solvers in order to reconstruct the original signal. Different aspects of the process of solving a discrete inverse problem will be investigated along with different options for each of the solvers. As stated earlier the model will disengage from the crystallography properties of the problem, but the experimental set-up described above with three detectors will still be used throughout the thesis. The problem will be grasped by first setting up a mathematical model for a simplified version of the experimental set-up. This initial model will be investigated thoroughly before moving on to expanding the model to more dimensions. After defining the model for the more complex set-up, discretizing and solving it, another aspect will be added in terms of the blurring that happens at the CCDs. This is done in order to get as close to the real world problem described above as possible. The goal is not to be able to solve a real world problem, but to reach a model that describes the mathematics behind the ray tracing problem. If we are able to do this and show that this problem is solvable, are we one step closer to solving a real world problem.

Chapter 2

Underlying Theory

This first chapter is a short introduction to the different mathematical tools that will be widely used throughout this thesis. The goal of the thesis work is to make a mathematical model describing the experiments made at the European Synchrotron Radiation Facility in Grenoble, France, and discretize this model in order to reach a large-scale system of linear equations to be solved - a so-called inverse problem. In order to solve this problem properties of the system matrix must be investigated and this is where the *Singular Value Decomposition* comes into action. After this analysis one can try to solve the problem. Different direct and iterative solvers will be used in this work and they are described in the third section of this chapter. The fourth section will describe different options for stopping criteria for the solvers and finally a stepwise guide on how to solve an inverse problem will be presented.

2.1 Inverse Problems

An inverse problem consists of reconstructing or solving for something unknown on the basis of external measurements. Inverse problems arise in e.g. medical imaging and geophysical measurements. A well-known inverse problem is the Radon-transform that is the basis of CT (computerized tomography) scanners.

The problem described in the introduction to this thesis does not fit the normal formulation of a tomography problem, but is still have similarities and it is an inverse problem. For further introduction to tomographic imaging see [7].

When discretizing an inverse problem a system of linear equations, $\boldsymbol{Ax} = \boldsymbol{b}$, arise. The system matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$. When $m > n$ the system is said to be overdetermined, opposite to underdetermined when $m < n$. Inverse problems are most often ill-posed problems, in the sense that they do not satisfy the definition of well-posed problem stated by Hadamard - see [3]. A well-posed problem satisfies the three crucial requirements. For one there must exist a solution, secondly this solution must be unique and third, the solution must be stabile in the sense that it depends continuously on the data. The Picard Condition states this. Later the Discrete Picard Condition will be discussed. Where an inverse problem is said to be ill-posed, is a discrete inverse problem ill-conditioned and the Discrete Picard Condition states when we are able to solve these ill-conditioned problems.

2.2 Singular Value Decomposition

An important mathematical tool for investigating properties of an inverse problem is the so-called *Singular Value Decomposition* - hereafter denoted SVD. It is similar to the *Singular Value Expansion* that is relevant in the continuous case. For more on this see [3]. In the SVD the model matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is expressed in terms of three matrices, $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{\Sigma}$:

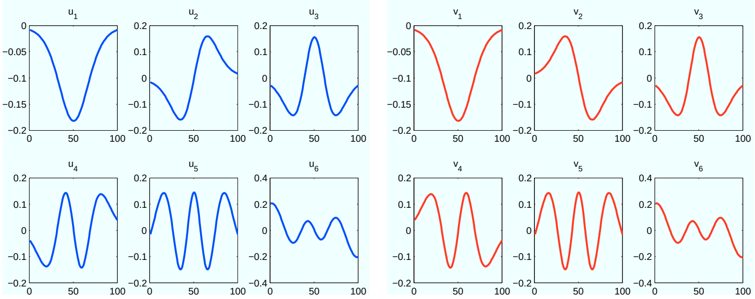
$$\boldsymbol{A} = \boldsymbol{U\Sigma V}^T = \sum_{i=1}^{n} \boldsymbol{u}_i \sigma_i \boldsymbol{v}_i.$$

The columns of $\boldsymbol{U}$, $\boldsymbol{u}_i$, and $\boldsymbol{V}$, $\boldsymbol{v}_i$, are called the left and right singular vectors respectively. $\boldsymbol{\Sigma}$ is a diagonal matrix with the singular values σ_i in the diagonal. The singular values are sorted in descending order. For $m \geq n$ is the solution $\boldsymbol{x}$ expressed in terms of the SVD

$$\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b} = \left(\boldsymbol{U\Sigma V}^T\right)^{-1}\boldsymbol{b} = \sum_{i=1}^{n} \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{v}_i. \tag{2.1}$$

We see that the singular vectors, especially the right singular vectors, has great importance in the reconstruction - they determine what we are able to reconstruct in $\boldsymbol{x}$.

The number of oscillations for each of the singular vectors will increase with their index. In Figure 2.1 both the left and right singular vectors are plotted



(a) First six left singular vectors. (b) First six right singular vectors.

Figure 2.1: Singular vectors of the `shaw` test problem.

for the system matrix of the test problem `shaw` from the *Regularization Toolbox* - see [4]. In the figure it is possible to see how the oscillations of the vectors increase. This phenomena means that we reconstruct $\boldsymbol{x}$ by adding a certain number of functions together, that each describe some variation in the solution.

Before moving on an important thing to mention is the *Discrete Picard Condition*. We know that the singular values is a descending sequence. But what happens with the fraction $(\boldsymbol{u}_i^T\boldsymbol{b})/\sigma_i$ as σ_i becomes smaller? $\boldsymbol{u}_i^T\boldsymbol{b}$ are also called the SVD coefficients of $\boldsymbol{b}$. In Figure 2.2 a so-called Picard plot is seen. The SVD coefficients and singular values are plotted with respect to their index i. The plot shows that the singular values $\sigma_{\mathbf{i}}$ descends as expected and that they until some index decrease slower than $\boldsymbol{u}_i^T\boldsymbol{b}$. But after this point, the coefficients $(\boldsymbol{u}_i^T\boldsymbol{b})/\sigma_i$ will no longer decay and the solution will be dominated by the larger SVD coefficents corresponding to the smaller singular values. In some cases the coefficients $(\boldsymbol{u}_i^T\boldsymbol{b})/\sigma_i$ will grow larger and larger and make the high frequency components dominant in the solution. As was seen in the plots of the singular vectors, will the number oscillations in $\boldsymbol{v}_i$ increase with its index, i.e., they start as low-frequent signals and ends as high-frequent signal. The noise in the right-hand side $\boldsymbol{b}$ is also high-frequent and therefore the above described issue will lead to the fact that the reconstruction by (2.1) will be dominated by noise.

The *Discrete Picard Condition*, that was defined and investigated in [5], states that the reconstruction is consistent with $\boldsymbol{x}^{\text{exact}}$ if the SVD coefficients, $|\boldsymbol{u}_i\boldsymbol{b}|$, decrease faster than the singular values until some index τ defined by the noise level. If the *Discrete Picard Condition* is satisfied the problem will be consistent with $\boldsymbol{x}^{\text{exact}}$ if we solve the problem by a regularized method. Regularization methods will be discussed in the coming section. The Picard Plot in Figure 2.2 shows that the discrete Picard condition is satisfied for the `shaw` test problem.

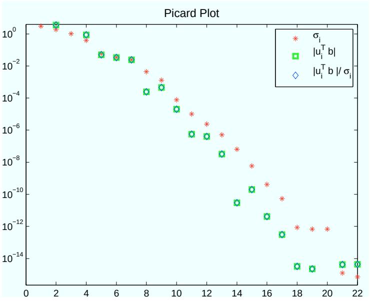


Figure 2.2: Picard plot for the `shaw` test problem.

2.3 Solution Methods

Different reconstruction methods will be considered throughout the thesis and in this section a short introduction to each of them is given. When working with a system of linear equations $\boldsymbol{Ax} = \boldsymbol{b}$ the naive solution can be computed by inverting $\boldsymbol{A}$. But this would lead to solutions influenced a lot by the noise in data, which the evaluations underneath shows.

$$\boldsymbol{x}^{\text{naive}} = \boldsymbol{A}^{-1}\boldsymbol{b} = \boldsymbol{A}^{-1}(\boldsymbol{b}^{\text{exact}} + \boldsymbol{e}) = \boldsymbol{A}^{-1}\boldsymbol{b}^{\text{exact}} + \boldsymbol{A}^{-1}\boldsymbol{e}. \tag{2.2}$$

We see that the solution will have two components - one consisting of the exact data and one consisting of inverted noise. The inverted noise will most often dominate the solution and the naive solution is therefore often useless. This fact is similar to what was concluded in the previous section about the SVD solution and the noisy part of the right-hand side $\boldsymbol{b}$. Besides this issue it will often not be computationally possible to invert $\boldsymbol{A}$ because of its size.

What can be done to solve this issue, is to regularize the solution by introducing filter factors in the SVD solution from (2.1). This is possible when working on smaller test problems - when the problem is large it is difficult to compute the SVD of the matrix $\boldsymbol{A}$. The regularized solution is given by

$$\boldsymbol{x}^{\text{reg}} = \sum_{i=1}^{n} \varphi_i \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{v}_i, \tag{2.3}$$

where the filter factors determine which and how much of each SVD component that should be included in the solution.

In the truncated SVD solution the filter factors are given by

$$\varphi_i = \begin{cases} 1, \text{ for } i = 1, \ldots, k \\ 0, \text{ for } i > k \end{cases} , \tag{2.4}$$

which means that we will include only the first k singular values and vectors in the solution. When finding the optimal value of the truncation parameter k, it is helpful to look at the Picard plot that was described in the previous section. The truncation parameter k will be chosen as the index i at which the SVD coefficients start to increase. This indicates that the SVD coefficients are decreasing faster than the singular values. This is the point that in the discrete Picard Condition is denoted τ.

In Tikhonov regularization the filter factors are given as

$$\varphi_i^{[\lambda]} = \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \approx \begin{cases} 1 & \text{for}\, \sigma_i \gg \lambda \\ \sigma_i^2/\lambda^2 & \text{for}\, \sigma_i \ll \lambda. \end{cases} \tag{2.5}$$

The filter factors for Tikhonov regularization are dependent on the regularization parameter λ. Unlike the truncated SVD that was described above the filter factors of Tikhonov regularization includes all SVD components in the solution, but dependent on the value of λ it emphasizes the SVD components with the largest corresponding singular value. This means that at some point the filter factors will dampen the high-frequency components of the SVD solution, but there will still be a small part of them included in the solution. Analogous to image reconstruction where the high-frequency components are important in order to reconstruct sharp edges and contrasts in an image, the high-frequency components in the diffraction problem will make sure that the transition from high intensities to low or zero intensity in the solution can be rapid.

Tikhonov regularization can be formulated as a least squares problem where we wish to minimize

$$\boldsymbol{x}^{\text{reg}} = \min_{\boldsymbol{x}}\{\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 + \lambda^2\|\boldsymbol{x}\|_2^2\}. \tag{2.6}$$

In this formulation it is unnecessary to compute the SVD, which can be advantageous.

The regularization methods just described use the SVD of the model matrix in the computation of the solution. But as the problem gets larger, it is not possible to compute the SVD or solve the Tikhonov problem. When this is the case we can use iterative methods to solve our problem. The first iterative method that will be considered is the Landweber method. In this method we start with an initial 'guess' on the solution, $\boldsymbol{x}^{[0]}$, and the iterates are then given as

$$\boldsymbol{x}^{[k+1]} = \boldsymbol{x}^{[k]} + \omega \boldsymbol{A}^T(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{[k]}), \quad k = 0, 1, 2, \ldots, \tag{2.7}$$

where ω is a constant that satisfies $0 < \omega < 2/\sigma_1^2$. The Landweber iterates can be expressed in terms of the SVD as well, where the filter factor for the k'th iterate is given by

$$\varphi_i^{[k]} = 1 - (1 - \omega\sigma_i^2)^k, \quad i = 1, 2, \ldots, n. \tag{2.8}$$

The landweber method belongs to a class of methods called *Simultaneous Iterative Reconstruction Techniques*, SIRT. For further readings on these methods see [3] or [6]. The other SIRT methods that will be considered in this project are Cimmino, DROP, CAV and SART. In general are the iterations of a SIRT method given as

$$\boldsymbol{x}^{[k+1]} = \boldsymbol{x}^{[k]} + \lambda_k \boldsymbol{T}\boldsymbol{A}^T\boldsymbol{M}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{[k]}). \tag{2.9}$$

The parameter ω in (2.7) is a special case of a fixed λ_k. The parameters that varies for the different SIRT methods are the matrices $\boldsymbol{T}$ and $\boldsymbol{M}$. All SIRT methods can be formulated as spectral decompositions, just as Landweber. The parameter λ_k is called the relaxation parameter and plays a great role in the performance of the SIRT methods. It can either be constant or adaptive, but it has to lie within the interval $[\epsilon, \frac{2}{\sigma_1^2} - \epsilon]$, with ϵ being an arbitrarily small number and σ_1 the first and largest singular value. The reason for not just looking at one instance of a SIRT method is that even though the methods are from the same class, they can still behave differently on a given problem.

The second class of iterative methods that will be considered is the *Algebraic Reconstruction Techniques*, ART. The classic ART method is called Kaczmarz's Method. The iterates of the ART methods are slightly more complex to describe than those of the SIRT methods, since for each iterate the method runs through all rows of $\boldsymbol{A}$. The iteration will run as follows:

$$\begin{aligned}
&\boldsymbol{x}^{[k^{(0)}]} = \boldsymbol{x}^{[k]} \\
&\text{for}\, i = 1, \ldots, m \\
&\qquad \boldsymbol{x}^{[k^{(i)}]} = \boldsymbol{x}^{[k^{(i-1)}]} + \frac{\boldsymbol{b}_i - \boldsymbol{a}_i^T\boldsymbol{x}^{[k^{(i-1)}]}}{\|\boldsymbol{a}_i\|_2^2}\boldsymbol{a}_i \\
&\text{end} \\
&\boldsymbol{x}^{[k+1]} = \boldsymbol{x}^{[k^{(m)}]}.
\end{aligned} \tag{2.10}$$

Kaczmarz' method has been observed to converge fast within the first iterations, and it has been used a lot in computerized tomography. The other two ART methods that will be considered in this project are randomized and symmetric Kaczmarz'. Both the SIRT and ART methods that will be used are implemented in the MATLAB AIRTools package.

The last iterative method that will be considered is the Conjugated Gradient Least Squares (CGLS) method. CGLS is a Krylov subspace method. The

Krylov subspace is given in terms of $\boldsymbol{A}$ and $\boldsymbol{b}$ as

$$\mathrm{K}_k = \mathrm{span}\{\boldsymbol{A}^T\boldsymbol{b}, (\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{A}^T\boldsymbol{b}, (\boldsymbol{A}^T\boldsymbol{A})^2\boldsymbol{A}^T\boldsymbol{b}, \ldots, (\boldsymbol{A}^T\boldsymbol{A})^{k-1}\boldsymbol{A}^T\boldsymbol{b}\}. \tag{2.11}$$

The CGLS method then finds the k'th iterate by solving

$$\boldsymbol{x}^{[k]} = \mathrm{argmin}_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2 \quad \text{s.t.} \quad \boldsymbol{x} \in \mathrm{K}_k. \tag{2.12}$$

According to [3] the CGLS method corresponds to a regularization method, just as TSVD and Tikhonov. The proof is simple and is based on the fact that the k'th iterate lies in the Krylov subspace and must therefore be a linear combination of the basis vectors of the subspace, which are

$$\boldsymbol{A}^T\boldsymbol{b}, (\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{A}^T\boldsymbol{b}, \ldots, (\boldsymbol{A}^T\boldsymbol{A})^{k-1}\boldsymbol{A}^T\boldsymbol{b},$$

such that

$$\boldsymbol{x}^{[k]} = c_1\boldsymbol{A}^T\boldsymbol{b} + c_2(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{A}^T\boldsymbol{b} + \ldots + c_k(\boldsymbol{A}^T\boldsymbol{A})^{k-1}\boldsymbol{A}^T\boldsymbol{b}. \tag{2.13}$$

This results in the filter factors being

$$\varphi_i^{[k]} = c_1\sigma_i^2 + c_2\sigma_i^4 + c_3\sigma_i^6 + \ldots + c_k\sigma_i^{2k}, \quad i = 1, \ldots, n. \tag{2.14}$$

This show us that we are able to formulate both the CGLS method and the SIRT methods as spectral filter methods as stated in (2.3). For further readings on the CGLS method see [3] or [9].

Common for the three classes of iterative methods is the concept of semi-convergence. When looking at the convergence of the methods, when carried out in practice, we will often see that within the first iterations the methods converge toward the exact solution, but at some iteration they reach the closest they can get to the solution. The iterates will after this diverge from the exact solution and converge toward the naive solution instead. This fact will be used in the stopping criteria introduced in the next section. Along with semi-convergence describes Hansen et.al. in [10] how non-negativity constraints can be imposed on the SIRT and ART methods described above. Unlike the CGLS method it is for these methods possible to impose the non-negativity constraint on an iterate and still use this solution in the next iteration. The non-negativity constraints will be considered throughout the thesis.

In order to be able to compare the performance of the iterative SIRT and ART methods the concept of work units (WU) were introduced in [6]. The concept of work units refers to the number of matrix-vector multiplications per iteration. So when a system matrix $\boldsymbol{A}$ of size $m \times n$ is considered is one work unit given as $\mathrm{WU} = 2mn$. This means that the SIRT and ART methods all use two WU per iteration, except for symmetric Kaczmarz' that uses the double amount of work units. The CGLS method also uses 2WU per iteration. The concept of work units will be used as a tool for comparison of the performance of the iterative methods throughout the thesis.

2.4 Stopping Criteria

Earlier we saw how visual inspection of the Picard plot could give an indication of the truncation parameter for TSVD. For the other methods described this is not an option, so how do we choose the optimal value of the regularization parameter or optimal number of iterations? Several methods are available - all based on different principles. Below three different methods will be considered. They are chosen on the basis of them being implemented in the MATLAB AIRTools package along with training algorithms for them. The training algorithms are functions that use a known data set - an exact solution and right-hand side - to train the parameters of the stopping rules. The goal is then if one has a real dataset, i.e., a noisy right-hand side, that has the same characteristics as the simulated dataset, then the trained value of the parameter will also be suitable for the real data. These training algorithms are also implemented for the relaxation parameter of the SIRT and ART methods and will be considered when we reach the point of using the algorithms for reconstruction.

Two of the stopping criteria are based on a rule called the α, β-rule. The rule was introduced in [1] by Elfving and Nikazad and is based on the application for the SIRT methods. Though in this setting it will be used on both the SIRT and ART methods, thus the derivation will take basis in the SIRT methods. The rule is based on the fact that we wish to have have monotonicity in our iterative solutions, such that

$$\|\boldsymbol{x}^{\text{exact}} - \boldsymbol{x}^{[k]}\|_2 > \|\boldsymbol{x}^{\text{exact}} - \boldsymbol{x}^{[k+1]}\|_2 \tag{2.15}$$

is obtained. The first iteration where this state is not present, is the iteration at which the method should stop. The rule states that this first index $k = k_{\alpha,\beta}$ where

$$\frac{d_{\alpha,\beta}}{\|\boldsymbol{r}^{[k]}\|_2} \leq \tau \|\boldsymbol{e}\|_2 \|\boldsymbol{M}^{1/2}\|_2, \tag{2.16}$$

is the optimal index and the routine should be stopped. In (2.16) $\boldsymbol{M}$ refers to the matrix of the iterations of the SIRT methods and

$$\boldsymbol{r}^{[k]} = \boldsymbol{M}^{1/2}(\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{[k]}) \tag{2.17}$$

$$d_{\alpha,\beta} = \left(\boldsymbol{r}^{[k]}\right)\left((2\alpha + \beta - 1)\boldsymbol{r}^{[k]} + (1-\beta)\boldsymbol{r}^{[k+1]}\right)^T . \tag{2.18}$$

It is the parameter τ in (2.16) that can be trained on a simulated set of data. The error level, $\|\boldsymbol{e}\|_2$ is a crucial factor for the stopping criterion. We will later see that the noise level for the problems considered is easily approximated.

2.4.1 The Discrepancy Principle

The Discrepancy Principle (DP) is the simplest of the three methods considered and is a special case of the α, β-rule with $(\alpha, \beta) = (0.5, 1)$. It is built on the fact that at some point the will residuals reach the level of the error in the data. After this has happened the solution will tend toward the naive solution. So to state it simple the discrepancy looks at the residuals and compares these to the error level. When the residuals are lower than the error level, such that,

$$\|\boldsymbol{A}\boldsymbol{x}^{[k]} - \boldsymbol{b}\|_2 \leq \tau \|\boldsymbol{e}\|_2, \tag{2.19}$$

the iterations will stop. τ is a 'safety' factor and $\|\boldsymbol{e}\|_2$ is the noise level. So once the regularized solution or iterates have reached a certain level, where the residual is close to the error level it will stop the algorithm. The disadvantage about DP is that in order to reach the optimal solution the error level must be known or be approximated very well. When working with real life data this is a difficult task.

