Universiteit
Leiden
The Netherlands

Error bounds for discrete tomography

Fortes, W.R.

Citation
Fortes, W. R. (2013, September 18). *Error bounds for discrete tomography*. Retrieved from https://hdl.handle.net/1887/21763

Version:	Corrected Publisher's Version
License:	Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden
Downloaded from:	https://hdl.handle.net/1887/21763

Note: To cite this publication please use the final published version (if applicable).

Cover Page

Universiteit Leiden

The handle http://hdl.handle.net/1887/21763 holds various files of this Leiden University dissertation.

Author: Fortes, Wagner Rodrigues
Title: Error bounds for discrete tomography
Issue Date: 2013-09-18

Error bounds for discrete tomography

Wagner Rodrigues Fortes

Error bounds for discrete tomography

Proefschrift

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van Rector Magnificus prof.mr. C.J.J.M. Stolker,
volgens besluit van het College voor Promoties
te verdedigen op woensdag 18 september 2013
klokke 13.45 uur

door

Wagner Rodrigues Fortes

geboren te Rio de Janeiro, Brazilië,
in 1983.

Samenstelling van de promotiecommissie:

promotor:	Prof. dr. K. J. Batenburg
promotor:	Prof. dr. ir. B. Koren, Eindhoven University of Technology
overige leden:	Prof. dr. R. Tijdeman Prof. dr. P. Stevenhagen Prof. dr. J. Sijbers, University of Antwerp, Belgium Dr. P. Balázs, University of Szeged, Hungary Prof. dr. L. Hajdu, University of Debrecen, Hungary

ISBN: 978-94-6191-854-3

©Wagner Rodrigues Fortes, Amsterdam, 2013
wfortes@gmail.com

Typeset using LaTeX
Printed by Ipskamp Drukkers
Fly on the cover from shutterstock.com

The research in this thesis has been supported by Erasmus Mundus and Leiden University.
It was carried out at Centrum Wiskunde & Informatica (CWI).

Universiteit Leiden

Erasmus Mundus

To my parents

Angela and Paulo

Contents

1. Introduction **1**
- 1.1. Discrete Tomography . . . 1
- 1.2. Difference between reconstructions . . . 2
- 1.3. Discrete reconstruction algorithms . . . 4
- 1.4. Feature detection . . . 4
- 1.5. Overview . . . 5

2. Bounds on the quality of reconstructed images in binary tomography **7**
- 2.1. Introduction . . . 7
- 2.2. Notation and model . . . 9
- 2.3. The central reconstruction . . . 11
- 2.4. Quality bounds for binary solutions . . . 12
  - 2.4.1. Elementary bounds based on the central reconstruction . . . 12
  - 2.4.2. Bounds based on rounding the central reconstruction . . . 14
  - 2.4.3. Bounds involving the number of ones in binary solutions . . . 16
- 2.5. Experiments and results . . . 17
  - 2.5.1. Concepts and interpretation . . . 19
  - 2.5.2. Error bounds for the grid model . . . 21
  - 2.5.3. Error bounds for the strip model . . . 21
  - 2.5.4. Error bounds for a particular reconstruction . . . 22
- 2.6. Discussion of the results . . . 23
- 2.7. Outlook and conclusions . . . 24

3. Error bounds on the reconstruction of binary images from low resolution scans **29**
- 3.1. Introduction . . . 29
- 3.2. Notation and concepts . . . 30
- 3.3. Error bounds . . . 32
- 3.4. Experiments and results . . . 33
- 3.5. Outlook and conclusions . . . 35

4. Bounds on the difference between solutions in noisy binary tomography **37**
- 4.1. Introduction . . . 37
- 4.2. Notation and model . . . 38

4.3. A bound on the difference between all binary solutions 38
4.4. Bounds based on rounding 40
4.5. Computations 42
4.6. Conclusions 43

5. Practical error bounds for binary tomography **45**
5.1. Introduction 45
5.2. Approach 46
5.3. Experiments 47
5.4. Discussion and Conclusions 49

6. Quality bounds for binary tomography with arbitrary projection matrices **51**
6.1. Introduction 51
6.2. Notation and the minimum norm solution 52
6.3. The Euclidean norm of binary solutions 53
6.4. A bound based on the minimum norm solution 56
6.5. Bounds based on rounding the minimum norm solution 57
6.6. Bounds based on a subsequent radius reduction 58
6.7. Numerical experiments 61
6.7.1. Tomography models 61
6.7.2. Bounds on the number of ones in binary solutions 63
6.7.3. Bounds on the difference between binary solutions and binary approximate solutions 65
6.7.4. Discussion of the results 66
6.8. Outlook and conclusions 67

7. Approximate discrete reconstruction algorithm **73**
7.1. Introduction 73
7.2. Notation and concepts 74
7.3. Algorithm description 75
7.4. Walkthrough example 77
7.5. The algorithm's proof 79
7.6. A threshold variation 81
7.7. Computations 82
7.7.1. Initial solution $\boldsymbol{x}^{(0)}$ 82
7.7.2. Non-null ghost images 83
7.8. Numerical experiments 83
7.8.1. Reconstruction comparison 85
7.9. Outlook and conclusions 91

8. A method for feature detection in binary tomography **93**
8.1. Introduction 93
8.2. Basic notation and model 94
8.3. Probe structure 95
8.4. Partially solving the probe problem 96

8.4.1. Probing by analyzing the binary solutions of the reduced linear system 97
8.4.2. Probing by analyzing the binary solutions of the original linear system 97
8.5. Numerical experiments 98
8.5.1. Homogeneous regions 98
8.5.2. Horizontal edges 99
8.6. Outlook and conclusion 100

Bibliography **103**

Summary **107**

Samenvatting **111**

Curriculum Vitae **115**

Chapter 1

Introduction

In this chapter we introduce the concept of discrete tomography and explain the basic concepts involved in this thesis. Furthermore, we describe the key problems considered in this thesis and outline the main results.

1.1. Discrete Tomography

Tomography is a technique for reconstructing an image from a series of projections of it. Such projections are acquired from a range of viewing angles. Images can be continuous or discrete in space. Space continuous 2-D images are often discretized in small squares, named *pixels*, which are assumed to have only one intensity value associated with it. Space discrete 2-D images are defined on a subset of $\mathbb{Z}^2$ and its points (or pixels) are assigned an intensity value. In both cases, reconstructed images are represented as a finite vector with each entry representing one pixel of the reconstruction region. All pixels outside of the region of interest are assumed to have zero intensity.

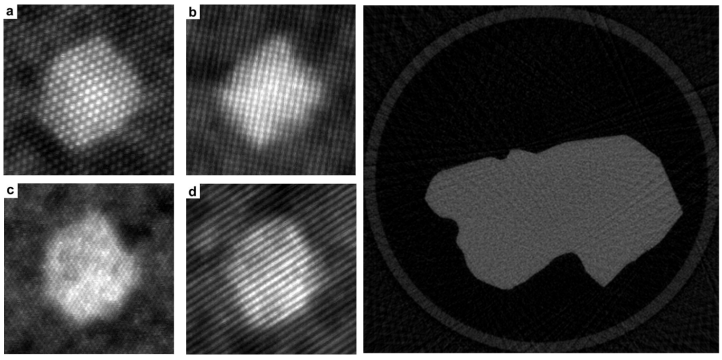
In this thesis, the term *discrete tomography* refers to the reconstruction of images with intensity values belonging to a small discrete subset of $\mathbb{R}$ and either continuous or discrete in space. Images with intensity values restricted to $\{0,1\}$ are called *binary images*. We refer to the corresponding reconstruction techniques as *binary tomography*.

The projection process is defined by the projection model and the set of angles and can be represented by a linear transformation. Given a set of projections, finding an image satisfying these projections is an inverse problem for which the existence and uniqueness of the solution is, in general, not guaranteed. The problem of finding an image from a set of projections is known as the *reconstruction problem*. Methods to find a solution to the reconstruction problem often yield an image that approximately satisfies the given set of projections. A solution or approximate solution of the reconstruction problem is called a *reconstruction*. If only a few projection angles are available, there may be large differences between the solutions of the *discrete reconstruction problem*, the reconstruction problem in discrete tomography.

An application of discrete tomography is the reconstruction of nano-crystals at atomic

resolution. In this problem, discrete atoms are positioned on a regular grid (in this case, it is a 3-D image). By using an electron microscope, 2-D projections are acquired from various angles, as displayed in Fig. 1.1(a) [1, 31].

Another application of binary tomography can be found in the imaging of diamonds. Diamonds are structures composed of only one element: carbon. The problem of reconstructing a diamond is a binary tomography problem where the carbon corresponds with the foreground and its absence corresponds with the background. Fibrous diamonds, however, may have cracks containing material with a small number of different densities [51]. In Fig. 1.1(b) we show a slice of a 3-D reconstruction of a diamond on a cylindrical holder. In Chapter 5 we present experiments with projections of a raw diamond.



(a) 2-D projections of a gold nanoparticle, reproduced with permission from [1].

(b) Tomographic reconstruction of a diamond from X-ray micro-CT data; courtesy of DiamCAD, Antwerp.

Figure 1.1: *Discrete tomography examples.*

There are also applications of discrete tomography in medical imaging, despite of its restricted use. Discrete tomography can be applied when some physical property of the organ of interest can be enhanced as compared to the surrounding tissue. As an example, see [29] for angiographic applications.

1.2. Difference between reconstructions

Despite the strong constraint imposed on the grey values in discrete tomography, many solutions of the reconstruction problem can exist, all corresponding to the same set of projections. If the projections are obtained by performing measurements on some unknown ground truth image, the reconstruction can then deviate substantially from the original image, as exemplified in Fig. 1.2. As a consequence, there is a need for a measurement on the difference between solutions of the reconstruction problem. As the ground truth is a solution by itself, this would also yield a bound on the reconstruction error with respect to the ground truth.

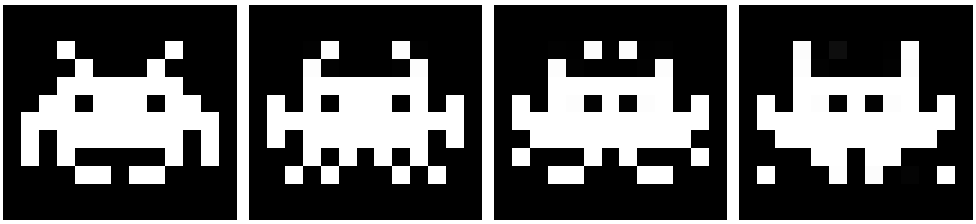


Figure 1.2: *All images have equal line sums in the vertical and horizontal directions.*

For images represented on a discrete grid, both lower and upper bounds have been obtained for the magnitude of changes in discrete solutions when the projections are slightly perturbed [2–4, 46]. For the case of binary image reconstruction from just two projections, horizontal and vertical, bounds on the difference between binary images having the same projections have been obtained by Van Dalen [44, 45].

In this thesis, we develop a series of computable upper bounds on the difference between reconstructions of binary images. Our approach is based on ideas initially proposed by Hajdu and Tijdeman [22]. When computing the absolute difference between corresponding pixels of two known binary images, one can identify whether such pixels differ or not, which allows the classification of a pixel of an image as correct or wrong with respect to the other image. Despite the fact that our methodology can be adapted to compute the cited bounds for non-binary images with small number of different grey values, we have restricted its development to binary images only.

In Chapter 2 we demonstrate that for parallel beam tomography, some projection models yield the property that the total intensity of all binary solutions of the reconstruction problem must be the same, and it can be computed directly from the projection data. Furthermore, all binary solutions lie on a hypersphere of which the center and radius are known. Based on these observations, a methodology is derived to compute an upper bound on the difference between any two binary solutions. As reconstruction algorithms may find an approximate solution, we also derive a bound on the difference between a given binary reconstruction and any binary solution.

Similar to the tomography problem, we investigate the problem of image reconstruction from low resolution scans. Video recordings, picture cameras and other devices record images subject to resolution restrictions. A sequence of recordings may be used to acquire an image with higher resolution than each recording individually. For such settings, Chapter 3 presents error bounds on the higher resolution reconstructed images.

Chapter 4 extends the previously developed bounds to deal with noise in the projection data. When perturbing the projection data, the reconstruction problem is likely to be inconsistent, having no solution. When solving inconsistent reconstruction problems, one finds solutions of an approximate problem. In this case, the set of solutions of the reconstruction problem may change significantly, yielding big differences between reconstructions due to the ill-posed nature of these inverse problems. Furthermore, the set of solutions may not contain the original image which generated the projection data. The error bounds developed

in this chapter provide bounds on the image error with respect to the solutions of the noiseless reconstruction problem, for which the projections are not known. Intending to make the error bounds useful in practice, we introduce parametrized approximations in these bounds in Chapter 5. The parameters are based on experiments with images in a controlled environment and then applied to a similar problem with real projection data. The resulting approximate bounds are no longer mathematical bounds, but still follow the same general behaviour as the true errors measured for phantom images.

So far, the error bounds were restricted to projections models with the property of conservation of total intensity of binary solutions. As a consequence, all binary solutions have the same total intensity. Chapter 6 studies the general case where any projection model can be used to compute reconstruction error bounds. In fact, these techniques can be used for any application modelled as an algebraic linear system of equations with binary solutions.

1.3. Discrete reconstruction algorithms

Discrete reconstruction algorithms exploit prior knowledge about the discreteness of the image that is being reconstructed. This can improve the reconstruction quality, or reduce the number of projections required while keeping the reconstruction quality the same. Another advantage of discrete tomography reconstruction algorithms is that the resulting reconstruction is an image that is already segmented.

A range of reconstruction algorithms for discrete tomography have been proposed in the literature, [6,9,28,40], however none of them comes with a guarantee that an exact solution of the discrete tomography problem is found for the general problem, from more than two directions. At the same time, the results of computational experiments show many cases of a near-optimal approximate solution, or even a reconstruction that is completely identical to the original image from which the projections were taken.

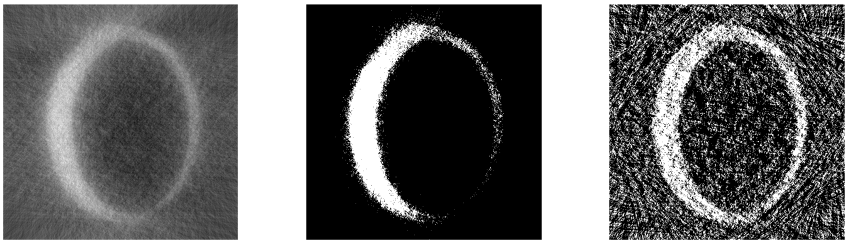
A major problem with these algorithms is the fact that the error - both in the reconstructed image (with respect to the ground truth image) and in the projected reconstructed image (with respect to the given projections) - depends on the particular problem instance and cannot be bounded sharply. We are not aware of any algorithm for which non-trivial bounds have been described for the difference between the projections of the reconstructed (discrete) image and the given projections.

In Chapter 7, we present a discrete approximate reconstruction algorithm that comes with such guarantees. The algorithm computes an image that has only grey values belonging to the given finite set (prior knowledge). It also guarantees that the difference between the given projections and the projections of the reconstructed discrete image is bounded. The bound is independent of the image size and proportional to the number of projection angles.

1.4. Feature detection

As mentioned in Section 1.2, the reconstruction problem may allow the existence of binary solutions that are substantially different from each other. Furthermore, Section 1.3 indicates how discrete reconstruction algorithms may not find an exact solution of the reconstruction

problem, but an approximation of it. When one is interested in finding a specific feature in binary tomography images, a specific reconstruction (either a solution or an approximate one) may not truly represent the original ground truth image. As an example, Figure 1.3 presents three different reconstructions of a ring-like shaped original image in which it is very difficult to determine, based on these reconstructions, whether the bottom right part of the ring is open.



(a) A continuous reconstruction (b) A binary reconstruction (c) Another binary reconstruction

Figure 1.3: *Reconstructions of a ring-like shaped object using three different reconstruction methods. Is the "ring" open?*

Even in cases when insufficient information is available to compute an accurate reconstruction of the complete image, it may still be possible to answer certain questions about the original image, or to determine certain features of it. Although finding a binary solution of the reconstruction problem is typically hard, it is often easier to prove that a solution with a specific feature *cannot* exist. For example, if the projections do not satisfy certain consistency conditions, a solution will certainly not exist.

When developing the error bounds for binary tomography, an existence condition for binary solutions of the reconstruction problem was found. In Chapter 8 we study the case where a pre-defined binary structure is enforced in the reconstruction problem and then a *consistency* condition for binary solutions is checked. By applying this methodology, it can be determined whether such a substructure can possibly occur, or whether it can certainly not occur in any binary image of the solution set.

1.5. Overview

As a conclusion of this introduction, we now provide a brief overview of the material contained in each of the next chapters.

In **Chapter 2**, we derive a series of upper bounds that can be used to guarantee the quality of a reconstructed binary image. The bounds limit the number of pixels that can be incorrect in the reconstructed image, in binary tomography, with respect to the original image. We provide several versions of these bounds, ranging from bounds on the difference between any two binary solutions of a tomography problem to bounds on the difference between

approximate solutions and the original object.

In **Chapter 3**, we consider the problem of reconstructing a high-resolution binary image from several low-resolution scans. Each of the pixels in a low-resolution scan yields the value of the sum of the pixels in a rectangular region of the high-resolution image. For any given set of such pixel sums, we derive an upper bound on the difference between a certain binary image which can be computed efficiently and *any* binary image that corresponds with the given measurements. We also derive a bound on the difference between any two binary images having these pixel sums.

In **Chapter 4** we expand the theory of error bounds for binary tomography of Chapter 2 to the case of noisy projection data. Despite the fact that the noiseless projection data is not available, we develop error bounds with respect to the solution set of the noiseless reconstruction problem.

In **Chapter 5** we show how the error bounds of Chapter 2 can be adapted to be useful for bounding the quality of experimental images. Our experimental results suggest that even though approximations have to be made due to noise and other errors in the data, the resulting bounds can still provide guidance on estimating the reconstruction quality in practice.

In **Chapter 6**, we present a series of computable bounds that can be used with any projection model. The approach developed in Chapter 2 is restricted to projection models where the corresponding matrix has constant column sums. We generalize these results and thereby broaden their applicability to include fan beam and cone beam projection models. In fact, the study presented here is not restricted to tomography and works for more general linear systems. We report the results of computational experiments for several phantom images, focused on parallel and fan beam projection models.

In **Chapter 7**, we develop a discrete approximate reconstruction algorithm. Our algorithm computes an image that has only grey values belonging to a given finite set. It also guarantees that the difference between the given projections and the projections of the reconstructed discrete image is bounded. The bound, which is computable, is independent of the image size.

In **Chapter 8**, we present a computational technique for discovering the possible presence of *features* (such as straight boundaries or homogeneous regions) in the unknown original image from its projections. We show that it is often possible to accurately identify the presence of certain features, even when insufficient information is available to compute an accurate reconstruction of the complete image.

Chapter 2

Bounds on the quality of reconstructed images in binary tomography

This chapter (with minor modifications) has been published as: K. J. Batenburg, W. Fortes, L. Hajdu, and R. Tijdeman. Bounds on the quality of reconstructed images in binary tomography. Discrete Applied Mathematics, Vol. 161(15), 2236–2251, 2013.

2.1. Introduction

Tomography is a technique for reconstructing an image of an object from a series of projections of this object, acquired from a range of viewing angles. The projection images are typically recorded using a scanning device, which can employ various types of beams (e.g., X-rays, neutrons, electrons) that traverse the object, after which a detector measures the result of the beam-object interaction. Provided that a large number of high-quality projections are available, sampled from a full range of angles, an accurate reconstruction of the object can be computed using a tomographic reconstruction algorithm [25, 33].

In practice, the set of angles for which projections are acquired is often limited. Due to dose constraints, it can be desirable to record as few projection images as possible, while still attaining sufficient image quality. Also, the angular range can be restricted by the particular scanning setup, such as in electron tomography, where the shape of the sample holder limits the angular range of the projections [36]. The resulting image reconstruction problems, based on just a small number of projections, are known as *limited data problems*.

For tomographic reconstruction from severely limited data, classical algorithms based on analytical inversion of the Radon transform, such as the Filtered Backprojection algorithm, often yield inferior reconstructions that are polluted by strong artefacts. In such cases, it makes sense to exploit available prior knowledge of the unknown object. Incorporation of this knowledge in the reconstruction algorithm can potentially result in a reduction

of the required number of projections, increased accuracy of the reconstruction, or an improved ability to deal with noisy projection data. A prior that has received much attention recently concerns the sparsity of the image, or of its gradient, which is exploited in the field of Compressed Sensing [14, 16, 41, 42].

A related, but more strict type of prior knowledge is exploited in *Discrete Tomography*, which focuses on the reconstruction of images that consist of a small, discrete set of grey values [27, 28]. The actual set of grey levels is typically assumed to be known in advance. Here, we focus specifically on the reconstruction of *binary* images, which consist of just two grey levels, 0 and 1. Several reconstruction algorithms have demonstrated the ability to reconstruct binary images from a very small number of projections, often even less than 6 [4, 6, 40].

Despite the strong constraint imposed on the grey values in discrete tomography, many valid solutions of the reconstruction problem can exist, all corresponding to the same set of projections. If the projections are obtained by performing measurements on some unknown ground truth object, the reconstruction can then deviate substantially from the true object. As a consequence, there is a need for an upper bound on the difference between binary solutions of the reconstruction problem. As the ground truth is a solution by itself, this would also yield a bound on the reconstruction error with respect to the ground truth.

A related problem in discrete tomography is the so-called *stability problem*, which deals with the question how the reconstruction changes if the projections are slightly perturbed. For images represented on a discrete grid, both lower and upper bounds have been obtained for the magnitude of such changes [2–4, 46]. For the case of binary image reconstruction from just two projections, horizontal and vertical, bounds on the difference between binary images having the same projections have been obtained by Van Dalen [44, 45].

In this chapter, we present a series of bounds which are highly general. Our bounds can be computed for any set of projections and in different geometrical settings, for lattice images as well as discretized continuous images. A key idea in deriving these bounds is an observation first made by Hajdu and Tijdeman in [22], concerning the fact that all binary solutions of the reconstruction problem must lie on a certain hypersphere, of which both the center and radius can be computed. The center of this hypersphere, which we call the *central reconstruction*, is the shortest real-valued solution of the tomography problem. This hypersphere construction leads directly to a simple bound on the distance between any two binary solutions, based on the triangle inequality. Stronger bounds can be derived by focusing on the distance between binary solutions of the tomography problem and the binary image that is obtained by *rounding* the central reconstruction. We derive several bounds that combine properties of the real-valued solution with combinatorial properties that are satisfied by the binary solutions. In particular, the fact that the sum of the pixel values in the unknown image is fully determined by the projection data can be used to improve the error bounds for binary images.

The structure of this chapter is as follows. Our notation is introduced in Section 2.2, after which a general class of reconstruction problems is introduced. We then discuss two specific examples of such problems, based on the grid model and strip model for computing the projections. In Section 2.3, the *central reconstruction* is introduced, which plays an important role in all results that follow. We discuss two different strategies for its computation. Section 2.4 contains the main results of this chapter: a series of quality bounds for binary solutions

of the tomography problem. It is divided in three parts: in Section 2.4.1, a general bound is derived on the difference between two binary images having a given set of projections. Section 2.4.2 deals with bounds that are based on properties of the binary images that are obtained by rounding the central reconstruction. These bounds are subsequently refined in Section 2.4.3 by including the knowledge of the total number of 1's in any binary solution, which can be determined from the projection data.

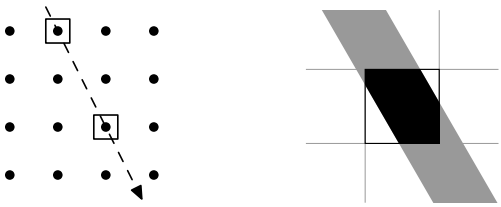
Section 2.5 presents a series of simulation experiments and their results. From these results, the practical value of the proposed bounds can be evaluated for different types of images. The results are further discussed in Section 2.6. Section 2.7 concludes the chapter.

2.2. Notation and model

Throughout the discrete tomography literature, several imaging models have been considered. In the *grid model*, an image is formed by assigning a value to each point in a regular grid. In the case of binary images, each point is assigned a value of either 0 or 1. Here, we consider square grids of the form $A = \{(i, j) \in \mathbb{Z}^2 : 1 \leq i, j \leq s\}$ for $s \in \mathbb{N}, s \geq 1$; see Fig. 2.1(a). For the grid model, we refer to the points in A as *pixels*. A binary image defined on A can be represented by a map $A \to \{0, 1\}$. A *projection* of an image f is formed by considering the set of parallel lines through one or more grid points in a certain direction $(a, b) \in \mathbb{Z}^2$, with $a \geq 0$ and (a, b) coprime, and summing the values of the points on each line. For a line given by the equation $ax - by = t$ ($t \in \mathbb{Z}$), the *line projection* p is defined as

$$p = \sum_{(x,y)\in A: ax-by=t} f(x, y).$$

The grid model can be used to model nanocrystals, that consist of discrete atoms positioned in a regular grid [1, 31].



(a) Grid model (b) Strip model

Figure 2.1: *Two different projection models.*

In many tomography applications, a *continuous* representation of the object is more realistic, as there is no intrinsic grid structure. In such cases, the unknown image is typically approximated by an image defined on a discrete pixel grid, using square pixels. A common model for computing the projections of such a pixelized image is the *strip model* [33, section 7.4.1], [50]. In the strip model, a projection is computed by considering a set of parallel

strips in a given direction and for each strip computing the weighted sum of all the pixels which intersect that strip with a weight equal to the intersection area of the strip and the pixel.

We now define some general notation. An *image* is represented by a vector $\boldsymbol{x} = (x_i) \in \mathbb{R}^n$. We refer to the entries of $\boldsymbol{x}$ as *pixels*, which correspond to unit squares in the strip model and to points in the grid model. The derivation of our main results does not depend on the particular projection model. Throughout this chapter we assume that all images are square, consisting of c rows and c columns, where $n = c^2$. A *binary image* corresponds with a vector $\bar{\boldsymbol{x}} \in \{0,1\}^n$.

For a given set of k projection directions, the *projection map* maps an image $\boldsymbol{x}$ to a vector $\boldsymbol{p} \in \mathbb{R}^m$ of *projection data*, where m denotes the total number of line measurements. As the projection map is a linear transformation, it can be represented by a matrix $\boldsymbol{W} = (w_{ij}) \in \mathbb{R}^{m \times n}$, called the *projection matrix*. Entry w_{ij} represents the weight of the contribution of x_j to projected line i. Note that for the grid model the projection matrix is a binary matrix, while for the strip model its entries are real values in $[0,1]$. The projection matrix $\boldsymbol{W}$ and vector $\boldsymbol{p}$ can be decomposed into k blocks as

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{W}^1 \\ \vdots \\ \boldsymbol{W}^k \end{pmatrix}, \quad \boldsymbol{p} = \begin{pmatrix} \boldsymbol{p}^1 \\ \vdots \\ \boldsymbol{p}^k \end{pmatrix}, \tag{2.1}$$

where each block $\boldsymbol{W}^d$ $(d = 1, \ldots, k)$ represents the projection map for a single direction and each block $\boldsymbol{p}^d$ represents the corresponding projection data.

From this point on, we assume that the projection matrix has the property that $\sum_{i=1}^{m} w_{ij} = k$ for all $j = 1, \ldots, n$. This property is certainly satisfied for the grid model, as every x_j is counted with weight 1 for exactly one line in each projection direction. The property is also satisfied for the strip projection model, as the total pixel weight for each projection angle is equal to the area of a pixel, which is 1. For most other projection models commonly used in tomography, such as the line model, where the weight of a pixel is determined by the length of its intersection with a line, this property is approximately satisfied, but not always exactly.

The *general reconstruction problem* consists of finding a solution of the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ for given projection data $\boldsymbol{p}$, i.e., to find an image that has the given projections. In *binary tomography*, one seeks a binary solution of the system. For a given projection matrix $\boldsymbol{W}$ and given projection data $\boldsymbol{p}$, let $S_{\boldsymbol{W}}(\boldsymbol{p}) = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}\}$, the set of all real-valued solutions corresponding with the projection data, and let $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p}) = S_{\boldsymbol{W}}(\boldsymbol{p}) \cap \{0,1\}^n$, the set of *binary solutions* of the system. As the main goal of incorporating prior knowledge of the binary grey levels in the reconstruction is to reduce the number of required projections, we focus on the case where m is small with respect to n, such that the real-valued reconstruction problem is severely underdetermined.

Despite the strong constraint that each pixel value x_i must belong to the set $\{0,1\}$, the binary reconstruction problem often does not have a unique solution. Instead of posing the uniqueness problem as a *yes/no* question, we focus in this chapter on the *number of different pixels* between different binary solutions. For example, if the reconstruction problem has no unique solution, but all pairs of solutions have at most 4 different pixels, then one can say

that *any* such solution must be close to the original object from which the projections have been obtained, even if the exact set of differences cannot be determined.

For any two vectors $\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}} \in \{0,1\}^n$, define the *difference set* $\mathrm{D}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \{i : \bar{u}_i \neq \bar{v}_i\}$ and the *number of differences* $\mathrm{d}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \#\mathrm{D}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}})$, where the symbol # denotes the cardinality operator for a finite set. Note that $\mathrm{d}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \|\bar{\boldsymbol{u}} - \bar{\boldsymbol{v}}\|_1$.