2.4.2 The Monotone Error Rule

Another method for stopping the algorithms at the optimal solution is the *Monotone Error Rule* (ME). It is another special case of the α, β-rule with $(\alpha, \beta) = (1, 0)$ such that

$$d_{\mathrm{ME}} = d_{1,0} = \left(\boldsymbol{r}^{[k]}\right)^T \left(\boldsymbol{r}^{[k]} + \boldsymbol{r}^{[k+1]}\right), \tag{2.20}$$

and the rule then states that the iterations should be stopped when

$$\frac{d_{\mathrm{ME}}}{\|\boldsymbol{r}^{[k]}\|_2} \leq \tau \|\boldsymbol{e}\|_2 \|\boldsymbol{M}^{1/2}\|_2 \tag{2.21}$$

As well as DP, ME also suffer under the fact that the error level has to be known or estimated well in order to reach a great result. ME can not be used together with the ART methods, but only with the SIRT methods.

2.4.3 Normalized Cumulative Periodogram

The last stopping criteria that will be considered is the Normal Cumulative Periodogram (NCP). As was done for the general α, β-rule the residual is defined as

$$\boldsymbol{r}^{[k]} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{[k]}. \tag{2.22}$$

We know that at some iteration $\boldsymbol{A}\boldsymbol{x}^{[k]}$ will get as close to $\boldsymbol{b}^{\text{exact}}$ as it can and hereafter it will tend toward the naive solution where the iterates will be dominated by noise. When this happens the residual will change and look more like noise. Therefore in NCP we will consider the residuals at each iteration, or for each value of the regularization parameter, as a time series. The Fourier transform of the residuals are defined as

$$\hat{\boldsymbol{r}}^{[k]} = \text{dft}(\boldsymbol{r}^{[k]}) = \left((\hat{\boldsymbol{r}}^{[k]})_1, (\hat{\boldsymbol{r}}^{[k]})_2, \ldots, (\hat{\boldsymbol{r}}^{[k]})_m\right)^T \quad \in \mathcal{C}^m. \tag{2.23}$$

We then define the power spectrum of $\boldsymbol{r}^{[k]}$ as

$$\hat{\boldsymbol{p}}^{[k]} = \left(|(\hat{\boldsymbol{r}}^{[k]})_1|^2, |(\hat{\boldsymbol{r}}^{[k]})_2|^2, \ldots, |(\hat{\boldsymbol{r}}^{[k]})_{q+1}|^2\right)^T, \tag{2.24}$$

where q is the largest integer such that $q \leq m/2$. The NCP is then given as

$$\boldsymbol{c}\left(\boldsymbol{r}^{[k]}\right)_i = \frac{(\hat{\boldsymbol{p}}^{[k]})_2 + \cdots + (\hat{\boldsymbol{p}}^{[k]})_{i+1}}{(\hat{\boldsymbol{p}}^{[k]})_2 + \cdots + (\hat{\boldsymbol{p}}^{[k]})_{q+1}}, \quad i = 1, \ldots, q. \tag{2.25}$$

The implementation of the NCP is made such that we wish to minimize the 2-norm between the NCP and the corresponding NCP for white noise $\boldsymbol{c}_{white} = (1/q, 2/q, \ldots, 1)^T$, i.e.,

$$d(k) = \|\boldsymbol{c}(\boldsymbol{r}^{[k]}) - \boldsymbol{c}^{\text{white}}\|_2. \tag{2.26}$$

When this is the case the NCP criterion will stop the iterative method.

2.5 Solving a Discrete Inverse Problem

1. Problem and corresponding data
   - From the given data we wish to reconstruct a specific signal
2. Mathematical model
   - Formulation of a mathematical model describing the process the signal has to go through in order to be like data

   - Discretization of the model in order to reach a system of linear equations

3. Regularization methods
   - In order to solve the problem the regularization methods described earlier in this section are used

4. Parameter Choice
   - Regularization parameters
   - Choice of stopping criteria and parameters for these methods

The steps just described will be the basis of the coming chapters.

Chapter 3

One-dimensional Model

This chapter will deal with the formulation of a simplified version of the diffraction problem and the process of solving this. But before moving on to the formulation of this model it is important to gain an understanding of the problem setting.

The laboratory experiment set-up involves a small sample of a polycrystal source material that is hit by X-rays from one side, and a set of three detectors placed on the other side of the source. Two of these detectors are placed close to the sample and the third is placed further away. This set-up is sketched in Figure 3.1. From this set-up the goal is to reconstruct the properties of the material in the sample, so we are dealing with an inverse problem.

In this project we introduce a simpler inverse problem than reconstruction of the material properties from the projection directly from the source. In our work, we introduce a source plane just in front of the material sample. This creates a new inverse problem on the source plane instead of the sample. The signal in this problem is the distribution of photons coming from the plane, when projected on top the source plane. The distribution of photons depends on what has happened with the X-rays when penetrating the sample. This intermediate step is illustrated in Figure 3.1, where a plane just immediately in front of the source has been introduced. From this distribution it is in principle possible to reconstruct the material parameters - but that is outside the scope of this

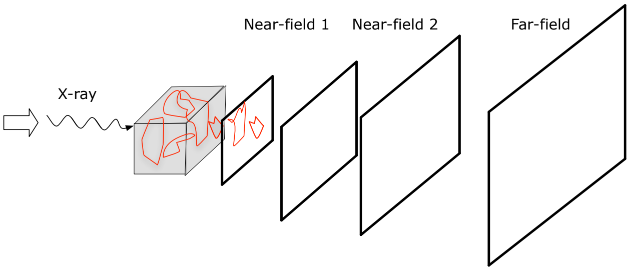


Figure 3.1: Illustration of the laboratory set-up and how it will be regarded in this project.

thesis work. Hence, in this thesis we forget that there is a real source behind the source plane, so the distribution of photons that we wish to reconstruct only has its presence on the source plane. This instance of the inverse problem is illustrated in Figure 3.2, where a new signal is present on the source plane that is independent of the rays from the source. This means that throughout the thesis inverse crime will be committed. Inverse crime arise when we use a forward model to create data, and hereafter use the same model to solve the problem. What justifies this choice is the fact that we primarily want to study and investigate the properties of the inversion method of going from data on the detector planes to a signal on the source plane. We are not interested in reconstructing the properties of the material itself.

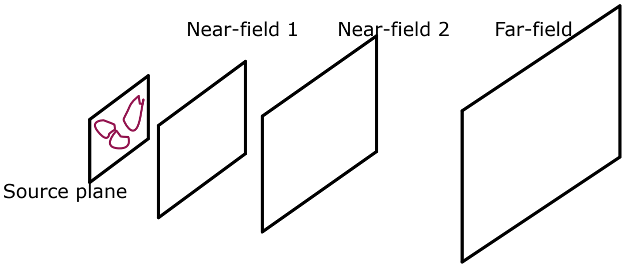


Figure 3.2: Illustration of how the problem has disengaged from crystallography and how the signal on the source plane is no longer dependent on the diffracted rays from the sample.

In this chapter a simplified version of the problem will be considered. Limiting the dimensions of the problem will give us a chance to set up a mathematical model that is simple and easy to discretize. This model can then be thoroughly investigated and be used as a basis for the more complex model. In this simplified version of the problem there is an emission axis, and detector axes are placed afterwards as illustrated in Figure 3.3. When we wish to reconstruct the intensity distribution at the source this set-up leads to dependence on two variables - where on the axis the photons are emitted and in what angle this happens.

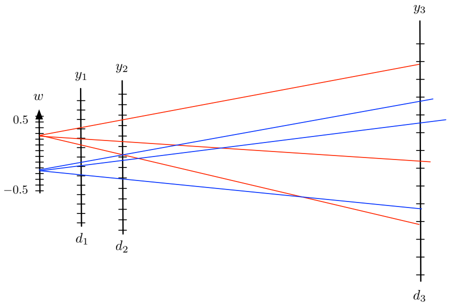


Figure 3.3: Problem set-up.

When the measurements were done, the set-up had three detectors, where the first two, called near-field detectors, are placed approximately one and two centimeters from the sample and the last one, a far-field detector, is placed 50 centimeters away. The detectors were CCDs with varying size, but they all had 2048 pixels in each direction. In Table 3.1 the dimensions of the detectors in this laboratory setting are stated along with the maximum angle each detector can detect. Figure 3.4 illustrates the situation. For the simulations conducted for this part of the project the set-up described in Table 3.2 will be used instead. It is a set-up where the three detectors all cover the same angle interval.

	Laboratory set-up		
	d_1	d_2	d_3
Distance	8 mm	18 mm	500 mm
Range	±1.54 mm	±4.61 mm	±51.2 mm
θ_{max}	±0.25	±0.28	±0.10

Table 3.1: Table of laboratory set-up.

As stated above is the third detector in the experiment conducted at the European Synchroton Radiation Facility in Grenoble meant as a far-field detector. The far-field will give rise to detections that will be like where they from a sin-

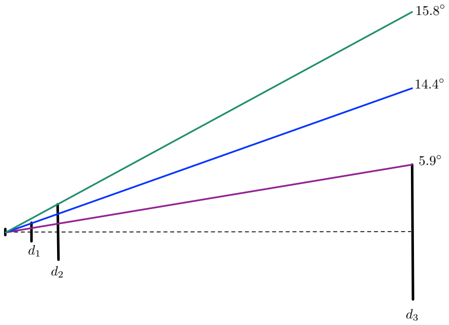


Figure 3.4: Detectable angles for the laboratory set-up.

Simulation set-up			
	d_1	d_2	d_3
Distance	8 mm	18 mm	500 mm
Range	±1.77 mm	±4.61 mm	±141.39 mm
θ_{max}	±0.28	±0.28	±0.28

Table 3.2: Table of simulation set-up.

gle point source. This means that the far-field detector will give us information about the angle distribution of the intensity distribution we wish to reconstruct. Realizing that the detectors do not cover the same angle interval, is in fact especially important for the far-field detector. The detections made at the far-field would only give us information about photons emitted within the angle interval that it can detect and would lead to lack of information about the remaining part of the angle interval. This would make the far-field less useful.

3.1 Accurate Forward Model

First step in the process of setting up the inverse problem of finding the intensity distribution at the source plane is to describe what happens in mathematical terms. This will lead to a mathematical model that can be discretized and hereafter an attempt to solve it and reach a reconstruction of the original intensity distribution can be made. Some assumptions are made in order to construct the model. As illustrated in Figure 3.3 the detectors are aligned, such that their spatial midpoints are horizontally equal. This will simplify the evaluations of the angles. Moreover in this simple first model we assume that the photons do not loose intensity as they pass through the detectors and no blurring occurs on the detectors.

The intensity distribution at the source will be described by the function f, that is dependent on the two variables w and θ:

$$f(w,\theta), \quad w \in [-0.5, 0.5] \quad \text{and} \quad \theta \in]-\pi/2, \pi/2[. \tag{3.1}$$

For a given point w_i on the source, photons are emitted in all different angles. At detector k, the photons will be detected on the CCD. This means that photons emitted at a certain angle interval will hit a certain pixel on the detector. Therefore the number of photons from a given point on the source plane, w_i, that hit the j'th pixel on the k'th detector will be given as

$$\Delta g_k(w_i, y_j) = \int_{\theta_{\text{start}}}^{\theta_{\text{end}}} f(w_i, \theta) d\theta. \tag{3.2}$$

θ_{start} and θ_{end} defines the angle interval the photons are emitted within, in order to hit the j'th interval on the k'th detector. These angles are dependent on the distance d_k of the detector:

$$\begin{aligned} \theta_{\text{start}} &= \arctan\left(\frac{w_i - y_{j-1/2}}{d_k}\right) \\ \theta_{\text{end}} &= \arctan\left(\frac{w_i - y_{j+1/2}}{d_k}\right). \end{aligned} \tag{3.3}$$

$[y_{j-1/2}, y_{j+1/2}]$ defines the j'th pixel, which is the interval around the j'th point on the k'th detector. This leads to the fact that the total light intensity detected at the j'th pixel on the detector is given by

$$g_k(y_j) = \sum_i \Delta g_k(w_i, y_j) = \sum_i \int_{\theta_{start}}^{\theta_{end}} f(w_i, \theta) d\theta \tag{3.4}$$

In Figure 3.5 three detections are seen at different distances. The detections are made in accordance with the model in (3.4). The function used for sampling values of the original intensity distribution is $f(w,\theta) = |\sin(w)\cos(\theta)|$, and the detections show that the oscillations of this function are repeated in the detections. It is also seen that the closer detectors reach higher values than the detectors further away from the source. This is because the detections at each pixel on the closer detectors will be hit by more photons, because they have not yet spread a lot.

Equation (3.4) that describes the total light intensity at a certain pixel on the k'th detector can be characterized as an inverse problem - we wish to find f from our knowledge of g, and as was stated in Section 2.1 inverse problems are ill-conditioned and difficult to solve. Moreover the dimensions of this problem are large and computation time becomes an issue when we wish to solve the problem.

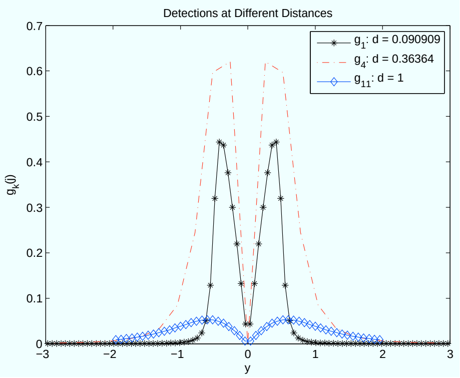


Figure 3.5: Detections at three different distances.

3.2 Discrete Forward Model

In order to discretize the problem to reach a system of linear equations as

$$\boldsymbol{Ax} = \boldsymbol{b}, \tag{3.5}$$

a proper discretization of our original solution, f, has to be chosen. The space coordinates w are in the domain $[-0.5, 0.5]$ - we can always scale it to have length 1 - and there should be N_w equidistant grid-points. This means that the grid spacing in this domain will be $h_w = 1/N_w$. For the angles $\theta \in [-\pi/2, \pi/2]$ we choose to have N_θ grid-points. The θ-domain must be symmetric around 0 radians, and have equidistant grid-points spaced with $h_\theta = \pi/N_\theta$. A discrete representation of the continuous function f, let us denote it $\boldsymbol{F}$, is introduced. $\boldsymbol{F}$ is a two-dimensional array and has dimensions $N_w \times N_\theta$, where each pixel has a constant value. Each detector is discretized with N_y^k grid points - k referring to the k'th detector - which means that the grid spacing can vary within the detectors. From the discretization of the detectors we are still able to find the angle-intervals for each of the integrals in 3.2, but in order to derive them, we must find the corresponding pixels in $\boldsymbol{F}$. The integral in (3.2) will therefore be an integral over a piecewise constant function with equidistant points defined by the θ-discretization. As seen in Figure 3.6 this corresponds to a sum of rectangle areas. We can not be sure that the angle intervals of the integral in (3.2) are in accordance with the grid intervals that was chosen by the discretization. Therefore the discretization has to take this into consideration and sometimes include just a portion of a pixel from $\boldsymbol{F}$ in the integral. This will lead to the most correct discretization. As the final step in the discretization of the intensity distribution function we choose to stack the columns of $\boldsymbol{F}$, such that

$$\boldsymbol{x} = \mathrm{vec}(\boldsymbol{F}). \tag{3.6}$$

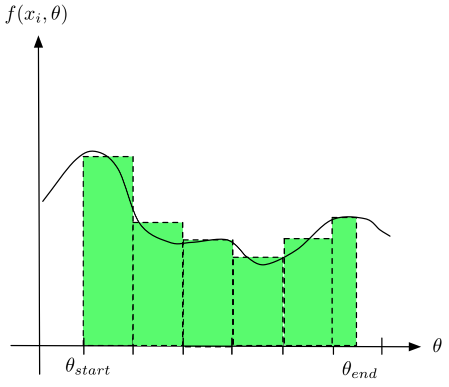


Figure 3.6: Discretization of integrals.

Similarly the right-hand side of the system of linear equations will then be the detections at each distance stacked on each other,

$$\boldsymbol{b}^{\text{exact}} = \text{vec}(\boldsymbol{G}), \tag{3.7}$$

where $\boldsymbol{G}$ is a matrix with the detections from the k'th detector in the corresponding column.

The model matrix $\boldsymbol{A}$ can now be derived by performing the forward operation for all unit vectors - the detections will then become the columns of $\boldsymbol{A}$. Hereafter will an exact solution be set up and the exact right-hand side of the system is found by a forward operation.

3.2.1 Noise

When adding noise to simulated data it is important to think about the physical matters behind the mathematical model. In this case we are detecting photons with a certain intensity, and these intensities are in a way counted at the CCD on the detector and then added together. Dealing with this kind of experiment we will assume that the noise is Poisson distributed. As opposed to Gaussian white noise, that is distributed with regards to two different parameters, the Poisson distribution only has one parameter, that is both the mean and variance. For each measurement, b_j^{exact}, the noisy version will be

$$b_j \backsim \mathcal{P}(b_j^{\text{exact}}). \tag{3.8}$$

Therefore the mean and the variance of the data are

$$E(b_j) = b_j^{\text{exact}} \tag{3.9}$$

$$V(b_j) = b_j^{\text{exact}}. \tag{3.10}$$

In Appendix A is a method on how to control the noise level presented along with an investigation of a suitable noise level for this problem.

In Section 2.4 different stopping criteria were introduced and in two of the methods, DP and ME, it is necessary to know the error level. When working with Poisson distributed noise, this is fortunately something that is easy to estimate. If we define the stochastic variable b_j as

$$b_j = b_j^{\text{exact}} + e_j, \tag{3.11}$$

the error term e_j will be dependent on the expected value of b_j, such that

$$e_j = b_j - b_j^{\text{exact}} = b_j - E(b_j). \tag{3.12}$$

The expected value of the squared error term is then

$$E(e_j^2) = E\left((b_j - E(b_j))^2\right) = V(b_j) = b_j^{exact}. \tag{3.13}$$

We are now able to find the expected value of the error level.

$$\begin{aligned} E(\|\boldsymbol{e}\|_2^2) &= E\left(\sum_{j=1}^{m} e_j^2\right) & (3.14)\\ &= \sum_{j=1}^{m} E(e_j^2) = \sum_{i=1}^{m} V(b_i) & (3.15)\\ &= \sum_{j=1}^{m} b_i^{\text{exact}} = \|\boldsymbol{b}^{\text{exact}}\|_1 \backsimeq \|\boldsymbol{b}\|_1 & (3.16) \end{aligned}$$

Table 3.3 shows how the estimated error level differs from the correct. When the number of elements in $\boldsymbol{b}$ increase - here denoted N - the difference becomes significantly smaller. When there are $N = 10^4$ points in $\boldsymbol{b}$ a relative error of 0.4% is reached, which must be said to very satisfying. Therefore it should be possible to use the stopping criteria introduced earlier by estimating the noise level in this way.

3.3 SVD Analysis

The SVD that was described in Section 2.2 can be used for more than achieving regularized solutions. It is also a powerful tool for understanding the properties

N	$\|\boldsymbol{e}\|_2$	$\sqrt{\|\boldsymbol{b}\|_1}$	Relative Error
10^2	$0.73 \cdot 10^4$	$0.70 \cdot 10^4$	8.6%
10^3	$2.11 \cdot 10^4$	$2.24 \cdot 10^4$	2.6%
10^4	$7.08 \cdot 10^4$	$7.07 \cdot 10^4$	0.4%

Table 3.3: Estimate of level of error

of the discrete inverse problem at hand. Having a closer look at the singular values and vectors of the system matrix can give us an indication about what to expect from our reconstructions.

First up is to see if the problem satisfies the *Discrete Picard Condition.* In Figure 3.7 the Picard plot of an instance of the problem is seen. The discrete Picard Condition is satisfied, since the SVD coefficients, $\boldsymbol{u}_i^T\boldsymbol{b}$, decay faster than the singular values until some point just before the 350'th singular value. Hereafter the machine precision is reached for the singular values and are therefore not reliable anymore. So now that the discrete Picard condition is satisfied, we know that the problem is solvable, and we can go on with our analysis.

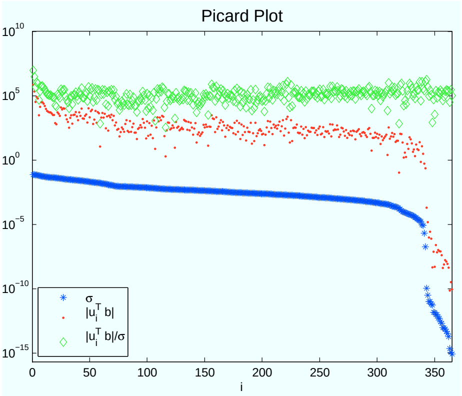


Figure 3.7: Picard Plot with a system matrix of size 450 × 441.

The left singular vectors are seen in Figure 3.8 . These singular vectors are of the same size as the detections, and are therefore 'split', such that we can see how they look for each detector. As was also seen in Section 2.2 the number of oscillations of these singular vectors increase with the index. Moreover we see that the oscillations for each singular vector is repeated on each of the detectors.

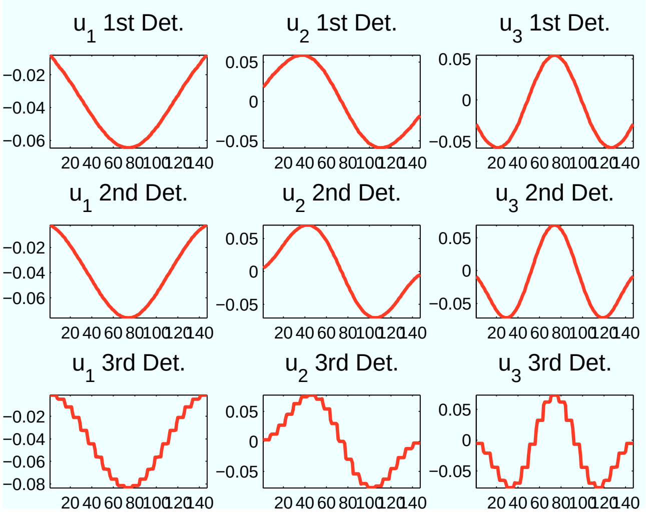


Figure 3.8: First three left singular vectors for each of the detectors.

Earlier we defined the sampling of the original intensity distribution function as the image $\boldsymbol{F}$. Considering the SVD solution, we can express $\boldsymbol{F}$ as

$$\boldsymbol{F} = \sum_i \frac{\boldsymbol{u}_i^T \boldsymbol{b}}{\sigma_i} \boldsymbol{V}_i, \tag{3.17}$$

where $\boldsymbol{V}_i$ is the i'th right singular vector reshaped to the same domain as $\boldsymbol{F}$. So the images of $\boldsymbol{V}_i$ are added together in order to reach a solution. Investigating these further will give an indication of what we are able to reconstruct. For a small test problem the SVD has been carried out, and in Figure 3.9 the images of the right singular vectors are seen for the problem set-up described in Table 3.2. The images of the singular vectors in Figure 3.9 show us that in the θ-dimension the solutions will be able to vary a lot. The number of oscillations in this direction increases with the number of the singular vector. But if we look at the other direction in the images, which corresponds to the w-direction in the original intensity distribution the same variation is not present. In fact it does not vary a lot, and it rarely oscillates. From the physics behind the experiment it is known that the farther away from the source the detectors are placed the more spatial information will be lost. In Section 3.4 this will be discussed and used in the solution process. So when the singular vectors do not seem to give us information about the spatial direction, it could imply that the detectors are placed too far from the source. A set-up with detectors closer is therefore tested. Table 3.4 describes the set-up that is simulated. This is a set-up that might not be possible when doing the actual laboratory experiments due to, e.g., equipment size. But for the sake of simulation and experiments it is useful.

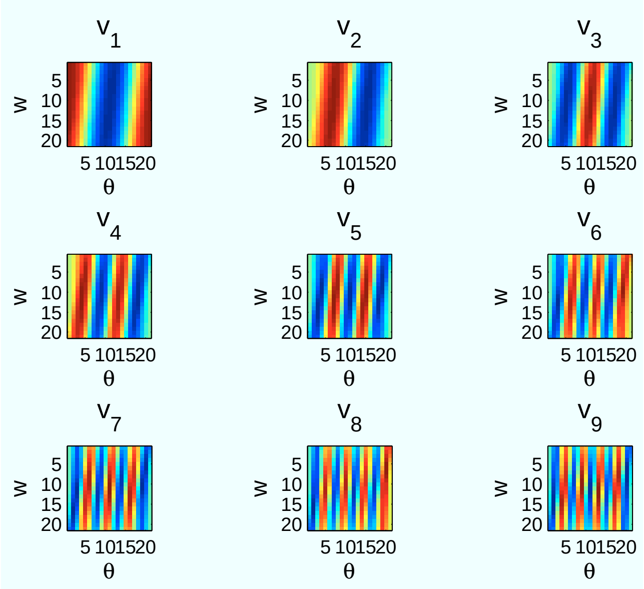


Figure 3.9: Images of right singular vectors for original experiment set-up.

Modified set-up			
	d_1	d_2	d_3
Test 1	1 mm	3 mm	500 mm

Table 3.4: Table of simulation set-up with modified detector distances.

Constructing the matrix with the test set-up described in the table and again having a look at the images of the right singular vectors gives no significantly increase in the variation in w. One could imagine that adding more detectors could help improve the issue, but looking at the detections made on the different detectors we see that they do not vary a lot - it is mainly the magnitude of the detected quantities that change. Since big improvements do not seem to appear by moving the detectors closer to the source, the original laboratory set-up will be used in the following sections.

3.4 Far-field

As described earlier the third detector should be used as a far-field detector to obtain the angle distribution of the original intensity distribution. Our intuition tells us that if we place the third detector far enough from the source then the photons emitted will be as coming from a single point source. This means that the angle and spatial information will separate. The intensity distribution only dependent on the angle θ, will be denoted $\tilde{f}$, and is given by

$$\tilde{f}(\theta) = \int_{-0.5}^{0.5} f(x, \theta) dx. \tag{3.18}$$

Since we have assumed that the photons emitted from the source do not loose intensity when passing through the detectors, we will use the detections at the far-field as they are. The detections are made on the grid specified on the far-field detector, so in order to get the angle distribution, this grid will be interpolated to the θ-grid.

The information obtained from the far-field detector will be used to minimize the number of columns in the model matrix $\boldsymbol{A}$. For every angle that is not present in the intensity distribution we are able to remove N_w columns from $\boldsymbol{A}$. One of the reasons why it is hard to solve this problem is the issue of the condition number of the model matrix. It is very high and makes the problem ill-conditioned. The condition number of an $m \times n$ matrix with regards to the Frobenius-norm is given by:

$$\kappa_{\mathrm{F}}(\boldsymbol{A}) = \|\boldsymbol{A}\|_{\mathrm{F}} \|\boldsymbol{A}^{-1}\|_{\mathrm{F}}. \tag{3.19}$$

We know that the norm of the matrix $\boldsymbol{A}$ is given by

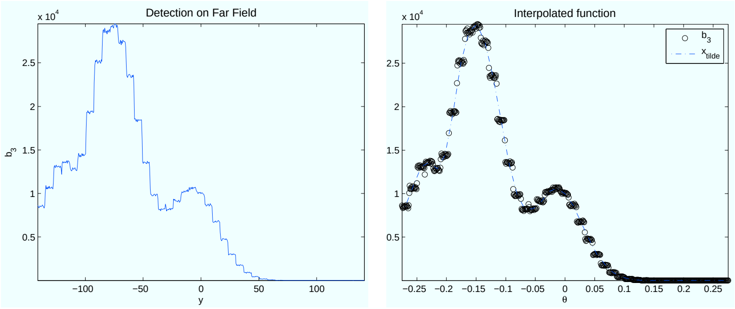
$$\|\boldsymbol{A}\|_{\mathrm{F}} = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2 \right)^{1/2}. \tag{3.20}$$

If we then remove a number of columns from $\boldsymbol{A}$ based on the knowledge of the angle distribution and denote this matrix $\boldsymbol{A}_c \in \mathbb{R}^{m \times n_c}$, we find that the condition number of this matrix is given by

$$\kappa_{\mathrm{F}}(\boldsymbol{A}_c) = \|\boldsymbol{A}_c\|_{\mathrm{F}} \|\boldsymbol{A}_c^{-1}\|_{\mathrm{F}}, \tag{3.21}$$

where

$$\|\boldsymbol{A}_c\|_{\mathrm{F}} = \left(\sum_{i=1}^{m} \sum_{j=1}^{n_c} |a_{ij}|^2 \right)^{1/2}. \tag{3.22}$$



(a) Detections at third detector.

(b) $\tilde{f}$ by interpolation.

Figure 3.10: Illustration of procedure for finding A_c.

Since $n_c \leq n$ must $\|\boldsymbol{A}_c\|_\mathrm{F}$ be equal to or smaller than $\|\boldsymbol{A}\|_\mathrm{F}$. We observed that this is also the case for the inverted matrices and therefore will $\kappa_\mathrm{F}(\boldsymbol{A}_c) \leq \kappa_\mathrm{F}(\boldsymbol{A})$. This means that it can be an advantage for us to remove the columns in $\boldsymbol{A}$ that correspond to angles that do not contribute to the intensity distribution function. Besides maybe reaching a lower condition number, we also gain a computational advantage since the size of the matrix will decrease with a factor N_w each time an angle does not contribute to the intensity distribution.

To illustrate the procedure above a small test problem is considered. It has an exact intensity distribution with the dimensions 21 × 21 and is constructed such that half the function is zero, in the sense that it is only the first half of the angles that actually contribute to the function. The number of grid points on the three detectors are 210. In Figure 3.10 a plot of the original data sampling with respect to the grid spacing at the third detector is seen to the left. For each grid point on the far-field detector a corresponding angle is found. The plot to the right shows the data sampling with respect to these grid points along with the interpolation that has been made from these and the data. Using the information from $\tilde{f}$ to leave out columns from $\boldsymbol{A}$, a smaller matrix $\boldsymbol{A}_c$ is obtained. In Figure 3.11 the difference between two such matrices are seen. The number of columns have decreased significantly, and we will therefore gain a computational advantage by solving

$$\boldsymbol{A}_c \boldsymbol{x}_c = \boldsymbol{b}, \tag{3.23}$$

instead of our original system. The set of angles contributing will be denoted Θ.

When the noisy signal is detected and a certain angle distribution has been chosen it is possible to interpolate the signal such that the angle distribution

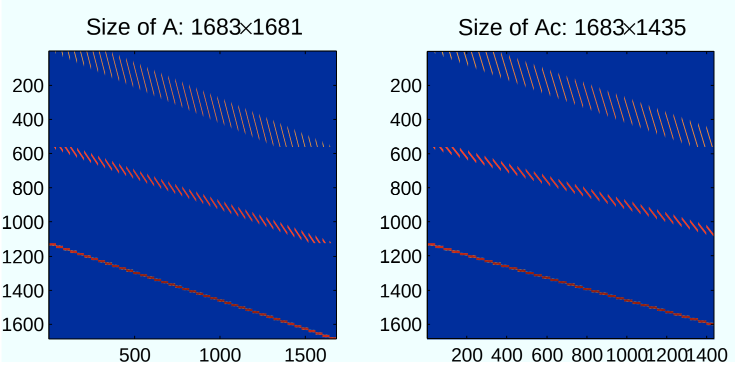


Figure 3.11: Left: Image of A. Right: Image of A_c.

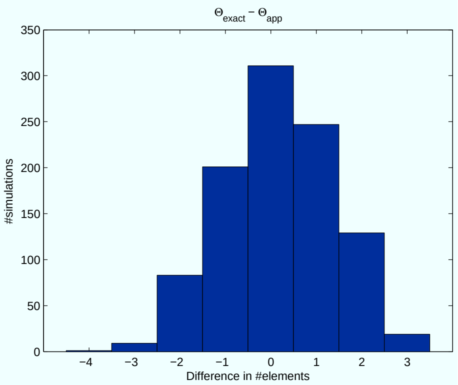


Figure 3.12: Difference between the correct and approximated Θ.

$\tilde{f}$ is approximated. As was explained earlier the angle distribution at the far-field can be used to find the angles that actually do contribute to the intensity distribution that we wish to reconstruct. So what is in fact aimed at, is a sparse representation of the solution - we only wish to solve the problem for the angles that are contributing. In order to find out if the noisy signal from the far-field can give us correct information about the angle distribution a number of simulations have been carried out on the same problem. For each simulation the noise realization is different. Figure 3.12 shows how the number of angles in Θ differs from the number of angles in Θ_{exact} - the exact set of angles contributing to the solution. The histogram show that most often a system will be reached that has the correct number of angles or one to little. This is important since the information from Θ is used to leave out columns of A to reach the system in (3.23). If too many columns are left out it will lead to a system that does not represent the real angle distribution. The 1000 simulations, that are the basis of the histogram gives a good clue about the correctness of Θ when using the noisy signal from the far-field to reach an angle distribution. Having this fact clear we are now able to solve (3.23) instead of (3.5) and by this hopefully gain a computational advantage.

3.5 Reconstructions

When reconstructing different factors have great importance for this problem of finding the intensity distribution function. First of all the ratio between the resolution of the spatial grid on the source and the detectors can have an impact. The appearance of the intensity distribution function could also influence the solution - some functions may be easier to reconstruct than other. All the iterative methods presented in Section 2.3 will therefore be considered and their performance discussed. These issues will be taken into consideration in these last sections of this chapter that revolve around the aspects of solving the simplified model.

In order to have a measure for how well the quality of the iterative solutions are, the relative errors will be considered throughout this section. It is given as

$$\frac{\|\boldsymbol{x}^{\text{exact}} - \boldsymbol{x}^{[k]}\|_2}{\|\boldsymbol{x}^{\text{exact}}\|_2}, \tag{3.24}$$

where $\boldsymbol{x}^{[k]}$ is the k'th iterate of the iterative methods. For the regularization methods TSVD and Tikhonov, $\boldsymbol{x}^{[k]}$ would refer to the solutions with the k'th regularization parameter. The following sections will each investigate a certain aspect of the reconstruction process.

3.5.1 Spatial Resolution

In order to see what influence the ratio between the spatial resolution on the source and the detectors have, simulations with a fixed resolution on the source have been carried out. The resolution on the three detectors is increased for each simulation. In Figure 3.13 the exact distribution is seen along with five solutions found by Tikhonov regularization with optimal regularization parameter. The resolution on the source is $N_w = 21$, which makes the resolution on the detectors 4, 6, 8, 10 and 12 times as big. These different set-ups give rise to systems that range from underdetermined to overdetermined. It is seen that as the resolution on the detectors increase the quality of the reconstructions increase as well. In order to reach good reconstructions we can see that the resolution on the detectors must be significantly larger than the resolution chosen for the spatial grid on the source. With basis in these experiments the resolution on the detectors will be kept at a level around ten or more times bigger than on the source. This will moreover often ensure an overdetermined system.

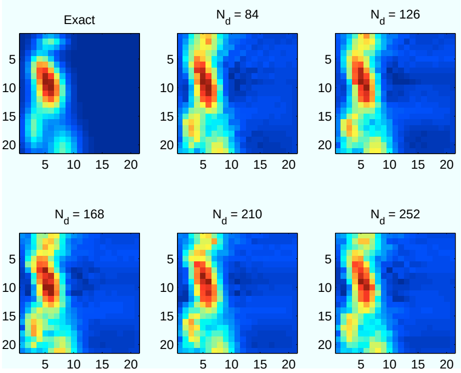


Figure 3.13: Solutions with Tikhonov regularization and increasing spatial resolution on the detectors.

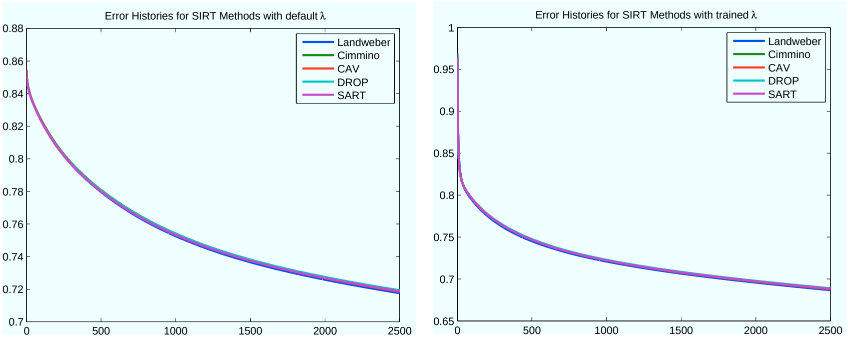
3.6 Performance of the Iterative Methods

In Section 2.3 different solution methods were described. They are all suitable for solving inverse problems, but the three classes of iterative methods are more suitable for large-scale problems. TSVD and Tikhonov work well as illustrators for certain issues. But this section will deal with how well the iterative methods perform compared to each other. The simulations done in the previous section will be used as a guideline for the set-up of the problem, so a fairly great resolution ratio for the source and detectors will be used. We will deal with a problem of size $N_w \times N_\theta = 41 \times 41$, and have a spatial resolution on the detectors as $N_d = 20N_w$. This set-up ensures that the system is overdetermined. The quality of the reconstructions will be measured by the relative error as given in (4.13). For testing the methods a certain test problem has been set up. It consists of a square discrete function representing the intensity distribution function f. The function has three Gaussian bells in the diagonal with increasing mean and variance. This problem set-up will help us give a view on whether the solution methods are good at placing the centers of the Gaussian bells at the right indices - spatial gridpoint and angle - but also to see if they are able to find the limits of the Gaussian bells. The results of the different classes of methods will be treated individually and hereafter compared.

As mentioned earlier are five different versions of the SIRT methods implemented in the MATLAB package AIRTools - Landweber, Cimmino, CAV, DROP and SART - and three different versions the ART methods - Kaczmarz, symmetric Kaczmarz and randomized Kaczmarz. Moreover the CGLS method will also be considered. This section of the project will deal with the performance of these different methods, and investigate if there is any difference in the performance of the methods. AIRTools also holds algorithms for training the relaxation parameter for the SIRT and ART methods. The goal with training is to find a near-optimal value for the relaxation parameter on a data set, where the exact solution is known. For every method the relaxation parameter has to be within a certain interval in order to be valid. In every iteration of the training algorithm, this interval is decreased based on the value of the errors from the previous iteration. Because the exact solution is known it is possible to find the error of every iterate. In the end an optimal value for the relaxation parameter is found. It will be tested whether the training of the relaxation parameter has any influence on the convergence.

3.6.1 SIRT Performance

First the SIRT methods will be considered. In Figure 3.14 the error histories of the given problem and noise realization are shown. The left plot shows the error histories for the iterates with the default value for the relaxation parameter, whereas in the right plot the error histories reflects the iterations with optimal relaxation parameter found by the training algorithm in AIRTools. In both cases non-negativity constraints have been imposed. Both plots show that the performance of the different SIRT methods are similar. They have the same slow convergence rate. Within the number of iterations considered here the iteration sequence does not reach the lowest level they can and the concept of semi-convergence is neither reflected in the error histories. Comparing the two plots it is seen that for all of the SIRT methods, the convergence toward the low level is faster when the relaxation parameter has been trained. Table 3.5 shows the difference of the default and trained value of the relaxation parameter. For the SIRT methods are the trained values of the relaxation parameter approximately twice as big as the default ones. Training the parameter does not influence the level of the relative error. The lowest level is just reached faster.



(a) Default λ (b) Trained λ

Figure 3.14: Error Histories for the SIRT methods.

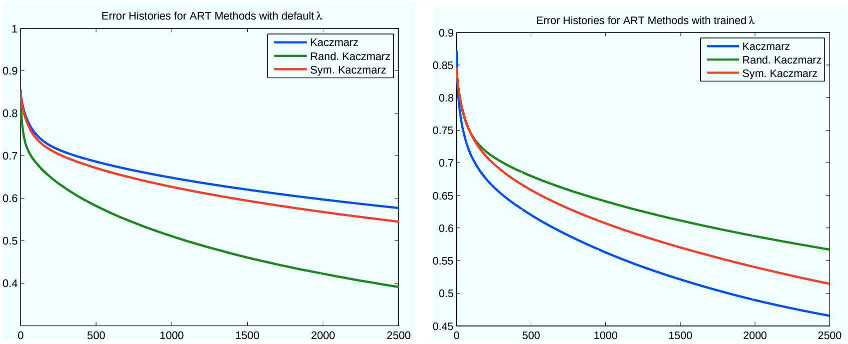
Method	Default λ	Trained λ
Cimmino	36.31	69.82
CAV	1.04	2.05
DROP	1.04	2.05
Landweber	678.04	1286.0
SART	1	1.97
Kaczmarz	0.25	0.15
Rand. Kaczmarz	1	1.24
Sym. Kaczmarz	0.25	0.19

Table 3.5: Default and trained values of relaxation parameter λ for the SIRT and ART methods.

3.6.2 ART Performances

Solving the same problem with the three different ART methods result in the error histories in Figure 3.15. Randomized Kaczmarz shows great performance in both cases. The level of error reached by this method is lower than for the two other ART methods. Especially when the relaxation parameter has been trained is a significant difference present in their individual performance.