2.3. The central reconstruction

As the projection matrix is typically not a square matrix, and also does not have full rank, it does not have an inverse. Recall that the *Moore-Penrose pseudo inverse* of an $m \times n$ matrix $\boldsymbol{A}$ is an $n \times m$ matrix $\boldsymbol{A}^\dagger$, which can be uniquely characterized by the two geometric conditions

$$\boldsymbol{A}^\dagger \boldsymbol{b} \perp \mathcal{N}(\boldsymbol{A}) \quad \text{and} \quad (I - \boldsymbol{A}\boldsymbol{A}^\dagger)\boldsymbol{b} \perp \mathcal{R}(\boldsymbol{A}) \quad \text{for all } \boldsymbol{b} \in \mathbb{R}^m,$$

where $\mathcal{N}(\boldsymbol{A})$ is the nullspace of $\boldsymbol{A}$ and $\mathcal{R}(\boldsymbol{A})$ is the range of $\boldsymbol{A}$, [12, page 15].

Let $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$. Then $\boldsymbol{x}^*$ has the property (see Chapter 3 of [11]) that it is the real-valued solution of minimal Euclidean norm of the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$, provided that the latter system is solvable. We call $\boldsymbol{x}^*$ the *central reconstruction* of $\boldsymbol{p}$. The central reconstruction plays an important role in the bounds we derive for the binary reconstruction problem. We will show in the next section that all binary solutions of the system have equal distance to $\boldsymbol{x}^*$, so that one can consider the central reconstruction as lying "in the middle" of all binary solutions.

As all bounds presented in this chapter depend on $\boldsymbol{x}^*$, accurate computation of $\boldsymbol{x}^*$ is necessary to compute the corresponding difference bounds. One approach to computing the central reconstruction of a consistent system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ is to use the QR decomposition of $\boldsymbol{W}^T$. We will only sketch the computation here and refer to [5] for details. For clarity of presentation, we assume that $\boldsymbol{W}$ has full row rank. In fact, this assumption is not satisfied for tomography, and the *extended QR decomposition* should be used. The QR decomposition factorizes the matrix $\boldsymbol{W}^T$ into an orthogonal matrix $\boldsymbol{Q}$ and an uppertriangular matrix $\boldsymbol{R}$ of full column rank, as

$$\boldsymbol{W}^T = \boldsymbol{Q}\begin{pmatrix}\boldsymbol{R}\\ \boldsymbol{0}\end{pmatrix}.$$

The central reconstruction is then given by $\boldsymbol{x}^* = \boldsymbol{Q}(\boldsymbol{R}^T)^{-1}\boldsymbol{p}$, which can be computed efficiently by first solving the system $\boldsymbol{R}^T\boldsymbol{y} = \boldsymbol{p}$ for $\boldsymbol{y}$ by back substitution, and then computing $\boldsymbol{x}^* = \boldsymbol{Q}\boldsymbol{y}$.

However, due to the size of the matrix $\boldsymbol{W}$, calculation of the QR decomposition is usually unpractical for large images. For the case $m < n$, the QR decomposition requires $O(n^3)$ operations. Moreover, the $n \times n$ matrix $\boldsymbol{Q}$ is typically dense, requiring a vast amount of computer memory. As an alternative, an iterative method for solving the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$, called *CGLS* (Conjugate Gradient Least Squares), can be used [39]. The CGLS algorithm can effectively exploit the sparse structure of the projection matrix to reduce the required computation time, and does not require storage of large, dense matrices. Apart from numerical errors, applying CGLS to the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ results, after convergence, in the computation of $\boldsymbol{W}^\dagger\boldsymbol{p}$, while not computing the matrix $\boldsymbol{W}^\dagger$ explicitly (see also [47]). For all experiments in Section 2.5, the time required to compute the central reconstruction varied from a few seconds up to two minutes on a standard PC, depending on the number of projections and image size.

2.4. Quality bounds for binary solutions

In all the results in the following subsections, we consider a fixed system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ corresponding to a binary tomography problem, and refer to the central reconstruction of this system as $\boldsymbol{x}^*$.

As a substantial number of bounds will be given throughout this chapter, we introduce the following notation that will be further defined in the remainder of the chapter:

- The expressions $a(i)$ $(i = 1,2,3,4)$ will represent bounds on the number of pixel differences between any two binary solutions of the reconstruction problem.
- The expressions $b(i)$ $(i = 1,2,3,4)$ will represent bounds on the number of pixel difference between a certain given binary image (not necessarily a solution) and any binary solution of the reconstruction problem.
- The expressions $c(i)$ $(i = 1,2)$ will represent bounds on the number of pixel differences between the *rounded central reconstruction* $\bar{\boldsymbol{r}}$ and any binary solution.

The bounds *within* each class a, b, and c represent upper bounds for the same distance measure and can therefore be compared.

2.4.1. Elementary bounds based on the central reconstruction

In this subsection, a first set of bounds are derived. They follow from the fact that the Euclidean distance between the central reconstruction and any binary solution of the reconstruction problem can be determined from the projections. We start by noticing that the Euclidean norm of any binary solution of the tomography problem is determined by the projection data:

Lemma 1. *Let* $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. *Then,* $\|\bar{\boldsymbol{x}}\|_2^2 = \frac{\|\boldsymbol{p}\|_1}{k}$.

Proof. By the definition of the ℓ_1-norm, $\|\boldsymbol{p}\|_1 = \sum_{i=1}^m |p_i| = \sum_{i=1}^m p_i$, since $p_i \geq 0$ $(i = 1,\ldots,n)$. Also,

$$\sum_{i=1}^{m} p_i = \sum_{i=1}^{m} \left(\sum_{j=1}^{n} w_{ij} \bar{x}_j \right) = \sum_{j=1}^{n} \left(\sum_{i=1}^{m} w_{ij} \right) \bar{x}_j = \sum_{j=1}^{n} k \bar{x}_j, \tag{2.2}$$

and therefore $\|\boldsymbol{p}\|_1 = k \sum_{j=1}^n \bar{x}_j$.

As $\bar{\boldsymbol{x}} \in \{0,1\}^n$, we have $\|\bar{\boldsymbol{x}}\|_2^2 = \|\bar{\boldsymbol{x}}\|_1 = \sum_{j=1}^n \bar{x}_j = \frac{\|\boldsymbol{p}\|_1}{k}$.

□

The following lemma illustrates the importance of the central reconstruction, the shortest real-valued solution in $S_{\boldsymbol{W}}(\boldsymbol{p})$, by showing that the binary solutions are the shortest among all integer solutions of the system.

Lemma 2. *Let* $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ *and* $\boldsymbol{y} \in S_{\boldsymbol{W}}(\boldsymbol{p}) \cap \mathbb{Z}^n$. *Then* $\|\bar{\boldsymbol{x}}\|_2 \leq \|\boldsymbol{y}\|_2$, *with equality if and only if* $\boldsymbol{y} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$.

Proof. Note that the statement is proved in [22], see Problem 2 and the subsequent paragraph. However, for the convenience of the reader we give the proof here.

We have

$$\|\bar{\boldsymbol{x}}\|_2^2 = \sum_{i=1}^{n} \bar{x}_i^2 = \sum_{i=1}^{n} \bar{x}_i = \sum_{i=1}^{n} y_i = \frac{\|\boldsymbol{p}\|_1}{k}. \tag{2.3}$$

Observing that

$$\sum_{i=1}^{n} y_i \leq \sum_{i=1}^{n} y_i^2 = \|\boldsymbol{y}\|_2^2, \tag{2.4}$$

with equality if and only if $\boldsymbol{y}$ is binary, yields the result.

□

Lemma 3. *Let* $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. *Then* $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 = \sqrt{\frac{\|\boldsymbol{p}\|_1}{k} - \|\boldsymbol{x}^*\|_2^2}$.

Proof. From the definition of $\boldsymbol{x}^*$ we have $(\bar{\boldsymbol{x}} - \boldsymbol{x}^*) \in \mathcal{N}(\boldsymbol{W})$, and $\boldsymbol{x}^* \perp (\bar{\boldsymbol{x}} - \boldsymbol{x}^*)$. Applying Pythagoras' Theorem and Lemma 1 yields

$$\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 = \frac{\|\boldsymbol{p}\|_1}{k} - \|\boldsymbol{x}^*\|_2^2. \tag{2.5}$$

□

Define $R = \sqrt{\frac{\|\boldsymbol{p}\|_1}{k} - \|\boldsymbol{x}^*\|_2^2}$. We will use this constant throughout the remainder of this chapter, and refer to R as the *central radius*. According to Lemma 3, any binary solution of the reconstruction problem is on the hypersphere centered in $\boldsymbol{x}^*$ with radius R.

Supposing the existence of at least two different binary solutions, Lemma 3 allows us to derive an upper bound for the number of pixel differences between those solutions.

Theorem 1. *Let* $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ *and put* $a(1) = 4R^2$. *Then* $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \leq a(1)$.

Proof. According to Lemma 3, we have $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 = \|\bar{\boldsymbol{y}} - \boldsymbol{x}^*\|_2 = R$. Therefore,

$$\|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2 \leq \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 + \|\bar{\boldsymbol{y}} - \boldsymbol{x}^*\|_2 = 2R.$$

As $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ are binary, we have $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) = \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_1 = \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2^2$. □

Using the triangle inequality, a simple bound can also be given for the distance between *any* binary image and a solution of the reconstruction problem, as follows:

Corollary 1. *Let* $\bar{\boldsymbol{v}} \in \{0,1\}^n$ *be a given binary image and put* $b(1) = (R + \|\bar{\boldsymbol{v}} - \boldsymbol{x}^*\|_2)^2$. *Then for any binary image* $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, *we have* $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{v}}) \leq b(1)$.

2.4.2. Bounds based on rounding the central reconstruction

The fact that all elements of $\bar{S}_W(\boldsymbol{p})$ have equal distance to the central reconstruction $\boldsymbol{x}^*$, combined with the facts that binary solutions are the shortest solutions among all integer solutions (Lemma 2) and that $\boldsymbol{x}^*$ is the shortest real-valued solution, suggests that binary solutions can often be found near $\boldsymbol{x}^*$. It is therefore natural to consider the image that is obtained by rounding each entry of $\boldsymbol{x}^*$ to the nearest binary value. In this section, we will derive several bounds based on the number of differences between a binary solution of the reconstruction problem and a binary image obtained by rounding $\boldsymbol{x}^*$.

For $\alpha \in \mathbb{R}$, let $\mathrm{bin}(\alpha) = \min(|\alpha|, |1-\alpha|)$. Put $T = \sqrt{\sum_{i=1}^{n} \mathrm{bin}^2(x_i^*)}$, i.e., the Euclidean distance from $\boldsymbol{x}^*$ to the nearest binary vector. We will use this constant throughout this chapter and refer to T as the *central rounding distance*.

Corollary 2. *If $R < T$, then $\bar{S}_W(\boldsymbol{p}) = \emptyset$.*

If $R = T$, then all solutions in $\bar{S}_W(\boldsymbol{p})$ can be obtained by rounding the values in $\boldsymbol{x}^*$ to the nearest binary values, and variations are only possible for the entries i where $x_i^* = \frac{1}{2}$.

Let $\mathcal{T} = \{\bar{\boldsymbol{v}} \in \{0,1\}^n : \|\bar{\boldsymbol{v}} - \boldsymbol{x}^*\|_2 = T\}$ and let $\bar{\boldsymbol{r}} \in \mathcal{T}$, i.e., $\bar{\boldsymbol{r}}$ is among the binary vectors that are nearest to $\boldsymbol{x}^*$ in the Euclidean sense.

If $R > T$ and $R - T$ is small, it is possible to say that a fraction of the rounded values are correct, i.e., to provide an upper bound on the *number* of pixel differences between any solution in $\bar{S}_W(\boldsymbol{p})$ and $\bar{\boldsymbol{r}}$. In most cases we can not say *which* rounded values are correct.

Lemma 4. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and let $\bar{\boldsymbol{v}} \in \{0,1\}^n$ be any binary vector.*
Then $\|\bar{\boldsymbol{v}} - \boldsymbol{x}^*\|_2^2 = T^2 + \sum_{i \in \mathrm{D}(\bar{\boldsymbol{v}}, \bar{\boldsymbol{r}})} |2x_i^* - 1|$.

Proof. We have the following identities:

$$\begin{aligned}
\|\bar{\boldsymbol{v}} - \boldsymbol{x}^*\|_2^2 &= \|\bar{\boldsymbol{v}} - \bar{\boldsymbol{r}} + \bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 \\
&= \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 + 2\langle \bar{\boldsymbol{r}} - \boldsymbol{x}^*, \bar{\boldsymbol{v}} - \bar{\boldsymbol{r}} \rangle + \langle \bar{\boldsymbol{v}} - \bar{\boldsymbol{r}}, \bar{\boldsymbol{v}} - \bar{\boldsymbol{r}} \rangle \\
&= T^2 + 2\langle \bar{\boldsymbol{r}} - \bar{\boldsymbol{v}}, \boldsymbol{x}^* \rangle + \|\bar{\boldsymbol{v}}\|_2^2 - \|\bar{\boldsymbol{r}}\|_2^2 \\
&= T^2 + 2\sum_{i=1}^{n} (\bar{r}_i - \bar{v}_i) x_i^* + \sum_{i=1}^{n} \bar{v}_i - \sum_{i=1}^{n} \bar{r}_i \\
&= T^2 + \sum_{i=1}^{n} (\bar{r}_i - \bar{v}_i)(2x_i^* - 1) \\
&= T^2 + \sum_{i \in \mathrm{D}(\bar{\boldsymbol{v}}, \bar{\boldsymbol{r}})} |2x_i^* - 1|.
\end{aligned}$$

□

Lemma 4 can be interpreted as follows: consider the set of entries where $\bar{\boldsymbol{r}}$ and $\bar{\boldsymbol{v}}$ are different. If we transform $\bar{\boldsymbol{r}}$ into $\bar{\boldsymbol{v}}$ by performing a sequence of single-entry changes (either from 0 to 1, or from 1 to 0), each time an entry i of $\bar{\boldsymbol{r}}$ is changed the squared Euclidean distance from the current vector to $\boldsymbol{x}^*$ increases by $s_i = |2x_i^* - 1|$.

Let π be a permutation of $\{1, \ldots, n\}$ such that $s_{\pi(1)} \leq s_{\pi(2)} \leq \ldots \leq s_{\pi(n)}$, which can be obtained by sorting the entries s_i in increasing order.

Corollary 3. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and let $\bar{\boldsymbol{v}} \in \{0,1\}^n$ be any binary vector. Then*

$$\|\bar{\boldsymbol{v}} - \boldsymbol{x}^*\|_2^2 \geq T^2 + \sum_{i=1}^{\ell} s_{\pi(i)},$$

where $\ell = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{v}})$.

Proof. According to Lemma 4 we have

$$\|\bar{\boldsymbol{v}} - \boldsymbol{x}^*\|_2^2 = T^2 + \sum_{i \in D(\bar{\boldsymbol{r}}, \bar{\boldsymbol{v}})} s_i \geq T^2 + \sum_{i=1}^{\ell} s_{\pi(i)}.$$

□

As the Euclidean distance from $\boldsymbol{x}^*$ to any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ is R, a bound can now be derived on the maximal number of pixels in $\bar{\boldsymbol{r}}$ that must be changed to move from $\bar{\boldsymbol{r}}$ to $\bar{\boldsymbol{x}}$.

Theorem 2. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$, $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. Put*

$$c(1) = \max \left\{ 0 \leq \ell \leq n : \sum_{i=1}^{\ell} s_{\pi(i)} \leq R^2 - T^2 \right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}}) \leq c(1)$.

Proof. As $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, we have $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 = R^2$. Applying Lemma 4, we find that

$$R^2 - T^2 = \sum_{i \in D(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}})} s_i \geq \sum_{i=1}^{\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}})} s_{\pi(i)},$$

which implies that $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}}) \leq c(1)$. □

The proof of Theorem 2 can be interpreted as follows: consider the set of entries where $\bar{\boldsymbol{r}}$ and $\bar{\boldsymbol{x}}$ are different. If we transform $\bar{\boldsymbol{r}}$ into $\bar{\boldsymbol{x}}$ by performing a sequence of single-entry changes (either from 0 to 1, or from 1 to 0), each time an entry i of $\bar{\boldsymbol{r}}$ is changed the squared Euclidean distance from the current vector to $\boldsymbol{x}^*$ increases by $s_i = |2x_i^* - 1|$. As all binary solutions of the reconstruction problem are on a hypersphere centered in $\boldsymbol{x}^*$ with radius R, we know that once we have crossed the boundary of this hypersphere, a binary solution can no longer be obtained by changing the values of additional entries that have not yet been changed. An upper bound on the number of differences between $\bar{\boldsymbol{r}}$ and $\bar{\boldsymbol{x}}$ can be obtained by counting the number of steps required to cross the hypersphere, each time choosing a pixel which results in the minimal increase of the distance to $\boldsymbol{x}^*$. The following two Corollaries follow directly from Theorem 2:

Corollary 4. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$, $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ and let $a(2) = 2c(1)$ with $c(1)$ defined as in Theorem 2. Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \leq a(2)$.*

Corollary 5. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and let $\bar{\boldsymbol{v}} \in \{0,1\}^n$ be a given binary image and let $b(2) = c(1) + \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{v}})$ with $c(1)$ defined as in Theorem 2. Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{v}}) \leq b(2)$.*

In fact, the bound from Corollary 4 can be sharpened by noting that we can assume that the sets $\mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})$ and $\mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{y}})$ are disjoint, as entries that occur in both sets do not contribute to the number of differences between $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$. This observation leads to the following Theorem:

Theorem 3. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$, $\bar{\boldsymbol{x}},\bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. Put*

$$a(3) = \max\left\{0 \le \ell \le n : \sum_{i=1}^{\ell} s_{\pi(i)} \le 2(R^2 - T^2)\right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) \le a(3)$.

Proof. Define $\hat{\boldsymbol{x}}$ by $\hat{x}_i = \bar{r}_i$ if $\bar{x}_i = \bar{y}_i$, and $\hat{x}_i = \bar{x}_i$ otherwise. Define $\hat{\boldsymbol{y}}$ analogously. Then $\mathrm{d}(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}) = \mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$, $\|\hat{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 \le R^2$, and $\|\hat{\boldsymbol{y}} - \boldsymbol{y}^*\|_2^2 \le R^2$. Hence,

$$2R^2 \ge \|\hat{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 + \|\hat{\boldsymbol{y}} - \boldsymbol{y}^*\|_2^2 = 2T^2 + \sum_{i \in D(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}})} s_i + \sum_{i \in D(\bar{\boldsymbol{r}},\hat{\boldsymbol{y}})} s_i.$$

As $D(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}})$ and $D(\bar{\boldsymbol{r}},\hat{\boldsymbol{y}})$ are disjoint, we have $2R^2 - 2T^2 \ge \sum_{i=1}^{\mathrm{d}(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}})+\mathrm{d}(\bar{\boldsymbol{r}},\hat{\boldsymbol{y}})} s_{\pi(i)}$. This implies that $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) = \mathrm{d}(\hat{\boldsymbol{x}},\hat{\boldsymbol{y}}) \le \mathrm{d}(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}}) + \mathrm{d}(\bar{\boldsymbol{r}},\hat{\boldsymbol{y}}) \le a(3)$. □

A similar bound can be derived for the case where a particular binary image $\bar{\boldsymbol{v}}$, not necessarily a solution of the tomography problem, is given. For this, we transform $\bar{\boldsymbol{v}}$ into $\bar{\boldsymbol{r}}$ and then we perform a sequence of single-entry changes in $\bar{\boldsymbol{r}}$ with the exclusion of the pixels that differ between $\bar{\boldsymbol{v}}$ and $\bar{\boldsymbol{r}}$ because they have already been counted as wrong pixels of $\bar{\boldsymbol{v}}$.

Theorem 4. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and let $\bar{\boldsymbol{v}} \in \{0,1\}^n$ be a given binary image. Consider the sequence $(\phi(1),\phi(2),\ldots,\phi(\tilde{n}))$ defined by removing all numbers i for which $\bar{v}_i \ne \bar{r}_i$ from the sequence $(\pi(1),\ldots,\pi(n))$. Put*

$$U = \max\left\{0 \le \ell \le \tilde{n} : \sum_{i=1}^{\ell} s_{\phi(i)} \le R^2 - T^2\right\}$$

and let $b(3) = U + \mathrm{d}(\bar{\boldsymbol{r}},\bar{\boldsymbol{v}})$. Then for any binary image $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, we have $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{v}}) \le b(3)$.

Proof. Define $\hat{\boldsymbol{x}}$ by $\hat{x}_i = \bar{r}_i$ if $\bar{x}_i = \bar{v}_i$, and $\hat{x}_i = \bar{x}_i$ otherwise. Similarly, define $\hat{\boldsymbol{v}}$ by $\hat{v}_i = \bar{r}_i$ if $\bar{x}_i = \bar{v}_i$, and $\hat{v}_i = \bar{v}_i$ otherwise. Then $\mathrm{d}(\hat{\boldsymbol{x}},\hat{\boldsymbol{v}}) = \mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{v}})$, $\mathrm{d}(\hat{\boldsymbol{v}},\bar{\boldsymbol{r}}) \le \mathrm{d}(\bar{\boldsymbol{v}},\bar{\boldsymbol{r}})$, and $\|\hat{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 \le R^2$. Hence, $R^2 \ge \|\hat{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 = T^2 + \sum_{i \in D(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}})} s_i$. As $D(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}})$ and $D(\bar{\boldsymbol{r}},\bar{\boldsymbol{v}})$ are disjoint, we have $R^2 - T^2 \ge \sum_{i=1}^{\mathrm{d}(\bar{\boldsymbol{r}},\hat{\boldsymbol{x}})} s_{\phi(i)}$. This implies that $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{v}}) = \mathrm{d}(\hat{\boldsymbol{x}},\hat{\boldsymbol{v}}) \le \mathrm{d}(\hat{\boldsymbol{x}},\bar{\boldsymbol{r}}) + \mathrm{d}(\bar{\boldsymbol{r}},\hat{\boldsymbol{v}}) \le U + \mathrm{d}(\bar{\boldsymbol{r}},\bar{\boldsymbol{v}})$. □

2.4.3. Bounds involving the number of ones in binary solutions

The fact that the ℓ_1-norm $\|\bar{\boldsymbol{x}}\|_1$ of any binary solution is determined by the projection data, can possibly be exploited to tighten the bounds on the number of differences between any binary solution of the reconstruction problem and the rounded central reconstruction. As the number of ones in both $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ and $\bar{\boldsymbol{r}} \in \mathcal{T}$ can be computed, the number of elements in $\mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})$ for which $\bar{\boldsymbol{r}}$ is 0 determines the number of elements in this set for which $\bar{\boldsymbol{r}}$ is 1.

For any two vectors $\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}} \in \{0,1\}^n$, define the sets $\mathrm{D}_0(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \{i : \bar{u}_i = 0 \wedge \bar{v}_i = 1\}$ and $\mathrm{D}_1(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \{i : \bar{u}_i = 1 \wedge \bar{v}_i = 0\}$. We also define $\mathrm{d}_0(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \#\mathrm{D}_0(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}})$ and $\mathrm{d}_1(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \#\mathrm{D}_1(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}})$.

If $\|\bar{\boldsymbol{x}}\|_1 = \|\bar{\boldsymbol{r}}\|_1$, we have $\mathrm{d}_0(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) = \mathrm{d}_1(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})/2$. In other words, in order to transform $\bar{\boldsymbol{r}}$ into $\bar{\boldsymbol{x}}$, the number of pixels of $\bar{\boldsymbol{r}}$ assigned to 0 that must be changed to 1 is equal to the number of pixels of $\bar{\boldsymbol{r}}$ assigned to 1 that must be changed to 0. In general, when $\|\bar{\boldsymbol{x}}\|_1 = \|\bar{\boldsymbol{r}}\|_1$ is not necessarily true, let $t = \|\bar{\boldsymbol{x}}\|_1 - \|\bar{\boldsymbol{r}}\|_1$. Hence, $\mathrm{d}_0(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) = \mathrm{d}_1(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) + t$, which gives $\mathrm{d}_0(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) - t/2 = \mathrm{d}_1(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) + t/2 = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})/2$. Therefore, $\mathrm{d}_0(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})/2 + t/2$ and $\mathrm{d}_1(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})/2 - t/2$.

Theorem 5. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and $t = \|\bar{\boldsymbol{x}}\|_1 - \|\bar{\boldsymbol{r}}\|_1$. Construct the sequence $(\pi_0(1), \pi_0(2), \ldots, \pi_0(n_0))$ by removing all numbers i for which $\bar{r}_i = 1$ from the sequence $(\pi(1), \ldots, \pi(n))$. Similarly, construct the sequence $(\pi_1(1), \pi_1(2), \ldots, \pi_1(n_1))$ by removing all numbers i for which $\bar{r}_i = 0$ from the sequence $(\pi(1), \ldots, \pi(n))$. Let*

$$c(2) = \max \left\{ \ell : \ \ell + t \text{ is even and } \sum_{i=1}^{\min((\ell+t)/2, n_0)} s_{\pi_0(i)} + \sum_{i=1}^{\min((\ell-t)/2, n_1)} s_{\pi_1(i)} \le R^2 - T^2 \right\}.$$

Then for any binary image $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, we have $\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \le c(2)$.

Proof. Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. Put $\tilde{\ell} = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})$. Then

$$R^2 - T^2 = \sum_{i \in D(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})} s_i = \sum_{i \in D_0(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})} s_i + \sum_{i \in D_1(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})} s_i \ge \sum_{i=1}^{(\tilde{\ell}+t)/2} s_{\pi_0(i)} + \sum_{i=1}^{(\tilde{\ell}-t)/2} s_{\pi_1(i)}$$

with $(\tilde{\ell} + t)/2 \le n_0$ and $(\tilde{\ell} - t)/2 \le n_1$, which implies that $\tilde{\ell} = \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \le c(2)$. □

Corollary 6. *Let $c(2)$ be as defined in Theorem 5 and define $a(4) = 2c(2)$. Then for any pair of binary images $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, we have $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \le a(4)$.*

Corollary 7. *Let $\bar{\boldsymbol{v}} \in \{0,1\}^n$ be a given binary image and let $b(4) = c(2) + \mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{v}})$ with $c(2)$ as defined in Theorem 5. Then for any binary image $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, we have $\mathrm{d}(\bar{\boldsymbol{v}}, \bar{\boldsymbol{x}}) \le b(4)$.*

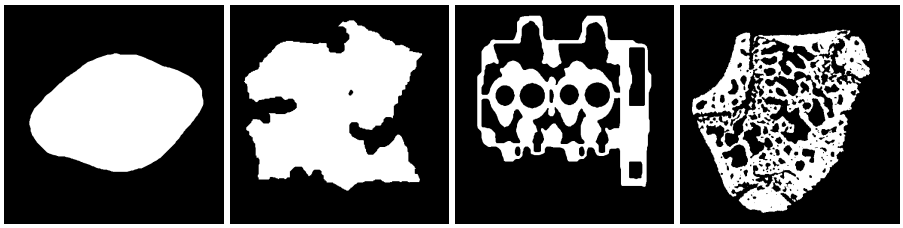
2.5. Experiments and results

A series of experiments have been performed to investigate the practical value of the bounds given in the several theorems and corollaries presented, for a range of images. The experiments are all based on simulated projection data obtained by computing the projections of the test images (so-called *phantoms*) in Fig. 2.2:

Phantom 1 represents a very simple, convex shaped object.

Phantom 2 represents an object with a more complex boundary. Also, the object is not convex and the boundary is fairly complex.

Phantom 3 represents a cross-section of a cylinder head in a combustion engine. It contains many holes and, as will become apparent from the results, it is more difficult to reconstruct accurately.



(a) Phantom 1 (b) Phantom 2 (c) Phantom 3 (d) Phantom 4

Figure 2.2: *Original phantom images used for the experiments.*

Phantom 4 was constructed from a micro-CT image of a rat bone, acquired with a SkyScan 1072 cone-beam micro-CT scanner.

All phantom images have a size of 512×512 pixels. To perform images with varying image size (smaller than 512×512), the phantoms have been downscaled to obtain binary images of the appropriate sizes.

In each experiment, the central reconstruction $\boldsymbol{x}^*$ was first computed using the CGLS algorithm. For some of the bounds, it is necessary to compute the rounded central reconstruction $\bar{\boldsymbol{r}}$ which was performed by rounding $\boldsymbol{x}^*$ to the nearest binary vector, choosing $\bar{r}_i = 1$ if $x_i^* = \frac{1}{2}$. Based on $\boldsymbol{x}^*$ and $\bar{\boldsymbol{r}}$, the various upper bounds described in Sections 2.4.1–2.4.3 were computed.

When presenting the results, we express the bounds on the pixel differences between two images as a *fraction* of the total number of image pixels. This allows for more straightforward interpretation of the results than using the absolute number of pixel differences. To aid in the identification of the bounds, Tables 2.1–2.3 provide a summary of all bounds and their respective theorems/corollaries.