Comparing the error level of the SIRT and ART methods it is seen that the ART methods in general, reach a lower error level than the SIRT methods within fewer iterations. Especially Randomized Kaczmarz seems to return great results. A further investigation is therefore of great interest. How does the method do compared to regular Kaczmarz for different noise realizations and different problem



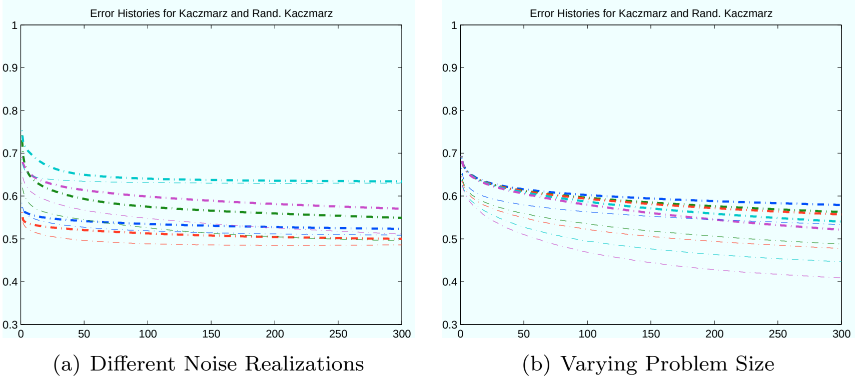
(a) Default λ (b) Trained λ

Figure 3.15: Error Histories for the ART methods.

sizes? The error histories of both Kaczmarz (bold lines) and randomized Kaczmarz for different noise realizations of a given problem are seen in Figure 3.16(a) . In all cases the errors of Randomized Kaczmarz reach a lower level than regular Kaczmarz and this level is reached within fewer iterations. The same is the case in Figure 3.16(b) where the different error curves reflect problems of different sizes - going from underdetermined to overdetermined problem. This investigation shows that in this case where the system matrix might lack information about the spatial variation randomized Kaczmarz does a good job. Another advantage of the results from randomized Kaczmarz, is that when all the stopping criteria take basis in the residuals of the current iterate, and semi-convergence is obtained fairly fast, it will be easier for these methods to find the optimal stopping time. When there is slow converge, as was seen for the SIRT methods above, it is even more difficult for these methods to find the optimal stopping iteration.

3.6.3 CGLS Performance

Finally the CGLS solution will be investigated. In Figure 3.17 the error history of the CGLS solutions is seen. For this method we see that the semi-convergence is reached within approximately 150 iterations. The error level reached by the CGLS method is a little higher than for the SIRT and ART methods. Within the amount if iterations considered in these test, the performancce of the CGLS method lies between the SIRT and ART methods. The lowest error reached was lower than for the SIRT methods. But since the point of semi-convergence is not reached for the SIRT methods, it will most likely be possible for them to reach a lower level than the CGLS method. But it will then be a choice between



(a) Different Noise Realizations (b) Varying Problem Size

Figure 3.16: Error histories for Kaczmarz and Randomized Kaczmarz.

computing time and quality of the solution. Since the ART method Kaczmarz' and the randomized version of this performs better than both the SIRT and CGLS methods these methods will be preferable.

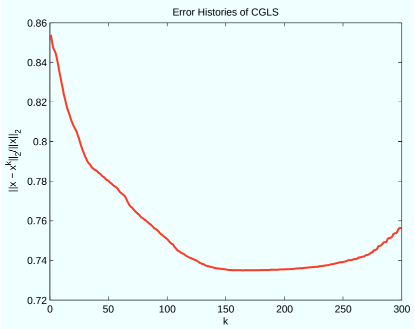


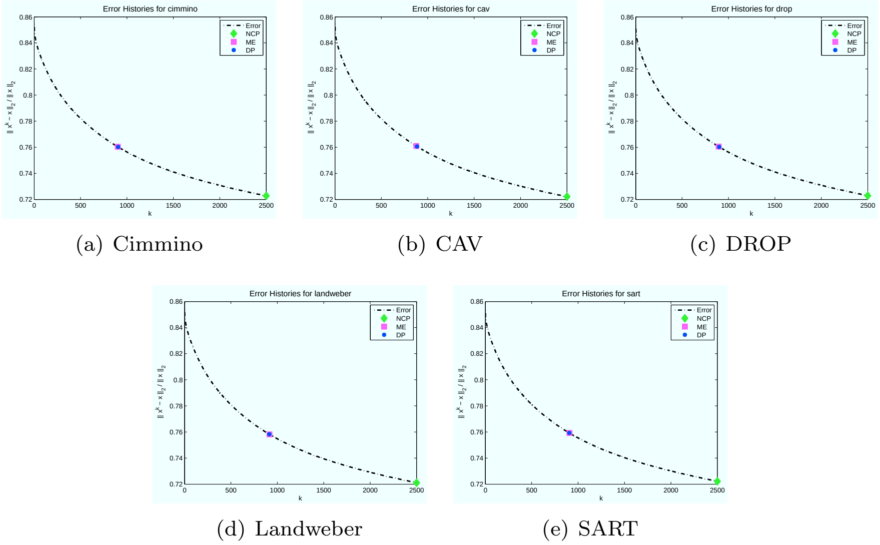
Figure 3.17: error histories of the CGLS method.

Now that we have shown that the ART methods return great results it could be tempting to just disregard the rest of the methods. But throughout the rest of the thesis all the methods will still be taken into consideration. There are several reasons for this. For one, all the methods behave differently on different problems and different noise realizations. Secondly; When we move on to looking at the problem with the source and detector axes replaced by planes, the properties of the system could change a lot and we do not know what influence this will have on the performance of the methods.

3.7 Working with Stopping Criteria

The stopping criteria described in Section 2.4 will now be tested on the full problem for reconstructing the intensity distribution function. First the SIRT methods will be the subject of our attention and then afterwards the ART methods. It is important to mention that for the ART methods it is only DP and NCP that will work.

In Figure 3.18 the error histories for the five SIRT methods are seen. For all five methods in the class the same conclusion can be made. All the stopping criteria perform equally well. The flat error curves make it difficult to determine when the error level has been reached. The difference in relative error is not significant for the different stopping times. In Figure 3.19 the error histories for the three ART methods are seen. For these methods DP and NCP performs equally good. In Table 3.6 and 3.7 a comparison of the optimal iteration number is done with the iteration number found by the different rules.



(a) Cimmino

(b) CAV

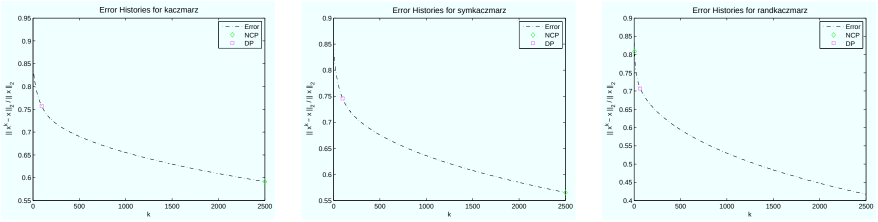
(c) DROP

(d) Landweber

(e) SART

Figure 3.18: Error Histories stopped at different times due to different stopping rules.

When training the parameter τ for ME and DP the conclusion is the same. The conclusions about the stopping rules are similar to the results in [6], where different other test problems were considered. This could indicate that the NCP criterion is a more robust way of stopping the methods. Although when



(a) Kaczmarz (b) Symmetric Kaczmarz (c) Randomized Kaczmarz

Figure 3.19: Error histories for the three ART methods when stopped by different stop rules.

		Default τ			Trained τ	
Method	k^*	NCP	ME	DP	ME	DP
Cimmino	k_{max}	2500	2500	442	901	905
CAV	k_{max}	2500	1090	174	884	877
DROP	k_{max}	2500	2500	430	895	899
Landweber	k_{max}	2500	2500	684	916	913
SART	k_{max}	2500	2500	538	906	905

Table 3.6: Optimal number of iterations along with the number of iterations found by the stopping rules for the five SIRT methods. k_{max} is 2500.

we were committing inverse crime and were able to train the parameter τ of DP and ME we were able to get quite good results with these methods as well. The properties of the noise in the data make it easy to get a good estimate of the noise level, which is a prerequisite to reach a good result with ME and DP. The stopping rules are all in some way based on the value of the residuals, and since we see that the convergence of especially the SIRT methods is very slow after a certain amount of iterations, one could imagine that the stopping rules have a hard time figuring out whether an optimal solution has been reached or not. This also means that the iteration numbers in Tables 3.6 and 3.7 do not reflect the difference in relative error for the optimal number of iterations and the iteration found by the stopping rule. Even though the iteration number chosen by the stopping rule might seem far from the optimal one, the difference in relative error for the corresponding solutions might be really small. In most cases this is the case when the NCP criterion is used. We see that for symmetric Kaczmarz the NCP criterion is performing really bad.

		Default τ		Trained τ
Method	k^*	NCP	ME	ME
Kaczmarz	$k_{\max}$	2500	185	94
Sym. Kaczmarz	$k_{\max}$	2500	184	98
Rand. Kaczmarz	$k_{\max}$	4	29	64

Table 3.7: Optimal number of iterations along with the number of iterations found by the stopping rules for the three ART methods. k_{max} 1000.

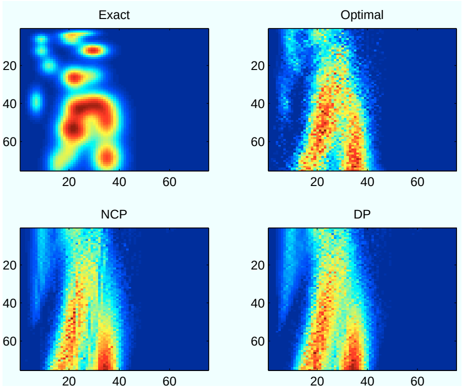


Figure 3.20: Exact solution, along with the optimal one, the one found by NCP and the one found by DP.

3.8 Realistic Problem

To end this chapter a problem with the correct resolution on the detectors will be solved using Randomized Kaczmarz and the NCP and DP stopping criteria. This means that the problem set-up will be as described in Table 3.2 and the resolution on each of the detectors are 2048 pixels. In order to reach an overdetermined system as was found crucial in Section 3.5.1, the number of grid points on the source is therefore 75. The same amount is given for the angle resolution. In Figure 3.20 the exact intensity distribution that has to be reconstructed is seen along with the solutions found by NCP and DP. The intensity distribution function is in this case very complex. The optimal solution is reached after approximately 50 iterations at a level of 0.38, when the error is measured by (4.13). The NCP stopping criterion stops the solution method after five iterations and DP stops after two iterations. This does not mean that the error level of the lower iteration steps is much different from the optimal solution - only about 0.2 higher. This experiment has shown us that it is possible to solve this ray-tracing problem, but only at a low satisfying level. The solutions found by using the stopping criteria revealed that the stopping criteria determine a solution close to the optimal one.

3.9 Summary

To summarize; The analysis of the one-dimensional problem has revealed some weaknesses about our model, but also some advantages. We saw how it was possible to solve the problem at a satisfying level even though the SVD analysis of the system matrix revealed that we might run into some problems for the spatial resolution. We could also conclude that in order to reach great results the resolution on the detectors must be significantly higher than on the source. That ensures an overdetermined system. Moreover the analysis of the experimental set-up revealed that the detectors of the original set-up do not cover the same angle interval, which could lead to inaccuracies when using the far-field to extract the angle distribution and also lead to columns of zeros in the system matrix. This would influence the condition number of the matrix and make the problem even more ill-posed. The great advantage about our model or experimental set-up is that the distribution of the noise makes it simple to estimate the noise level, such that the stopping rules for the iterative methods are applicable and returns great results. Furthermore we saw how both the SIRT and ART methods returned useful results. The results were better than the ones achieved with the CGLS method, but partially because of the non-negativity that was imposed on the other methods. The analysis of the simple model will now work as a basis for setting up and understanding the more complex two-dimensional model that will be the subject of the rest of the thesis.

Chapter 4

Two-dimensional Model

The previous chapter dealt with a simplified version of the real experimental set-up. The model set-up and the analysis of the model matrix revealed matters that were interesting and challenging. Having dealt with these matters before moving on to the two-dimensional model in this chapter will hopefully give us an advantage when dealing with a more complex problem set-up. This chapter will deal with the set-up of a mathematical model for the two-dimensional experimental set-up and the discretization process of this model. This will be followed by a thorough analysis of the model set-up. In the end of the chapter an attempt at solving the problem will be carried out with the deterministic methods described earlier.

4.1 Accurate Forward Model

In figure 4.1 is an illustration of how the two-dimensional problem will be regarded in this project shown. For each pixel on the source plane the photons are emitted in a cone-shape. The radius of the circle at the end of the cone is dependent on the distance between the source plane and the detector. The direction of the rays after they have been diffracted in the polycrystal is determined by the properties of the material. In mathematical terms this means that the directions are discrete and a priori knowledge. In order to keep the model

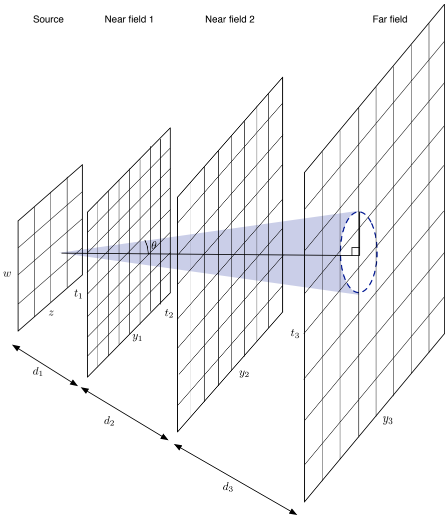


Figure 4.1: Illustration of the real problem set-up from the European Synchroton Radiation Facility in Grenoble, France.

as generic as possible, we will keep this direction continuous in the definition of the model. In this setting light or rays will be emitted from each midpoint of a pixel on the source plane and this can happen in any direction.

The intensity distribution function at the source is dependent on four variables - the point (z, w) at which the ray is emitted and in what direction it is emitted given by a set of angles (ϕ, θ). The intensity distribution function will again be denoted f and the domain of the dependent variables is in the continuous setting given by

$$w, z \in [-0.5, 0.5], \quad \phi \in [0, 2\pi], \quad \theta \in [0, \pi/2]. \tag{4.1}$$

The signal from the source will lead to detections g_k on the k'th detector. The axes on the detectors are denoted (y_k, t_k). For each pixel on the detector we will add up the photons that come from all the pixels on the source. The contribution on a detector pixel from a certain source pixel will be given by

$$\Delta g_k(w_i, z_j, y_{k_l}, t_{k_m}) = \int_{\theta_1}^{\theta_2} \int_{\phi_1}^{\phi_2} f(w_i, z_j, \phi, \theta) d\phi d\theta. \tag{4.2}$$

θ_1, θ_2, ϕ_1 and ϕ_2 defines the boundaries of the integration area for this specific detector pixel. Thus the detection at a specific pixel will be given by

$$g_k(y_{k_l}, t_{k_m}) = \sum_{j=1}^{N_s} \sum_{j=1}^{N_s} \Delta g_k(w_i, z_j, y_{k_l}, t_{k_m}). \tag{4.3}$$

This continuous model will be the basis of the discrete model that will be derived in the next section. It is assumed that the detectors and the source are symmetric around 0 and have the same grid resolution in both directions. Moreover the source and detectors are assumed to be aligned such that when the source and the detectors are parallel, a straight line can go through origo of all of them.

4.2 Discrete Forward Model

When the one-dimensional model was set up and discretized, the thinking behind was to consider one pixel on a detector and for this pixel add up the photons coming from all pixels on the source, dependent on what angle intervals the pixel gave rise to. The same track of thoughts will be used when discretizing this larger problem. But since it is not in the same way straightforward to find the angle intervals for which to integrate, the problem will be considered slightly different. Each pixel on the detector is split into $p \times q$ smaller sub-pixels and a quadrature method is used to calculate the integrals of (4.2). Each sub-pixel gives rise to a certain value of ϕ and θ, and it is in these values the function f are then sampled in order to reach the value of the integrals. Figure 4.2 illustrates this.

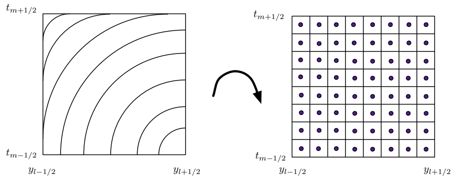


Figure 4.2: Each pixel on the detector is split into $p \times q$ sub-pixels.

We wish to discretize in order to reach a system of linear equations like

$$\boldsymbol{Ax} = \boldsymbol{b}.$$

Therefore the domains of the four variables of the intensity distribution function are discretized such that there is N_s grid points in each direction on the source and N_ϕ and N_θ grid points respectively. $\boldsymbol{x}$ is therefore a vector representing $f(z_i, w_j, \phi_m, \theta_n)$, $i, j = 1, \dots, N_s$, $j = 1, \dots, N_w$, $m = 1, \dots, N_\phi$ and $n = 1, \dots, N_\theta$. Each combination of i, j, m and n gives rise to an element in $\boldsymbol{x}$. The number of elements in $\boldsymbol{x}$ will then be $N_s^2 \cdot N_\phi \cdot N_\theta$.

On each detector there is $N_d \times N_d$ pixels, so on the k'th detector there will be N_d^2 observations in total. This leads to a total number of $3N_d^2$ observations, when dealing with the laboratory set-up with three detectors. Thus the right-hand side of the system $\boldsymbol{b}$ will have $3N_d^2$ elements. This results in a system matrix of dimensions $3N_d^2 \times (N_s^2 \cdot N_\phi \cdot N_\theta)$. If the discrete version of the intensity distribution function is denoted $\boldsymbol{F}$ and the discrete image of g is denoted $\boldsymbol{G}$, the integral in (4.2) in the discrete setting will be given by

$$\Delta \boldsymbol{G}_k(w_i, z_j, y_{k_l}, t_{k_m}) = h_\phi \cdot h_\theta \sum_{r=1}^{q} \sum_{s=1}^{p} \boldsymbol{F}(w_i, z_j, \phi_r, \theta_s). \tag{4.4}$$

p and q refers to the number of quadrature points on the sub-pixel and h_ϕ and h_θ the grid spacing in the discretization of the domains of ϕ and θ. Equivalent is the value of a specific pixel on the detector given by

$$\boldsymbol{G}_k(y_{k_l}, t_{k_m}) = \sum_{i=1}^{N_s} \sum_{j=1}^{N_s} h_\phi \cdot h_\theta \sum_{r=1}^{q} \sum_{s=1}^{p} \boldsymbol{F}(w_i, z_j, \phi_r, \theta_s). \tag{4.5}$$

For each detector a subproblem is reached such that

$$\boldsymbol{A}_k \boldsymbol{x} = \boldsymbol{b}_k, \tag{4.6}$$

where $\boldsymbol{b}_k = vec(\boldsymbol{G}_k)$. $\boldsymbol{A}_k$ describes the diffraction of rays from the source plane to the k'th detector. By 'stacking' the right-hand sides of each problem on top of each other and doing the same for the system matrices we reach the final system of linear equations. By approximating the integrals of each pixel in the way described above, we introduce some discretization errors. The way to minimize the discretization errors are, just as in any general case, to refine the grid spacing. In this case the grid spacing of the angles. In the discretization of the one-dimensional model, we were able to find the exact angle interval over which to integrate and in that way minimize the discretization error. As Figure 4.2 shows it would have been a cumbersome task to find the ϕ-interval to integrate over for each θ-angle.

The problem set-up that will be considered throughout the next chapters will be based on the data given in Table 3.1. Just as in the one-dimensional case we operate with two near-field detectors and one far-field detector. At the experiments carried out in Grenoble the number of pixels on the detectors were 2048 × 2048. It will not be possible to reach this resolution due to limitations of memory space and computation time. These issues will be discussed further in Chapter 7.1.

In the one-dimensional case we saw that the dimensions of the detectors, as they are in the real experiment, do not give rise to detection of the same rays, because they do not cover the same angle interval. It was straightforward to find the maximum angle in the one-dimensional problem and that is also the case for the two-dimensional problem. If each of the detectors are assumed to be square and the cartesian grid is defined from $[-b^k, b^k]$, then the maximum value for θ for each of the detectors is given in terms of the first set of points on the source (z_1, w_1) and b^k:

$$\theta^k_{\text{max}} = \arctan\left(\frac{\sqrt{(b^k - w_1)^2 + (b^k - z_1)^2}}{d}\right). \tag{4.7}$$

For the sake of simulations, the following sections will therefore be based on a problem, where the dimensions of the detectors are adjusted, such that they all cover the same angle interval. Expressed more explicitly, this means that θ^k_{max} will be the same for all three detectors. If this restriction is not imposed on the problem it will give rise to a lot of columns of zeros in the system matrix. These columns will lead to a higher singularity of the system matrix and result in an even more ill-posed problem.

When doing experiments with this set-up it is important to have the investigations of the simpler problem in mind. We could conclude that in order to reach fairly good reconstructions an overdetermined problem should be reached. But since the integrals of the model given in (4.2) are approximated by a sum defined by the quadrature points of each of the detector pixels as illustrated in Figure 4.2, it is necessary that there is a fine resolution for angular variables in the discrete setting of the model. Consequently the number of columns in $\boldsymbol{A}$ will be large and the resolution at the detectors have to be extremely high in order to reach an overdetermined system. In the rest of this chapter a highly underdetermined problem will therefore be considered instead.

4.3 Test Problems

It is difficult to set up a realistic test problem reflecting a real polycrystal. But nevertheless two different test problems will be considered. The first test problem simulates a material that has been pulverized. The second test problem simulates a polycrystal that can be exposed to strain, temperature variations, etc. The test problems are basic and approximated versions of a real polycrystal. If we are able to solve the diffraction problem at a satisfying level for the simple test problems and can conclude that the errors of the iterative methods are decreasing until the point of semi-convergence, we can conclude that the possiblity of being able to solve a real world problem is increasing.

The test problem that corresponds to looking at a material that has been pulverized has no variation in the angle ϕ - the light will always spread out in a cone with the same intensity in the radial direction. The spatial variation for (z, w) will be described by a two-dimensional Gaussian distribution. The variation in θ will also be described by a Gaussian distribution. So what we end up with is a test problem where the variables are separated, such that

$$\begin{aligned} f(z,w,\phi,\theta) &= g(z,w)h(\theta) \quad \text{where} \\ g(z,w) &= \exp\left(-\frac{1}{2}\left(\frac{w-w_0}{\sigma_w}\right)^2\right)\exp\left(-\frac{1}{2}\left(\frac{z-z_0}{\sigma_z}\right)^2\right) \\ h(\theta) &= \exp\left(-\frac{1}{2}\left(\frac{\theta-\theta_0}{\sigma_\theta}\right)^2\right) \end{aligned} \tag{4.8}$$

This will result in a test problem where the light emitted from a certain pixel on the source will come from the same normal distribution, but just weighted by the value of the pixel defined by the Gaussian distribution of g. It would be ideal to define g as a δ-function, such that light is only emitted from a certain pixel. By letting the variances σ_z and σ_w go toward zero, g will tend toward a δ-function. When the test problem is constructed as in (4.8), each source pixel sends out the same cone of light, just with different intensities.

In Figure 4.3 the three detections of this certain problem are shown. Clearly the cone shape of the signal is reflected in these detections and we see that the intensities of the light is the same at all three detectors, which is consistent with our assumption that the rays do not loose intensity as they pass through the detectors. The test problem could be made more advanced by having different values of θ_0 for different pixels. In Figure 4.3 the mean angle θ_0 is set to $\theta_{max}/2$.

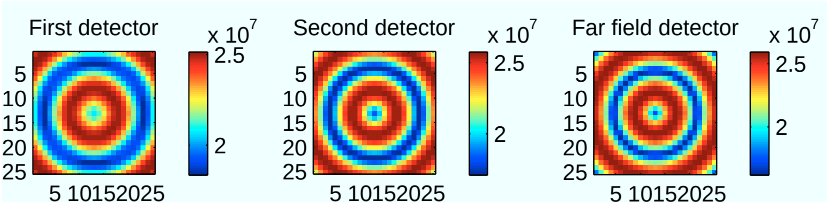


Figure 4.3: Detections of the two-dimensional test problem with a pulverized material.

The second test problem corresponding to a polycrystal material will be simulated in the same way as the first, but now the problem is no longer invariant in ϕ. The distribution in ϕ will also be modeled with basis in the Gaussian distribution. The goal is to have something that will give rise to detections that are drops or splotches on the detectors. On the far-field the detections will then be circles with different radii. On the perimeter of the circles at the detectors, the intensity will be varying. So the second test problem is modeled such that

$$\begin{aligned}
f(z,w,\phi,\theta) &= g(z,w)h(\theta)r(\phi) \quad \text{with} \\
g(z,w) &= \exp\left(-\frac{1}{2}\left(\frac{w-w_0}{\sigma_w}\right)^2\right)\exp\left(-\frac{1}{2}\left(\frac{z-z_0}{\sigma_z}\right)^2\right) \\
h(\theta) &= \sum_i \exp\left(-\frac{1}{2}\left(\frac{\theta-\theta_i}{\sigma_\theta}\right)^2\right) \\
r(\phi) &= \sum_i \exp\left(-\frac{1}{2}\left(\frac{\phi-\phi_i}{\sigma_\phi}\right)^2\right). \qquad (4.9)
\end{aligned}$$

The number of spikes for h and r is chosen at random. By making the standard deviations σ_ϕ and σ_θ small the spikes of the function will be isolated and therefore give rise to small blobs on the detectors. In Figure 4.4 the detections of the second test problem are seen.

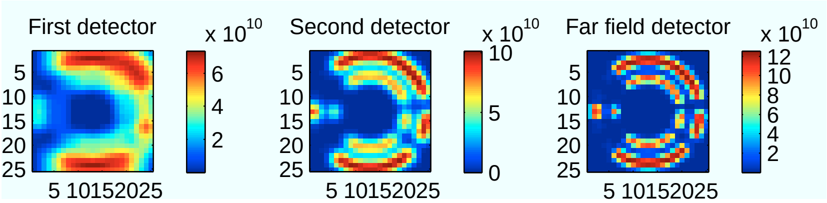


Figure 4.4: Detections of the two-dimensional test problem with a firm source material.

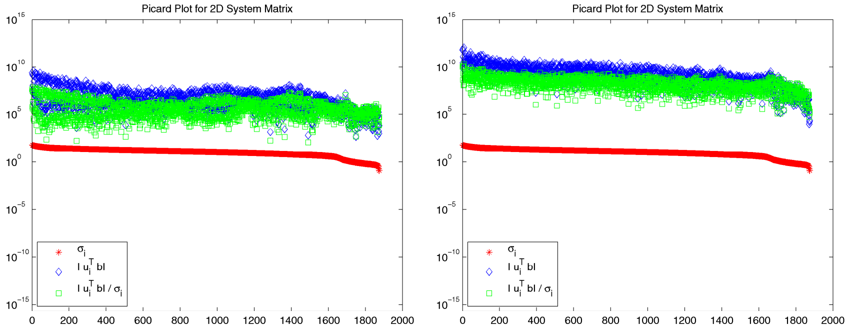
The test problems will be influenced by Poisson distributed noise as described in Section 3.2.1. The two test problems just described will be the basis of the reconstructions later in this chapter. It is important to mention that these setups are meant for simulating and testing. This means that if we can solve these problems and show that the error is decreasing, then the possibily of being able to a real world problem with the model that has been set up, is increasing.

4.4 SVD Analysis

Just as was done in the one-dimensional case the properties of the system matrix will be investigated by means of the SVD. In this case where an underdetermined problem is considered, such that $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ with $m \leq n$, is the SVD given by

$$A = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \sum_{i=1}^{m} \boldsymbol{u}_i \sigma_i \boldsymbol{v}_i^T. \tag{4.10}$$

The system matrix of this two-dimensional problem is large and it is therefore not straightforward to compute the SVD of the system matrix as it was done when working with the one-dimensional problem. At first it will be investigated whether the Picard condition is satisfied or not. However the m singular values and corresponding singular vectors will be considered. We will work on a problem set-up with dimensions given by $N_s = 10$, $N_\phi = N_\theta = 25$ and $N_d = 25$. The result is a system matrix of size 1875×62500. In Figure 4.5 a Picard plot with the 1875 singular values are seen for each of the test problems. All the singular values are above the computer accuracy, which indicates that the problem is surprisingly well-conditioned. Along with the singular values, the absolute value of the SVD coefficients are plotted. From the plots it is seen that the Discrete Picard Condition is satisfied for both test problems and the problems are solveable.



(a) Test problem 1 (b) Test problem 2

Figure 4.5: Picard plot with the amount of singular values computationally capable.

In Figure 4.6 the left singular vectors of the system matrix are shown. Since the left singular vectors are in accordance with the dimensions of the detectors they are reshaped to the sizes of these. This corresponds to what was done for the right singular vectors in the one-dimensional case. Just as was the case for the left singular vectors of the one-dimensional problem, the left singular vectors also repeats on every detector for the two-dimensional case. In Appendix B.1 the first sixteen left singular vectors are seen. This shows the oscillations of the vectors increase by the columns in $\boldsymbol{U}$. Moreover it is important to mention that the singular values come in groups or chunks of approximately four. Thus the singular vectors also 'belong together' four and four. When looking at the images of the left singular vectors in correspondance with the groups of singular values, one can see that they together add up to a symmetric image.

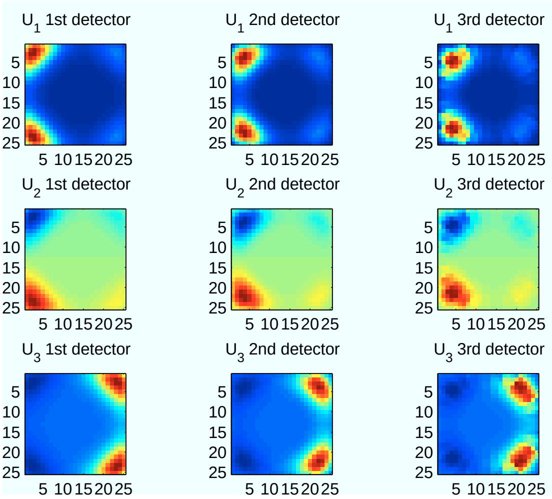


Figure 4.6: Left singular vectors repeated on the three detectors.

Just as we have reshaped the left singular vectors, the right singular vectors should be reshaped to the size of $\boldsymbol{F}$ - the four dimensional array representing the sampled intensity distribution function. Therefore is it difficult to overall see how the behavior of the left singular vectors are. In Figure 4.7 the images of $\boldsymbol{v}_i$, $i = 1, \ldots, 9$, are seen for a fixed point (z, w). Thus what we see is the right singular vector for (ϕ, θ). The variation is not significant, except at some specific combinations of (ϕ, θ). Similar results are found for other fixed values of (z, w). In Figure 4.8 the right singular vectors are now displayed for a fixed pair of (ϕ, θ). So this is the resolution of $\boldsymbol{v}_i$ in (z, w). Due to the small resolution on the source - it is only 10 × 10 pixels - it is difficult to conclude anything about the right singular vectors in this perspective. But the look of them could indicate that the same problem regarding the variation in the spatial resolution is present for the two-dimensional case as it was for the one-dimensional problem.

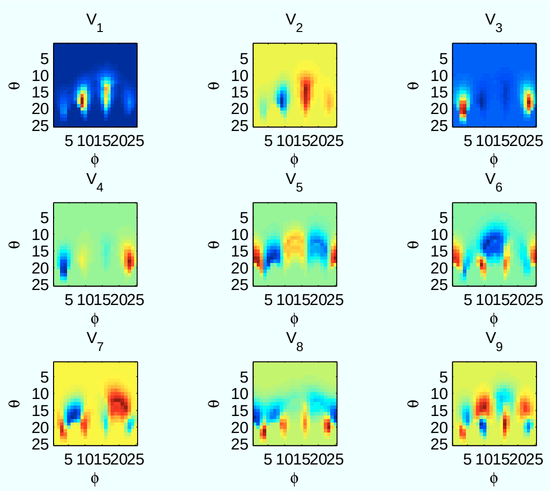


Figure 4.7: Right singular vectors from (ϕ, θ).

The SVD analysis of the system matrix revealed that our test problems satisfy the Discrete Picard Condition and we should therefore be able to solve them both. The left and right singular vectors showed us that we might have problems reconstructing the spatial variations of the solution, just as was the case with the one-dimensional problem.

4.5 Far-field

Just as was done in the one-dimensional problem, the detector farthest away from the source, denoted the far field detector, will be used as a way to lighten the problem in a computational way. Since the distance between the source and the far field is huge compared to the size of the material, it will be assumed that the detection on the far field is seperated from the spatial variance that is on the source. This means that what we see on the far field is a detection of the angle distribution. The angle distribution will, comparable to the one-dimensional problem, be denoted $\tilde{f}$. When this distribution is known it is possible to exclude some of the angles from the original intensity distribution function f, and remove the corresponding columns in $\boldsymbol{A}$. This procedure does not change the right hand side, so no information is lost. The way it is done is by interpolating the signal at the far field to the grid specified by the grid of the angles ϕ and θ from the discretization of the problem.

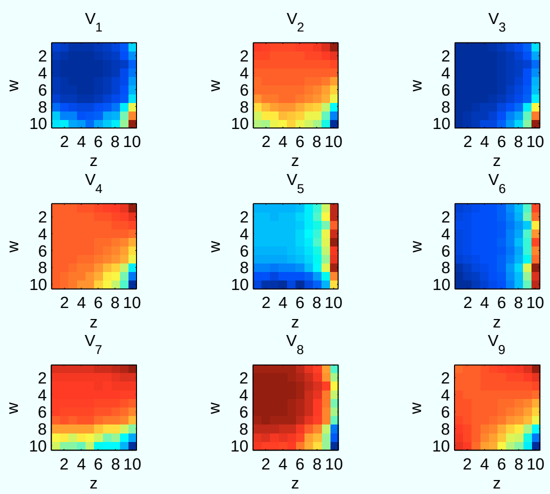


Figure 4.8: Right singular vectors from (z, w).

For each midpoint of a pixel on the far field it is possible to find the corresponding pair of angles (ϕ, θ). We therefore know that the cartesian coordinates of the far field is given by

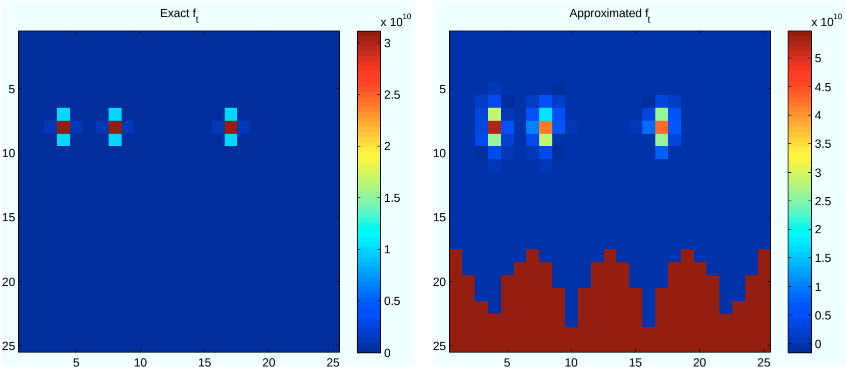
$$\begin{aligned} y_{l,m} &= d_3 \cdot \arctan(\theta_l)\cos(\phi_m) \\ t_{l,m} &= d_3 \cdot \arctan(\theta_l)\sin(\phi_m). \\ l, m &= 1, \ldots, N_d \end{aligned} \tag{4.11}$$

Correspondingly the cartesian grid points of the angle grids will be given by

$$\begin{aligned} y_{i,j}^{grid} &= d_3 \cdot \arctan(\theta_i)\cos(\phi_j) \\ t_{i,j}^{grid} &= d_3 \cdot \arctan(\theta_i)\sin(\phi_j), \\ i &= 1, \ldots, N_\phi, \quad j = 1, \ldots, N_\theta. \end{aligned} \tag{4.12}$$

By interpolating the two-dimensional detection at the far field from the first grid set to the second one, an estimate of the angle distribution will be reached.

In Figure 4.9 is the exact angle distribution of the polycrystal problem seen along with the approximated one. It is clearly an approximation, but the main characteristics of the angle distribution is present. It is useful for the sake of excluding some angles from the solution. Each time a pair of angles (ϕ, θ) is not present in the solution we can exclude N_s^2 columns from $\boldsymbol{A}$. The example in this case is a sparse solution where only some of the angle pairs (ϕ, θ) contribute to the solution. The size of the system matrix $\boldsymbol{A}$ will in this case decrease



(a) Exact $\tilde{f}$ (b) Approximated $\tilde{f}$

Figure 4.9: Exact and approximated angle distribution.

significantly, going from 1875 × 62500 to 1875 × 33750. In the image of the approximated $\tilde{f}$ there is a large area in the bottom that is red. This area corresponds to the points (y^{grid}, t^{grid}) that are outside the domain of the far field detector. These points arise because we defined the maximum detectable angle as in (4.7). This θ_{max} gives rise to a certain radius of the cone on the detector. But for the values of ϕ not corresponding to the corner pixels at the far field, the rays will hit outside the detector. This is what the red area in the image reflects, and because we do not have any detections of these certain pairs of angles, we are not able to exclude any of these from the solution.

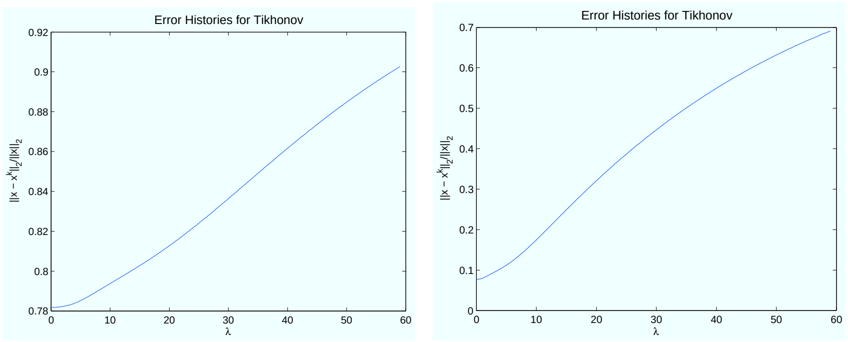
4.6 Reconstruction

When we solved the one-dimensional problem in Chapter 3 we were able to look at the reconstructions and from this also make a judgement about the quality of it. This is going to be a little be more tricky when solving the problem for all four dimensions. Therefore the error measurement given by the relative error

$$\frac{\|\boldsymbol{x}^{\text{exact}} - \boldsymbol{x}^{[k]}\|_2}{\|\boldsymbol{x}^{\text{exact}}\|_2}, \tag{4.13}$$

will be an even more important tool for us now than it was earlier. The only reason why we are actually able to use this error measurement is due to the fact that so far all data has been simulated by a forward operation with the system matrix and hereafter influenced by noise. This is what is called inverse crime, where an attempt to reconstruct the signal is based on data, that was generated from the original signal. But for the sake of understanding the properties and potential problematics about the reconstructions, this crime has to be committed. The noise model will be as described in Section 3.2.1. Therefore we will also be able to use the different stopping criteria with the ART and SIRT methods.

4.6.1 Tikhonov Solution



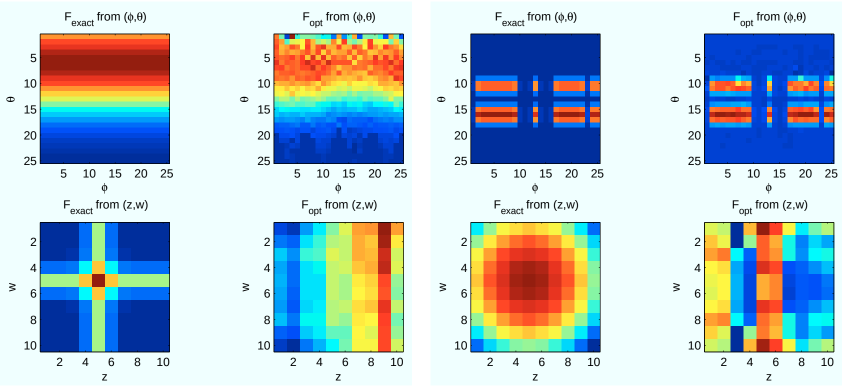
(a) Test problem 1

(b) Test problem 2

Figure 4.10: Error histories for varying values of the regularization parameter.

As a first attempt at solving the two-dimensional problem the Tikhonov solution will be investigated. In Figure 4.10 the relative errors for varying values of the regularization parameter are seen for both test problems. We see that the optimal solution is found for $\varphi_i = 1$, $i = 1, \ldots, m$, so the filter factors of the regularized solution will all be one. This is the case for both test problems and it is in accordance with the Picard plot Figure 4.5. None of the singular values are close to zero or lower than the machine precision, and the decay of the SVD coefficients were faster than the decay of the singular values. Therefore we wish to include all of the singular values in the reconstruction. In Figure 4.11 the exact solution is seen along with the optimal one found be the Tikhonov regularization. The images show the solutions for both a fixed pair of coordinates (z, w) and angles (ϕ, θ). The solution reflects what was concluded in the matrix analysis in Section 4.4 regarding the ability to reconstruct the spatial variation of the intensity distribution function - the system matrix holds more information about what happens in the angle distribution than in the spatial distribution. If we take aother look at the images of the right singular vectors in Figure 4.7 and compare the look of these with the exact image of $\boldsymbol{F}$ from the angle perspective for the second test problem, one can see that there is some resemblance. For the first test problem this is not at all the case. The resemblance between the singular vectors and the angle distribution for the second test problem can be a part of the explanation of why the second test problem reaches a much lower error level than the first.