$\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}})$	Result	Mathematical expression
$a(1)$	Theorem 1	$4R^2$
$a(2)$	Corollary 4	$2\max\left\{0 \le \ell \le n : \sum_{i=1}^{\ell} s_{\pi(i)} \le R^2 - T^2\right\}$
$a(3)$	Theorem 3	$\max\left\{0 \le \ell \le n : \sum_{i=1}^{\ell} s_{\pi(i)} \le 2(R^2 - T^2)\right\}$
$a(4)$	Corollary 6	$2\max\left\{\ell : \ \ell + s \text{ is even and } \sum_{i=1}^{(\ell+s)/2} s_{\pi_0(i)} + \sum_{i=1}^{(\ell-s)/2} s_{\pi_1(i)} \le R^2 - T^2\right\}$

Table 2.1: *List of symbolic expressions used for the bounds on the number of pixel differences between binary solutions, their respective theorems/corollaries and mathematical expressions.*

$\mathrm{d}(\bar{\boldsymbol{v}},\bar{\boldsymbol{x}})$	Result	Mathematical expression
$b(1)$	Corollary 1	$(R+\|\bar{\boldsymbol{v}}-\boldsymbol{x}^*\|_2)^2$
$b(2)$	Corollary 5	$\max\left\{0\le\ell\le n:\ \sum_{i=1}^{\ell} s_{\pi(i)}\le R^2-T^2\right\}+\mathrm{d}(\bar{\boldsymbol{r}},\bar{\boldsymbol{v}})$
$b(3)$	Theorem 4	$\max\left\{0\le\ell\le n:\ \sum_{i=1}^{\ell} s_{\phi(i)}\le R^2-T^2\right\}+\mathrm{d}(\bar{\boldsymbol{r}},\bar{\boldsymbol{v}})$
$b(4)$	Corollary 7	$\max\left\{\ell:\ \ell+s \text{ is even and } \sum_{i=1}^{(\ell+s)/2} s_{\pi_0(i)}+\sum_{i=1}^{(\ell-s)/2} s_{\pi_1(i)}\le R^2-T^2\right\}+\mathrm{d}(\bar{\boldsymbol{r}},\bar{\boldsymbol{v}})$

Table 2.2: *List of symbolic expressions used for the bounds on the number of pixel differences between a given $\bar{\boldsymbol{v}}\in\{0,1\}^n$ and any binary solution, their respective theorems/corollaries and mathematical expressions.*

$\mathrm{d}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})$	Result	Mathematical expression
$c(1)$	Theorem 2	$\max\left\{0\le\ell\le n:\ \sum_{i=1}^{\ell} s_{\pi(i)}\le R^2-T^2\right\}$
$c(2)$	Theorem 5	$\max\left\{\ell:\ \ell+s \text{ is even and } \sum_{i=1}^{(\ell+s)/2} s_{\pi_0(i)}+\sum_{i=1}^{(\ell-s)/2} s_{\pi_1(i)}\le R^2-T^2\right\}$

Table 2.3: *List of symbolic expressions used for the bounds on the number of pixel differences between $\bar{\boldsymbol{r}}$ and any binary solution, their respective theorems/corollaries and mathematical expressions.*

All graphs presented in the following subsections use a logarithmic scale for the error bounds. In some cases, the bound may become very small, or even 0, resulting in a point on the graph that cannot be plotted. These points are simply removed from the plot, causing the graph to be disconnected.

The remainder of this section is structured as follows. First, the concepts of central reconstruction, central radius, and rounded central reconstruction are illustrated for a concrete example in Section 2.5.1. Next, experimental results for the grid model and the strip model are presented in Sections 2.5.2 and 2.5.3, respectively. In Section 2.5.4, we consider a scenario where a binary reconstruction has been computed by a certain reconstruction algorithm, and we are interested in bounding the error with respect to the phantom for this particular reconstruction.

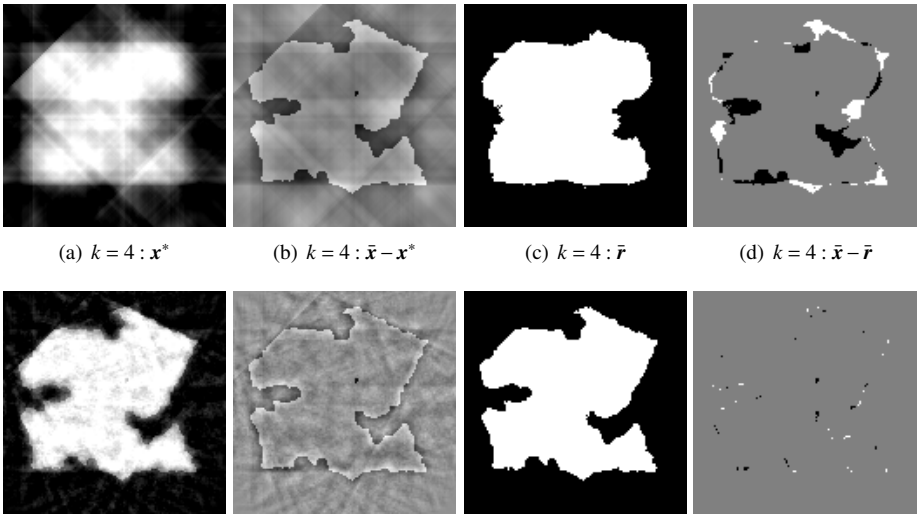
2.5.1. Concepts and interpretation

Fig. 2.3(a) shows the central reconstruction $\boldsymbol{x}^*$ of size 128×128, based on $k=4$ projections of the binary image $\bar{\boldsymbol{x}}$ corresponding to Phantom 2. The projection matrix $\boldsymbol{W}$ was formed

using the grid model for the directions $\{(0,1),(1,0),(1,1),(1,-1)\}$. The central reconstruction was computed using the CGLS algorithm, as explained in Section 2.3. The difference between the phantom image and the central reconstruction is shown in Fig. 2.3(b), where the grey levels are scaled between -1 (black) and 1 (white). Note that this difference vector $\bar{\boldsymbol{x}} - \boldsymbol{x}^*$ is in the nullspace of $\boldsymbol{W}$, as $\bar{\boldsymbol{x}}$ and $\boldsymbol{x}^*$ are both solutions of the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$. Therefore, the difference image can be considered as a *switching component* in the sense that it can be added to any other image without changing the projections in the four chosen directions.

The *central radius R* corresponds to the Euclidean norm of the difference vector $\bar{\boldsymbol{x}} - \boldsymbol{x}^*$, which is around 31.3152 in this case. Fig. 2.3(c) shows the rounded central reconstruction $\bar{\boldsymbol{r}}$, which is formed by rounding each pixel of $\boldsymbol{x}^*$ to the nearest binary value. The *central rounding distance T* corresponds with the Euclidean norm of the difference $\bar{\boldsymbol{r}} - \boldsymbol{x}^*$, which is around 25.5704 in this case. Note that $\bar{\boldsymbol{r}}$ is usually *not* a solution of the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$. The difference between the phantom $\bar{\boldsymbol{x}}$ and the image $\bar{\boldsymbol{r}}$ is shown in Fig. 2.3(d). This difference image is a three-valued image, with pixel values from the set $\{-1,0,1\}$. The bounds from Section 2.4.2 and onwards are based on bounding the norm of this difference image.

Similar images for the case of $k = 16$ projections are shown in Fig. 2.3e-h, based on the direction set $\{(1,0),(0,1),(1,1),(1,-1),(1,2),(1,-2),(2,1),(2,-1),(1,3),(1,-3),(2,3),$ $(2,-3),(3,1),(3,-1),(3,2),(3,-2)\}$. It can be clearly observed that, as the number of projections increases, the central reconstruction and the rounded central reconstruction both become better approximations of the phantom image. For the case of 16 projections, the difference between $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{r}}$ is already surprisingly small (approximately 0.45% of the total number of pixels), even though the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ is highly underdetermined.



(a) $k = 4 : \boldsymbol{x}^*$ (b) $k = 4 : \bar{\boldsymbol{x}} - \boldsymbol{x}^*$ (c) $k = 4 : \bar{\boldsymbol{r}}$ (d) $k = 4 : \bar{\boldsymbol{x}} - \bar{\boldsymbol{r}}$

(e) $k = 16 : \boldsymbol{x}^*$ (f) $k = 16 : \bar{\boldsymbol{x}} - \boldsymbol{x}^*$ (g) $k = 16 : \bar{\boldsymbol{r}}$ (h) $k = 16 : \bar{\boldsymbol{x}} - \bar{\boldsymbol{r}}$

Figure 2.3: *Illustration of various images that are related to the bounds.*

2.5.2. Error bounds for the grid model

In the grid model, a projection direction is represented by a pair of integers $(a,b) \in \mathbb{Z}^2$, such that $\gcd(a,b) = 1$ and $a \geq 0$. Let $\mathcal{A}$ be the set of all such pairs. For any positive integer c, put $\mathcal{A}_c := \{(a,b) \in \mathcal{A} : \max(a,|b|) = c\}$ and order the elements of $\mathcal{A}_c$, firstly by increasing value of a, secondly by increasing value of $|b|$, and thirdly by decreasing value of b. For example, $\mathcal{A}_3 = \{(1,3),(1,-3),(2,3),(2,-3),(3,1),(3,-1),(3,2),(3,-2)\}$. For any positive integer c, the ordered set $\mathcal{D}_c$ is formed by concatenating $\mathcal{A}_1,\ldots,\mathcal{A}_c$; for example, $\mathcal{D}_3 = \{(0,1),(1,0),(1,1),(1,-1),(1,2),(1,-2),(2,1),(2,-1),(1,3),(1,-3),(2,3),(2,-3),$ $(3,1),(3,-1),(3,2),(3,-2)\}$. To perform an experiment with k projection angles, the first k directions were selected from the set $\mathcal{D}_{20}$. This means that when the number of directions is increased, the old set of directions is always included in the new set of directions.

Experiments have been performed based on the three phantom images, scaled to sizes of 32×32, 128×128 and 512×512 respectively, varying the number of projection directions. The maximum number of projection directions for each image size is of 16, 64 and 200, respectively.

The first set of results are shown in Fig. 2.6, where bounds on the distance between any two binary solutions of the reconstruction problem, bounds on the number of differences between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$, and the exact error between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$ are jointly plotted. The bounds a and c were obtained by computing the minimum of all bounds in Tables 2.1 and 2.3 for each test case, i.e., $a = \min\{a(1),a(2),a(3),a(4)\}$ and $c = \min\{c(1),c(2),c(3),c(4)\}$. In Fig. 2.7, the individual bounds $a(1)$–$a(4)$ are shown for the same experiments.

2.5.3. Error bounds for the strip model

The experiments for the strip model have been performed using projection angles selected to coincide with the projection directions specified for the grid model. The projection angles were selected in this way to make the two models comparable. Projections have been computed based on sets of parallel strips, each strip having a width that equals the pixel size.

Experiments have been performed based on the four phantom images, scaled to sizes of 32×32, 128×128 and 512×512 respectively, varying the number of projection directions. The maximum number of projection directions for each image size is of 16, 64 and 200, respectively.

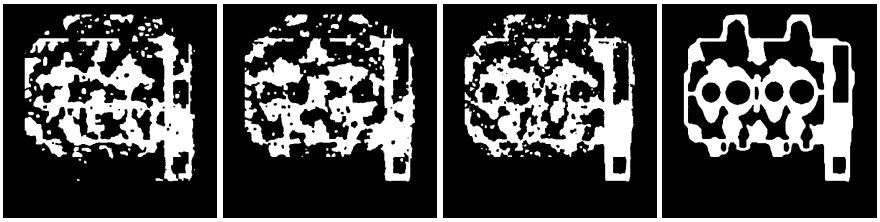
For the sake of compactness, we include the results for Phantoms 1 and 4. We did not observe strong deviations in the general behaviour of the bounds for the two other phantoms.

The first set of results are shown in Fig. 2.8, where bounds on the distance between any two binary solutions of the reconstruction problem, bounds on the number of differences between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$, and the exact error between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$ are jointly plotted. The bounds a and c were obtained by computing the minimum of all bounds in Tables 2.1 and 2.3 for each test case, i.e., $a = \min\{a(1),a(2),a(3),a(4)\}$ and $c = \min\{c(1),c(2),c(3),c(4)\}$. In Fig. 2.9, the individual bounds $a(1)$–$a(4)$ are shown for the same experiments.

2.5.4. Error bounds for a particular reconstruction

So far, the experiments were focused on bounding the difference between any two binary solutions, or the difference between $\bar{\boldsymbol{r}}$ and any binary solution. In practice, it can also be important to know bounds on the difference between any binary solution and a *particular* binary image, computed by a certain reconstruction algorithm. As the problem of computing a binary solution of the reconstruction problem is usually very hard, we will not assume that such a binary reconstruction is an exact solution to the reconstruction problem.

Several algorithms have been proposed in the literature for reconstructing binary images from their projections, see, e.g., [6, 9, 28, 40]. As an example, we focus here on the Discrete Algebraic Reconstruction Technique (DART), which has recently been proposed as a promising reconstruction algorithm for discrete tomography [7, 9]. The binary reconstruction computed by DART is not guaranteed to be an exact solution of the tomography problem.



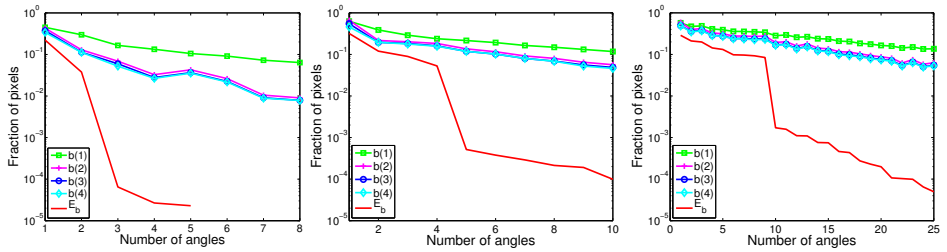
(a) $k = 4$ (b) $k = 6$ (c) $k = 8$ (d) $k = 10$

Figure 2.4: *DART reconstructions of Phantom 3 from 4, 6, 8 and 10 projections.*

DART reconstructions have been computed for Phantoms 1, 2 and 3, using the strip model with projection angles equally distributed between 0 and 180 degrees. As an illustration of the reconstruction results, Fig. 2.4 shows reconstructions of Phantom 3 for an increasing number of projections.

The bounds $b(1),\ldots,b(4)$ were computed, which bound the fraction of different pixels between the DART reconstruction and the original phantom. The results are shown in Fig. 2.5. The figure also contains graphs for the actual fraction of pixel differences E_b between the DART reconstruction $\bar{\boldsymbol{v}}$ and the phantom.

It can be observed that the true number of pixel differences between the DART reconstructions and the corresponding phantom images is much lower than the bounds. For the bounds $b(2)$, $b(3)$ and $b(4)$, this can be understood from the fact that the construction of the bounds involves using the triangle inequality to go from $\bar{\boldsymbol{v}}$ to $\bar{\boldsymbol{x}}$ through $\bar{\boldsymbol{r}}$. For a small number of projection images, the rounded central reconstruction $\bar{\boldsymbol{r}}$ is much further away from the phantom than the DART reconstruction. Still, meaningful guaranteed error estimates can be obtained from these bounds. For example, for Phantom 2, the graph shows that if just 10 projections are available, a guarantee can be given that no more than 10% of the pixels in the DART reconstruction are wrong with respect to the unknown phantom image.



(a) Phantom 1 (b) Phantom 2 (c) Phantom 3

Figure 2.5: *Error bounds for DART reconstruction of the three phantoms (size* 512×512*) as function the number of projection angles. Note that the graphs for* $b(2)$*,* $b(3)$*, and* $b(4)$ *have strong overlap and can hardly be distinguished.*

2.6. Discussion of the results

Despite the facts that the four phantoms have strong differences in shape and morphology, and that the grid and strip models are quite different, the results shown in Figs. 2.6–2.9 are surprisingly consistent throughout all experiments. Most of the bounds become smaller as the number of projection directions is increased but monotonicity is not a property of all the bounds presented in Section 2.4.

From the difference between the bounds presented in Section 2.4.1 and the bounds based on the rounded central reconstruction, we see that in most cases the phantom $\bar{\boldsymbol{x}}$ is substantially closer to $\bar{\boldsymbol{r}}$ than to $\boldsymbol{x}^*$.

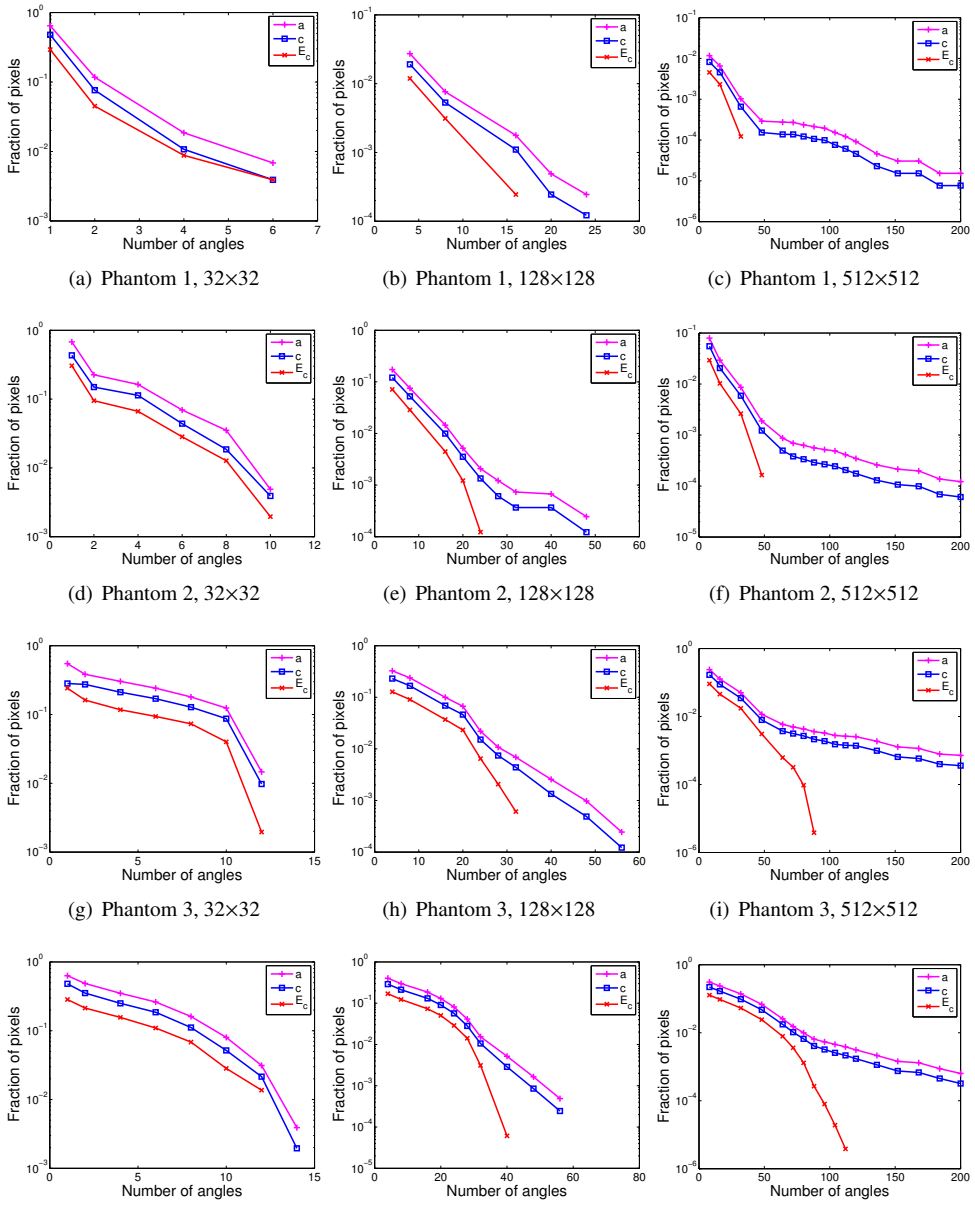
In Figs. 2.6 and 2.8, it can be observed that the true fraction of pixel differences between the phantom image $\bar{\boldsymbol{x}}$ and the rounded central reconstruction $\bar{\boldsymbol{r}}$, denoted by E_c, is often approximated quite well by the bound c, in particular for the grid model. This indicates that with respect to $\bar{\boldsymbol{r}}$, the bounds presented in this chapter can be quite sharp.

In Fig. 2.8(a), parts of the graphs for the bounds a and c, for more than 6 projections, are missing. In fact, in this case all of them are zero, such that they cannot be displayed in the logarithmic scale. This illustrates that our theorems for bounding the distance between any two binary solutions can be used to prove uniqueness of a binary solution, even when the corresponding real-valued system of equations is underdetermined.

In most of the experiments, the graphs for $a(2)$ and $a(4)$ are almost indistinguishable and the bound $a(3)$ is smaller than $a(4)$ but it can happen, as in Fig. 2.9(a), that this is not the case. In Fig. 2.9(a) the bound $a(4)$ proves uniqueness from 6 projection directions or more while the bound $a(3)$ does not for 12, 14 and 16 projections. Although $a(3)$ is the sharpest bound in most of the experiments, this is not true in general and all bounds should be used to obtain the best result. The bounds $a(2)$, $a(3)$ and $a(4)$ use similar principles and their computation can be performed simultaneously.

2.7. Outlook and conclusions

In this chapter, we have presented a range of general bounds on the accuracy of reconstructions in binary tomography, with respect to the unknown original object. The bounds can be computed within reasonable time and give guarantees on the number of pixels that can be different between any two binary solutions of the tomography problem, on the difference between an image obtained by rounding the central reconstruction and any binary solution, and on the difference between *any* binary image and any binary solution. The experimental results show that by using these bounds, one can prove that the number of differences between binary reconstructions must be very small, even when the corresponding real-valued system of equations is severely underdetermined. In order to make these bound practically useful, our results will have to be extended to deal with noisy projection data, which we will incorporate in future research.



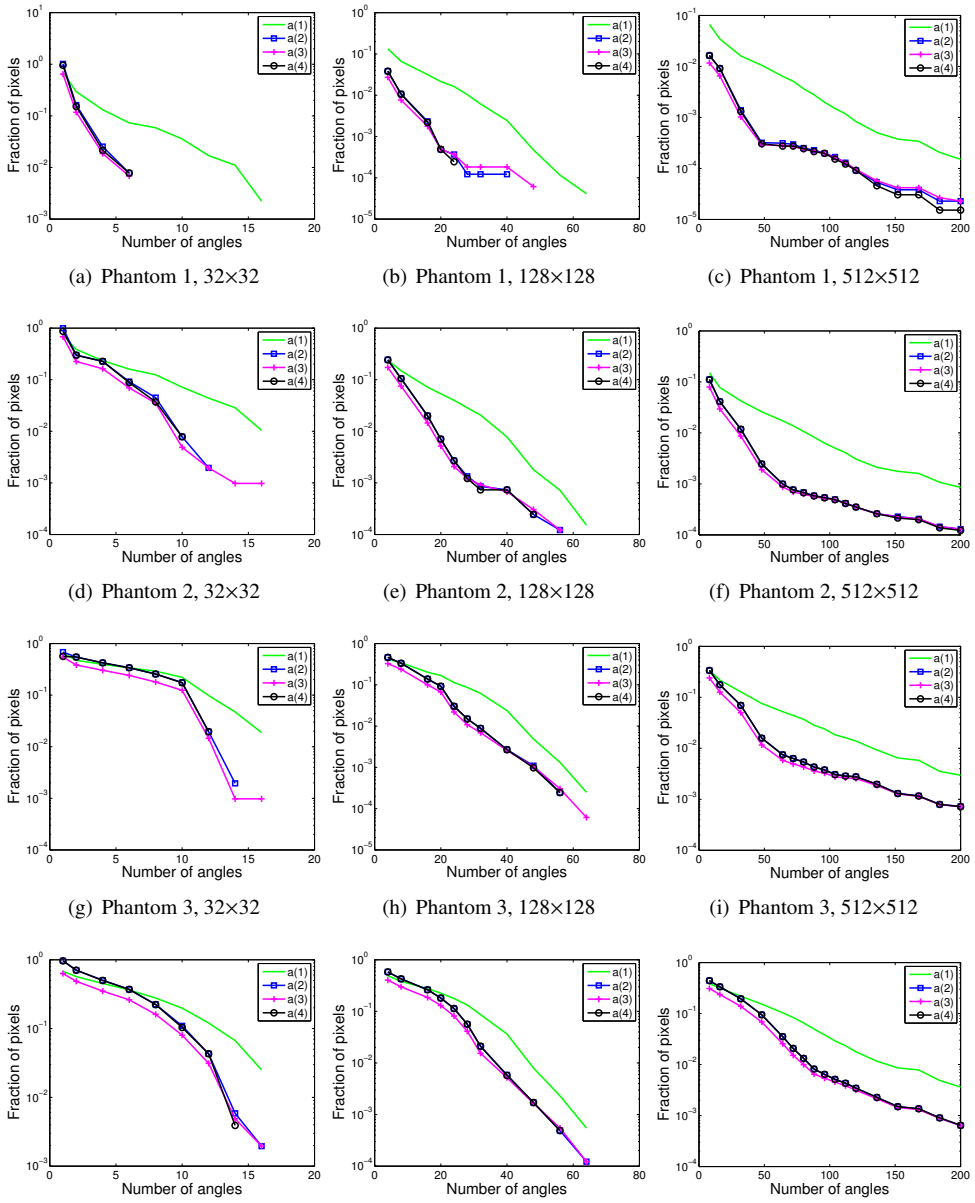
(a) Phantom 1, 32×32 (b) Phantom 1, 128×128 (c) Phantom 1, 512×512

(d) Phantom 2, 32×32 (e) Phantom 2, 128×128 (f) Phantom 2, 512×512

(g) Phantom 3, 32×32 (h) Phantom 3, 128×128 (i) Phantom 3, 512×512

(j) Phantom 4, 32×32 (k) Phantom 4, 128×128 (l) Phantom 4, 512×512

Figure 2.6: *Grid model: computed bounds as a function of the number of projection directions.*



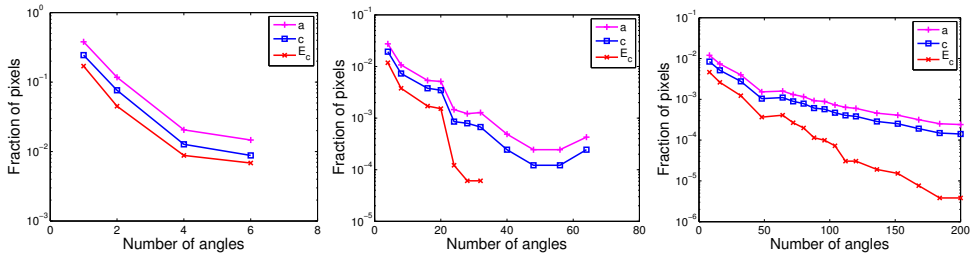
(a) Phantom 1, 32×32 (b) Phantom 1, 128×128 (c) Phantom 1, 512×512

(d) Phantom 2, 32×32 (e) Phantom 2, 128×128 (f) Phantom 2, 512×512

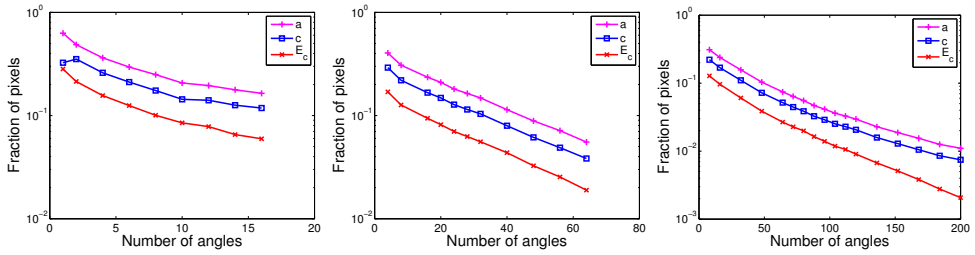
(g) Phantom 3, 32×32 (h) Phantom 3, 128×128 (i) Phantom 3, 512×512

(j) Phantom 4, 32×32 (k) Phantom 4, 128×128 (l) Phantom 4, 512×512

Figure 2.7: *Grid model: computed bounds as a function of the number of projection directions. Note that the graphs for $a(2)$ and $a(4)$ have strong overlap and can hardly be distinguished.*

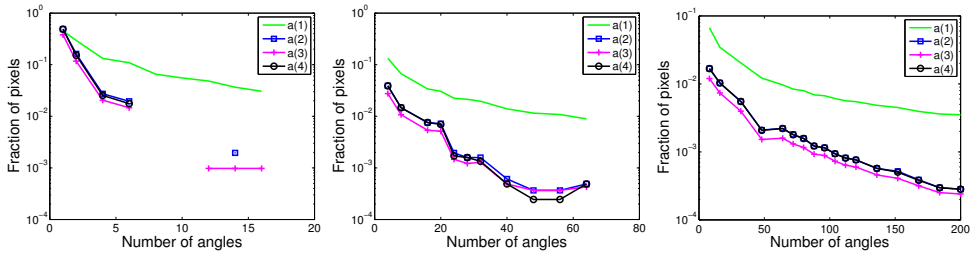


(a) Phantom 1, 32×32 (b) Phantom 1, 128×128 (c) Phantom 1, 512×512

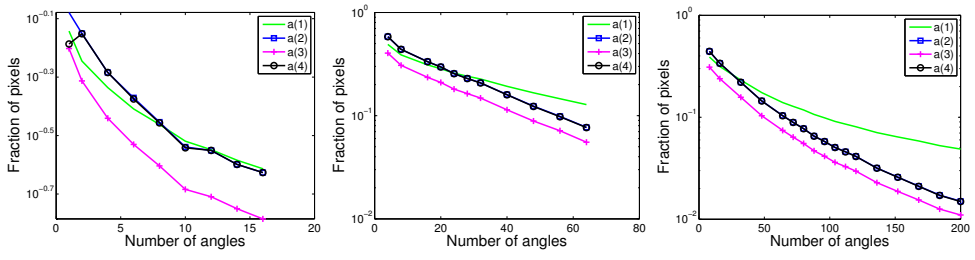


(d) Phantom 4, 32×32 (e) Phantom 4, 128×128 (f) Phantom 4, 512×512

Figure 2.8: *Strip model: computed bounds as a function of the number of projection directions for Phantoms 1 and 4.*



(a) Phantom 1, 32×32 (b) Phantom 1, 128×128 (c) Phantom 1, 512×512



(d) Phantom 4, 32×32 (e) Phantom 4, 128×128 (f) Phantom 4, 512×512

Figure 2.9: *Strip model: computed bounds as a function of the number of projection directions for Phantoms 1 and 4. Note that the graphs for $a(2)$ and $a(4)$ have strong overlap and can hardly be distinguished.*

Chapter 3

Error bounds on the reconstruction of binary images from low resolution scans

This chapter (with minor modifications) has been published as: W. Fortes and K. J. Batenburg. Error bounds on the reconstruction of binary images from low resolution scans. In Berciano, A., DÃŋaz-Pernil, D., Kropatsch, W., Molina-Abril, H., Real, P. (eds.), Proceedings of the Fourteenth International Conference on Computer Analysis of Images and Patterns (CAIP), Lecture Notes in Computer Science, Vol. 6855, 152–160. Heidelberg, 2011. Springer.

3.1. Introduction

Black-and-white images, also called *binary images*, occur in a wide range of imaging applications. In many such applications, the images are actually acquired as grey level images by a scanning device. When scanning text, for example, binary characters are often scanned by a grey level scanner. When taking pictures of numberplates using a low resolution digital camera, the structure of the binary characters may even be unrecognizable in the resulting grey level images. Another example can be found in the single-pixel camera, which has recently been proposed within the framework of compressive sensing. Instead of recording individual fine-resolution pixels, such a camera records the total intensity over various areas of the object being photographed [34, 49].

If several such grey level images are available, each representing a low resolution scan of some unknown "original" binary image, one can attempt to reconstruct the binary image by combining the information from multiple scans [8, 17, 18]. In particular, if the relative position of the different scans is well-known, this may lead to a high quality reconstruction. However, if the number of low resolution images available is relatively small in comparison with the resolution needed to properly represent the binary image, this reconstruction

problem can be highly underdetermined. In such cases, many binary images can exist that correspond with the same scanned grey level data. At present, no useful bounds are available that can guarantee that the reconstructed image is actually close to the unknown original image.

In Chapter 2, we presented bounds for binary image reconstruction in *tomography* (i.e. from projection data) that allow to bound the error between any two binary solutions, and therefore the error between the reconstructed binary image and the unknown original image. The proposed methodology is quite general and can potentially be extended to other imaging problems. As an intermediate step towards a general framework for bounding errors in binary image reconstruction, we apply the key concepts here to the problem of reconstructing binary images from low resolution scans.

3.2. Notation and concepts

Let $A \subset \mathbb{Z}^2$ be a finite set, called the *reconstruction area*. We consider the problem of reconstructing a binary image defined on A, represented by a function $F : A \to \{0,1\}$. A high resolution binary image defined on A will be reconstructed from several low resolution scans. The value of each pixel in such a scan corresponds with the summed intensity over all pixels in the corresponding region of the binary image. For simplicity, we assume here that the boundaries of the low resolution pixels coincide exactly with pixel boundaries in the high resolution binary image. We call a set $S \subset A$ a *window* of the reconstruction area. Let $\mathcal{S} = 2^A$, the set of all windows of A. We call a set $\mathbf{S} \subset \mathcal{S}$ of windows a *partition* of A if
(*i*) $S \cap T = \emptyset$ for all $S, T \in \mathbf{S}$ and
(*ii*) $\bigcup_{S \in \mathbf{S}} S = A$. We are also interested in the subsets of $\mathcal{S}$ which satisfy the property (*i*) but do not necessarily satisfy (*ii*). Such a subset will be called a *partial partition*. For $S \subset A$, define

$$P_F(S) = \sum_{(i,j) \in S} F(i,j). \tag{3.1}$$

We refer to the values $P_F(S)$ as *window-sums*. Note that our model for computing the window sums does not take certain properties of the imaging system, such as the detector point spread function, into account. However, the proposed methodology can easily be extended to include such effects, as long as they are linear. The *reconstruction problem* consists of finding an image F that has prescribed window-sum for a set $\mathbf{S}$ of windows. The existence and uniqueness of the solution of the general reconstruction problem is not guaranteed, in general.

To simplify the notation, the reconstruction problem can be formulated using linear algebra notation, which will be used in the forthcoming sections. Since there is an one-to-one mapping, say χ, from A to $\{1,\ldots,n\}$, the image F can be represented as a vector $\boldsymbol{x} = (x_j) \in \mathbb{R}^n$, where $n = \#A$ is the cardinality of A. We refer to the entries of $\boldsymbol{x}$ as *pixels*. A *binary image* on A corresponds with a vector $\bar{\boldsymbol{x}} \in \{0,1\}^n$.

For a given set $\mathbf{S} \subset \mathcal{S}$ and an image $\boldsymbol{x} \in \mathbb{R}^n$, the combined set of window sums results in a vector $\boldsymbol{p} = (p_i) \in \mathbb{R}^m$, where m represents the number of window-sums taken. As the operator $P_F(S)$ is linear, the mapping from an image to its window sums can be represented

by a matrix $\boldsymbol{W} = (w_{ij}) \in \mathbb{R}^{m\times n}$, which we call the *scan matrix*. The entry w_{ij} represents the weight of the contribution of x_j to the window-sum i.

Then, the general reconstruction problem can be stated as finding a solution of the system

$$\boldsymbol{Wx} = \boldsymbol{p}$$

for given window-sum data $\boldsymbol{p}$. In the binary image reconstruction problem, one seeks a binary solution of the system. For a given scan matrix $\boldsymbol{W}$ and a window-sum vector $\boldsymbol{p}$, let $\mathcal{T}_{\boldsymbol{W}}(\boldsymbol{p}) := \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{Wx} = \boldsymbol{p}\}$, the set of all real-valued solutions corresponding with the given data, and let $\bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p}) := \mathcal{T}_{\boldsymbol{W}}(\boldsymbol{p}) \cap \{0,1\}^n$, the set of *binary solutions* of the system.

As the scan matrix is typically not a square matrix, and also does not have full rank, it does not have an inverse. We recall that the *Moore-Penrose pseudo inverse* of an $m \times n$ matrix $\boldsymbol{A}$ is an $n \times m$ matrix $\boldsymbol{A}^\dagger$, which can be uniquely characterized by the two geometric conditions

$$\boldsymbol{A}^\dagger \boldsymbol{b} \perp \mathcal{N}(\boldsymbol{A}) \quad \text{and} \quad (I - \boldsymbol{A}\boldsymbol{A}^\dagger)\boldsymbol{b} \perp \mathcal{R}(\boldsymbol{A}) \quad \forall \boldsymbol{b} \in \mathbb{R}^m,$$

where $\mathcal{N}(\boldsymbol{A})$ is the nullspace of $\boldsymbol{A}$ and $\mathcal{R}(\boldsymbol{A})$ is the range of $\boldsymbol{A}$, [12, page 15].

Let $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$. Then $\boldsymbol{x}^*$ also has the property (see Chapter 3 of [11]) that it is the minimal Euclidean norm solution of the system $\boldsymbol{Wx} = \boldsymbol{p}$, if it exists. We call $\boldsymbol{x}^*$ the *central reconstruction* of $\boldsymbol{p}$. The central reconstruction plays an important role in the bounds we derive for the binary reconstruction problem.

The description of the general reconstruction problem given above is quite broad and we will now specify the scan model by which we define the scan matrix $\boldsymbol{W}$ and the window-sum vector $\boldsymbol{p}$, in order to model the problem of reconstructing high resolution images from low resolution scans.

Put $A = \{(i,j) \in \mathbb{Z}^2 |\ 0 \le i < l, \le j < h\}$. Let $1 \le p \le l$, and $1 \le q \le h$. For $0 \le i < l$, $0 \le j < h$, define a rectangular set of pixels of size $p \times q$ by

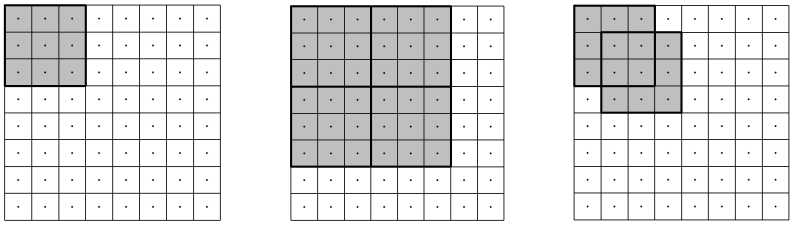
$$S_{i,j}^{p,q} = \{(i+c, j+r) |\ 0 \le c < p, 0 \le r < q\}.$$

Each pixel in a low resolution scan corresponds to a *window* in our framework. It provides information about the summed intensity in a rectangular set of pixels of the scanned high resolution image. Adjacent low resolution pixels are connected and do not overlap. For $0 \le a < p$ and $0 \le b < q$, define

$$\mathbf{S}^{a,b} = \{S_{a+ip,b+jq}^{p,q} |\ a + ip < l, b + jq < h\}. \tag{3.2}$$

Each set $\mathbf{S}^{a,b}$ is a partial partition. Its elements correspond to pixels of the low resolution image. Let us now assume that several such low resolution images are available. Then the total set $\boldsymbol{S}$ of window-sums consists of the union of partial partitions $\mathbf{S}^d := \mathbf{S}^{a_d,b_d}$ for $d \in \{1,\dots,k\}$. These concepts are illustrated in Fig. 2.1. Fig. 2.1a shows a single window $S_{a,b}$, whereas Fig. 2.1b shows the corresponding partial partition $\mathbf{S}^{a,b}$ formed by a tiling of its translates, where windows that cross the boundary of the image are not allowed. Fig. 2.1c shows two windows that are in separate partial partitions.

For $1 \le d \le k$, define the set of indices of the pixels x_j that were scanned by a partial partition $\mathbf{S}^d$ as $I_d := \{j | \chi^{-1}(j) \in \cup_{S \in \mathbf{S}^d} S\}$ and its complement $\bar{I}_d := \{1,\dots,n\} \backslash I_d$.



(a) Scan window.

(b) A partial partition formed by translates of a scan window.

(c) Scan windows for two different pairs (a, b).

Figure 3.1: *Rectangular scanning.*

As already mentioned, this linear scanning model can be modeled by a linear system of equations $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$. The matrix $\boldsymbol{W}$ and the window-sum $\boldsymbol{p}$ can be decomposed into k blocks as

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{W}^1 \\ \vdots \\ \boldsymbol{W}^k \end{pmatrix}, \quad \boldsymbol{p} = \begin{pmatrix} \boldsymbol{p}^1 \\ \vdots \\ \boldsymbol{p}^k \end{pmatrix}, \tag{3.3}$$

where each block $\boldsymbol{W}^d$ $(d = 1, \ldots, k)$ represents the scanning of the image with a rectangular window as defined by $\mathbf{S}^d$ and each block $\boldsymbol{p}^d$ represents the corresponding window-sums $P_F(S)$ for $S \in \mathbf{S}^d$.

3.3. Error bounds

Without loss of generality, we assume that all pixels in A are contained in at least one window. Clearly, no bounds can be given for those pixels that are not scanned at all, and they are removed from the analysis. As each set $\mathbf{S}^d$ samples a collection of disjoint subsets of A, the norm of the scanned binary image can be bounded from above by the available window-sums:

Proposition 1. *Let* $\bar{\boldsymbol{x}} \in \bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p})$. *Then,* $\|\bar{\boldsymbol{x}}\|_2^2 = \|\bar{\boldsymbol{x}}\|_1 \leq \|\boldsymbol{p}^d\|_1 + \#\bar{I}_d$ *for all* $1 \leq d \leq k$.

The norm of any binary solution can therefore be estimated by summation of the window-sums in $\boldsymbol{p}^d$ and its accuracy increases with the number of scanned pixels included in the partial partition $\boldsymbol{S}^d$.

In the next Theorem we will use Prop. 1 to show that all binary solutions of the linear system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ have bounded distance to the central reconstruction $\boldsymbol{x}^*$.

Lemma 5. *Let* $\bar{\boldsymbol{x}} \in \bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p})$ *and* $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$. *Put* $R := \min_{1 \leq d \leq k} R_d$, *where* $R_d := \sqrt{\|\boldsymbol{p}^d\|_1 + \#\bar{I}_d - \|\boldsymbol{x}^*\|_2^2}$. *Then,* $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 \leq R$.

Proof. From the definition of $\boldsymbol{x}^*$ we have $(\bar{\boldsymbol{x}} - \boldsymbol{x}^*) \in \mathcal{N}(\boldsymbol{W})$, and $\boldsymbol{x}^* \perp (\bar{\boldsymbol{x}} - \boldsymbol{x}^*)$. Combining Pythagoras' theorem and Prop. 1 yields the theorem. □

We will now consider the image that is obtained by rounding each entry of $\boldsymbol{x}^*$ to the nearest binary value. Let $\langle\alpha\rangle = \min(|\alpha|, |\alpha - 1|)$ for $\alpha \in \mathbb{R}$, and put $U = \sqrt{\sum_{i=1}^{n} \langle x_i^* \rangle^2}$, i.e., the Euclidean distance from $\boldsymbol{x}^*$ to the nearest binary vector.

Let $\bar{\boldsymbol{r}} \in \{0,1\}^n$ such that $\|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2 = U$, i.e., $\bar{\boldsymbol{r}}$ is among the binary vectors that are nearest to $\boldsymbol{x}^*$ in the Euclidean sense. If $R > U$ and $R - U$ is small, it is possible to say that a fraction of the rounded values are correct, i.e., to provide an upper bound on the *number* of pixel differences between any solution in $\bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p})$ and $\bar{\boldsymbol{r}}$.

In most cases we can not say *which* rounded values are correct. Suppose that $\bar{\boldsymbol{x}} \in \bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p})$ and that $\bar{r}_i = 1$ whereas $\bar{x}_i = 0$. Note that we have $x_i^* \geq \frac{1}{2}$. Put $\tilde{\boldsymbol{r}} := \bar{\boldsymbol{r}}$ and then set $\tilde{r}_i$ to 0. We then have $\|\tilde{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 = \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 - |x_i^* - 1|^2 + |x_i^*|^2 = \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 + 2x_i^* - 1$. Similarly, if $\bar{r}_i = 0$, then the squared Euclidean distance increases by $1 - 2x_i^*$ by setting pixel i to 1. Each time an entry i of $\bar{\boldsymbol{r}}$ is changed, the squared Euclidean distance to $\boldsymbol{x}^*$ increases by $b_i := |2x_i^* - 1|$.

As the Euclidean distance from $\boldsymbol{x}^*$ to $\bar{\boldsymbol{x}}$ is no greater than R, a bound can now be derived on the maximal number of pixels in $\bar{\boldsymbol{r}}$ that must be changed to move from $\bar{\boldsymbol{r}}$ to $\bar{\boldsymbol{x}}$.

Let us order the values b_i $(i = 1, \ldots, n)$ such that $b_i \leq b_{i+1}$ for $1 \leq i \leq n-1$. Assuming that $\bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p}) \neq \emptyset$, we have $R \geq \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2$ and the change of s entries of $\bar{\boldsymbol{r}}$ would increase the distance between $\bar{\boldsymbol{r}}$ and $\boldsymbol{x}^*$ such that $R^2 \geq \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 + \sum_{j=1}^{s} b_j$.

Theorem 6. *Let $\bar{\boldsymbol{r}}$, $\bar{\boldsymbol{x}}$ and b_i $(i = 1, \ldots, n)$ be as defined above. Choose s such that*

$$\sum_{i=1}^{s} b_i \leq R^2 - \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 < \sum_{j=1}^{s+1} b_j. \tag{3.4}$$

Then at most s pixels can have the wrong value in $\bar{\boldsymbol{r}}$ with respect to $\bar{\boldsymbol{x}}$ and at least $n - s$ pixels must have the correct value.

Proof. Due to the increasing order of the b_i's, changing more than s pixels in $\bar{\boldsymbol{r}}$ will result in a vector $\tilde{\boldsymbol{r}}$ for which $\|\tilde{\boldsymbol{r}} - \boldsymbol{x}^*\|_2 > R$, which cannot be an element of $\bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p})$. □

Theorem 6 bounds the number of pixel differences between $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{r}}$, and between $\bar{\boldsymbol{y}}$ and $\bar{\boldsymbol{r}}$. When using these two bounds to determine an upper bound on the number of differences between $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$, we can assume that these two sets of pixel differences are disjoint, as otherwise the difference between $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ will only be smaller. This observation leads to the following corollary:

Corollary 8. *Let $\bar{\boldsymbol{r}}$ and b_i $(i = 1, \ldots, n)$ be as defined above. Let $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{\mathcal{T}}_{\boldsymbol{W}}(\boldsymbol{p})$. Choose t such that*

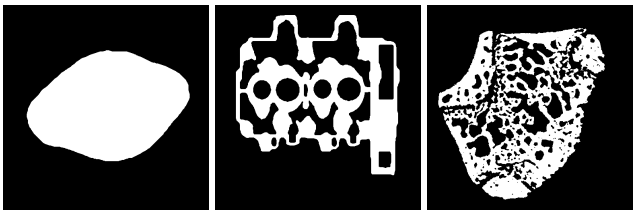
$$\sum_{i=1}^{t} b_i \leq 2(R^2 - \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2) < \sum_{j=1}^{t+1} b_j. \tag{3.5}$$

Then at most t pixels can be different between $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$.

3.4. Experiments and results

A series of experiments were performed to investigate the practical value of the bounds given in Theorem 6 and Corollary 8, for several test images. The experiments are all

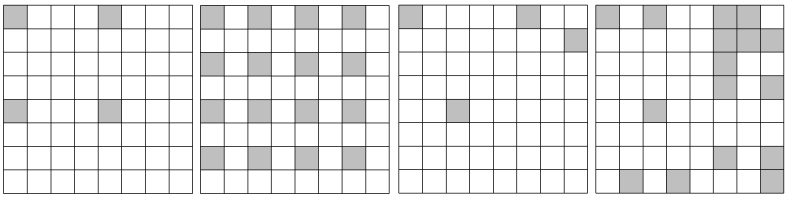
based on simulated data obtained by computing the scanning of the test images (so-called *phantoms*) in Fig. 3.2. All phantoms have a size of 512×512 pixels.



(a) Phantom 1 (b) Phantom 2 (c) Phantom 3

Figure 3.2: *Original phantom images used for the experiments.*

In each experiment, the central projection $\boldsymbol{x}^*$ was first computed using the CGLS algorithm [39]. The binary vector $\bar{\boldsymbol{r}}$ was computed by rounding $\boldsymbol{x}^*$ to the nearest binary vector (choosing $\bar{r}_i = 0$ if $x_i^* = \frac{1}{2}$). The upper bound s from Theorem 6 on the number of differences between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$ was then computed, followed by a bound on the fraction of pixel differences $U := \frac{s}{n}$, and the actual fraction of differences $E := \frac{e}{n}$, where e is the number of pixel differences between $\bar{\boldsymbol{r}}$ and $\bar{\boldsymbol{x}}$. The upper bound t from Corollary 8 on the number of differences any two binary solutions of $\boldsymbol{Wx} = \boldsymbol{p}$ was then computed, followed by the computation of the fraction of pixel differences $V := \frac{t}{n}$. Due to space limitations, we



(a) 4 points-regular (b) 16 points-regular (c) 4 points-random (d) 16 points-random

Figure 3.3: *Distribution of starting points in a first scan-window of size* 8×8.

only show the results for Phantom 3. The results for the other two phantoms are in line with the observations made for the third phantom. In all experiments, a square window was used. Note that the position of a partial partition $\mathbf{S}^{a,b}$ with respect to the high resolution image is completely determined by the pair (a, b), which we call a *starting point*. Each low resolution image of the high resolution binary image corresponds to a different starting point. In the experiments, we distinguish between regularly and randomly distributed starting points, where the regular case corresponds to a low resolution scanner that is gradually shifted across the high resolution image, and the random case corresponds to a device that moves irregularly (or an object that moves in such a way); see Fig. 3.3. In Fig. 3.4, the three error measures V, U and E are plotted as a function of the number of starting points for window size of 8×8

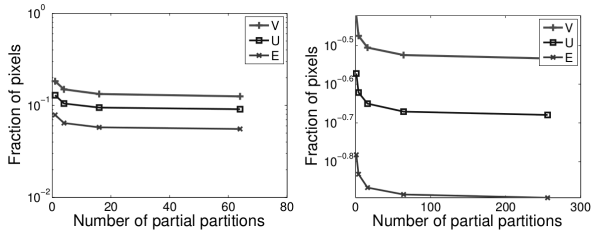
and 32×32 and for both regularly and randomly distributed starting points. Note that for a larger window size, more starting points are required to obtain similar error bounds.

Various observations can be made from the graphs in Fig. 3.4. Even if the number of starting points is much smaller than the number of pixels in the window, meaning that the reconstruction problem is severely underdetermined, it is still possible to guarantee that only a limited fraction of pixels can be different between binary solutions. Although the given bounds U are clearly not sharp when compared to the real error E, rounding the central reconstruction yields a binary image that is in many cases guaranteed to be rather close to the original image. For example, for window size 8×8 and randomly distributed starting points, having just 16 low resolution images available (resulting in a system of equations that is underdetermined by a factor of 4) can still guarantee that the rounded central reconstruction is within 10% of the original binary image.

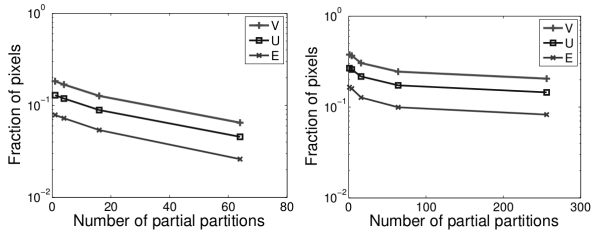
Fig. 3.5 illustrates the key concepts involved in the proposed bounds. The top row shows the central reconstruction for window sizes 8×8 and 32×32, with regularly and randomly distributed starting points. Here, the number of starting points is chosen as a fixed fraction of $\frac{1}{4}$ times the number of pixels in the window. In this way, all four reconstruction problems can be described by roughly the same number of equations. The middle row shows the difference images between the central reconstruction and the phantom, whereas the bottom row shows the difference images between the rounded central reconstruction and the phantom.

3.5. Outlook and conclusions

In this chapter, we have presented general bounds on the accuracy of reconstructions of binary images from several low resolution graylevel scans, with respect to the unknown original image. The bounds can be computed efficiently and give guarantees on the number of pixels that can be different between any two binary reconstructions that satisfy given window-sums, and on the difference between a particular binary image, obtained by rounding the central projection to the nearest binary vector, and any binary image satisfying the window-sums. The experimental results show that by using these bounds, one can prove that the number of differences between binary reconstructions must be small, even when the corresponding real-valued system of equations is severely underdetermined. This work represents an extension of the methodology set up in Chapter 2, which is a step towards a set of general bounds for binary image reconstruction problems that allow various forms of image sampling and incorporation of noisy measurements.



(a) window: 8×8, regular (b) window: 32×32, regular



(c) window: 8×8, random (d) window: 32×32, random

Figure 3.4: *computed bounds as a function of the number of partial partitions for Phantom 3; V: bound on the distance between any two binary solutions from Cor. 8; U: bound on the distance between any binary solution and the rounded central reconstruction* $\bar{\mathbf{r}}$ *from Thm. 6; E: true error between the rounded central reconstruction and the binary phantom* $\bar{\mathbf{x}}$.

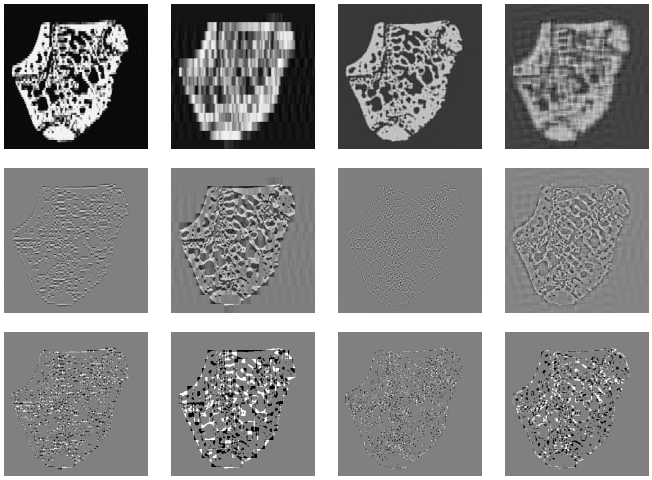


Figure 3.5: *Illustrations of the key concepts for Phantom* 3*;* **From left to right***:* 8×8 *window, regular;* 32×32 *window, regular;* 8×8 *window, random;* 32×32 *window, random;* **From top to bottom***: central reconstruction; difference between the central reconstruction and the phantom; difference between the rounded central reconstruction and the phantom.*

Chapter 4

Bounds on the difference between solutions in noisy binary tomography

4.1. Introduction

In Computed tomography one wants to find an image which satisfies a given set of projections. When this set of projections is obtained from a real scanning device, the projections are usually corrupted with noise and other artefacts. Due to this data perturbation, it is highly likely that there is no image satisfying the projection set. In any attempt to reconstruct an image which approximately satisfies the acquired measurements, the data perturbation is propagated into the reconstructed image which may not truly represent the original scanned object.

In Chapter 2 we studied error bounds for binary tomography with noiseless data, which are not applicable for noisy projection data. In this chapter, we aim to develop error bounds on the difference between binary solutions of the noiseless reconstruction problem even though the available projection set is perturbed. Also, following the bounds of Chapter 2, we bound the number of wrong pixels of a certain binary reconstruction with respect to the binary solutions of the noiseless problem.

A key idea of this chapter also relies on the fact that all binary solutions of the noiseless reconstruction problem must lie on a certain hypersphere. However, the center and radius of this hypersphere *cannot* be computed. Nevertheless, we compute the center and radius of a hypersphere which contains the set of binary solutions of the noiseless reconstruction problem. The center of this hypersphere is the least squares solution of the noisy tomography problem. Its radius is computed taking into account a bound on the distance between the centers of the two mentioned hyperspheres.

This chapter is structured as follows. Our notation is introduced in Section 4.2, after which a general class of reconstruction problems is introduced. In Section 4.3 we bound

the difference between any two binary solutions of the noiseless reconstruction problem by bounding the Euclidean norm of the binary solutions of the noisy problem and the minimum norm solution of the noiseless reconstruction problem. Section 4.4 deals with bounds that are based on properties of the binary images that are obtained by rounding the central reconstruction. Section 4.5 presents the computational challenges and Section 4.6 concludes the chapter.

4.2. Notation and model

Throughout the discrete tomography literature, several imaging models have been considered: the grid model, the strip model, the line model, etc. [33, section 7.4.1]. We now define some general notation. An *image* is represented by a vector $\boldsymbol{x} = (x_i) \in \mathbb{R}^n$. A *binary image* corresponds with a vector $\bar{\boldsymbol{x}} \in \{0,1\}^n$. For a given set of k projection directions, the *projection map* maps an image $\boldsymbol{x}$ to a vector $\boldsymbol{p} \in \mathbb{R}^m$ of *projection data*, where m denotes the total number of line measurements. As the projection map is a linear transformation, it can be represented by a matrix $\boldsymbol{W} = (w_{ij}) \in \mathbb{R}^{m \times n}$, called the *projection matrix*. Entry w_{ij} represents the weight of the contribution of x_j to projected line i. From this point on, we assume that the projection matrix has the property that $\sum_{i=1}^{m} w_{ij} = k$ for all $j = 1, \ldots, n$, i.e., constant column sums.

The *general reconstruction problem* consists of finding a solution of the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ for given projection data $\boldsymbol{p}$, i.e., to find an image that has the given projections. In *binary tomography*, one seeks a binary solution of the system. For a given projection matrix $\boldsymbol{W}$ and given projection data $\boldsymbol{p}$, let $S_{\boldsymbol{W}}(\boldsymbol{p}) = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}\}$, the set of all real-valued solutions corresponding with the projection data, and let $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p}) = S_{\boldsymbol{W}}(\boldsymbol{p}) \cap \{0,1\}^n$, the set of *binary solutions* of the system. As the main goal of incorporating prior knowledge of the binary grey levels in the reconstruction is to reduce the number of required projections, we focus on the case where m is small with respect to n, such that the real-valued reconstruction problem is severely underdetermined.

For this chapter, the given projection vector $\hat{\boldsymbol{q}}$ is perturbed by noise with $\hat{\boldsymbol{q}} = \boldsymbol{p} + \boldsymbol{\eta}$ where $\boldsymbol{p}$ is the unknown noiseless projection data and $\boldsymbol{\eta}$ is the unkown vector of noise. As the system $\boldsymbol{W}\boldsymbol{x} = \hat{\boldsymbol{q}}$ is typically inconsistent, one may try to solve the it approximately. Alternatively, it can be useful to consider a *consistent* linear system with a righthand side that is different, but close to $\boldsymbol{p}$. We therefore study the consistent linear system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{q}$, where $\boldsymbol{q}$ is the orthogonal projection of $\hat{\boldsymbol{q}}$ onto the range of $\boldsymbol{W}$. As $\boldsymbol{p}$ is not available, we consider the use of an estimated upper bound for the distance between $\boldsymbol{p}$ and $\boldsymbol{q}$ measured by some norm, i.e., $\delta \geq \|\boldsymbol{p} - \boldsymbol{q}\|$ is given for a defined norm.