One could ask why the solution does not reach an error level closer to zero when all components of the SVD is included. A part of the explanation for this lies in the noise that has been added to the right hand side of the system. But a lot of information is also missing when an underdetermined system is considered. In this case 1875 SVD components are included in the solution, but there are 62500 right singular vectors in total. The remaining 60625 singular vectors also hold some information. For the first test problem, this information would probably have helped on the quality of the solutions. What more to note about these Tikhonov solutions, is that they are based on the SVD of a matrix that reflects a system where the resolution of the angle grids are two and a half times bigger than for the spatial grid. This could lead to some distortions and incongruence between the solutions on each of these grids.



(a) Test problem 1

(b) Test problem 2

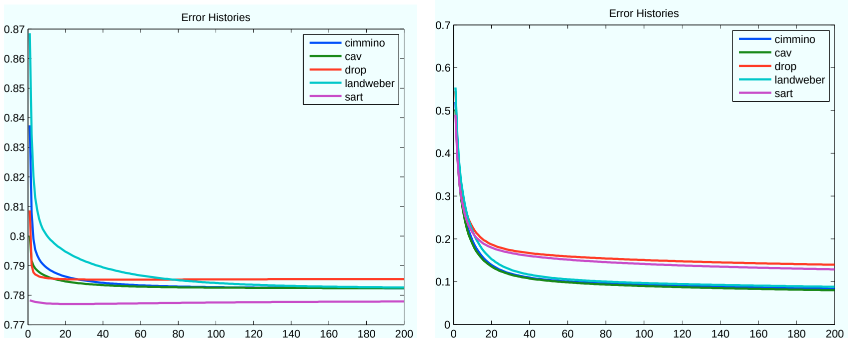
Figure 4.11: Exact solution along with the optimal solution found by Tikhonov.

4.6.2 Solutions of the iterative methods

This section will deal with the performance of the iterative methods - SIRT, ART and CGLS. The performance of the stopping criteria introduced in Section 2.4 will also be tested. Due to the fact that the problem is well conditioned, we will expect that the iterative methods all reach the same or a slightly lower level of relative error than the Tikhonov regularization was capable of. On the SIRT and

ART methods non-negativity constraints are imposed due to the non-negative properties of the intensity distribution function.

Figure 4.12 shows the error histories of both test problems when solved by the five different SIRT methods. It is easily concluded that the lowest error for each of the problems is the same as the one reached by Tikhonov regularization. The corresponding optimal solutions are seen in Figure B.2. The solutions behave in the same way as regularized Tikhonov solutions. Though for the first test problem the SART method seem to get somehow closer to the correct solution - at least for the angle distribution. This is also reflected in the error histories where the SART method reaches the lowest level of the five methods. It is not significantly lower, only about 1%. The second test problem does not show any significant difference for the five different solvers. We can conclude that the SIRT methods solve the problem just as well as the SVD based solution. The advantage of using the iterative methods is that the SVD does not need to be computed which is a time and memory consuming process. For the ART methods and the CGLS exactly the same behavior is seen. The error plots are found in Figure 4.13 and 4.14. The solutions for all the methods can be found in Appendix B.



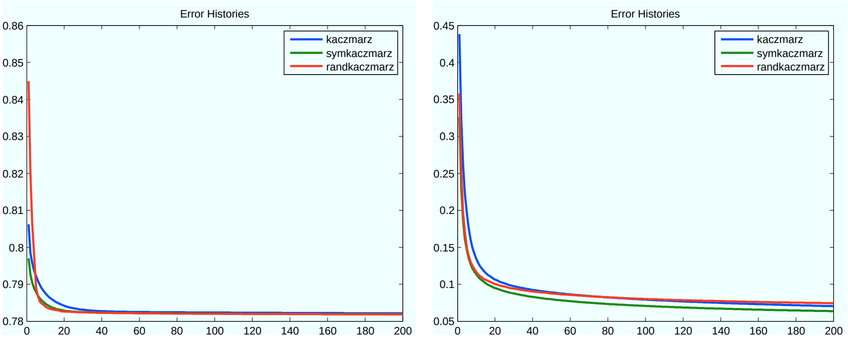
(a) Test problem 1

(b) Test problem 2

Figure 4.12: Error Histories for the five different SIRT methods.

4.6.3 Stopping Criteria

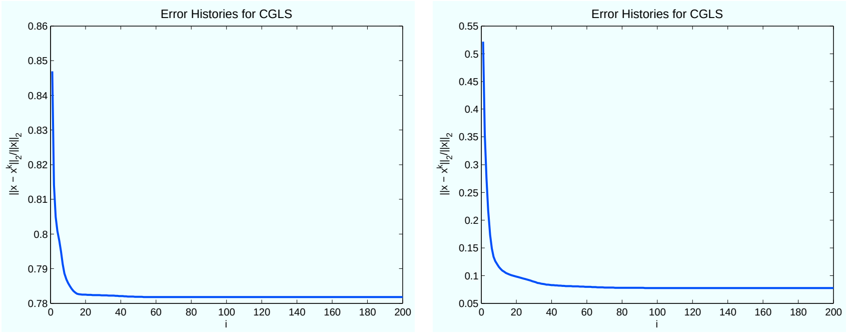
Just as was done for the one-dimensional problem, the stopping criteria introduced in Section 2.4 will be used when solving the problem. The reconstructions in the previous section were based on the fact that the exact solution is known



(a) Test problem 1

(b) Test problem 2

Figure 4.13: Error Histories for the three different ART methods.



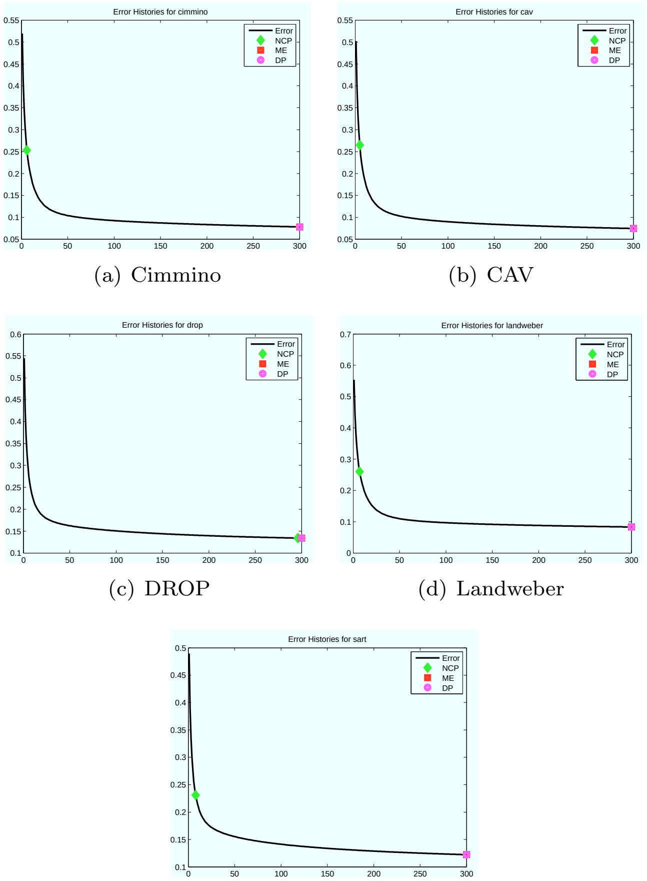
(a) Test problem 1

(b) Test problem 2

Figure 4.14: Error Histories for the CGLS method.

and we are therefore able to determine the optimal solution by the error measure introduced (4.13), but in the real world this is not the case. The stopping criteria can help us choose the correct iteration at which to stop. Two of the three stopping criteria need an estimate of the error level. The estimate of this level was found in Section 3.2.1 and is also applicable for this problem.

The error histories for the five SIRT methods for the second test problem are plotted in Figure 4.15. On the graphs the stopping times are indicated for the three stopping criteria, NCP, DP and ME. In this case the parameter τ is trained for the stopping criteria DP and ME and it is seen how these two methods stop the iterative method at the very last iteration in all cases. Almost similar for all methods is also the iteration at which the NCP criterion stop the methods. The results of the stopping criteria are all similar for the SIRT methods. This could originate from the fact that the progress of the error histories - and therefore also the solutions - are identical for the SIRT methods. Though the NCP criterion stops the SIRT methods earlier, the difference in relative error is not significant and it is therefore just as good a choice as ME and DP. For the ART methods it is possible to use the NCP and ME criteria. In Appendix B.5 it is possible to see the corresponding plots for the three ART methods. The behavior of the stopping criteria is the same as for the SIRT methods - NCP stops around the fifth iteration for all three methods, whereas DP chooses the maximum number of iterations possible. Thus we can conclude that our stopping criteria return



(a) Cimmino

(b) CAV

(c) DROP

(d) Landweber

(e) SART

Figure 4.15: Error histories stopped at different times due to different stopping rules.

reliable results when simple test problems are considered.

4.7 Summary

The formulation of the mathematical model for the two-dimensional problem was based on the work done on the one-dimensional model. The discretization process was a bit more tricky than for the one-dimensional problem, but it was doable and resulted in underdetermined test problems. The SVD analysis revealed - highly unexpected - that this problem is better conditioned than the simple one-dimensional model and in fact defined as a well-conditioned problem due to the nice behavior of the singular values. The nice conditions of the problem also mean that there is really no need for regularization methods. When solving with Tikhonov's method the regularization parameter was equal to zero, which meant that all elements of the SVD should be included in the solution. That no need for regularization is necessary also means that the level at which the two Tikhonov solutions reached for the two different test problems, is the level at which we will be able to solve the problems with other solvers. The iterative methods are more computationally efficient since they do not need the SVD of the system matrix in order to solve the problem. Overall the performance of the different iterative solvers was the same, only with a few methods performing slightly better than the others dependent on the test problem. In the next section an extension of the problem will be considered that will lead to a decrease in the nice properties of this system.

Chapter 5

Blurring

As stated earlier the detectors in the laboratory set-up are CCDs with the same amount of pixels on each of them but with different pixel sizes on each of the CCDs. A CCD is also the main component of a camera, and just like a picture can get blurred due to atmosphere noise, shaky conditions as the picture was taken etc., the detections of the experiment can also get blurred. This chapter will describe a simple blurring process on the two-dimensional problem that was described and dealt with in the previous chapter. The blurring model that will be used in this chapter is described in, e.g., [8].

5.1 Theory

As stated above blurring can for example occur when a picture is taken and the photographer is shaking a bit, and the photons do not only hit the pixel on the CCD they were supposed to, but also hit other pixels. This kind of blurring is called motion blur. The concept of blurring can be expressed as an inverse problem

$$\boldsymbol{A}^{\mathrm{bl}}\boldsymbol{x} = \boldsymbol{b}, \tag{5.1}$$

where we wish to find the deblurred image, $\boldsymbol{x}$, from the blurred image $\boldsymbol{b}$ and the knowledge of the blurring that is stored in $\boldsymbol{A}^{\mathrm{bl}}$. The different kinds of blurring all give rise to a certain *Point Spread Function* (PSF) array, that describes the blurring of a specific point or pixel. The PSF should reflect the blurring that happens to a specific pixel in the non-blurred image. This means that if we had an image with only one non-zero pixel, the PSF array would be an image of that pixel after blurring.

From the PSF array we can construct the model matrix $\boldsymbol{A}^{\text{bl}}$. The assumptions made about the boundary conditions are represented in $\boldsymbol{A}^{\text{bl}}$. The boundary conditions will reflect what is assumed about the image outside the small field that we are trying to capture. Three common choices of boundary conditions are zero, periodic and reflexive boundary conditions. The different boundary conditions all result in a certain structure of the blurring matrix. This could be a Toeplitz or Hankel structure or a mix of the two. When dealing with blurring on the CCDs of the diffraction problem, reflexive boundary conditions will be assumed. Zero boundary conditions will often lead to artifacts on the pixels closest to the edge. And due to the small dimensions of the examples, these artifacts will therefore be very significant. The reflexive boundary conditions will on the other hand limit these artifacts in the reconstructions and give us a nice structure of the blurring matrix. The reflexive boundary conditions give rise to a matrix of blocks with a so-called Toeplitz-plus-Hankel structure. For further readings on PSF arrays and the construction of the model matrix, see [8].

5.2 Blurring Model

As for now we know that our detections are given by

$$\boldsymbol{b}^{\text{exact}} = \boldsymbol{A}^{\text{dif}}\boldsymbol{x}^{\text{exact}}, \tag{5.2}$$

where $\boldsymbol{A}^{\text{dif}}$ refers to the system matrix of the diffraction from the previous chapter. The blurred detections will then be given by

$$\boldsymbol{b}^{\text{bl}} = \boldsymbol{A}^{\text{bl}}\boldsymbol{b}^{\text{exact}}, \tag{5.3}$$

with $\boldsymbol{A}^{\text{bl}}$ being the blurring matrix. Poisson noise will hereafter be added in the way described in Section 3.2.1, such that

$$\boldsymbol{b} = \boldsymbol{b}^{\text{bl}} + \boldsymbol{e}. \tag{5.4}$$

The noise can be seen as additive, therefore a noise vector is introduced. This means that our final discrete problem ends up being

$$\boldsymbol{A}^{\text{bl}}\boldsymbol{A}^{\text{dif}}\boldsymbol{x} = \boldsymbol{b}. \tag{5.5}$$

The blurring on each of the CCDs are not necessarily the same, but it is assumed that it has the same behavior on each of the detectors. For stating and defining the blurring that happens at each CCD will we look at what happens at just one CCD. This could be the first detector, and for this is the mathematical model given as

$$\boldsymbol{A}_1^{\text{bl}}\ \boldsymbol{A}_1^{\text{dif}}\boldsymbol{x} = \boldsymbol{b}_1. \tag{5.6}$$

The blurring matrix will be square such that $\boldsymbol{A}_1^{\text{bl}} \in \mathbb{R}^{N_d^2 \times N_d^2}$. The blurring that happens at the CCDs of the diffraction problem is assumed to be Gaussian and have the same variance in the horizontal and vertical direction. The variance is set to two pixels in each direction. The basis of this assumption is the fact that in [2] it is explained how the PSF have been estimated prior to the experiments, by making measurements at cobber grains. Due to the assumption will the PSF array be separable, such that it is given by

$$\boldsymbol{P} = \boldsymbol{c}\boldsymbol{r}^T = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} \begin{bmatrix} r_1 & r_2 & \cdots & r_n \end{bmatrix}. \tag{5.7}$$

When the PSF is applied to each of the pixels this will result in a matrix with blocks defined by a Kronecker product. This means that the blurring matrix is given by

$$\boldsymbol{A}_1^{\text{bl}} = \boldsymbol{A}_r \otimes \boldsymbol{A}_c = \begin{bmatrix} a_{11}^{(r)}\boldsymbol{A}_c & a_{12}^{(r)}\boldsymbol{A}_c & \cdots & a_{1n}^{(r)}\boldsymbol{A}_c \\ a_{21}^{(r)}\boldsymbol{A}_c & a_{22}^{(r)}\boldsymbol{A}_c & \cdots & a_{2n}^{(r)}\boldsymbol{A}_c \\ \vdots & \vdots & & \vdots \\ a_{n1}^{(r)}\boldsymbol{A}_c & a_{n2}^{(r)}\boldsymbol{A}_c & \cdots & a_{nn}^{(r)}\boldsymbol{A}_c \end{bmatrix}, \tag{5.8}$$

where $\otimes$ defines the Kronecker Product. The elements of $\boldsymbol{A}_r$ and $\boldsymbol{A}_c$ are given by $\boldsymbol{r}$ and $\boldsymbol{c}$ respectively. Depending on the boundary conditions considered $\boldsymbol{A}_r$ and $\boldsymbol{A}_c$ will have a special structure. For the reflexive boundary conditions, that will be considered in the following, will $\boldsymbol{A}_r$ and $\boldsymbol{A}_c$ be Toeplitz-plus-Hankel matrices.

The blurring that happens at the first detector has now been defined. Since it is assumed that the blurring that happens at the remaining detectors are similar, although maybe not with the same parameters, we can now define the full blurring matrix as

$$\boldsymbol{A}^{\text{bl}} = \begin{bmatrix} \boldsymbol{A}_1^{\text{bl}} & 0 & 0 \\ 0 & \boldsymbol{A}_2^{\text{bl}} & 0 \\ 0 & 0 & \boldsymbol{A}_3^{\text{bl}} \end{bmatrix}, \tag{5.9}$$

where $\boldsymbol{A}_k^{\mathrm{bl}}$, $k = 1, 2, 3$, is defined by Kronecker products. In Figure 5.1 the detections of the polycrystal test problems are seen. First the exact detections, then the blurred one and finally the noisy detection. It is seen how the blurring has affected the pixels, such that the value of the pixels on the exact detection is distributed on the neighbor-pixels. In this case a Gaussian PSF with standard deviation two has been used, such that the value of each pixel is on average spread in a range of two pixels in all directions. The corresponding image of the powder test problem can be found in Appendix C on Figure C.1.

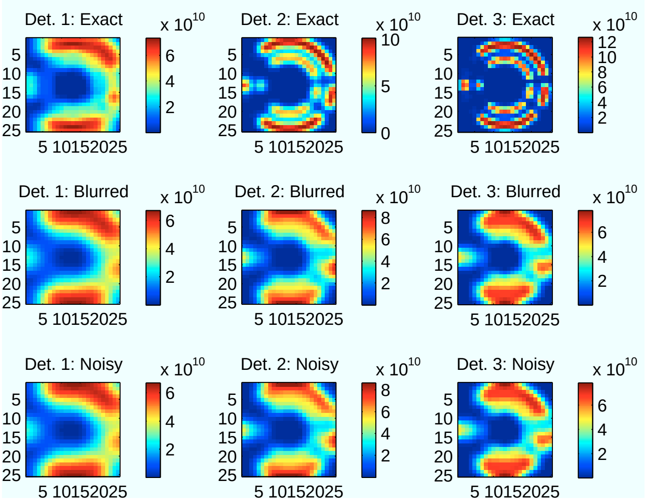


Figure 5.1: Top row: Exact detections. Middle row: Blurred detections. Bottom row: Noisy and blurred detections

As described earlier, all the simulations and experiments of this thesis work have been carried out by performing the forward operation with the system matrix on a known problem and hereafter by adding noise to the right-hand side. This kind of inverse crime will also be committed when we are working with the blurring model. It is important to note that the model described above is not necessarily the best way to construct a blurred image, that can be used for solving the problem. The optimal way of doing it, is to use the forward model to blur a large version of the image, that we wish to reconstruct, and hereafter sort of cut out a smaller image from the blurred. This will remove the artifacts that can occur at the edges of the blurred image. Since the blurring in this model is a part of another forward operation, it is not possible to blur in the most optimal way.

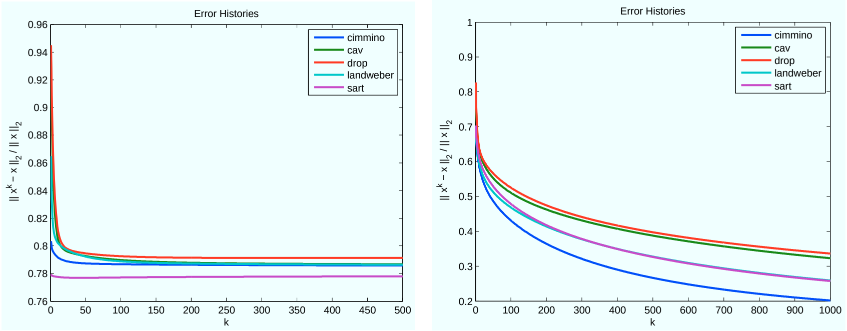
In Chapter 4 a noisy version of the problem was solved. We could conclude that we were able to solve the problem to a certain error level dependent on the test problem at hand. If we are now able to solve the problem when the blurring is present and show that the errors decrease, then we are one step closer to solving a real world problem.

5.3 Reconstruction

The iterative solvers will also be used for solving this problem and it is expected that the solutions are now of worse quality than the solutions that was seen in Section 4.6. The relative error reached a level of approximately 0.8 and 0.1 for the two test problems respectively for all the solution methods.