4.3. A bound on the difference between all binary solutions

Lemma 3 from Chapter 2 allows us to compute the radius of the hypersphere which contains all binary solutions of the noiseless problem. The computation of the radius depends on $\|\boldsymbol{x}^*\|_2$ and $\|\bar{\boldsymbol{x}}\|_2$, the Euclidean norm of the minimum norm solution and the Euclidean norm of any binary solution of $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$, respectively. Recall that all binary solutions of $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$

have the same length, given by $\|\bar{\boldsymbol{x}}\|_2^2 = \frac{\|\boldsymbol{p}\|_1}{k}$ as proved in Lemma 1.

This section starts by showing that the Euclidean norm of any binary solution of $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ can be bounded even if $\boldsymbol{p}$ itself is not available, but only a perturbed version of $\boldsymbol{p}$.

Lemma 6. *Let $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^m$ and $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. Assume that $\delta_1 \geq \|\boldsymbol{p} - \boldsymbol{q}\|_1$ and define $U_{\bar{\boldsymbol{x}}} = \sqrt{\frac{\|\boldsymbol{q}\|_1 + \delta_1}{k}}$. Then $\|\bar{\boldsymbol{x}}\|_2 \leq U_{\bar{\boldsymbol{x}}}$.*

Proof. Write $\boldsymbol{p} = \boldsymbol{q} + \boldsymbol{p} - \boldsymbol{q}$ and use the triangle inequality to obtain

$$\|\boldsymbol{p}\|_1 \leq \|\boldsymbol{q}\|_1 + \|\boldsymbol{p} - \boldsymbol{q}\|_1, \tag{4.1}$$

and therefore

$$\|\boldsymbol{p}\|_1 \leq \|\boldsymbol{q}\|_1 + \delta_1. \tag{4.2}$$

Using Eq. (4.2) with $\|\bar{\boldsymbol{x}}\|_2^2 = \frac{\|\boldsymbol{p}\|_1}{k}$ and applying Lemma 1 of Chapter 2, we find that

$$\|\bar{\boldsymbol{x}}\|_2^2 \leq \frac{\|\boldsymbol{q}\|_1 + \delta_1}{k}, \tag{4.3}$$

as required. □

The next two lemmas are meant to develop a lower bound for $\|\boldsymbol{x}^*\|_2$.

Lemma 7. *Let $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^m$, $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$ and $\boldsymbol{z}^* = \boldsymbol{W}^\dagger \boldsymbol{q}$, where $\boldsymbol{W}^\dagger$ denotes the Moore-Penrose inverse of $\boldsymbol{W}$ [12].. Assume that $\|\boldsymbol{p} - \boldsymbol{q}\|_2 \leq \delta_2$. Then $\|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2 \leq \|\boldsymbol{W}^\dagger\|_2 \delta_2$.*

Proof. As the ℓ_2-norm is a subordinate norm, the following inequality property holds

$$\|\boldsymbol{W}^\dagger(\boldsymbol{p} - \boldsymbol{q})\|_2 \leq \|\boldsymbol{W}^\dagger\|_2 \|\boldsymbol{p} - \boldsymbol{q}\|_2 \leq \|\boldsymbol{W}^\dagger\|_2 \delta_2. \tag{4.4}$$

□

Lemma 8. *Let $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^m$, $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$ and $\boldsymbol{z}^* = \boldsymbol{W}^\dagger \boldsymbol{q}$. Assume $\|\boldsymbol{p} - \boldsymbol{q}\|_1 \leq \delta_1$ and $\|\boldsymbol{p} - \boldsymbol{q}\|_2 \leq \delta_2$. Define*

$$L_{\boldsymbol{x}^*} = \max\left\{\|\boldsymbol{z}^*\|_2 - \|\boldsymbol{W}^\dagger\|_2 \delta_2, 0\right\}.$$

Then $L_{\boldsymbol{x}^*} \leq \|\boldsymbol{x}^*\|_2$.

Proof. A reverse triangle inequality is used to give

$$\left|\|\boldsymbol{z}^*\|_2 - \|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2\right| \leq \|\boldsymbol{x}^*\|_2. \tag{4.5}$$

From Lemma 7 we have $\|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2 \leq \|\boldsymbol{W}^\dagger\|_2 \delta_2$ which in combination with Eq. (4.5) gives the bound $\|\boldsymbol{x}^*\|_2 \geq \|\boldsymbol{z}^*\|_2 - \|\boldsymbol{W}^\dagger\|_2 \delta_2$. Hence

$$\max\left\{\|\boldsymbol{z}^*\|_2 - \|\boldsymbol{W}^\dagger\|_2 \delta_2, 0\right\} \leq \|\boldsymbol{x}^*\|_2. \tag{4.6}$$

□

Supposing the existence of at least two different binary solutions of the reconstruction problem $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$, Lemmas 6 and 8 allow us to derive an upper bound for the Euclidean distance between those solutions.

Theorem 7. *Let $\boldsymbol{p},\boldsymbol{q} \in \mathbb{R}^m$, $\bar{\boldsymbol{x}},\bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ and $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$. Define $U_{\bar{\boldsymbol{x}}}$ as in Lemma 6, and $L_{\boldsymbol{x}^*}$ as in Lemma 8. Define $R_{\boldsymbol{p}}^2 = U_{\bar{\boldsymbol{x}}}^2 - L_{\boldsymbol{x}^*}^2$. Then $\|\bar{\boldsymbol{y}} - \bar{\boldsymbol{x}}\|_2 \leq 2R_{\boldsymbol{p}}$.*

Proof. From Theorem 3 of Chapter 2, we have $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 = \|\bar{\boldsymbol{x}}\|_2^2 - \|\boldsymbol{x}^*\|_2^2$. Using the bound $U_{\bar{\boldsymbol{x}}}$ for $\|\bar{\boldsymbol{x}}\|_2$ of Lemma 6 and the bound $L_{\boldsymbol{x}^*}$ for $\|\boldsymbol{x}^*\|_2$ of Lemma 8, both depending on $\boldsymbol{q}$, we have

$$\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 \leq R_{\boldsymbol{p}}^2. \tag{4.7}$$

Equation 4.7 says that any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ is *not outside* of the n-dimensional hypersphere with radius $R_{\boldsymbol{p}}$ and center $\boldsymbol{x}^*$. Therefore,

$$\|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2 \leq \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 + \|\bar{\boldsymbol{y}} - \boldsymbol{x}^*\|_2 = 2R_{\boldsymbol{p}}. \tag{4.8}$$

□

Corollary 9. *Let $\bar{\boldsymbol{x}},\bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ and define $R_{\boldsymbol{p}}$ as in Theorem 7. Then* $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) \leq 4R_{\boldsymbol{p}}^2$.

Proof. As $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ are binary, we have $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) = \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2^2 \leq 4R_{\boldsymbol{p}}^2$. □

4.4. Bounds based on rounding

The first result of this section shows that any binary solution of $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ is *not outside* of a hypersphere centered in $\boldsymbol{z}^*$, the minimal Euclidean norm solution of $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{q}$, of which the radius can be computed based on the given perturbed projection data.

Theorem 8. *Let $\boldsymbol{p},\boldsymbol{q} \in \mathbb{R}^m$, $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$ and $\boldsymbol{z}^* = \boldsymbol{W}^\dagger \boldsymbol{q}$. Suppose that $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. Put $R = \sqrt{\|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2^2 + \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2}$. Then $\|\bar{\boldsymbol{x}} - \boldsymbol{z}^*\|_2 = R$.*

Proof. As $\boldsymbol{W}\boldsymbol{x}^*, \boldsymbol{W}\boldsymbol{z}^* \in \mathcal{R}(\boldsymbol{W}) = \{\boldsymbol{W}\boldsymbol{x} : \boldsymbol{x} \in \mathbb{R}^n\}$, we also have $(\boldsymbol{W}\boldsymbol{x}^* - \boldsymbol{W}\boldsymbol{z}^*) \in \mathcal{R}(\boldsymbol{W})$. Therefore $\boldsymbol{x}^* - \boldsymbol{z}^* = \boldsymbol{W}^\dagger(\boldsymbol{p} - \boldsymbol{q})$ is the minimal Euclidean norm solution of $\boldsymbol{W}\boldsymbol{x} = (\boldsymbol{p} - \boldsymbol{q})$. Consequently, $(\boldsymbol{x}^* - \boldsymbol{z}^*) \perp (\bar{\boldsymbol{x}} - \boldsymbol{x}^*) \in \mathcal{N}(\boldsymbol{W})$.

Using Pythagoras' theorem we have

$$\|\bar{\boldsymbol{x}} - \boldsymbol{z}^*\|_2^2 = \|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2^2 + \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2. \tag{4.9}$$

□

Although we cannot compute the radius $R = \sqrt{\|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2^2 + \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2}$ from Theorem 8, we are going to bound it from above. In order to do that, we need to find a way to bound the term $\|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2$. Notice that an upper bound for the term $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2$ is given in the proof of Theorem 7.

Corollary 10. *Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$, $\boldsymbol{z}^* = \boldsymbol{W}^\dagger \boldsymbol{q}$, and $R_{\boldsymbol{p}}$ defined as in Theorem 7. Then $\|\bar{\boldsymbol{x}} - \boldsymbol{z}^*\|_2 \leq R_{\boldsymbol{q}}$, where $R_{\boldsymbol{q}} = \sqrt{\left[\min\left\{\|\boldsymbol{W}^\dagger\|_2\delta_2, \sqrt{\frac{\|\boldsymbol{q}\|_1+\delta_1}{k}} + \|\boldsymbol{z}^*\|_2\right\}\right]^2 + R_{\boldsymbol{p}}^2}$.*

Proof. In Theorem 8 we have shown that

$$\|\bar{\boldsymbol{x}} - \boldsymbol{z}^*\|_2^2 = \|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2^2 + \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2. \tag{4.10}$$

Using the upper bounds from Lemma 7 and the proof of Theorem 7,

$$\|\bar{\boldsymbol{x}} - \boldsymbol{z}^*\|_2^2 \le \left[\min\left\{\|\boldsymbol{W}^\dagger\|_2\delta_2,\, \sqrt{\frac{\|\boldsymbol{q}\|_1 + \delta_1}{k}} + \|\boldsymbol{z}^*\|_2\right\}\right]^2 + R_{\boldsymbol{p}}^2, \tag{4.11}$$

completing the proof. □

Let $\langle\alpha\rangle = \min(|\alpha|, |\alpha - 1|)$ for $\alpha \in \mathbb{R}$, and put $T_{\boldsymbol{q}} = \sqrt{\sum_{i=1}^n \langle z_i^*\rangle^2}$, i.e., the Euclidean distance from $\boldsymbol{z}^*$ to the nearest binary vector. Let $\mathcal{T} = \{\bar{\boldsymbol{v}} \in \{0,1\}^n : \ \|\bar{\boldsymbol{v}} - \boldsymbol{z}^*\|_2 = T_{\boldsymbol{q}}\}$ and let $\bar{\boldsymbol{r}} \in \mathcal{T}$, i.e., $\bar{\boldsymbol{r}}$ is among the binary vectors that are nearest to $\boldsymbol{z}^*$ in the Euclidean sense.

If $T_{\boldsymbol{q}} < R_{\boldsymbol{q}}$ and $R_{\boldsymbol{q}} - T_{\boldsymbol{q}}$ is small, it is possible to say that a fraction of the rounded values are correct, i.e., to provide an upper bound on the *number* of pixel differences between any solution in $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ and $\bar{\boldsymbol{r}}$. In most cases we can not say *which* rounded values are correct.

For any two vectors $\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}} \in \{0,1\}^n$, define the *difference set* $\mathrm{D}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \{i : \bar{u}_i \ne \bar{v}_i\}$ and the *number of differences* $\mathrm{d}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \#\mathrm{D}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}})$, where the symbol # denotes the cardinality operator for a finite set. Note that $\mathrm{d}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \|\bar{\boldsymbol{u}} - \bar{\boldsymbol{v}}\|_1$.

Lemma 9. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and let $\bar{\boldsymbol{v}} \in \{0,1\}^n$ be any binary vector.*
Then $\|\bar{\boldsymbol{v}} - \boldsymbol{z}^*\|_2^2 = T_{\boldsymbol{q}}^2 + \sum_{i \in \mathrm{D}(\bar{\boldsymbol{v}}, \bar{\boldsymbol{r}})} |2z_i^* - 1|$.

Proof. We have the following identities:

$$\begin{aligned}
\|\bar{\boldsymbol{v}} - \boldsymbol{z}^*\|_2^2 &= \|\bar{\boldsymbol{v}} - \bar{\boldsymbol{r}} + \bar{\boldsymbol{r}} - \boldsymbol{z}^*\|_2^2 \\
&= \|\bar{\boldsymbol{r}} - \boldsymbol{z}^*\|_2^2 + 2\langle \bar{\boldsymbol{r}} - \boldsymbol{z}^*, \bar{\boldsymbol{v}} - \bar{\boldsymbol{r}}\rangle + \langle \bar{\boldsymbol{v}} - \bar{\boldsymbol{r}}, \bar{\boldsymbol{v}} - \bar{\boldsymbol{r}}\rangle \\
&= T_{\boldsymbol{q}}^2 + 2\langle \bar{\boldsymbol{r}} - \bar{\boldsymbol{v}}, \boldsymbol{z}^*\rangle + \|\bar{\boldsymbol{v}}\|_2^2 - \|\bar{\boldsymbol{r}}\|_2^2 \\
&= T_{\boldsymbol{q}}^2 + 2\sum_{i=1}^n (\bar{r}_i - \bar{v}_i) z_i^* + \sum_{i=1}^n \bar{v}_i - \sum_{i=1}^n \bar{r}_i \\
&= T_{\boldsymbol{q}}^2 + \sum_{i=1}^n (\bar{r}_i - \bar{v}_i)(2z_i^* - 1) \\
&= T_{\boldsymbol{q}}^2 + \sum_{i \in \mathrm{D}(\bar{\boldsymbol{v}}, \bar{\boldsymbol{r}})} |2z_i^* - 1|.
\end{aligned}$$

□

Define $s_i = |2z_i^* - 1|$ and let π be a permutation of $\{1, \ldots, n\}$ such that $s_{\pi(1)} \le s_{\pi(2)} \le \ldots \le s_{\pi(n)}$, which can be obtained by sorting the entries s_i in increasing order.

Theorem 9. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$, $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. Put*

$$U = \max\left\{0 \le \ell \le n : \ \sum_{i=1}^{\ell} s_{\pi(i)} \le R_{\boldsymbol{q}}^2 - T_{\boldsymbol{q}}^2\right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}}) \le U$.

Proof. As $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, we have $\|\bar{\boldsymbol{x}} - \boldsymbol{z}^*\|_2^2 \leq R_{\boldsymbol{q}}^2$. Applying Lemma 9, we find that

$$R_{\boldsymbol{q}}^2 - T_{\boldsymbol{q}}^2 \geq \sum_{i \in D(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}})} s_i \geq \sum_{i=1}^{\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}})} s_{\pi(i)},$$

which implies that $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{r}}) \leq U$. □

Similarly to Theorem 9, another bound on the difference between any two binary solutions of $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{q}$ can be derived.

Theorem 10. *Let* $\bar{\boldsymbol{r}} \in \mathcal{T}$, $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$. *Put*

$$V = \max \left\{ 0 \leq \ell \leq n : \sum_{i=1}^{\ell} s_{\pi(i)} \leq 2(R_{\boldsymbol{q}}^2 - T_{\boldsymbol{q}}^2) \right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \leq V$.

Proof. Define $\hat{\boldsymbol{x}}$ by $\hat{x}_i = \bar{r}_i$ if $\bar{x}_i = \bar{y}_i$, and $\hat{x}_i = \bar{x}_i$ otherwise. Define $\hat{\boldsymbol{y}}$ analogously. Then $\mathrm{d}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}) = \mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}})$, $\|\hat{\boldsymbol{x}} - \boldsymbol{z}^*\|_2^2 \leq R_{\boldsymbol{q}}^2$, and $\|\hat{\boldsymbol{y}} - \boldsymbol{z}^*\|_2^2 \leq R_{\boldsymbol{q}}^2$. Hence,

$$2R_{\boldsymbol{q}}^2 \geq \|\hat{\boldsymbol{x}} - \boldsymbol{z}^*\|_2^2 + \|\hat{\boldsymbol{y}} - \boldsymbol{z}^*\|_2^2 = 2T_{\boldsymbol{q}}^2 + \sum_{i \in D(\bar{\boldsymbol{r}}, \hat{\boldsymbol{x}})} s_i + \sum_{i \in D(\bar{\boldsymbol{r}}, \hat{\boldsymbol{y}})} s_i.$$

As $D(\bar{\boldsymbol{r}}, \hat{\boldsymbol{x}})$ and $D(\bar{\boldsymbol{r}}, \hat{\boldsymbol{y}})$ are disjoint, we have $2R_{\boldsymbol{q}}^2 - 2T_{\boldsymbol{q}}^2 \geq \sum_{i=1}^{d(\bar{\boldsymbol{r}}, \hat{\boldsymbol{x}}) + d(\bar{\boldsymbol{r}}, \hat{\boldsymbol{y}})} s_{\pi(i)}$. This implies that $d(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) = d(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}) \leq d(\bar{\boldsymbol{r}}, \hat{\boldsymbol{x}}) + d(\bar{\boldsymbol{r}}, \hat{\boldsymbol{y}}) \leq V$. □

4.5. Computations

In several intermediate bounds developed in this chapter, and in particular in Corollary 10, the computation of the ℓ_2-norm of $\boldsymbol{W}^\dagger$, the Moore-Penrose pseudo-inverse of the projection matrix, is required. Computing the pseudo-inverse $\boldsymbol{W}^\dagger$ is expensive in terms of number of operations and memory requirements. A computer equipped with 8 GB memory ran out of memory when computing the matrix $\boldsymbol{W}^\dagger$ for images larger than 64×64. There is, however, a way to compute $\|\boldsymbol{W}^\dagger\|_2$ without computing $\boldsymbol{W}^\dagger$ explicitly.

A matrix norm based on a vector norm satisfies certain properties that are analogous to properties of this vector norm [30]. As the ℓ_2- matrix norm is unitary invariant, using the SVD decomposition of $\boldsymbol{W}^\dagger$ we have

$$\|\boldsymbol{W}^\dagger\|_2 = \|\boldsymbol{V}\boldsymbol{\Sigma}^\dagger\boldsymbol{U}^T\|_2 = \|\boldsymbol{\Sigma}^\dagger\|_2 = \sigma_{\max}(\boldsymbol{W}^\dagger),$$

where $\boldsymbol{V} \in \mathbb{R}^{n \times n}$ is unitary, $\boldsymbol{\Sigma}^\dagger \in \mathbb{R}^{n \times m}$ is a rectangular diagonal matrix with non-negative entries, $\boldsymbol{U}^T \in \mathbb{R}^{m \times m}$ is unitary, and $\sigma_{\max}(\boldsymbol{W}^\dagger)$ is the largest singular value of $\boldsymbol{W}^\dagger$. The matrix $\boldsymbol{\Sigma}^\dagger$ is the pseudo-inverse of $\boldsymbol{\Sigma}$, which is formed by replacing every non-zero diagonal entry by its reciprocal and transposing the resulting matrix. The diagonal entries of $\boldsymbol{\Sigma}$ are the singular values of $\boldsymbol{W}$. Therefore,

$$\|\boldsymbol{W}^\dagger\|_2 = \frac{1}{\sigma_{\min}(\boldsymbol{W})}.$$

There are computational techniques for computing the smallest singular value of a matrix without a complete SVD decomposition, such as the *power iteration* [48] (also known as *Von Mises iteration*) applied to the shifted matrix $\boldsymbol{W}\boldsymbol{W}^{\boldsymbol{T}} - \lambda_{\max}\boldsymbol{I}$ with $\lambda_{\max}$ being the largest eigenvalue of $\boldsymbol{W}\boldsymbol{W}^{\boldsymbol{T}}$. The singular values of $\boldsymbol{W}$ are the square roots of the non-negative eigenvalues of $\boldsymbol{W}\boldsymbol{W}^{\boldsymbol{T}}$. However, due to the ill-posed nature of the reconstruction problem, the smallest singular value can often not be distinguished from small values originated from numerical errors.

4.6. Conclusions

The nature of the reconstruction problem and the numerical errors in the computation make the computed smallest singular value of $\boldsymbol{W}$ unsuitable for effective use in the bounds presented in this chapter. Very small computed values for $\sigma_{\min}$, which can possibly be attributed to numerical errors, result in computed upper bounds that are too big to be useful in practice. In an attempt to overcome this problem, we will demonstrate in the next chapter how approximations can be made that avoid the computation of $\|\boldsymbol{W}^{\dagger}\|_2$, resulting in non-guaranteed bounds that can still be useful in practice.

Chapter 5

Practical error bounds for binary tomography

This chapter (with minor modifications) has been published as: W. Fortes, J. Sijbers and K. J. Batenburg. Practical error bounds for binary tomography. In the proceedings of the 1st International Conference of Tomography of Materials and Structures, 97–100, 2013.

5.1. Introduction

In Computed Tomography (CT), an image of a scanned object is formed by reconstructing an image of the object from a series of its projections. The image represents some physical property of the original object, usually the attenuation coefficient, which in turn is related to the compositions of the object [26]. The reconstructed image is typically not an exact representation of the original object, for various reasons. Firstly, the measured data itself is noisy and may contain various imaging artefacts, propagating into the reconstructed image. Secondly, the reconstruction algorithm itself may not be exact, such that even for noiseless projections it does not result in a reconstructed image that matches these projections. Finally, when using a relatively small number of projections, the reconstruction problem is inherently underdetermined. This means that many solutions may exist, each satisfying the projection data [35]. To draw quantitative conclusions about the scanned object based on the reconstructed image, it is essential to determine how well the reconstruction corresponds with the original object, which we refer to as the accuracy of the reconstruction. At present, there is a gap in the ability to determine reconstruction accuracy. Based on the point-spread-function of forward projection and consecutive reconstruction, resolution estimates can be derived for the reconstructed image [15]. However, such a local resolution measure does not impose any bound on the global difference between the reconstructed image and the ground truth. If a large number of projections are available, linear algebra arguments can be used to test if the pixelized reconstruction is uniquely determined by the measured data [37, p.27]. If only a small number of projections are available, experimental validation of reconstruction accuracy, using known phantoms, is the only option.

In Chapter 2, we have developed a mathematical approach that enables the computation of upper bounds on the reconstruction accuracy when the original object is homogeneous, i.e. corresponding to a binary image. These error bounds cannot be used directly on experimental images, as they are based on the assumption of perfect, noiseless projection data. In this chapter we show how one of these error bounds can be adapted to be useful for bounding the quality of experimental images.

5.2. Approach

The reconstruction problem in tomography can be approximated by a system of linear equations $\boldsymbol{W}\boldsymbol{v} = \boldsymbol{p}$ where $\boldsymbol{v} \in \mathbb{R}^n$ denotes a vector of unknown pixel values, $\boldsymbol{p} \in \mathbb{R}^m$ denotes a vector containing the measured projection data and $\boldsymbol{W} \in \mathbb{R}^{m \times n}$ denotes the discretized projection operator [33, Chapter 7].

In Chapter 2, it was demonstrated that for parallel beam tomography, the norm $\|\boldsymbol{x}\|_2$ of all *binary* solutions of the system $\boldsymbol{W}\boldsymbol{v} = \boldsymbol{p}$ must be the same, and it can be computed directly from the projection data. This can be seen from the fact that the sum of the projected intensities in any direction equals the sum of the image pixels. So, we can determine the sum of the pixel values based on the projection data. As $0^2 = 0$ and $1^2 = 1$, this sum is equal to the sum of *squared* pixel values for any binary image $\boldsymbol{x}$, and therefore the norm of all binary solutions is identical.

In fact, it was shown in Chapter 2 that all binary solutions $\boldsymbol{x} \in \{0,1\}^n$ lie on a hypersphere centered in the minimum norm solution $\boldsymbol{x}^*$ and having radius $R = \sqrt{\|\boldsymbol{x}\|_2^2 - \|\boldsymbol{x}^*\|_2^2}$; see Fig. 5.1. Based on these observations, a methodology was derived to compute an upper bound on the difference between any two binary solutions, as well as an upper bound on the difference between a given binary image and any binary solution, all based on the computation of $\boldsymbol{x}^*$ and R. The computation of the bounds depends on the hyperspherеâĂŹs center $\boldsymbol{x}^*$, corresponding to the shortest real-valued solution of the tomography equations. If the projection data contains no noise or other errors, this solution can be computed efficiently using iterative methods, such as the Conjugate Gradient Least Squares (CGLS) algorithm [39].

Now suppose that $\boldsymbol{b} \in \{0,1\}^n$ represents the ground truth and $\boldsymbol{p} = \boldsymbol{W}\boldsymbol{b}$ represents the ideal projections of this object. In practice, the measured projection data $\boldsymbol{q}$ is contaminated with noise and therefore $\boldsymbol{x}^*$ cannot be computed, as it requires knowledge of $\boldsymbol{p}$. Instead, we consider a different hypersphere containing all binary solutions of the noiseless problem. This hypersphere is centered in $\boldsymbol{z}^*$, the minimum norm least squares solution of the available reconstruction problem $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{q}$, and with radius S based on the theorem below:

Theorem 11. $S^2 = \|\boldsymbol{z}^*\|_2^2 - 2\|\boldsymbol{x}^*\|_2\|\boldsymbol{z}^*\|_2 \cos\theta + \|\boldsymbol{x}\|_2^2$.

Proof. Based on Theorem 8 we know that $\|\boldsymbol{x} - \boldsymbol{z}^*\|_2^2 = \|\boldsymbol{x}^* - \boldsymbol{z}^*\|_2^2 + \|\boldsymbol{x} - \boldsymbol{x}^*\|_2^2$. From Chapter 2 we have $\|\boldsymbol{x} - \boldsymbol{x}^*\|_2^2 = \|\boldsymbol{x}\|_2^2 - \|\boldsymbol{x}^*\|_2^2$. Furthermore, as the ℓ_2-norm is an inner product with underlying field $\mathbb{R}$, we have

$$\begin{aligned} S^2 &= \|\boldsymbol{x} - \boldsymbol{z}^*\|_2^2 \\ &= \|\boldsymbol{x}^*\|_2^2 - 2\|\boldsymbol{x}^*\|_2\|\boldsymbol{z}^*\|_2 \cos\theta + \|\boldsymbol{z}^*\|_2^2 + \|\boldsymbol{x}\|_2^2 - \|\boldsymbol{x}^*\|_2^2 \\ &= \|\boldsymbol{z}^*\|_2^2 - 2\|\boldsymbol{x}^*\|_2\|\boldsymbol{z}^*\|_2 \cos\theta + \|\boldsymbol{x}\|_2^2. \end{aligned}$$

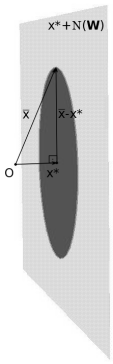


Figure 5.1: *All binary solutions $\boldsymbol{x}$ lie on a hypersphere centered in $\boldsymbol{x}^*$.*

□

The theorem expresses the value of S as a function of several terms that can be approximated based on the noisy projection data $\boldsymbol{q}$. The angle θ refers to the angle between the vectors $\boldsymbol{x}^*$ and $\boldsymbol{z}^*$. As we cannot compute $\|\boldsymbol{x}^*\|_2$, $\cos\theta$ and $\|\boldsymbol{x}\|_2$, we approximate these three terms. We assume that the noise has a mean value of 0, such that the total summed intensity of any binary solution can still be computed approximately from the projections as $\frac{\|\boldsymbol{q}\|_1}{k}$, where k denotes the number of projections. Moreover, we approximate the norm $\|\boldsymbol{x}^*\|_2$ by $\|\boldsymbol{z}^*\|_2$. Finally, $\cos\theta$ was experimentally computed for several different settings yielding a value close to 1, which depends on the noise level, image size, and the general shape of the object. The radius S is then approximately given by

$$S^2 \approx (\alpha - 1)\|\boldsymbol{z}^*\|_2^2 + \frac{\|\boldsymbol{q}\|_1}{k},$$

where α is a value close to 0 which depends on the specific problem setting and has to be calibrated based on simulation experiments using phantom images that have similar noise characteristics and similar general shape as the true object.

The computation of the bounds are the same as given in Chapter 2, but using the hypersphere centered in $\boldsymbol{z}^*$ and radius S as given above.

5.3. Experiments

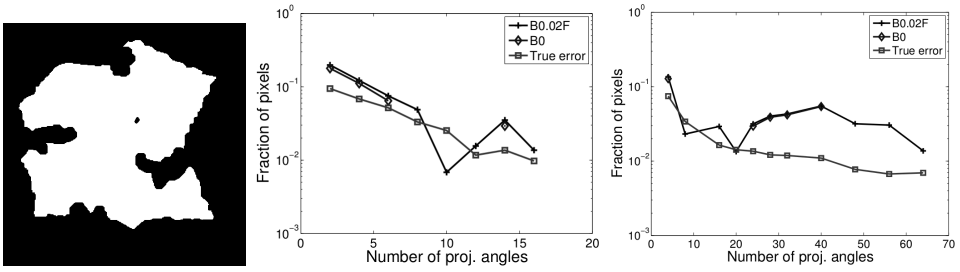
Simulation experiments have been performed to determine if the error bounds for noiseless data can be computed based on noisy projection data. Here we present the result of one such experiment.

One of the bounds in Chapter 2 concerns an upper bound on the number of pixel differences between any binary solution of the tomography problem $\boldsymbol{Wv} = \boldsymbol{p}$ and the *rounded shortest real-valued solution* $\boldsymbol{r}$. The image $\boldsymbol{r}$ can be computed from the projection data by applying the CGLS algorithm and rounding each entry of the result to the nearest binary

number. The advantage of this particular bound, is that it can be verified without knowledge of *all* binary solutions. Other bounds given in the same chapter deal with the differences between any *two* binary solutions, but these bounds cannot be verified based on a single phantom image. Therefore, we focus here on the bound with respect to $\boldsymbol{r}$.

Simulated projections were computed based on downsampled versions of the phantom in Fig. 5.2(a), using a strip model for the projection operator [33, Section 7.4.1] and equiangular projections. The phantom was downsampled to binary images of size 32×32 and 128×128, respectively, and all experiments were carried out at both these sizes. A moderate amount of additive Gaussian noise was then applied to the projections, yielding the vector $\boldsymbol{q}$, to be used for computing the bounds.

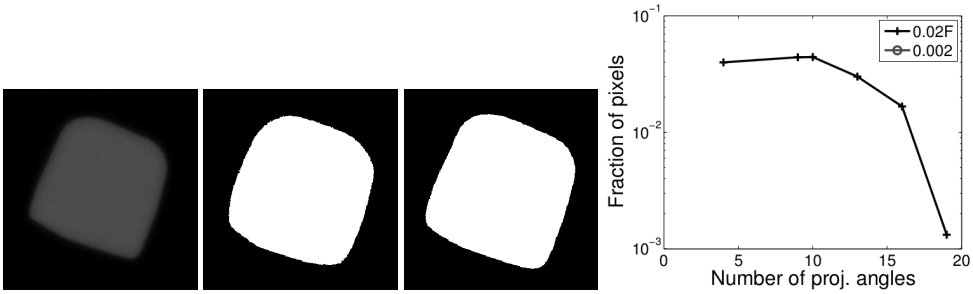
The vector $\boldsymbol{z}^*$ was computed using the CGLS algorithm. The parameter α was found to be dependent on image size, type of phantom, noise level and number of projections; it was set to two different values, 0 and 0.02, in our experiments. In Fig. 5.2(b) and 5.2(c), we show the relative number of pixels (as a fraction of the total number) that differ between the phantom image in Fig. 5.2(a) and the result of rounding $\boldsymbol{z}^*$ to the nearest binary image, which is marked by the label “true error" (red curve), as a function of the number of projection angles. The estimated error bound for the two values of α are labelled by $B0$ ($\alpha = 0$, blue curve) and $B0.02$ ($\alpha = 0.02$, black curve). The blue curve could not be plotted for most of the angles, as the resulting squared radius S^2 became negative. The black curve, however, can be computed for all angles and tracks the true error rather well: although the shape of the curve is somewhat irregular, the estimated error stays within an order of magnitude from the true error for all experiments.



(a) Binary phantom of size 512×512 (b) Computed error bounds for a downscaled phantom of size 32×32 (c) Computed error bounds for downscaled phantom of size 128×128

Figure 5.2: *Error bounds for simulated noise projection data.*

In Fig. 5.3, some results are shown for an experimental micro-CT dataset of a diamond, acquired using a Scanco micro-CT 40 X-ray scanner based on 500 projections. Fig. 5.3(a) shows a FBP-reconstructed slice based on all 500 projections, while Fig. 5.3(b) and 5.3(c) show binary reconstructions computed by applying CGLS and rounding the result based on 10 and 20 projections, respectively. The corresponding approximated error as a function of the number of projection angles is shown in Fig. 5.3(d), based on a value of $\alpha = 0.02$.



(a) Slice of a diamond from an experimental micro-CT dataset

(b) Binary reconstruction from 10 projections

(c) Binary reconstruction from 20 projections

(d) Computed error bound on the difference between the binary reconstruction and the true image, as function of the number of angles

Figure 5.3: *Reconstructions and error bounds for a slice of diamond*

5.4. Discussion and Conclusions

The approach provided here is the first technique for estimating a global image error in binary image reconstruction that can be applied to a set of noisy projections. The experimental results demonstrate that error estimates for binary tomography can be computed based on noisy, non-ideal projection data. The computed estimates have similar properties to the theoretical estimates that can only be computed if perfect, noiseless projection data is available. Further research is needed to establish how the factor α should be chosen in various scenarios, to further validate the approach, and to scale it up to larger images.

Chapter 6

Quality bounds for binary tomography with arbitrary projection matrices

This chapter (with minor modifications) has been submitted for publication as: K. J. Batenburg and W. Fortes. Quality bounds for binary tomography with arbitrary projection matrices. Discrete Applied Mathematics, Special Issue: DGCI 2013.

6.1. Introduction

Binary tomography deals with the problem of reconstructing a binary image from its projections [28]. Projection images of an object are typically recorded using a scanning device, which employs a beam that is transmitted through the object (e.g. photons, electrons). An array of detectors records the beam intensity after the beam-object interaction, resulting in a projection of the object. Due to dose constraints or geometrical constraints on the angles for which projections can be acquired, the set of angles for which projections are acquired is often limited [25, 33]. By exploiting the fact that the reconstructed image must be binary, it is often possible to compute useful reconstructions even if just a few projections are available [28]. However, such underdetermined binary tomography problems can have a large number of binary solutions, making it important to have a quality measure for the reconstruction with respect to the unknown original image.

Our results are related to the stability problem in discrete tomography, which has been studied by several authors [2–4, 44–46] for specific projection models or specific sets of projection directions.

In Chapter 2, more general results were obtained allowing the computation of error bounds between images in binary tomography for any number of projection angles but limited to projection models for which the associated *projection matrix* has constant column sums (i.e. identical sums for all columns). The property of constant column sums holds in

particular cases (e.g. the strip model for parallel beam tomography), but limits the application of the results to restricted cases. Although the bounds can still be approximated if this assumption is not completely satisfied, it is then no longer clear if they really provide a quality *guarantee*.

In this chapter, we derive error bounds that are more general than those in Chapter 2, as they can be applied to basically any problem modelled as an underdetermined algebraic linear system of equations. Although our focus is on tomography, our results are more general. We therefore consider the following general problem, of finding a binary vector that satisfies

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{6.1}$$

a consistent and underdetermined linear system of algebraic equations with $\boldsymbol{A} = (a_{ij}) \in \mathbb{R}^{m\times n}$, $m < n$, the vector of unknowns $\boldsymbol{x} = (x_j) \in \mathbb{R}^n$ and the right-hand side $\boldsymbol{b} = (b_i) \in \mathbb{R}^m$.

Finding a binary solution of Eq. (6.1) is often a very difficult problem and several binary solutions may exist. A given binary solution does not have to be close to another binary solution. In practice, the righthand side vector $\boldsymbol{b}$ is often obtained from an *original* binary vector $\boldsymbol{x}$ by a certain measurement procedure, modelled as the matrix $\boldsymbol{A}$. For a given measurement vector $\boldsymbol{b}$, it is unlikely that all binary solutions are representative solutions of the specific problem which yielded $\boldsymbol{b}$, since some solutions of Eq. (6.1) may be meaningless for physical problems. In such cases, it can be important to know how different these solutions can be. If one can give a bound on the maximum difference between two solutions, this also bounds the maximum difference between the *ground truth vector*, from which the vector of measured data $\boldsymbol{b}$ was obtained, and any other solution.

This chapter is structured as follows. In Section 6.2, we establish the notation which will be used throughout this chapter. In Section 6.3, different versions of bounds on the Euclidean norm of binary solutions are introduced. In Section 6.4, a general bound is derived on the difference between two binary solutions. Section 6.5 deals with bounds that are based on properties of the binary vectors that are obtained by rounding the minimum norm solution. These bounds are refined with a different approach in Section 6.6. Section 6.7 presents a series of simulation experiments for fan and parallel beam binary tomography and their results. From these results, the practical value of the proposed bounds can be evaluated for different kinds of problems. Section 6.8 concludes the chapter.

6.2. Notation and the minimum norm solution

For a given matrix $\boldsymbol{A}$ and given right-hand side $\boldsymbol{b}$, let $S_{\boldsymbol{A}}(\boldsymbol{b}) = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}\}$, the set of all real-valued solutions corresponding with the given data. A *binary vector* corresponds with a vector $\bar{\boldsymbol{x}} \in \{0,1\}^n$. Let $\bar{S}_{\boldsymbol{A}}(\boldsymbol{b}) = S_{\boldsymbol{A}}(\boldsymbol{b}) \cap \{0,1\}^n$, the set of *binary solutions* of the system.

Throughout this chapter, we use the vector $\boldsymbol{0}_t \in \mathbb{R}^t$ (for an integer $t > 0$), to denote a column vector consisting of t 0's, the vector $\boldsymbol{1}_t \in \mathbb{R}^t$ to denote a column vector consisting of t 1's and the identity matrix $\boldsymbol{I}_t \in \mathbb{R}^{t\times t}$. However, we often use the vectors $\boldsymbol{0}$ and $\boldsymbol{1}$ and the identity matrix $\boldsymbol{I}$ without specifying their dimension, as it does not compromise the understanding and clarity of the proofs.

For any two vectors $\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}} \in \{0,1\}^n$, define the *difference set* $\mathrm{D}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \{i : \bar{u}_i \neq \bar{v}_i\}$ and the *number of differences* $\mathrm{d}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \#\mathrm{D}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}})$, where the symbol # denotes the cardinality operator

for a finite set. Note that $\mathrm{d}(\bar{\boldsymbol{u}}, \bar{\boldsymbol{v}}) = \|\bar{\boldsymbol{u}} - \bar{\boldsymbol{v}}\|_1$.

For the following sections, consider the problem of finding a binary solution of a fixed linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ called the *binary solution problem*.

As the matrix $\boldsymbol{A}$ is not a square matrix, and may not have full rank, it does not have an inverse. Recall that the *Moore-Penrose pseudo inverse* of an $m \times n$ matrix $\boldsymbol{A}$ is an $n \times m$ matrix $\boldsymbol{A}^\dagger$, which can be uniquely characterized by the two geometric conditions

$$\boldsymbol{A}^\dagger \boldsymbol{b} \perp \mathcal{N}(\boldsymbol{A}) \quad \text{and} \quad (\boldsymbol{I} - \boldsymbol{A}\boldsymbol{A}^\dagger)\boldsymbol{b} \perp C(\boldsymbol{A}), \quad \forall \boldsymbol{b} \in \mathbb{R}^m,$$

where $\mathcal{N}(\boldsymbol{A})$ is the nullspace of $\boldsymbol{A}$ and $C(\boldsymbol{A})$ is the *column space* of $\boldsymbol{A}$ [12, page 15].

Let $\boldsymbol{x}^* = \boldsymbol{A}^\dagger \boldsymbol{b}$. Then $\boldsymbol{x}^*$ also has the property (see Chapter 3 of [11]) that it is a real-valued solution of minimal Euclidean norm of the system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, provided that such a solution exists. The minimum norm solution plays an important role in the bounds we will derive for the binary solution problem. Several methods are available for the computation of the minimum norm solutions. One approach is computing the QR decomposition of $\boldsymbol{A}^T$, see [5] for details. However, in several cases an iterative method, such as CGLS [39], is more suitable.

6.3. The Euclidean norm of binary solutions

The bounds on the difference between binary solutions that will be introduced in the upcoming sections are functions of the Euclidean norm of the solutions of the binary solution problem (6.1). In this section we present lower and upper bounds on the Euclidean norm (also referred to as *length*) of all binary solutions of the equation system (6.1).

If the matrix $\boldsymbol{A}$ has constant column sum k, the length of any binary solution can be determined directly from the data $\boldsymbol{b}$, as shown in Chapter 2. Let $\bar{\boldsymbol{x}} \in \{0,1\}^n$ and $\boldsymbol{b} = \boldsymbol{A}\bar{\boldsymbol{x}}$. Then

$$\sum_{i=1}^{m} b_i = \mathbf{1}_m^T \boldsymbol{b} = \mathbf{1}_m^T \boldsymbol{A}\bar{\boldsymbol{x}} = k\mathbf{1}_n^T \bar{\boldsymbol{x}} = k \sum_{j=1}^{n} \bar{x}_j.$$

and hence $\|\bar{\boldsymbol{x}}\|_2^2 = \frac{\sum_{j=1}^{n} b_i}{k}$. This result also proves that if there is more than one binary vector satisfying $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, all of them have the same length. However, if the matrix $\boldsymbol{A}$ does not have the property of constant column sums, the binary solutions of problem (6.1) may differ in length.

There are cases in which the matrix $\boldsymbol{A}$ almost has the property of constant column sums except for a very small discrepancy. It happens, for instance, due to numerical approximations. For such cases, there is a trivial way to obtain upper and lower bounds for the length of any binary solution, provided that a few conditions on $\boldsymbol{b}$ and $\boldsymbol{A}$ are satisfied:

Theorem 12. *Let $\sum_{i=1}^{m} b_i \geq 0$, $\bar{\boldsymbol{x}} \in \bar{S}_A(\boldsymbol{b})$ and put $\boldsymbol{v}^T = \mathbf{1}^T \boldsymbol{A}$. Define $\delta^+ = \max_{1 \leq i \leq n} v_i$ and $\delta^- = \min_{1 \leq i \leq n} v_i$, the maximum and the minimum column sums of $\boldsymbol{A}$, respectively. Suppose that $\delta^- > 0$. Then $\left\lceil \frac{\sum_{i=1}^{m} b_i}{\delta^+} \right\rceil \leq \|\bar{\boldsymbol{x}}\|_2^2 \leq \left\lfloor \frac{\sum_{i=1}^{m} b_i}{\delta^-} \right\rfloor$.*

Proof.

$$\sum_{i=1}^{m} b_i = \mathbf{1}^T \boldsymbol{b} = \mathbf{1}^T \boldsymbol{A}\bar{\boldsymbol{x}} = \boldsymbol{v}^T \bar{\boldsymbol{x}} = \sum_{j=1}^{n} v_j \bar{x}_j, \tag{6.2}$$

and therefore

$$\frac{\sum_{i=1}^{m} b_i}{\delta^+} \leq \sum_{j=1}^{n} \bar{x}_j \leq \frac{\sum_{i=1}^{m} b_i}{\delta^-}. \tag{6.3}$$

As $\bar{\boldsymbol{x}} \in \{0,1\}^n$, we have $\|\bar{\boldsymbol{x}}\|_2^2 = \|\bar{\boldsymbol{x}}\|_1 = \sum_{j=1}^{n} \bar{x}_j$ and

$$\left\lceil \frac{\sum_{i=1}^{m} b_i}{\delta^+} \right\rceil \leq \|\bar{\boldsymbol{x}}\|_2^2 \leq \left\lfloor \frac{\sum_{i=1}^{m} b_i}{\delta^-} \right\rfloor. \tag{6.4}$$

□

However, if the difference $\left\lfloor \frac{\sum_{i=1}^{m} b_i}{\delta^-} \right\rfloor - \left\lceil \frac{\sum_{i=1}^{m} b_i}{\delta^+} \right\rceil$ is not very small, the bounds computed by Theorem 12 can be very different from the length of a binary solution $\bar{\boldsymbol{x}}$. In order to overcome this restriction, we introduce, in the following theorems, bounds on the length of any binary solution which will be used in the remainder of this chapter.

Lemma 10. *Let $\bar{\boldsymbol{x}} \in \bar{S}_A(\boldsymbol{b})$, $\boldsymbol{x}^* = \boldsymbol{A}^\dagger \boldsymbol{b}$ and $\boldsymbol{e}^T = \mathbf{1}^T(\boldsymbol{I} - \boldsymbol{A}^\dagger \boldsymbol{A})$. Then $\|\bar{\boldsymbol{x}}\|_2^2 = \mathbf{1}^T \boldsymbol{x}^* + \boldsymbol{e}^T \bar{\boldsymbol{x}}$.*

Proof. Let $\boldsymbol{y}_{LS}^T = \mathbf{1}^T \boldsymbol{A}^\dagger$ be the transpose of the minimum norm least squares solution of the linear system $\boldsymbol{A}^T \boldsymbol{y} = \mathbf{1}$. We have

$$\boldsymbol{A}^T \boldsymbol{y}_{LS} = \mathbf{1} - \boldsymbol{e},$$

with $\boldsymbol{e} = \mathbf{1} - \boldsymbol{A}^T \boldsymbol{y}_{LS} = (\boldsymbol{I} - \boldsymbol{A}^\dagger \boldsymbol{A})^T \mathbf{1}$, the residual. Note that in the case of constant column sums we have $\boldsymbol{e} = \mathbf{0}$.

Left-multiplying the equality $\boldsymbol{A}\bar{\boldsymbol{x}} = \boldsymbol{b}$ by the vector $\boldsymbol{y}_{LS}^T$, we obtain

$$\boldsymbol{y}_{LS}^T \boldsymbol{A}\bar{\boldsymbol{x}} = (\mathbf{1}^T - \boldsymbol{e}^T)\bar{\boldsymbol{x}} = \boldsymbol{y}_{LS}^T \boldsymbol{b},$$

which yields

$$\begin{aligned} \mathbf{1}^T \bar{\boldsymbol{x}} &= \boldsymbol{y}_{LS}^T \boldsymbol{b} + \boldsymbol{e}^T \bar{\boldsymbol{x}} \\ &= \mathbf{1}^T \boldsymbol{x}^* + \boldsymbol{e}^T \bar{\boldsymbol{x}}. \end{aligned}$$

Since $\bar{\boldsymbol{x}} \in \{0,1\}^n$, we have $\mathbf{1}^T \bar{\boldsymbol{x}} = \|\bar{\boldsymbol{x}}\|_1 = \|\bar{\boldsymbol{x}}\|_2^2$. □

Lemma 10 can be interpreted as follows: any solution of the underdetermined problem (6.1) can be written as the sum of a vector orthogonal to $\mathcal{N}(\boldsymbol{A})$ (i.e., the minimum norm solution) and the orthogonal projection of itself onto $\mathcal{N}(\boldsymbol{A})$. Hence, the sum of the elements of a solution is equal to the sum of the elements of the minimum norm solution plus the sum of the elements of the orthogonal projection of this solution onto the null space of $\boldsymbol{A}$. The orthogonal projector onto $\mathcal{N}(\boldsymbol{A})$ is given by $\boldsymbol{P} = (\boldsymbol{I} - \boldsymbol{A}^\dagger \boldsymbol{A})$.

Define $\boldsymbol{e}^T = \mathbf{1}^T(\boldsymbol{I} - \boldsymbol{A}^\dagger \boldsymbol{A})$, a *correction vector*, which will be frequently used throughout this chapter.

Lemma 10 states that $\|\bar{\boldsymbol{x}}\|_2^2 = \mathbf{1}^T \boldsymbol{x}^* + \boldsymbol{e}^T \bar{\boldsymbol{x}}$, which cannot be computed exactly without knowing $\bar{\boldsymbol{x}}$. However, it is possible to bound $\|\bar{\boldsymbol{x}}\|_2^2$ as follows:

Theorem 13. *Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ and $\boldsymbol{x}^* = \boldsymbol{A}^{\dagger}\boldsymbol{b}$. Then*

$$\left\lceil \mathbf{1}^T\boldsymbol{x}^* + \sum_{i\in\{j:e_j<0\}} e_i \right\rceil \leq \|\bar{\boldsymbol{x}}\|_1 \leq \left\lfloor \mathbf{1}^T\boldsymbol{x}^* + \sum_{i\in\{j:e_j>0\}} e_i \right\rfloor.$$

Proof. By Lemma 10, we have $\|\bar{\boldsymbol{x}}\|_2^2 = \mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\bar{\boldsymbol{x}}$, allowing $\|\bar{\boldsymbol{x}}\|_2^2$ to be bounded as

$$\mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{n} \min_{\bar{y}_i\in\{0,1\}} e_i\bar{y}_i \leq \|\bar{\boldsymbol{x}}\|_2^2 \leq \mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{n} \max_{\bar{y}_i\in\{0,1\}} e_i\bar{y}_i, \tag{6.5}$$

resulting in $\left\lceil \mathbf{1}^T\boldsymbol{x}^* + \sum_{i\in\{j:e_j<0\}} e_i \right\rceil \leq \|\bar{\boldsymbol{x}}\|_2^2 \leq \left\lfloor \mathbf{1}^T\boldsymbol{x}^* + \sum_{i\in\{j:e_j>0\}} e_i \right\rfloor$, as desired. □

The bounds given in Theorem 13 are based on the idea of implicitly selecting binary vectors to minimize or maximize the sum of the elements of the correction vector $\boldsymbol{e}$. However, this approach does not consider using the number of ones of these binary vectors to bound the length of the binary solutions.

Let ρ be a permutation of $\{1,\ldots,n\}$ such that $e_{\rho(1)} \geq e_{\rho(2)} \geq \ldots \geq e_{\rho(n)}$, which can be obtained by sorting the entries e_i in non-increasing order.

Theorem 14. *Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b}) \setminus \{\mathbf{0},\mathbf{1}\}$. Suppose that $\mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\mathbf{1} < n$. There is a unique $1 \leq \ell < n$ such that*

$$\begin{aligned} &(C1) \quad \ell+1 \quad > \mathbf{1}^T\boldsymbol{x}^* + \sum_{j=1}^{\ell+1} e_{\rho(j)} \quad \textit{and} \\ &(C2) \quad \ell \quad \leq \mathbf{1}^T\boldsymbol{x}^* + \sum_{j=1}^{\ell} e_{\rho(j)}. \end{aligned}$$

For this ℓ, we have $\|\bar{\boldsymbol{x}}\|_1 = \|\bar{\boldsymbol{x}}\|_2^2 \leq \ell$.

Proof. From Lemma 10, we have

$$\|\bar{\boldsymbol{x}}\|_1 = \mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\bar{\boldsymbol{x}} = \mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{n} e_i\bar{x}_i = \mathbf{1}^T\boldsymbol{x}^* + \sum_{i\in\{j:\bar{x}_j=1\}} e_i \leq \mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{\|\bar{\boldsymbol{x}}\|_1} e_{\rho(i)}. \tag{6.6}$$

Consider the concave function $f(\ell) = \mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{\ell} e_{\rho(i)}$ and the function $g(\ell) = \ell$, for $\ell = 1,\ldots,n-1$. From Eq. (6.6), we know that $g(\|\bar{\boldsymbol{x}}\|_1) \leq f(\|\bar{\boldsymbol{x}}\|_1)$, and from the assumption of the Theorem, we have $f(n) = \mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\mathbf{1} < n = g(n)$. As we want an upper bound for $\|\bar{\boldsymbol{x}}\|_1$, we now try to find the largest value of ℓ for which $g(\ell) \leq f(\ell)$. We distinguish the following cases:

(i) $g(\ell) > f(\ell)$ for all $\ell = 1,\ldots,n$. Then there is no ℓ satisfying condition (C2), therefore this case cannot occur.

(ii) $g(\ell) \leq f(\ell)$ for $\ell = 1,\ldots,s$ and $g(\ell) > f(\ell)$ for $\ell = s+1,\ldots,n$. Then (C1) and (C2) are jointly satisfied only for $\ell = s$.

(iii) $g(\ell) > f(\ell)$ for $\ell = 1,\ldots,t-1$; $g(\ell) \leq f(\ell)$ for $\ell = t,\ldots,s$; $g(\ell) > f(\ell)$ for $\ell = s+1,\ldots,n$. Then (C1) and (C2) are jointly satisfied only for $\ell = s$.

If $g(\ell) < f(\ell)$ for all $\ell = 1,\ldots,n-1$, then $\mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\mathbf{1} \not< n$, which does not satisfies the assumptions of the Theorem. □

Example 1. *Consider the linear system of equations* $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ *with the set of binary solutions* $\bar{S}_{\boldsymbol{A}}(\boldsymbol{b}) = \{\bar{\boldsymbol{x}}^{(1)}, \bar{\boldsymbol{x}}^{(2)}, \bar{\boldsymbol{x}}^{(3)}\}$ *such that*

$$\boldsymbol{A} = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}, \boldsymbol{b} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \bar{\boldsymbol{x}}^{(1)} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \bar{\boldsymbol{x}}^{(2)} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \text{ and } \bar{\boldsymbol{x}}^{(3)} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}.$$

In order to apply Theorem 14 and obtain an upper bound on the Euclidean norm of any binary solution of the given linear system, the correction vector $\boldsymbol{e} = (\boldsymbol{I} - \boldsymbol{A}^\dagger\boldsymbol{A})^T\mathbf{1}$ *must be computed. The explicit computation of the pseudo-inverse* $\boldsymbol{A}^\dagger$ *is not necessary, as indicated in the proof of Lemma 10. Hence, we have* $\boldsymbol{e}^T = \begin{pmatrix} 0.25 & -0.5 & 0.25 & 0.25 & 0.25 \end{pmatrix}$ *and* $\boldsymbol{x}^{*T} = \begin{pmatrix} 0.625 & 0.75 & 0.125 & 0.625 & 0.125 \end{pmatrix}$.

We check the conditions (C1) and (C2) of Theorem 14 for $1 \leq \ell < 5$*, and find that they are satisfied only for* $\ell = 3$:

$$4 > \mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{4} e_{\rho(i)} = 2.25 + 1 = 3.25 \quad \text{and} \quad 3 \leq \mathbf{1}^T\boldsymbol{x}^* + \sum_{i=1}^{3} e_{\rho(i)} = 2.25 + 0.75 = 3$$

which gives, for any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$, $\|\bar{\boldsymbol{x}}\|_2^2 \leq 3$.

6.4. A bound based on the minimum norm solution

In this section, a first bound is derived on the distance between solutions of the binary reconstruction problem, which follows from the fact that the Euclidean distance between the minimum norm solution and any binary solution of Eq. (6.1) can be bounded by an expression based on the minimum norm solution and the elements of the correction vector.

Lemma 11. *Let* $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$. *Then* $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 = \sqrt{\|\bar{\boldsymbol{x}}\|_2^2 - \|\boldsymbol{x}^*\|_2^2}$.

Proof. From the definition of $\boldsymbol{x}^*$ we have $(\bar{\boldsymbol{x}} - \boldsymbol{x}^*) \in \mathcal{N}(\boldsymbol{A})$, and $\boldsymbol{x}^* \perp (\bar{\boldsymbol{x}} - \boldsymbol{x}^*)$. Applying Pythagoras' Theorem and Lemma 10 yields

$$\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 = \|\bar{\boldsymbol{x}}\|_2^2 - \|\boldsymbol{x}^*\|_2^2. \tag{6.7}$$

□

For $\bar{\boldsymbol{x}} \in \{0,1\}^n$, define $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) = \sqrt{\mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\bar{\boldsymbol{x}} - \|\boldsymbol{x}^*\|_2^2}$. Notice that if $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ we have $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) = \sqrt{\|\bar{\boldsymbol{x}}\|_2^2 - \|\boldsymbol{x}^*\|_2^2}$. The function $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$ will be used throughout this chapter, and we refer to $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$ as the *central radius of* $\bar{\boldsymbol{x}}$. According to Lemma 11, any binary solution $\bar{\boldsymbol{x}}$ of problem (6.1) is on the hypersphere centered in $\boldsymbol{x}^*$ with radius $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$. Notice that different binary solutions of problem (6.1) may lie on different hyperspheres.

In further theorems, we are interested in computing $R \in \mathbb{R}$ such that $R \geq \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$ for all $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, i.e, we want an upper bound for all $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$ with $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$. To this end, the bound R can be computed by Lemma 11 combined either with Theorem 12, 13 or 14. If we also consider the lower bound from Theorem 13 (or 12), we obtain two radii defining a spherical shell centered in $\boldsymbol{x}^*$ containing all binary solutions.

Supposing the existence of at least two different binary solutions, the upper bound R of the central radius allows us to derive an upper bound for the number of entry differences between those solutions.

Theorem 15. *Let* $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ *and* $R \geq \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{u}})$*, for all* $\bar{\boldsymbol{u}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$*. Then* $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \leq 4R^2$.

Proof. According to Lemma 11, we have $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 = \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$ and $\|\bar{\boldsymbol{y}} - \boldsymbol{x}^*\|_2 = \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{y}})$. Therefore,

$$\|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2 \leq \|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2 + \|\bar{\boldsymbol{y}} - \boldsymbol{x}^*\|_2 = \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) + \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{y}}) \leq 2R. \tag{6.8}$$

As $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ are binary, we have $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) = \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_1 = \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2^2$. □

6.5. Bounds based on rounding the minimum norm solution

In this section we repeat several results of Chapter 2 (without their proof), which can be extended to the case of non-constant column sums of the projection matrix, simply by using an upper bound on the central radius instead of using the exact central radius (as in Chapter 2). In the experiments of Section 6.7, these bounds will be compared to the new bounds introduced in Section 6.6. The results of this section are expressed as theorems and are illustrated by an example.

The fact that the elements of $\bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ surround the minimum norm solution $\boldsymbol{x}^*$, suggests that binary solutions can often be found near $\boldsymbol{x}^*$. It is therefore natural to consider the vector that is obtained by rounding each entry of $\boldsymbol{x}^*$ to the nearest binary value. The following bounds are based on the number of differences between a binary solution of problem (6.1) and a binary vector obtained by rounding $\boldsymbol{x}^*$.

Let $\mathcal{T} = \{\bar{\boldsymbol{w}} \in \{0,1\}^n : \|\bar{\boldsymbol{w}} - \boldsymbol{x}^*\|_2 \leq \|\bar{\boldsymbol{u}} - \boldsymbol{x}^*\|_2, \text{ for all } \bar{\boldsymbol{u}} \in \{0,1\}^n\}$ and let $\bar{\boldsymbol{r}} \in \mathcal{T}$, i.e., $\bar{\boldsymbol{r}}$ is among the binary vectors that are nearest to $\boldsymbol{x}^*$ in the Euclidean sense. Put $T = \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2$, i.e., the Euclidean distance from $\boldsymbol{x}^*$ to the nearest binary vector. If $R > T$ and $R - T$ is small, it is possible to say that a fraction of the rounded values are correct, i.e., to provide an upper bound on the *number* of entry differences between any solution in $\bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ and $\bar{\boldsymbol{r}}$. In most cases we cannot say *which* rounded values are correct.