The results of the reconstructions from the blurred detections can be seen in Appendix C. But in Figure 5.2-5.4 the error histories for both test problems are shown when solving with the different classes of iterative methods. Comparing these error histories with the corresponding ones of the non-blurred problem in Section 4.6, we see that for the first test problem is the level at which the relative errors start to stagnate the same no matter what iterative method that is considered. Similar for all the methods is that the iteration at which the error level is stagnating is higher for the blurred version than the non-blurry version. For the second test problem the level of relative error that is reached by the methods when solving the blurred version is a little higher than for the non-blurry version. Again the lowest error level is reached after more iterations than in the non-blurred problem setting.

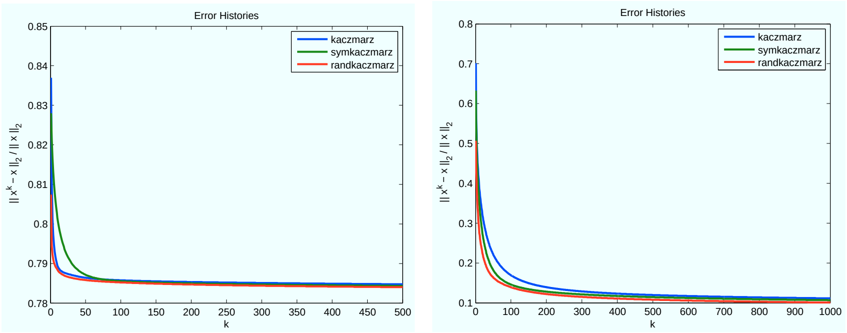
From the investigations done in this section we can conclude that we are also able to solve the test problems of the diffraction problem when the factor of blurring is taken into account in the model. As a final test the next section will deal with test problems that are much more complicated than the two considered so far.



(a) Test problem 1

(b) Test problem 2

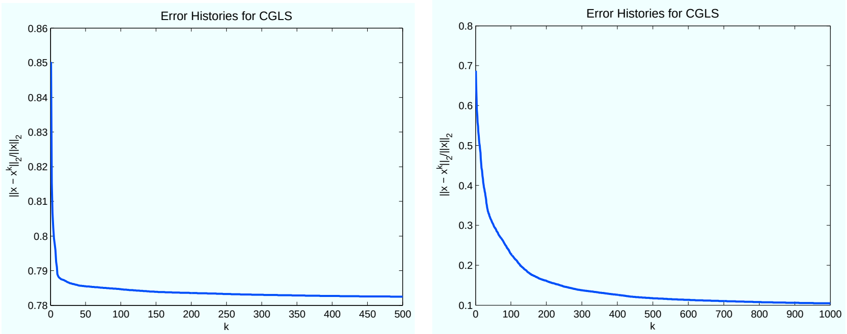
Figure 5.2: Error histories for the five SIRT methods.



(a) Test problem 1

(b) Test problem 2

Figure 5.3: Error histories of the three ART methods.



(a) Test problem 1

(b) Test problem 2

Figure 5.4: Error histories of the CGLS method.

Chapter 6

Complex Problem

The test problems considered so far have been approximations to what we assume about a polycrystal material. These test problems were considered in order to show that the solvers used in the reconstruction, returned solutions that converged. We were able to show that these problems were solvable and that the solutions in fact led to semi-convergence - even though the increase in relative error happened after a lot of iterations. Even after adding one more aspect to the problem in the form of blurring of the detections at the near- and far-fields, the test problems were solvable. This chapter will deal with a much more complex test problems than what has been considered so far. If we can show the convergence for these, we are close to being able to solve a real world problem.

In the test problems earlier the variation on the spatial grid was described by a Gaussian distribution - it was smooth and controlled. In this more complex test problems this will not be case. Instead the pixels on the source that will emit rays will be selected randomly, and the intensity of it will be chosen in the same way. For each of the source pixels the parameters of the angular Gaussian distribution will also be chosen at random, which mean that from each source pixel, the rays will emit in a unique way.

As a start though, test problems with two point sources moving closer and closer will be considered. This will give us an indication about how good the solvers are at reconstructing the spatial variation and how good it is at distinguishing the two points that move closer and closer. Figure 6.1 shows the error histories of this problem when solved with the five different SIRT methods. The error histories plotted are for one noise realization. In order to make sure that the results are representative for the general behavior of the methods, each instance of the two point source problem is solved for several different noise realizations. For the two test problems considered earlier the point at which the curve of the error histories started stagnating was between the 50th and 100th iteration. So making the test problem this much more complex has had a major impact on the course of convergence. Figure 6.2 shows the error histories when the SIRT methods are given a trained relaxation parameter as input. Similar to what was concluded earlier is the convergence of the iterations with a trained value for the relaxation parameter faster than with the default value. When the two point sources are moved closer and closer together the level of the error curves increase, which mean that the error in general is increased. For the iterative methods it gets harder and harder to distinguish the two point sources that emit light.

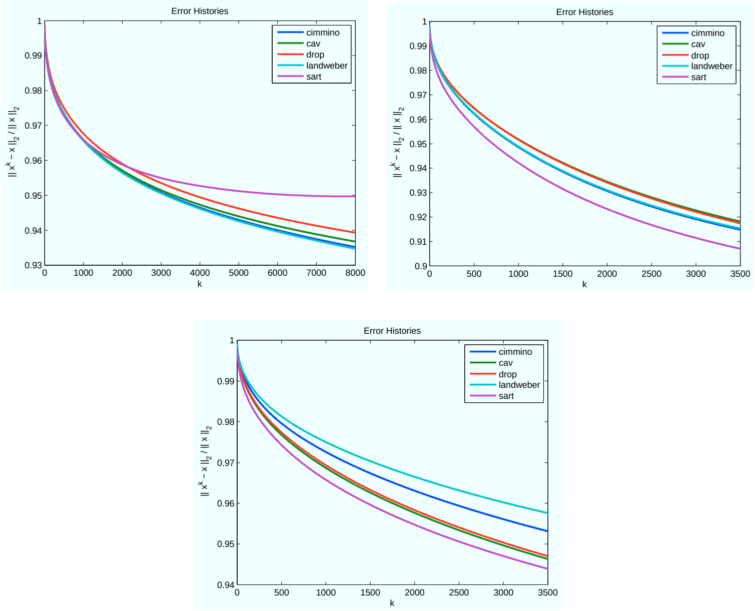


Figure 6.1: Error histories of the two point source problem when solved with the SIRT methods.

The error histories for the two point source problems showed us that the convergence against the point at which the relative error stagnates is now much slower

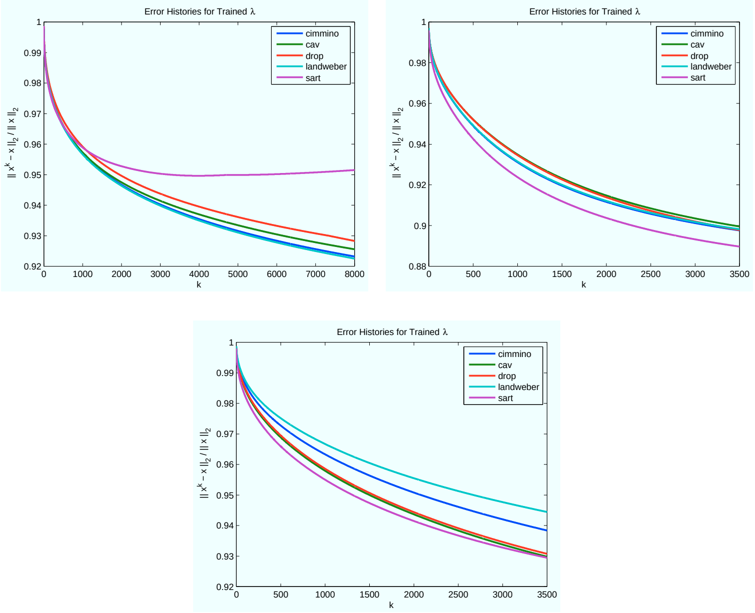
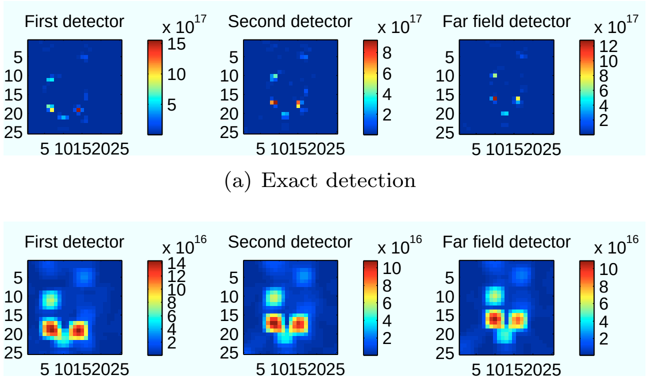


Figure 6.2: Error histories of the two point source problem when solved with the SIRT methods with a trained value for the relaxation parameter λ.

than the earlier test problems with little variation in the spatial directions on the source. The number of iterations needed is significantly higher than for the simple test problems considered earlier. The corresponding results of the ART and CGLS methods are seen in Appendix D.

As a last test a totally random problem will be generated. The exact right-hand side will be found by forward operation with $\boldsymbol{A}$ and hereafter the right-hand side will be influenced by a blurring matrix as described in Section 5.2. In Figure 6.3 the detections of the complex test problem is shown - both the exact and the noisy and blurred detections. On each detector the signal is blurred with a Gaussian PSF with standard deviation two. The difference between the exact detection and the noisy detections look very significant in this case. When we wish to reconstruct the intensity distribution function giving rise to these detections it is important to have in mind that just above we saw that the amount of iterations needed for reconstructing the two point sources problem was much larger than for the two simple test problems considered earlier. This is probably also the case for this problem. Again both the SIRT and ART solvers are considered as well as the CGLS method.



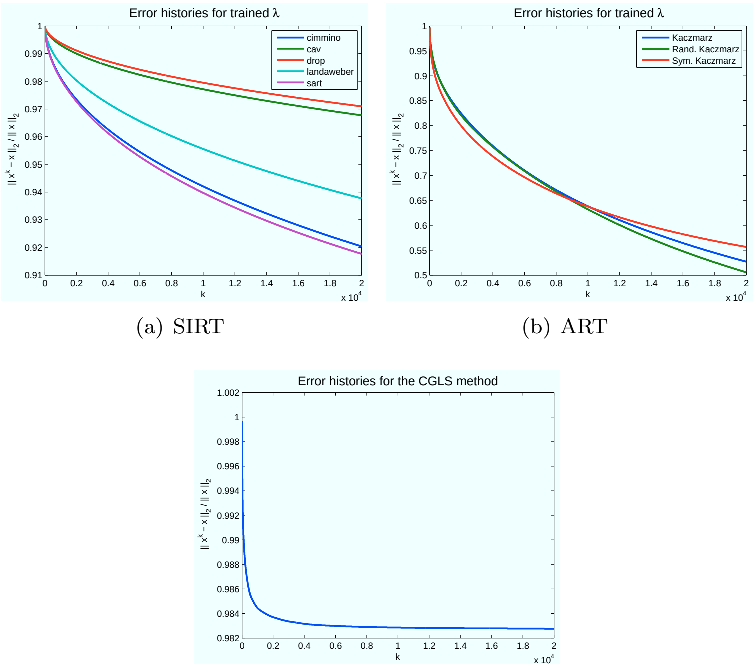
(a) Exact detection

(b) Blurry and noisy detection

Figure 6.3: Detections of the complex problem.

Figure 6.4 shows the error histories of the iterative methods divided by class. The relaxation parameter has been trained for the SIRT and ART methods. The convergence of the five SIRT methods are very much alike and as we saw earlier is the convergence rate slow. Comparing the convergence of the SIRT and ART methods we see that the three ART methods converge significantly faster than the SIRT methods. What is worth noticing about the error histories of the ART methods is that the training of the relaxation parameter seems to have improved the performance of all methods. The progress of the errors for the CGLS method is also rather slow and it seems like it stagnates earlier than the SIRT and ART methods. The error histories illustrate once again that the ART methods converge significantly faster than the SIRT methods. The number of iterations considered for the methods are the same, but the ART methods reach an error level approximately twice as low as the SIRT methods - and for CGLS as well. One has to bear in mind that symmetric Kaczmarz' uses the double amount of work units than all the other iterative methods. But since it is randomized Kaczmarz' that reaches the lowest level this is of no great importance in this comparison. For the test problems considered in Chapter 4 it was in some sense straightforward to also make a visual inspection of the quality of the reconstructions. Since the angular distribution for the complex test problems varies for each pixel on the source it would be a cumbersome task to make any comparisons.

What can be concluded from these simulations is that the convergence of the iterative methods will be slow no matter what method is chosen when a complex solution is the goal. Although the convergence is slow it is present and if we had had a higher spatial resolution, a lower error level might have been reached. Moreover the ART methods showed great performance when comparing with the



(a) SIRT

(b) ART

(c) CGLS

Figure 6.4: Error histories of the iterative methods.

other iterative methods considered. The convergence is significantly faster and they outperform the other methods. They reach a relative error of half as much as the other classes of methods within the same amount of iterations. Due to the concept of work units that was introduced earlier, we can therefore conclude that the ART methods are preferable when a complex problem is considered.

Chapter 7

Conclusion

The goal of this thesis work was to gain an understanding of the General Ray Tracing problem by setting up a mathematical model describing it in the context of the experiments carried out at the European Synchrotron Radiation Facility. It was important for us that the forward model was formulated in general terms and that the laboratory set-up was used as an example of this problem.

Throughout the thesis the steps of Section 2.5 was used as a guideline. The first step in the process is to gain an understanding of the underlying problem - in this case, the experiment carried out at the European Synchrotron Radiation Facility. Secondly we formulated the mathematical model for the problem. At first a simplified version was formulated and analyzed. This model and the results of the analysis was then used as a point of reference for further development of the model. For both the simplified and the more complex model the analysis by means of the SVD revealed a weakness about the model. Due to the properties of the far-field, the intensity distribution is described well in the angular directions, whereas the spatial variation is less evident. When the iterative methods were used for solving simple test problems a satisfying error level was reached within few iterations. It was also shown that making the model more advanced by introducing blurring at the CCD's on the detectors, made the problem ill-conditioned, but it was still solvable when simple test problems where considered, although more iterations were needed in order to reach a satisfying level of relative error. No significant difference was seen in the performance of the iterative methods. The model that was used to add noise to the right-hand side of the system had great properties for estimating noise levels for use in the stopping criteria used. For the simple test problems this meant that the Discrepancy Principle and the Monotone Error Rule returned good solutions, whereas the performance of the NCP criterion was more unsteady.

When we moved on to solving problems of greater complexity with a lot of variation in the spatial directions the performance of the iterative methods decreased significantly. But the methods of the ART type showed great performance compared to the other classes of methods and that without extra computational cost.

All in all the goal of this thesis has been reached - a model for the General Ray Tracing Problem has been formulated and analyzed, and we have seen that for the special application of diffraction, we are able to solve the inverse problem that arise.

7.1 Future Work

The work done in this thesis is a basis study of the diffraction problem. This study has given rise to a lot of issues that can be treated in further studies. The first and main thing that could be improved is the routine for setting up the forward model. Concerning the two-dimensional problem the main issue was the low resolution on both the detectors and especially the source. At the experiment done at ESRF in Grenoble the resolutions on the detectors were 2048×2048 pixels. In the simulations done in this thesis, the resolution was limited by both the memory capacities of Matlab and the computational effort that had to be put into it. So a future improvement of the project could be to implement the routines in another language. This could possibly lead to lower running times but also memory-wise could it be advantageous. The problems considered in the two-dimensional section were all underdetermined and we saw that the reconstructions could only reach a certain level of relative error dependent on the problem.

One should bear in mind that the resolution of the detectors in the laboratory set-up will give rise to a large-scale system with emphasize in large. Dealing with matrices of this size it could be interesting to look at other methods for solving the problem than the deterministic ones that have been used throughout this thesis work. This could be a stochastic solver or a hybrid method, that combines a stochastic and deterministic method. The information in the far field could also be used in an efficient way when working with stochastic solvers. In this thesis it has been showed that it is possible to determine the angle distribution from the detections at the far field. This information could be used as a priori knowledge for a stochastic solver. A stochastic solver needs less memory in order to reach a solution.

Appendix A

Noise Level

This appendix deals with another way to formulate the noise model for the diffraction problem and will moreover give an argument for the level of noise added in the simulations.

As described in [3] can the noise in data be Poisson distributed such that

$$b_j \sim \mathcal{P}\left(\eta b_j^{\text{exact}}\right)/\eta. \tag{A.1}$$

The parameter η will decide the level of noise in data. For those samplings where b_j^{exact} is zero, the Poisson distribution is not defined. So for these elements in $\boldsymbol{b}^{exact}$, the elements in $\boldsymbol{b}$ will also be set to zero. But there will most likely be some background noise present, such that no measurements will be zero. Therefore Gaussian white noise with standard deviation 1 is added, such that

$$\boldsymbol{b}_j = \mathcal{P}(\eta \boldsymbol{b}_j^{\text{exact}})/\eta + \texttt{rand(1,1)}. \tag{A.2}$$

When we wish to find the angle distribution from the far field as described in Section 3.4 it is then important to take this background noise into consideration. For now an experiment will be carried out by solving one instance of the problem, but by adding noise at different levels it will be investigated how much influence the noise level has on the solution. In Figure A.1 five solutions of the Tikhonov regularization with optimal regularization paramter are seen along with the exact intensity distribution function. When the value of η is big the noise level is low. The five different solutions show us that a suitable noise level could be found for $\eta = 1$. In the following simulations of the problem we will therefore use

$$b_j = \mathcal{P}(b_j^{\text{exact}}) + \texttt{rand(1,1)}. \tag{A.3}$$

When we wish to use the stopping criteria described in Section 2.4, the noise level will be estimated by

$$\|\boldsymbol{e}\|_2 = \sqrt{\|\boldsymbol{b}\|_1}, \tag{A.4}$$

since the expected value of the Gaussian white noise is 0.

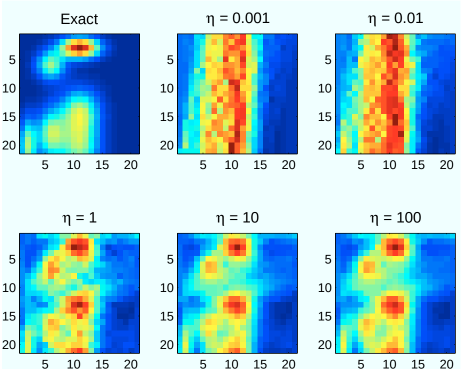


Figure A.1: Solutions for $\eta = 10^{-3}, 10^{-2}, 1, 10^{2}, 10^{3}$.

Appendix B

Two-dimensional Problem

B.1 Left Singular Vectors

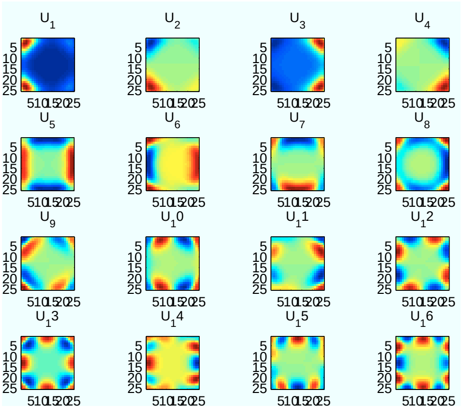
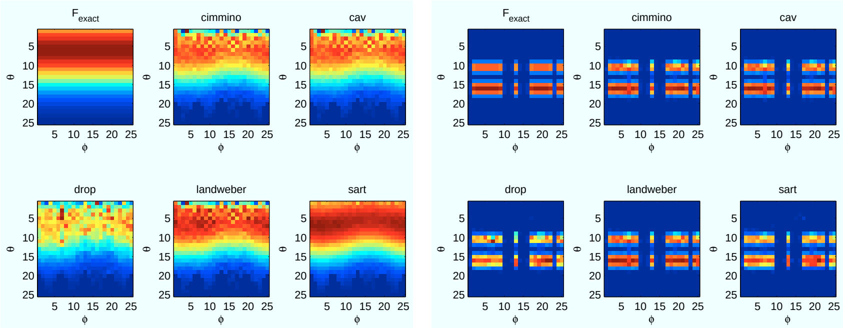


Figure B.1: First sixteen left singular vectors.

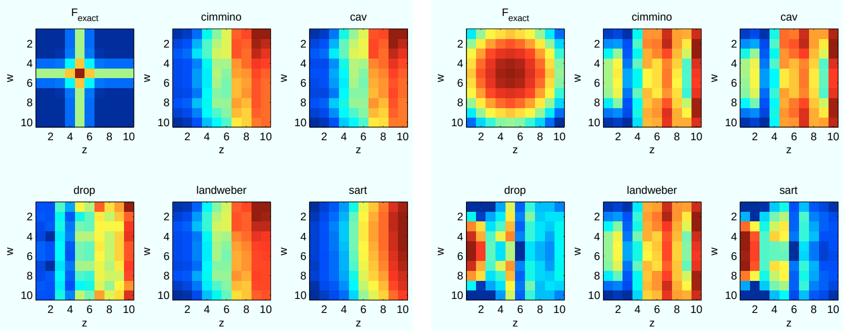
B.2 SIRT Solutions



(a) Test problem 1

(b) Test problem 2

Figure B.2: Reconstructions when seen from (ϕ, θ).

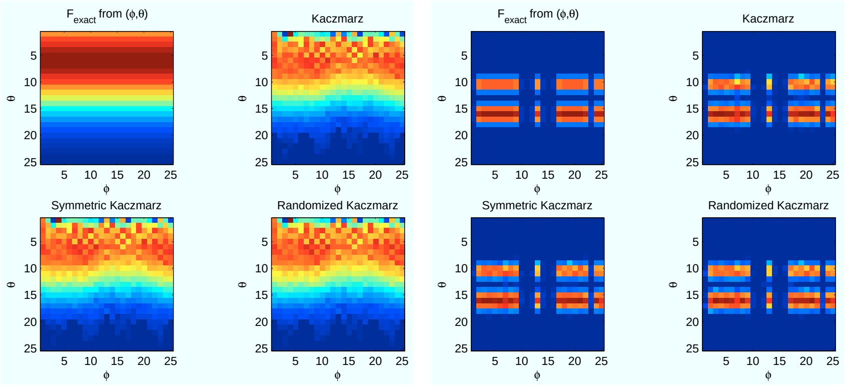


(a) Test problem 1

(b) Test problem 2

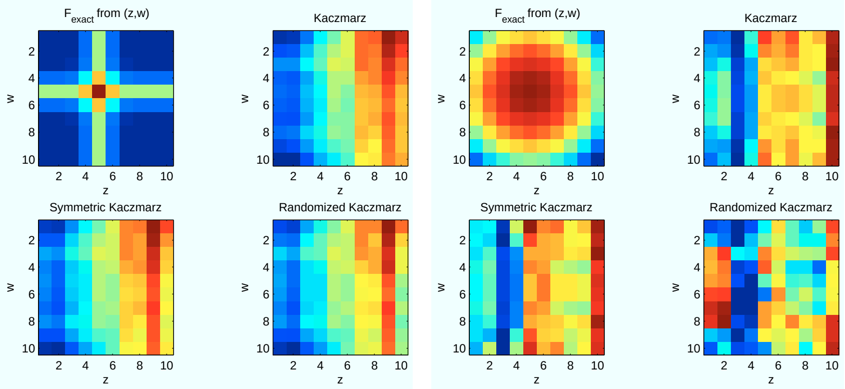
Figure B.3: Reconstructions when seen from (z, w).

B.3 ART Solutions



(a) Test problem 1 (b) Test problem 2

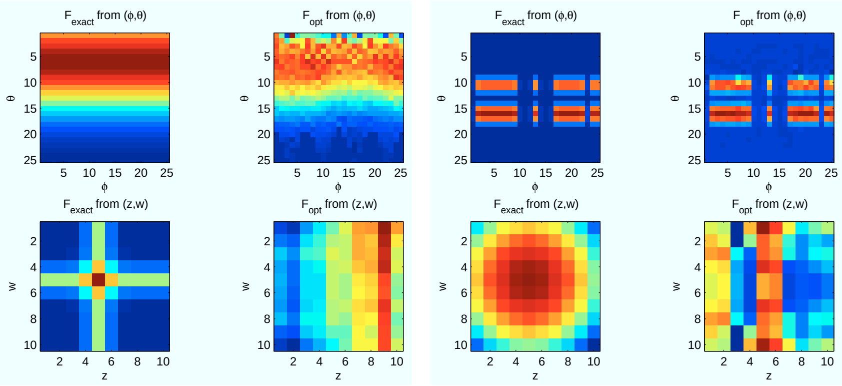
Figure B.4: Reconstructions when seen from (φ, θ).



(a) Test problem 1 (b) Test problem 2

Figure B.5: Reconstructions when seen from (z, w).

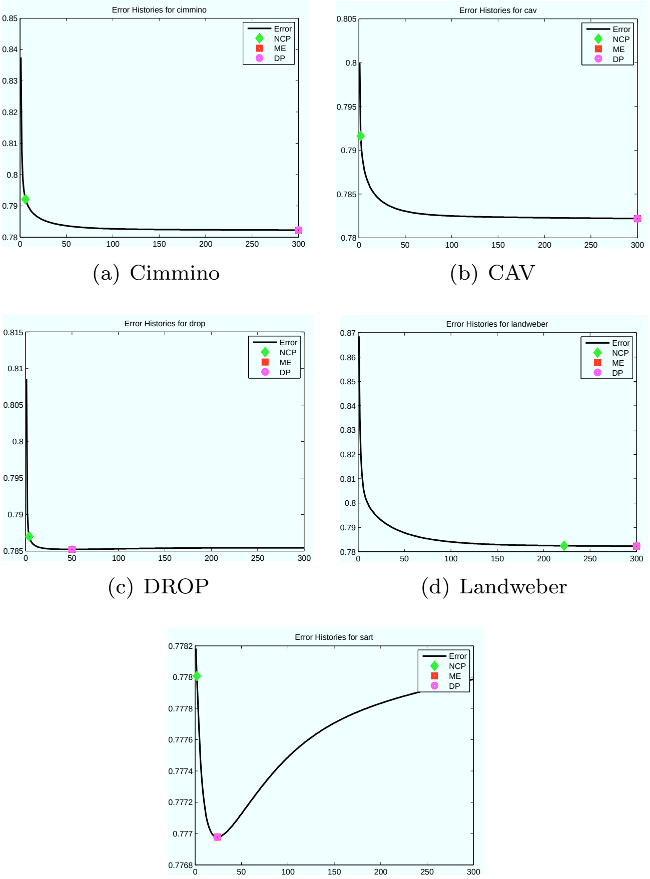
B.4 CGLS Solution



(a) Test problem 1 (b) Test problem 2

Figure B.6: Reconstructions by the CGLS method.

B.5 Stopping Criteria 2D



(a) Cimmino

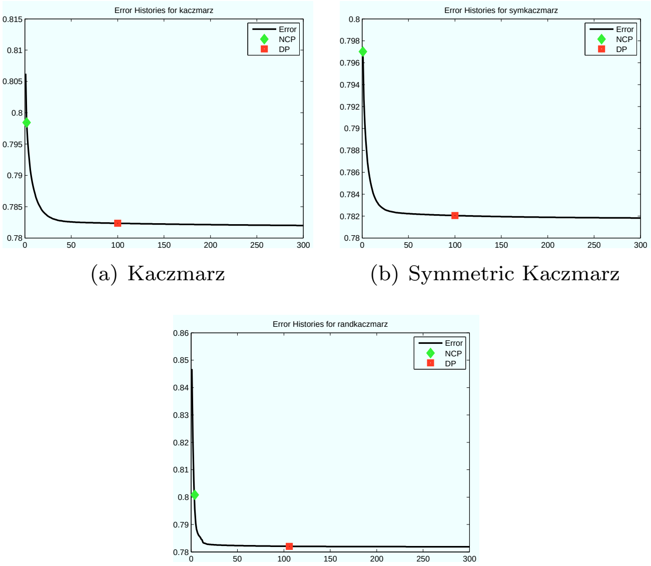
(b) CAV

(c) DROP

(d) Landweber

(e) SART

Figure B.7: Error Histories of the first test problem stopped at different times due to different stopping rules.

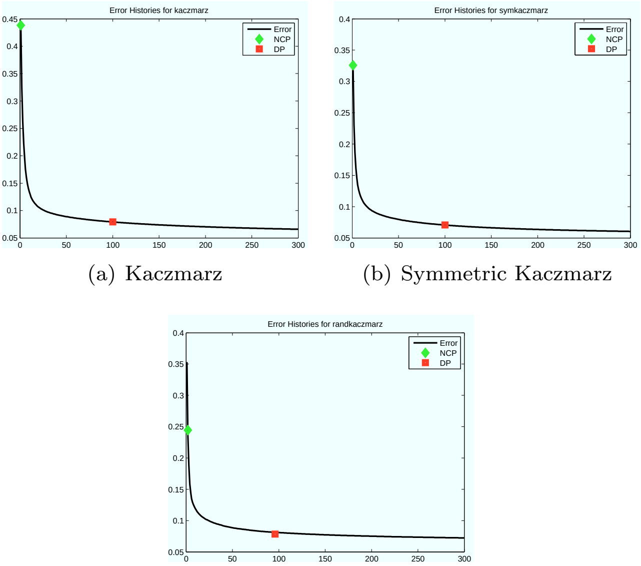


(a) Kaczmarz

(b) Symmetric Kaczmarz

(c) Randomized Kaczmarz

Figure B.8: Error histories of the first test problem for the three ART methods when stopped by different stop rules.



(a) Kaczmarz

(b) Symmetric Kaczmarz

(c) Randomized Kaczmarz

Figure B.9: Error histories of the second test problem for the three ART methods when stopped by different stop rules.

Appendix C

Blurring

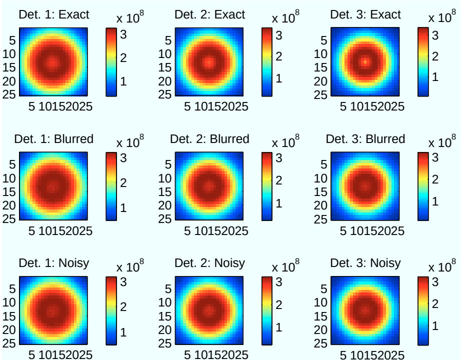
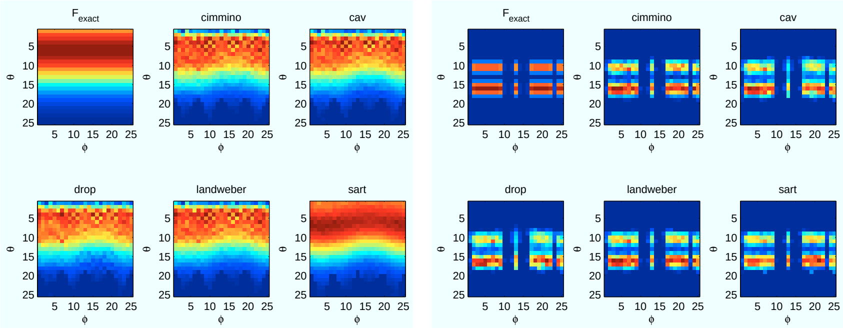


Figure C.1: Top row: Exact detections. Middle row: Blurred detections. Bottom row: Noisy and blurred detections

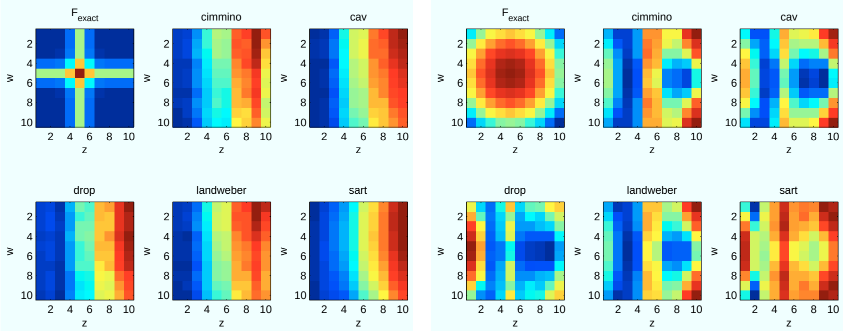
C.1 SIRT Solutions



(a) Test problem 1

(b) Test problem 2

Figure C.2: Reconstructions when seen from (φ, θ).

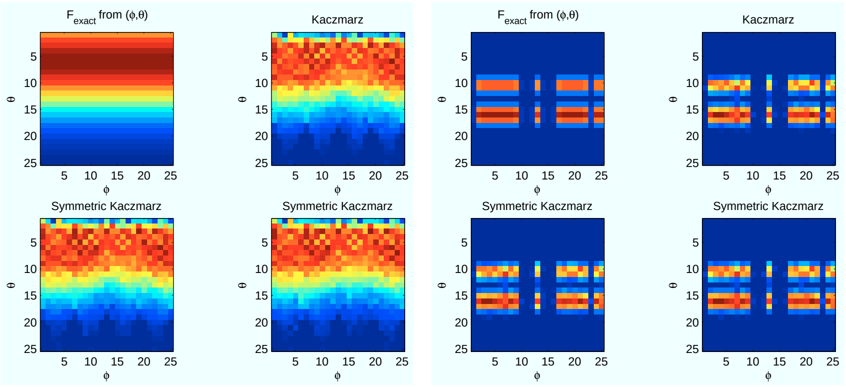


(a) Test problem 1

(b) Test problem 2

Figure C.3: Reconstructions when seen from (z, w).

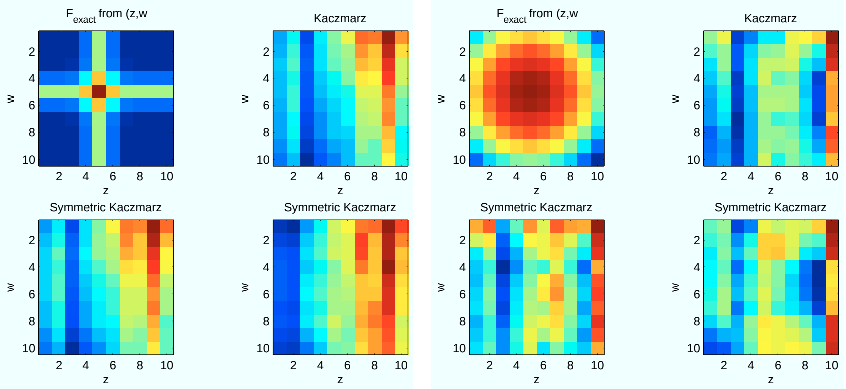
C.2 ART Solutions



(a) Test problem 1

(b) Test problem 2

Figure C.4: Reconstructions when seen from (φ, θ).

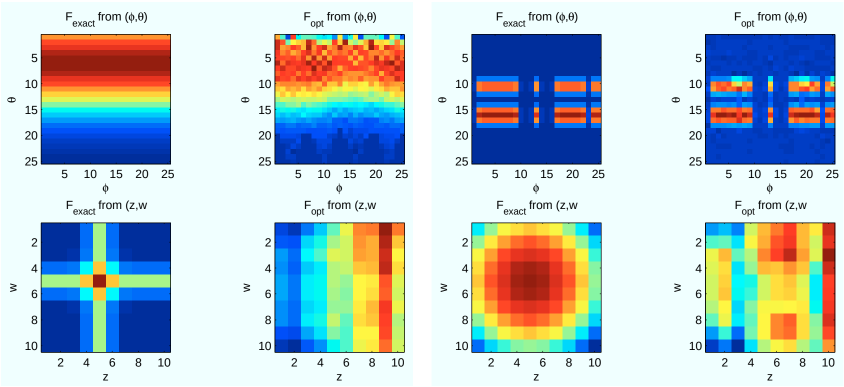


(a) Test problem 1

(b) Test problem 2

Figure C.5: Reconstructions when seen from (z, w).

C.3 CGLS Solutions



(a) Test problem 1

(b) Test problem 2

Figure C.6: Reconstructions from both (φ, θ) and (z, w).

Appendix D

Complex Problem

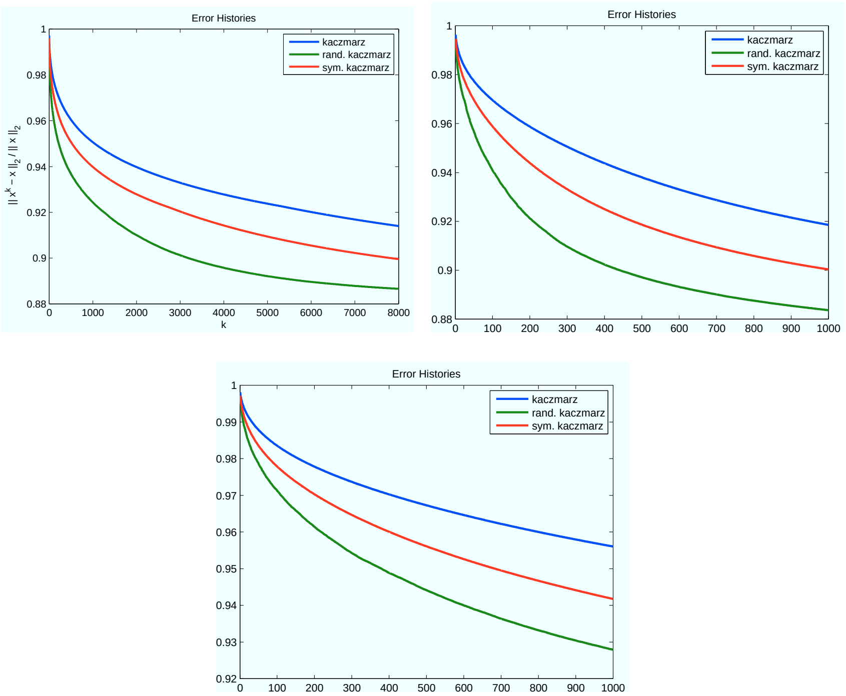


Figure D.1: Error histories of the two point source problem when solved with the ART methods.

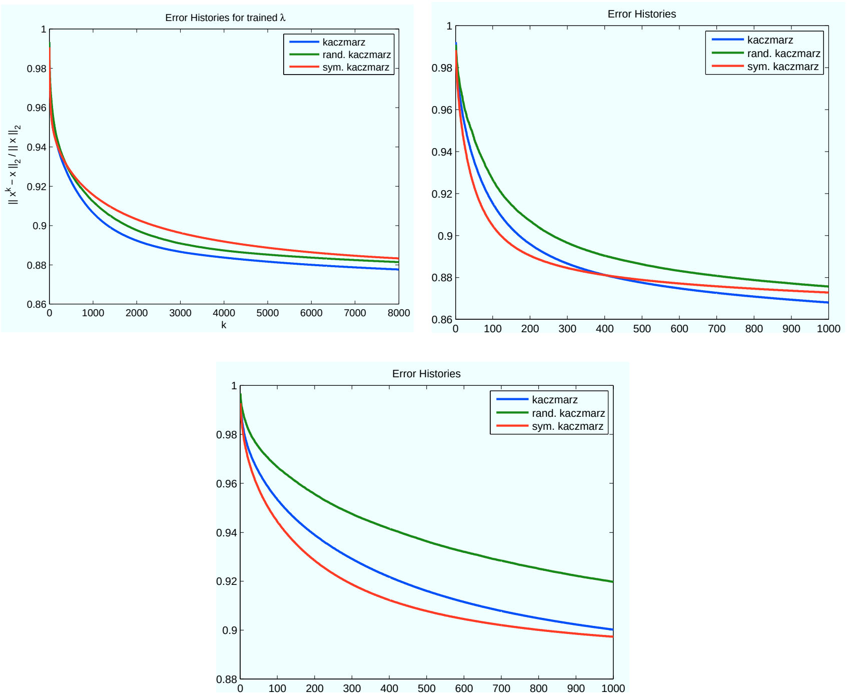


Figure D.2: Error histories of the two point source problem when solved with the ART methods with a trained value for the relaxation parameter λ.

Bibliography

[1] Tommy Elfving and Touraj Nikazad. Stopping Rules for Landweber-type Iteration. *Inverse Problems*, 23(4):1417–1432, 2007.

[2] Søren Schmidt et. al. Direct observation of 3-D grain growth in al-0.1% mn. *Scripta Materiala*, 59(5):491–494, 2008.

[3] P. C. Hansen. *Discrete Inverse Problems - Insight and Algorithms.* SIAM, 2010.

[4] Per Christian Hansen. http://www2.imm.dtu.dk/ ˜pch/regutools/.

[5] Per Christian Hansen. The Discrete Picard Condition for Discrete Ill-posed Problems. *BIT Numerical Mathematics*, 30(4):658–672, 1990.

[6] Per Christian Hansen and Maria Saxild-Hansen. Air tools - a matlab Package of Algebraic Iterative Reconstruction Techniques. 2010.

[7] A. C. Kak and Malcolm Slaney. *Principles of Computerized Tomographic Imaging.* Society of Industrial and Applied Mathematics, 2001.

[8] D.P. O'leary P. C. Hansen, J.G. Nagy. *Deblurring Images - Matrices, Spectra and Filtering.* Siam, 2008.

[9] Jonathan Richard Shewchuk. An Introduction to the Conjugate Gradient Method Without the Agonizing Pain. 1994.

[10] P.C. Hansen & T. Nikazad T. Elfving. Semi-convergence and relaxation parameters for projected SIRT algorithms.