Lemma 12. *Let* $\bar{\boldsymbol{r}} \in \mathcal{T}$ *and* $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$*. Then* $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) - T^2 = \sum_{i \in \mathrm{D}(\bar{\boldsymbol{x}},\bar{\boldsymbol{r}})} |2x_i^* - 1|$.

Define $\beta_i = |2x_i^* - 1|$ and let π be a permutation of $\{1,\ldots,n\}$ such that $\beta_{\pi(1)} \leq \beta_{\pi(2)} \leq \ldots \leq \beta_{\pi(n)}$, which can be obtained by sorting the entries β_i in increasing order.

Theorem 16. *Let $\bar{\boldsymbol{r}} \in \mathcal{T}$ and $R \geq \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{u}})$, for all $\bar{\boldsymbol{u}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$. Put*

$$U = \max\left\{0 \leq \ell \leq n : \sum_{i=1}^{\ell} \beta_{\pi(i)} \leq R^2 - T^2\right\}.$$

Then for any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, we have $\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \leq U$.

Example 2. *Using the same data as given in Example 1, we compute*

$$\boldsymbol{x}^* = \begin{pmatrix} 0.65 \\ 0.75 \\ 0.125 \\ 0.625 \\ 0.125 \end{pmatrix}, \quad \bar{\boldsymbol{r}} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \quad \boldsymbol{\beta} = \begin{pmatrix} 0.25 \\ 0.5 \\ 0.75 \\ 0.25 \\ 0.75 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\beta}_{\pi} = \begin{pmatrix} 0.25 \\ 0.25 \\ 0.5 \\ 0.75 \\ 0.75 \end{pmatrix},$$

where $\bar{\boldsymbol{r}}$ is obtained by rounding the entries of $\boldsymbol{x}^*$ to binary values. Then, we compute $T^2 = \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2^2 = 0.375$ and, from Lemma 11 and Example 1, $R^2 = 1.625 \geq \|\bar{\boldsymbol{x}}\|_2^2 - \|\boldsymbol{x}^*\|_2^2$.

Applying Theorem 16, we verify, for $0 \leq \ell \leq 5$, that

$$\sum_{i=0}^{3} \beta_{\pi(i)} = 1 \leq 1.25 = R^2 - T^2 \quad \text{and} \quad \sum_{i=0}^{4} \beta_{\pi(i)} = 1.75 \not\leq 1.25 = R^2 - T^2.$$

Then for any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, we have $\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \leq 3$.

Theorem 17. *Let $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, $\bar{\boldsymbol{r}} \in \mathcal{T}$ and $R \geq \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{u}})$, for all $\bar{\boldsymbol{u}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$. Put*

$$U = \max\left\{0 \leq \ell \leq n : \sum_{i=1}^{\ell} \beta_{\pi(i)} \leq 2\left(R^2 - T^2\right)\right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \leq U$.

6.6. Bounds based on a subsequent radius reduction

In this section we do not use the upper bound for the length of binary solutions given by Theorem 12 or 14, but the one from Theorem 13 only. Recall from the proof of Theorem 13 that a binary vector $\bar{\boldsymbol{z}}$ can be constructed (not necessarily a solution of the binary solution problem), which maximizes the central radius, i.e., $\bar{\boldsymbol{z}} \in \{\bar{\boldsymbol{w}} \in \{0,1\}^n : \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{w}}) \geq \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{u}}),$ for all $\bar{\boldsymbol{u}} \in \{0,1\}^n\}$. After computing the largest central radius $\mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}})$, a radius reduction is performed based on a function depending on the correction vector $\boldsymbol{e}$ and the binary vector $\bar{\boldsymbol{r}}$ closest to the minimum norm solution $\boldsymbol{x}^*$ (the center of the hypersphere containing all binary solutions). For the next theorems, recall that $\boldsymbol{e}^T = \mathbf{1}^T(\boldsymbol{I} - \boldsymbol{A}^{\dagger}\boldsymbol{A})$ and $T = \|\bar{\boldsymbol{r}} - \boldsymbol{x}^*\|_2$.

Lemma 13. *Let $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \{0,1\}^n$. Then, $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) = \mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{y}}) - \sum_{i=1}^{n} e_i(\bar{y}_i - \bar{x}_i)$.*

Proof. As $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) = \mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\bar{\boldsymbol{x}} - \|\boldsymbol{x}^*\|_2^2$ and $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{y}}) = \mathbf{1}^T\boldsymbol{x}^* + \boldsymbol{e}^T\bar{\boldsymbol{y}} - \|\boldsymbol{x}^*\|_2^2$, we have

$$\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{y}}) - \mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) = \boldsymbol{e}^T(\bar{\boldsymbol{y}} - \bar{\boldsymbol{x}}).$$

Hence, $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) = \mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{y}}) - \sum_{i=1}^n e_i(\bar{y}_i - \bar{x}_i)$. □

Lemma 14. *Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, $\bar{\boldsymbol{r}} \in \mathcal{T}$ and $\hat{\alpha}_i = e_i(\bar{z}_i - \bar{x}_i)$, for $i = 1,\ldots,n$. Then*

$$\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2 \geq \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} (\hat{\alpha}_i + \beta_i).$$

Proof. From Lemma 11, we have $\|\bar{\boldsymbol{x}} - \boldsymbol{x}^*\|_2^2 = \mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}})$. Applying Lemma 12, we find that

$$\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{x}}) - T^2 \quad = \quad \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - \sum_{i=1}^n \hat{\alpha}_i - T^2 = \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} \beta_i.$$

It is straightforward to check that $\hat{\alpha}_i \geq 0$ and $\sum_{i\in\{j:x_j=z_j\}} \hat{\alpha}_i = 0$. Hence we have

$$\begin{aligned}
\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2 &= \sum_{i=1}^n \hat{\alpha}_i + \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} \beta_i \\
&= \sum_{i\in \mathrm{D}(\bar{\boldsymbol{x}},\bar{\boldsymbol{z}}) \bigcup \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} \hat{\alpha}_i + \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} \beta_i \\
&\geq \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} (\hat{\alpha}_i + \beta_i).
\end{aligned}$$

□

Lemma 15. *Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, $\bar{\boldsymbol{r}} \in \mathcal{T}$ and $\alpha_i = e_i(\bar{z}_i - |\bar{r}_i - 1|)$, for $i = 1,\ldots,n$. Then*

$$\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2 \geq \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} (\alpha_i + \beta_i).$$

Proof. By the definition of $\boldsymbol{\alpha}$,

$$\alpha_i = e_i(\bar{z}_i - |\bar{r}_i - 1|) = e_i(\bar{z}_i - \bar{x}_i) = \hat{\alpha}_i, \quad \text{for all } i \in \mathrm{D}(\bar{\boldsymbol{x}},\bar{\boldsymbol{r}}). \tag{6.9}$$

From Lemma 14 and Eq. (6.9), we find that

$$\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2 \geq \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} (\hat{\alpha}_i + \beta_i) = \sum_{i\in \mathrm{D}(\bar{\boldsymbol{r}},\bar{\boldsymbol{x}})} (\alpha_i + \beta_i),$$

as desired. □

Let $\gamma_i = \alpha_i + \beta_i$ and ϕ be a permutation of $\{1,\ldots,n\}$ such that $\gamma_{\phi(1)} \leq \gamma_{\phi(2)} \leq \ldots \leq \gamma_{\phi(n)}$, which can be obtained by sorting the entries γ_i in increasing order.

Theorem 18. *Let $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ and $\bar{\boldsymbol{r}} \in \mathcal{T}$. Define $\alpha_i = e_i(\bar{z}_i - |\bar{r}_i - 1|)$ and $\gamma_i = \alpha_i + \beta_i$, for $i = 1, \ldots, n$. Put*

$$U = \max\left\{0 \le \ell \le n : \sum_{i=1}^{\ell} \gamma_{\phi(i)} \le \mathcal{R}_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}})^2 - T^2\right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \le U$.

Proof. From Lemma 15 we have

$$\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2 \quad \ge \quad \sum_{i \in \mathrm{D}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})} (\alpha_i + \beta_i) \ge \sum_{i=1}^{\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})} \gamma_{\phi(i)},$$

which implies that $\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \le U$. □

Theorem 18 is based in the following: consider the set of entries where $\bar{\boldsymbol{r}}$ and $\bar{\boldsymbol{x}}$ are different. If we transform $\bar{\boldsymbol{r}}$ into $\bar{\boldsymbol{x}}$ by performing a sequence of single-entry changes, each time an entry i of $\bar{\boldsymbol{r}}$ is changed the squared Euclidean distance from the current vector to $\boldsymbol{x}^*$ increases by $\beta_i = |2x^*_i - 1|$ and the current central radius reduces by $\alpha_i = e_i(\bar{z}_i - |\bar{r}_i - 1|)$.

Example 3. *Using the same data given in Example 1, we compute $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}})$, the largest central radius that can be obtained by $\bar{\boldsymbol{z}} \in \{0,1\}^n$, as given by Theorem 13.*

$$\bar{\boldsymbol{z}} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad \boldsymbol{\alpha} = \begin{pmatrix} 0.25 \\ 0 \\ 0 \\ 0.25 \\ 0 \end{pmatrix}, \quad \boldsymbol{\gamma} = \begin{pmatrix} 0.5 \\ 0.5 \\ 0.75 \\ 0.5 \\ 0.75 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\gamma}_\phi = \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.75 \\ 0.75 \end{pmatrix}.$$

With $\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2 = 1.875 - 0.375 = 1.5$, we apply Theorem 18 and verify, for $0 \le \ell \le 5$, that

$$\sum_{i=0}^{3} \gamma_{\phi(i)} = 1.5 \le 1.5 \quad \text{and} \quad \sum_{i=0}^{4} \gamma_{\phi(i)} = 2.25 \not\le 1.5.$$

Then for any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$, we have $\mathrm{d}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \le 3$.

Theorem 19. *Let $\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}} \in \bar{S}_{\boldsymbol{A}}(\boldsymbol{b})$ and $\bar{\boldsymbol{r}} \in \mathcal{T}$. Define $\alpha_i = e_i(\bar{z}_i - |\bar{r}_i - 1|)$ and $\gamma_i = \alpha_i + \beta_i$, for $i = 1, \ldots, n$. Put*

$$U = \max\left\{0 \le \ell \le n : \sum_{i=1}^{\ell} \gamma_{\phi(i)} \le 2\left(\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2\right)\right\}.$$

Then $\mathrm{d}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}) \le U$.

Proof. From Lemma 15 we have

$$2\left(\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2\right) \ge \sum_{i \in \mathrm{D}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}})} \gamma_i + \sum_{i \in \mathrm{D}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{y}})} \gamma_i = \sum_{i \in \mathrm{D}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}})} \gamma_i + \sum_{i \in \mathrm{D}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{x}}) \bigcap \mathrm{D}(\bar{\boldsymbol{r}}, \bar{\boldsymbol{y}})} 2\gamma_i. \tag{6.10}$$

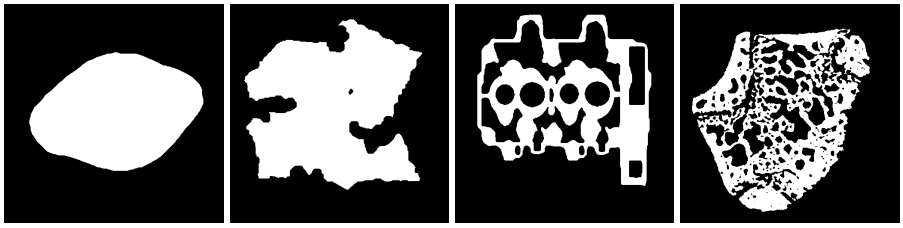
As $\gamma_i \geq 0$, for all $i = 1, \ldots, n$, then

$$2\left(\mathcal{R}^2_{\boldsymbol{A},\boldsymbol{b}}(\bar{\boldsymbol{z}}) - T^2\right) \geq \sum_{i \in \mathrm{D}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})} \gamma_i \geq \sum_{i=1}^{\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})} \gamma_{\phi(i)}, \tag{6.11}$$

which implies that $\mathrm{d}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) \leq U$. □

6.7. Numerical experiments

A series of experiments was performed to investigate the practical value of the bounds given in the theorems and corollaries presented in the previous sections, where the bounds were evaluated for a range of problems. The experiments have been performed for two basic acquisition geometries: parallel beam and fan beam tomography. All experiments are based on simulated projection data obtained by computing the projection data from the test images (so-called *phantoms*) in Fig. 6.1. We refer to Chapter 2 for information on the origin of these phantoms.



(a) Phantom 1 (b) Phantom 2 (c) Phantom 3 (d) Phantom 4

Figure 6.1: *Original phantom images used for the experiments.*

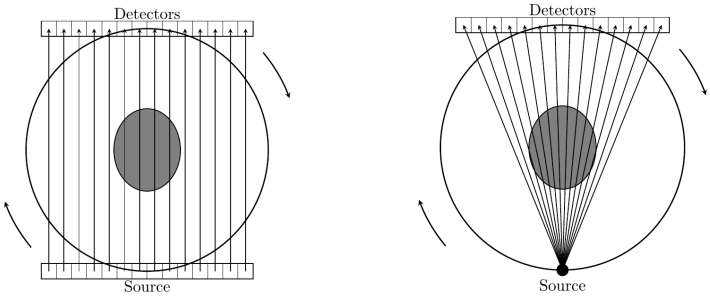
All phantoms have a size of 512×512 pixels. To perform experiments for images with varying image size (smaller than 512×512), the phantoms have been downscaled to obtain binary images of the appropriate sizes.

The remainder of this section is structured as follows. Brief descriptions of parallel beam tomography and fan beam tomography are presented in section 6.7.1. The quality of the bounds on the *length* of binary solutions is evaluated in section 6.7.2. Experiments with bounds on the difference between binary solutions for the tomography problem are presented in section 6.7.3.

6.7.1. Tomography models

Throughout the tomography literature, several imaging models have been considered [33, section 7.4.1]. The unknown image is often approximated by an image defined on a discrete pixel grid. In *parallel beam tomography*, a projection is computed by considering a set of parallel rays in a given direction and computing a weighted sum of all the pixels that

intersect with each ray, see Fig. 6.2(a). We select projection angles equally spaced between 0°and 180°.



(a) Parallel beam geometry (b) Fan beam geometry

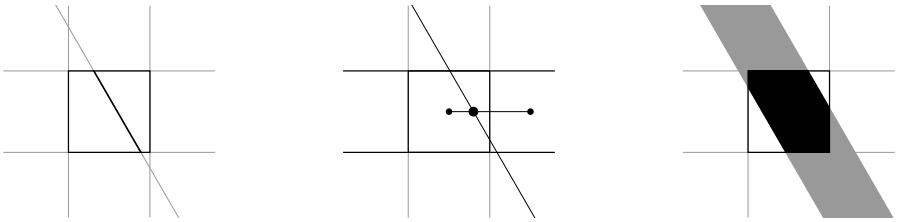
Figure 6.2: *Tomography geometry*

In *fan beam tomography*, a point source emits a set of rays in all directions and an array of detectors measures the weighted sum of all the pixels that intersect with each ray, see Fig. 6.2(b). In our case, we assume that the detector is flat, i.e., all measurements are performed on a detector that follows a straight line. The phantom image is centered around the point of rotation of the source and detector. As the source and detector are positioned further away from the center of ration, the the fan beam geometry becomes more and more similar to a parallel beam geometry. We select projection angles equally spaced between 0°and 360°.

The intersection between a pixel and a ray can be computed in different ways, each leading to a different model for the imaging process. A common model for computing the projections of a pixelized image is the *line model*. In the line model, the weight a_{ij}, defined by the intersection between beam i and pixel j is equal to the intersection length between the line (beam) and the pixel, Fig. 6.3(a).

In parallel beam tomography, for the case where the projection is aligned with the horizontal and vertical axis, the weight function of the line model has two discontinuities. Due to floating point errors, these can easily lead to pixel weights set to 0, where in fact they should be set to 1, or vice versa. The weighting scheme introduced by Joseph [32] does not have this drawback. Here, the weights a_{ij} are the interpolation coefficients obtained when tracing the line row by row (or column by column, depending on the projection angle), and applying linear interpolation between the centers of the two adjacent pixels, as shown in Fig. 6.3(b).

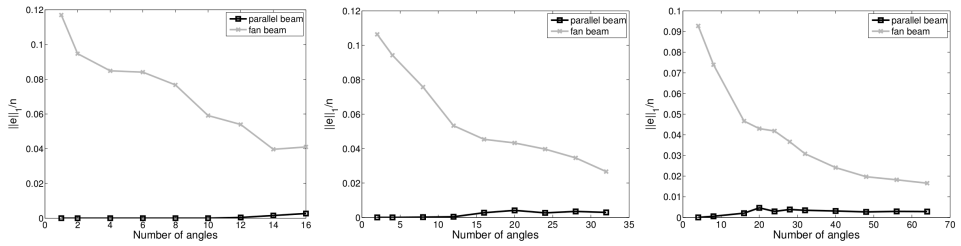
The *strip model* differs from the line model because the beam is a strip instead of a line. The weight a_{ij} is determined by the intersection area between strip i and the pixel j. For both line and Joseph's models, the column sums of their respective projection matrix is not constant while it is constant for the strip model in parallel beam tomography. Despite the fact that a projection matrix for the line model in parallel beam tomography does not have constant column sums, the variance of the column sums is small. The line model in fan beam tomography shows stronger difference as compared with the line model from parallel beam tomography. To show this, we have computed the *correction vector* $\boldsymbol{e}$ and plotted the sum



(a) Line model (b) Joseph's model (c) Strip model

Figure 6.3: *Projection models*

of the absolute value of its entries divided by its number of entries for several projection angles, as shown in Fig. 6.4. Recall that when the projection matrix has constant column sums, the vector $\boldsymbol{e}$ is a null vector.



(a) Image size: 32×32 (b) Image size: 64×64 (c) Image size: 128×128

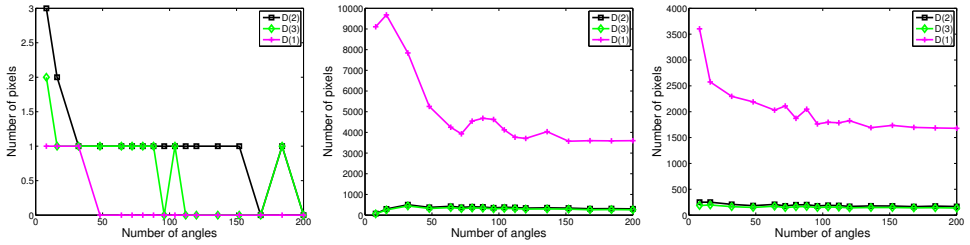
Figure 6.4: *Relative ℓ_1-norm of the correction vector $\boldsymbol{e}$ for the line model in parallel and fan beam tomography.*

We will show experimental results for parallel beam tomography with the three projection models presented (line, Joseph's and strip). For fan beam tomography we use only the line model.

6.7.2. Bounds on the number of ones in binary solutions

The linear system (6.1) may have binary solutions with different lengths. In Section 6.4 we have presented different ways of computing an upper bound for the length of any binary solution. The expressions $D(i)$, $(i = 1,2,3)$ represent the difference between the computed upper bound for the number of ones of any binary solution (Theorems 12, 13 and 14, respectively) and the actual number of ones of the binary image used to construct the projections. A comparison between these bounds is shown in Fig. 6.5 for Phantom 4 with size 512×512. Fig. 6.5 includes three graphs, each one for a different parallel beam tomography model.

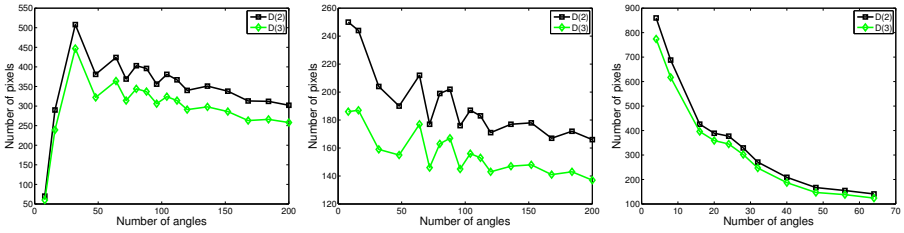
The graph in Fig. 6.5(a) corresponds to the strip model, for which the projection matrix has the property of constant column sums except for numerical errors. Figures 6.5(b) and 6.5(c) correspond to the line and Joseph's models.



(a) Strip model (b) Line model (c) Joseph's model

Figure 6.5: *Absolute difference between the bound for the number of ones in any binary solution and the number of ones in the phantom.*

It can be observed that for the strip model, the most basic bound is typically smaller than the other two, more refined, bounds. However, the number of pixels by which this bound differs from the other bounds is very small compared with the total number of pixels. The graphs for the line and Joseph's models show that the bound given in Theorem 12 can give very high bounds with a significant difference with respect to the other two bounds, which are close to each other. The bounds for the line and Joseph's model can be better visualized in Fig. 6.6. A comparison between these bounds is shown in Fig. 6.6 for Phantom 4 with size 512×512 for parallel beam tomography. Fig. 6.6 also includes the same bounds for Phantom 4 of size 128×128 with the line model for fan beam tomography.

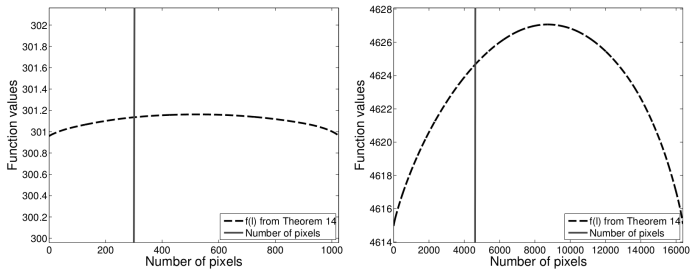


(a) Parallel beam: line model (b) Parallel beam: Joseph's model (c) Fan beam: line model

Figure 6.6: *Absolute difference between the bound for the number of ones in any binary solution and the number of ones of the phantom.*

The upper bound for the number of ones in binary solutions given in Theorem 14 is computed by the intersection of two functions. One of these functions is the linear function $g(\ell) = \ell$, which corresponds with the number of ones in a binary solution, while the other function is a concave function, which defines an upper bound on the number of ones. An example of a plot of these two functions can be seen in Fig. 6.7 for Phantom 4, using Joseph's model. The Phantom used for Fig. 6.7(a) has size 32×32 and 10 projection angles were used. For Fig. 6.7(b) the Phantom has size 128×128 with 32 projection angles. The results for fan beam tomography are somewhat similar to the ones of parallel beam tomography and are not shown here. Note that the graphs of the linear function g are almost vertical, due to the scale of the plots.

From Fig. 6.7(a), we have $300 < f(\ell) < 302$ for $0 \le \ell < n$, so $\|\bar{\boldsymbol{x}}\|_1 = 301$. For the example in Fig. 6.7(b) the bound bound is not exact, but it is very close to the actual number of ones (4615) of the original phantom.



(a) Phantom 4 (32×32), 10 projections (b) Phantom 4 (128×128), 32 projections

Figure 6.7: *Intersection between the two functions determines an upper bound for the length of any binary solution.*

There is no guarantee that increasing the number of projection angles decreases the bound for the number of ones in the binary solutions. This implies that the bound for the number of ones can be recomputed every time a new angle is added and the smallest selected to generate the error bounds. This has not been done in the graphs of this chapter.

6.7.3. Bounds on the difference between binary solutions and binary approximate solutions

We now focus on the computation of the actual quality bounds for solutions of the binary reconstruction problem. In each experiment, the minimum norm solution $\boldsymbol{x}^*$ was first computed using the CGLS algorithm. For some bounds, it is necessary to compute the rounded central reconstruction $\bar{\boldsymbol{r}}$ which was performed by rounding $\boldsymbol{x}^*$ to the nearest binary vector, choosing $\bar{r}_i = 1$ if $x_i^* = \frac{1}{2}$. Based on $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{r}}$, the various upper bounds described in Sections 6.3-6.6 were computed.

When presenting the results, we express the bounds on the pixel differences between two images as a *fraction* of the total number of image pixels. This allows for more straightforward interpretation of the results than using the absolute number of pixel differences. As a substantial number of bounds will be given throughout this chapter, we introduce the following notation:

- The expressions $U_d(i)$ $(i = 1, 2, 3)$ will represent bounds on the number of pixel differences between any two binary solutions of the reconstruction problem.
- The expressions $U_s(i)$ $(i = 1, 2)$ will represent bounds on the number of pixel differences between the *rounded central reconstruction* $\bar{\boldsymbol{r}}$ and any binary solution.

The bounds *within* each class U_d and U_s represent upper bounds for the same distance measure and can therefore be compared. The expression E_s denotes the number of pixel differences between the rounded central reconstruction and the phantom.

The expressions $U_d(1)$ is computed by using Theorem 15, $U_d(2)$ by using Theorem 17 and $U_d(3)$ by using Theorem 19. The expressions $U_s(i)$, $(i = 1, 2)$ are computed by using Theorems 16 and 18 respectively, with the bound on the length of binary solutions from Theorem 14.

Several graphs presented in this section use a logarithmic scale for the error bounds. In some cases, the bound may become very small, or even 0, resulting in a point on the graph that cannot be plotted. These points are simply removed from the plot, causing the graph to be disconnected.

Parallel beam tomography

Experiments have been performed based on the four phantom images, scaled to sizes of 32×32, 128×128 and 256×256 respectively, varying the number of projection directions. The first set of results are shown in Fig. 6.8, where bounds $U_s(1)$ and $U_s(2)$ on the number of differences between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$, and the exact error between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$ are jointly plotted. In Fig. 6.9, the bounds $U_d(1) - U_d(3)$ on the distance between any two binary solutions of the reconstruction problem are shown for the same experiments. For both Fig. 6.8 and Fig. 6.9 we used Joseph's model, which has results similar to the line model.

Fan beam tomography

Experiments have been performed based on the four phantom images, scaled to sizes of 32×32, 128×128 and 256×256 respectively, varying the number of projection directions. The first results are shown in Fig. 6.10, where bounds $U_s(1)$ and $U_s(2)$ on the number of differences between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$, and the exact error between $\bar{\boldsymbol{r}}$ and the phantom image $\bar{\boldsymbol{x}}$ are jointly plotted. In Fig. 6.11, the bounds $U_d(1) - U_d(3)$ on the distance between any two binary solutions of the reconstruction problem are shown for the same experiments.

6.7.4. Discussion of the results

Despite the facts that the four phantoms have strong differences in shape and morphology, and that the tomography models are quite different, the results shown in Figs. 6.8-6.11 are consistent throughout all experiments. In general, the bounds become smaller as the number of projection angles is increased.

From the difference between the bounds based on Section 6.4 and the bounds based on the rounded central reconstruction, we see that in most cases the phantom $\bar{\boldsymbol{x}}$ is substantially closer to $\bar{\boldsymbol{r}}$ than to $\boldsymbol{x}^*$.

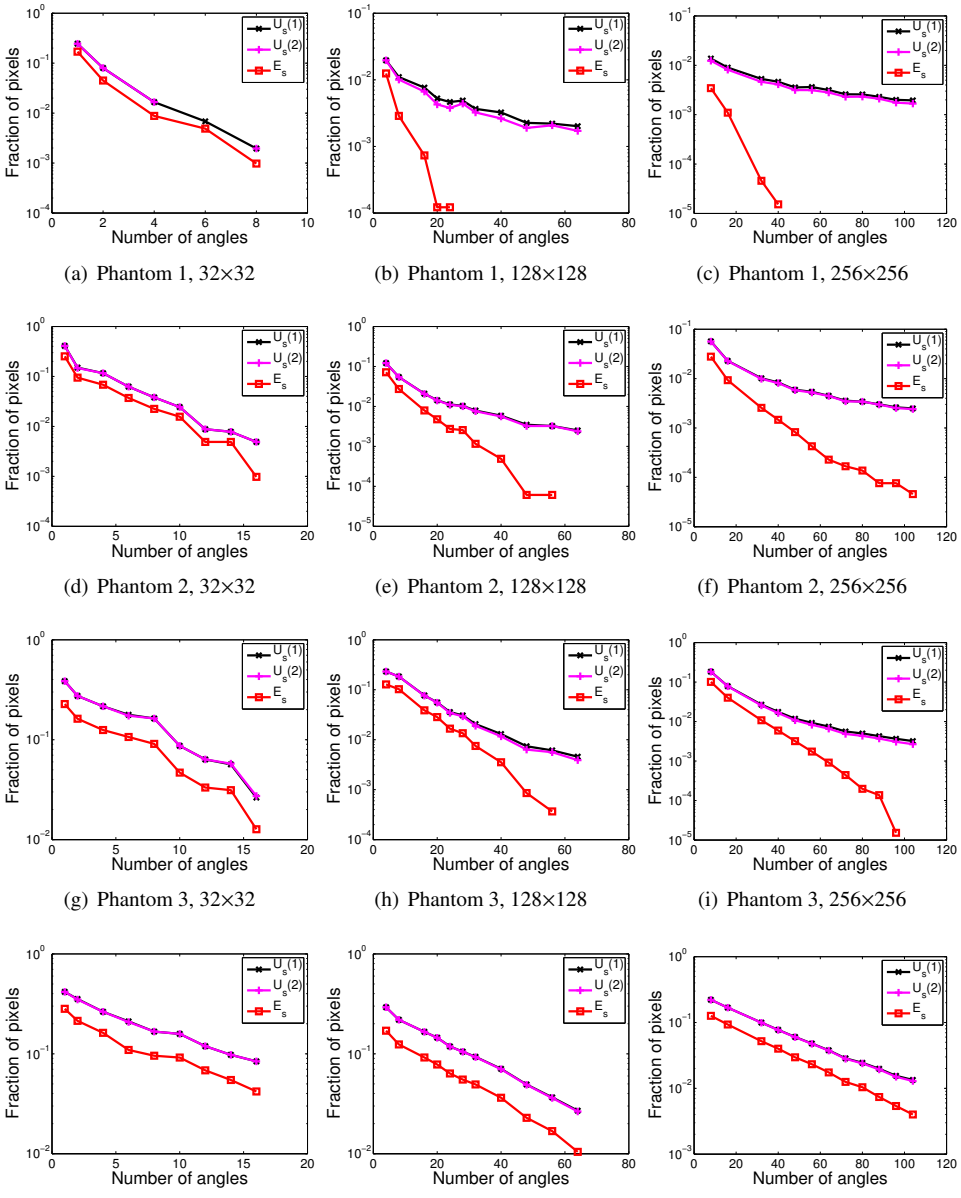
In Figs. 6.8 and 6.10, it can be observed that the true fraction of pixel differences between the phantom image $\bar{\boldsymbol{x}}$ and the rounded central reconstruction $\bar{\boldsymbol{r}}$, denoted by E_s, is sometimes well approximated by the bound U_s , in particular for small images. Although bounding errors for the line model for fan beam tomography is more challenging than for parallel beam tomography, the bounds are reasonably low and effective.

In some of the figures, parts of the graph for $U_s(2)$ are missing, caused by zero values that cannot be displayed in the logarithmic scale. This can even occur due to numerical inaccuracies in the computation of the bounds if the true, correct bound is slightly larger than 0.

6.8. Outlook and conclusions

In this chapter, we have presented a range of general bounds on the accuracy of binary solutions, with respect to the unknown original vector. The bounds are based on an approach initiated in Chapter 2, where the we presented bounds for the case that the projection matrix has constant column sums. In the present chapter, bounds have been derived that are much more general as they do not depend on this assumption.

Our bounds can be computed within reasonable time and give guarantees: (i) on the number of vector entries that can be different between any two binary solutions of an underdetermined problem and (ii) on the difference between a vector obtained by rounding the central reconstruction and any binary solution. The experimental results for parallel beam and fan beam tomography show that by using these bounds, one can prove that the number of differences between binary solutions of the reconstruction problem must be very small, even if the corresponding real-valued system of equations is severely underdetermined.



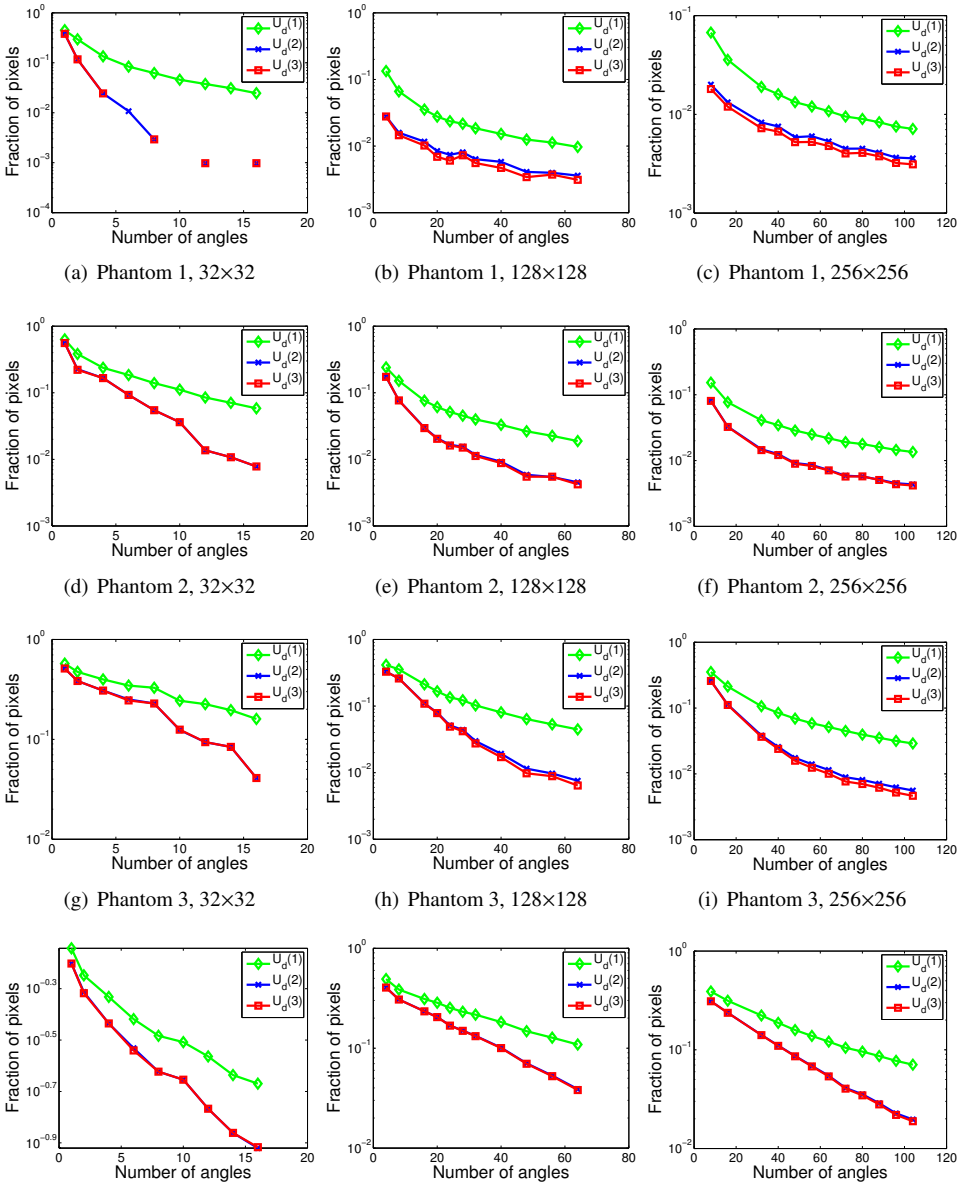
(a) Phantom 1, 32×32 (b) Phantom 1, 128×128 (c) Phantom 1, 256×256

(d) Phantom 2, 32×32 (e) Phantom 2, 128×128 (f) Phantom 2, 256×256

(g) Phantom 3, 32×32 (h) Phantom 3, 128×128 (i) Phantom 3, 256×256

(j) Phantom 4, 32×32 (k) Phantom 4, 128×128 (l) Phantom 4, 256×256

Figure 6.8: *Parallel beam, Joseph's model: computed U_s bounds as a function of the number of projection angles.*



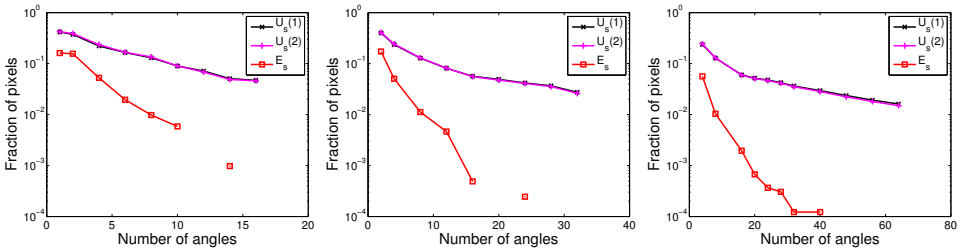
(a) Phantom 1, 32×32 (b) Phantom 1, 128×128 (c) Phantom 1, 256×256

(d) Phantom 2, 32×32 (e) Phantom 2, 128×128 (f) Phantom 2, 256×256

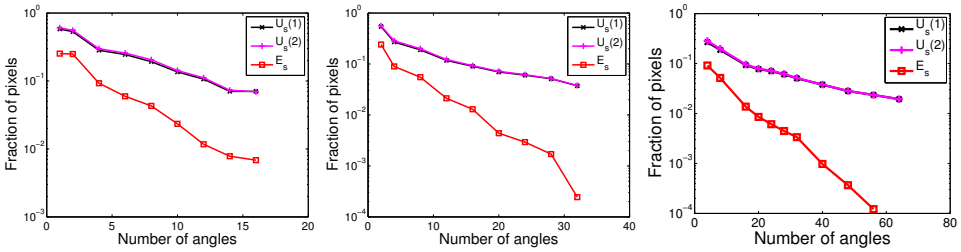
(g) Phantom 3, 32×32 (h) Phantom 3, 128×128 (i) Phantom 3, 256×256

(j) Phantom 4, 32×32 (k) Phantom 4, 128×128 (l) Phantom 4, 256×256

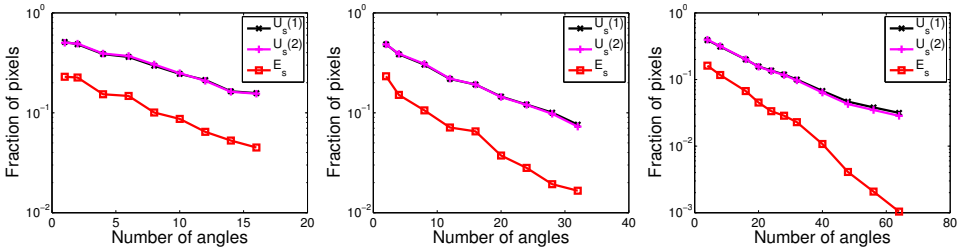
Figure 6.9: *Parallel beam, Joseph's model: computed U_d bounds as a function of the number of projection angles.*



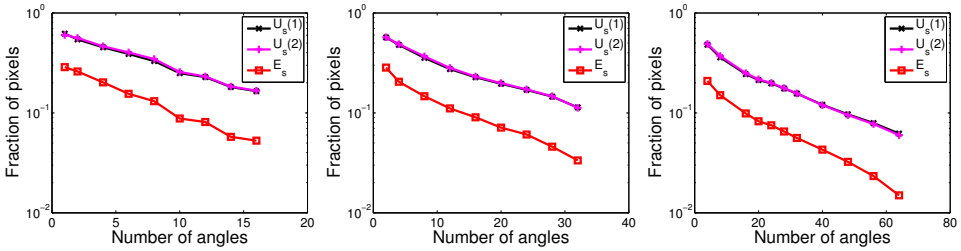
(a) Phantom 1, 32×32 (b) Phantom 1, 64×64 (c) Phantom 1, 128×128



(d) Phantom 2, 32×32 (e) Phantom 2, 64×64 (f) Phantom 2, 128×128

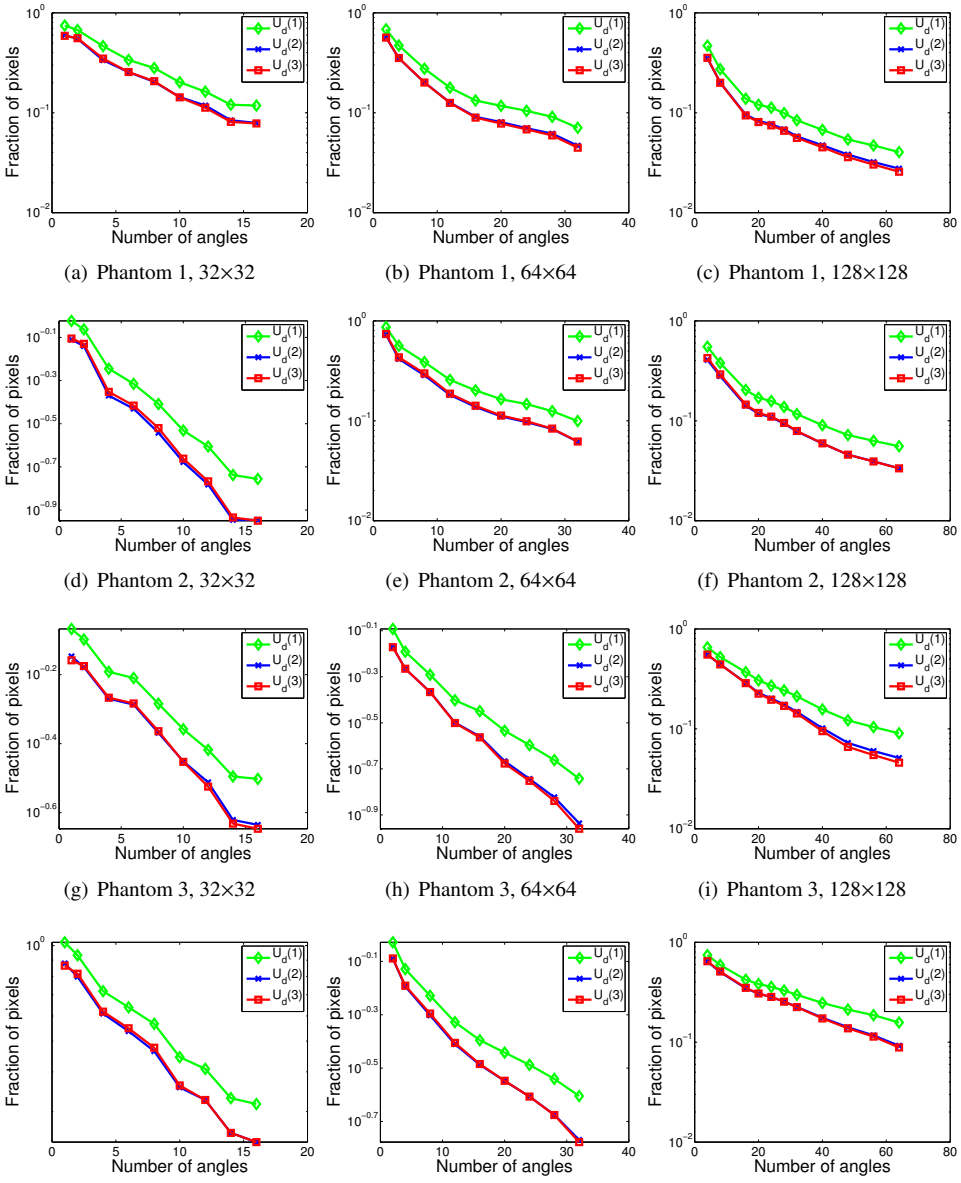


(g) Phantom 3, 32×32 (h) Phantom 3, 64×64 (i) Phantom 3, 128×128



(j) Phantom 4, 32×32 (k) Phantom 4, 64×64 (l) Phantom 4, 128×128

Figure 6.10: *Fan beam, line model: computed U_s bounds as a function of the number of projection angles.*



(a) Phantom 1, 32×32 (b) Phantom 1, 64×64 (c) Phantom 1, 128×128

(d) Phantom 2, 32×32 (e) Phantom 2, 64×64 (f) Phantom 2, 128×128

(g) Phantom 3, 32×32 (h) Phantom 3, 64×64 (i) Phantom 3, 128×128

(j) Phantom 4, 32×32 (k) Phantom 4, 64×64 (l) Phantom 4, 128×128

Figure 6.11: *Fan beam, line model: computed U_d bounds as a function of the number of projection angles.*

Chapter 7

Approximate discrete reconstruction algorithm

This chapter (with minor modifications) has been published as: K. J. Batenburg, W. Fortes, and R. Tijdeman. Approximate discrete reconstruction algorithm. Fundamenta Informaticae, Vol. 125(3–4), 239–259, 2013.

7.1. Introduction

Discrete tomography deals with tomographic reconstruction of greyscale images for which the set of possible grey levels is discrete and small [27,28]. An image of an unknown object is reconstructed from a series of *projections* of the object, taken along a range of angles. Contrary to general computed tomography (CT) [25, 33], which requires a large number of projections to obtain an accurate reconstruction, the constraint on the set of grey levels in discrete tomography enables accurate reconstruction from a relatively small number of projections, depending on the properties of the object [9, 27, 28]. A projection for a single angle can be modeled as a collection of *line sums* along parallel lines through the object, where the exact definition of "line sum" depends on the particular model used for the object and the projection process. The *grid model*, where the object is modeled as an image defined on a discrete set of points and line sums are taken along lattice lines through these grid points, can be used to model the position of atoms in crystalline nanostructures [1, 31]. Other projection models, such as the *line* and *strip* models [33, section 7.4.1], [50], are formed by discrete approximation of integral operators that compute integrals of a function defined on the plane. Such an integral model can be used to model the projection process in tomography at lower magnifications than the atomic level, such as X-ray tomography. A range of reconstruction algorithms for discrete tomography have been proposed in the literature [6, 9, 28, 40]. Most of these algorithms are specifically designed for the case of *binary* tomography, where only two grey levels are allowed in the reconstruction.

None of these algorithms comes with a guarantee that an exact solution of the discrete tomography problem is always found, which is not surprising considering the fact that the

reconstruction problem is NP-hard for the grid model case [20]. At the same time, the results of computational experiments suggest that in many cases, a solution is found that is near-optimal, or even completely identical to the original object from which the projections were taken, even if the number of projections is very small (less than 10, say). A principal problem with these algorithms is the fact that the error made in the reconstruction depends on the particular problem instance and cannot be bounded sharply. One notable exception is the algorithm proposed in [20], where the ratio between the *number* of 1's (i.e. grid points having a value of 1) in the unknown ground truth object and the reconstructed object is bounded in the optimal solution of certain relaxed variants of the binary reconstruction problem. We are not aware of any algorithm for which nontrivial bounds have been described for the difference between the projections of the reconstructed (discrete) image and the given projections.

In this chapter, we propose a discrete approximate reconstruction algorithm that comes with such guarantees, based on bounds derived in [23], which are in turn based on the Beck-Fiala Theorem [10]. Our algorithm computes an image that has only grey values belonging to the given finite set. It also guarantees that the difference between the given projections and the projections of the reconstructed discrete image is bounded. The bound, which is explicitly computable, is independent of the image size and scales linearly with the number of projection angles. The algorithm combines techniques from combinatorics with algebraic methods for the solution of linear equation systems.

This chapter is structured as follows. Section 2 introduces notation and describes the main concepts and mathematical objects used throughout the chapter. In Section 3, the basic algorithm is introduced. A proof for the upper bound on the projection error of the reconstructed image is given in Section 4. Section 5 presents several variations of the algorithm, which can improve its computational performance in practice. Computational and numerical aspects are discussed in Section 6. In Section 7, results are presented for a series of simulation experiments based on three phantom images, including both binary images and an image with three grey levels. Finally, Section 8 provides an outlook and conclusions.

7.2. Notation and concepts

Tomography concerns the reconstruction of an object from its *projections*. The exact definition of the term "projection" can vary, but in general it is related to the set of line integrals (or surface integrals along a tight strip) through the object in a particular direction. We refer to the specific model used to determine such integrals as the *imaging model*. Throughout the discrete tomography literature, several imaging models have been considered, such as the line, strip and Joseph's model [32]. Since the algorithm developed here can be used for any imaging model, we do not specify any model for the development of the theory, only for the experiments.

The *unknown object* that one wants to reconstruct is typically approximated by an image defined on a discrete pixel grid. The image that we want to reconstruct is unknown but we do know its grey values, which belong to a discrete set. Let $D = \{d_1, \ldots, d_s\} \subset \mathbb{R}^s$ be a given set of grey values such that $d_1 < d_2 < \ldots < d_s$ with s small. Also, let $d = \max_{i=1,\ldots,s-1}(d_{i+1} - d_i)$. An *image* is represented by a vector $\boldsymbol{x} = (x_i) \in \mathbb{R}^n$. We refer to the entries of $\boldsymbol{x}$ as *pixels*,

corresponding to elements in the discrete pixel grid. A *discrete image*, in this chapter, corresponds with a vector $\bar{\boldsymbol{x}} \in D^n$.

For given set of k projection directions, each consisting of w measured values, the *projection map* maps an image $\boldsymbol{x}$ to a vector $\boldsymbol{p} \in \mathbb{R}^m$ of projection data, where $m = kw$ denotes the total number of measurements. As the projection map is a linear transformation, it can be represented by a matrix $\boldsymbol{W} = (w_{ij}) \in \mathbb{R}^{m \times n}$, called the *projection matrix*. The entry w_{ij} represents the weight of the contribution of pixel x_j to the projection value p_i, which typically satisfies $0 \leq w_{ij} \leq 1$. For two images $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$, we refer to $\boldsymbol{W}(\boldsymbol{x} - \boldsymbol{y})$ as the *projection difference* of $\boldsymbol{x}$ and $\boldsymbol{y}$, and to its largest component in absolute value $\|\boldsymbol{W}(\boldsymbol{x} - \boldsymbol{y})\|_\infty$ as the *projection distance*.

The *general reconstruction problem* consists of finding a solution of the system

$$\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p} \tag{7.1}$$

for given projection data $\boldsymbol{p} = (p_i)$, i.e., to find an image that has the given projections. In *discrete tomography*, one seeks a solution of the general reconstruction problem which belongs to D.

Throughout the chapter, we use the symbol $\mathbf{0}$ to denote a column vector $(0, \ldots, 0)^T$ consisting of 0's, where its dimension is clear from the context. Also, we often use $\kappa = \|\boldsymbol{W}\|_1 = \max_{j=1,\ldots,n} \sum_{i=1}^{m} |w_{ij}|$.

For the following sections, consider the problem of finding a solution, belonging to the set D^n, of a fixed linear system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ called the *discrete reconstruction problem*. We call $\bar{\boldsymbol{x}} \in D^n$ an *approximate discrete solution* if its projection difference with respect to a solution of the general reconstruction problem is small.

7.3. Algorithm description

The algorithm presented here aims to reconstruct an image that only contains grey values belonging to the given set D and ensures that its projections have a projection distance from the given projections smaller than κd. The parameter $\kappa = \|\boldsymbol{W}\|_1$ depends on the projection model and the number of projection angles, while $d = \max_{i=1,\ldots,s-1}(d_{i+1} - d_i)$ is the largest difference between two consecutive values in D. The bounds on the projection distance do not depend on the image size.

The algorithm requires an initial grey scale image $\boldsymbol{x}^{(0)} \in [d_1, d_s]^n$ and uses the projections $\boldsymbol{p} = \boldsymbol{W}\boldsymbol{x}^{(0)}$ of the initial image as the given projections for the reconstruction. If a pixel $x_j^{(0)}$ already belongs to the set D, it is left unchanged by the algorithm. For every pixel j such that $x_j^{(0)} \notin D$, there is a unique i such that $d_i < x_j^{(0)} < d_{i+1}$. In the reconstructed image $\bar{\boldsymbol{x}} \in D^n$, this pixel j satisfies $\bar{x}_j \in \{d_i, d_{i+1}\}$, so a pixel can either be assigned the first grey level larger than its initial value, or the first grey level smaller than its initial value.

In each iteration, a new image $\boldsymbol{x}^{(t)}$ is computed that has a projection distance smaller than κd with respect to the initial image $\boldsymbol{x}^{(0)}$. Whenever the current image $\boldsymbol{x}^{(t)}$ contains pixels in D, the next image $\boldsymbol{x}^{(t+1)}$ will contain these pixels with the same values as the current image. Furthermore, the current image $\boldsymbol{x}^{(t)}$ contains more pixels belonging to D than the previous image $\boldsymbol{x}^{(t-1)}$.

The iteration step adds a specific *ghost image* to the current image $\boldsymbol{x}^{(t)}$. A ghost image for a given projection matrix is an image in the null space of that matrix. The ghost image may change the value of pixels not in D but not the ones in D, and is constructed in such a way that the number of pixels for which the value is in D *increases*. Therefore, the number of such iteration steps is at most n, the total number of pixels.

In each iteration, some line sums remain unchanged. Such line sums are associated with equations in the system (7.1) for which the sum of the coefficients, of the pixels currently not in D, is not smaller than κ. The remaining line sums may have an error of at most κd with respect to the projections of the initial image.

In order to specify the ghost image, define the following sets

Definition 1. *For $i = 1,\ldots,m$, define $L_i = \{1 \le j \le n : w_{ij} \neq 0\}$, the set of indices of pixels that occur in the ith equation of the linear system (7.1). Based on the current image $\boldsymbol{x}^{(t-1)}$ in iteration $t-1$, we can define $\bar{I}_D^{(t)} = \{1 \le j \le n : x_j^{(t-1)} \notin D\}$, the set of indices of pixels with values not in D in $\boldsymbol{x}^{(t-1)}$ and its complement $I_D^{(t)} = \{1,\ldots,n\} \setminus \bar{I}_D^{(t)}$. Another important set for the algorithm is $G^{(t)} = \{L_i : \sum_{j \in L_i \cap \bar{I}_D^{(t)}} w_{ij} \ge \kappa$, for $i = 1,\ldots,m\}$, the family of sets L_i of pixel indices, such that the sum of the corresponding coefficients is at least κ for pixels not in D in $\boldsymbol{x}^{(t-1)}$.*

Definition 2. *Let $\boldsymbol{x}^{(t-1)}$ be the reconstructed image that resulted from iteration $t-1$. Consider the following homogeneous linear system of equations*

$$\sum_{j \in L_i \cap \bar{I}_D^{(t)}} w_{ij} y_j^{(t)} = 0 \quad \text{for all } L_i \in G^{(t)} \tag{7.2}$$

$$y_j^{(t)} = 0 \quad \text{for all } j \notin (\cup_{L_i \in G^{(t)}} L_i) \cap \bar{I}_D^{(t)}. \tag{7.3}$$

Let $\boldsymbol{A}^{(t)}$ be the matrix associated with the linear system corresponding to the equations (7.2) and (7.3). An image $\boldsymbol{y}^{(t)} \in \mathcal{N}(\boldsymbol{A}^{(t)}) = \{\boldsymbol{x} : \boldsymbol{A}^{(t)}\boldsymbol{x} = \boldsymbol{0}\}$ is called a ghost image *associated with $\boldsymbol{A}^{(t)}\boldsymbol{y}^{(t)} = \boldsymbol{0}$.*

In iteration t of the algorithm, a ghost image associated with $\boldsymbol{A}^{(t)}\boldsymbol{y}^{(t)} = \boldsymbol{0}$ is computed, which is subsequently used to form the new reconstruction $\boldsymbol{x}^{(t)}$. Pseudocode for the procedure that uses this ghost image is shown in Procedure 1. As a result of applying Procedure 1, a new reconstruction is formed for which at least one new pixel belongs to D.

Procedure 1 *Calculate $\boldsymbol{x}^{(t)}$*

```
λ_min ← ∞
for j ∈ Ī_D^(t) do
    Compute the smallest λ > 0 s.t. x_j^(t−1) + λ y_j^(t) ∈ D
    if λ < λ_min then
        λ_min ← λ
    end if
end for
x^(t) ← x^(t−1) + λ_min y^(t)
```

If $\mathcal{N}(\boldsymbol{A}^{(t)}) = \{\mathbf{0}\}$, hence $\boldsymbol{y}^{(t)} = \mathbf{0}$, Procedure 1 has no effect. Another technique must be employed to change the current value of pixels not yet in D. In that case, all pixel values not in D are rounded to their nearest value in D, after which the algorithm terminates. The algorithm ends with a final vector $\bar{\boldsymbol{x}} \in D^n$ satisfying the given bounds. This part of the algorithm is shown in Procedure 2.

Procedure 2 *Calculate* $\bar{\boldsymbol{x}}$

for $j \in \{1, \ldots, n\}$ **do**
  if $x_j^{(t)} \in D$ **then**
    $\bar{x}_j \leftarrow x_j^{(t)}$
  else
    Assign $\bar{x}_j$ the element of D which is nearest to $x_j^{(t)}$
  end if
end for

The iterative algorithm which obtains a vector in D^n with projection error at most κd is given in the flowchart represented in Fig. 7.1.

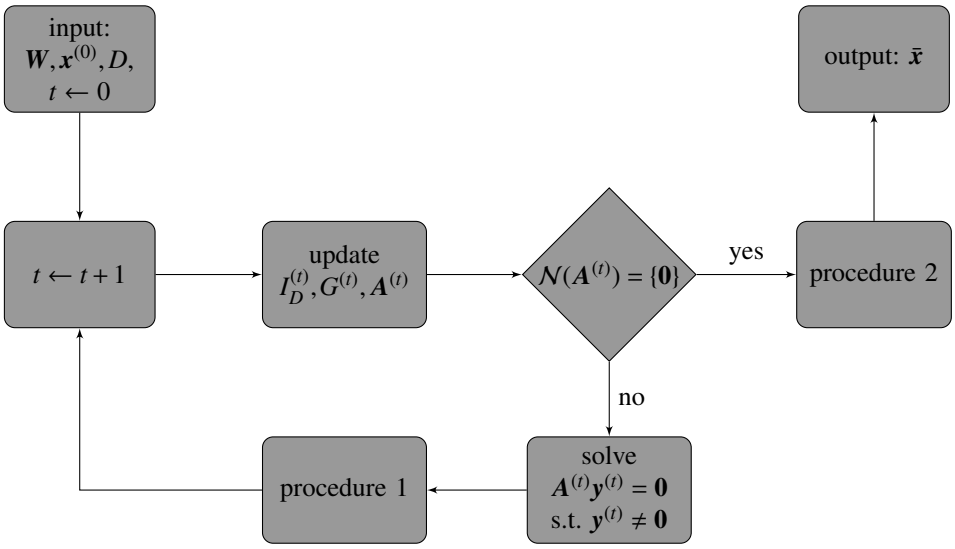


Figure 7.1: *This flowchart describes the discrete reconstruction algorithm which obtains a discrete image* $\bar{\boldsymbol{x}}$ *in* D *with projection distance of at most* κd *with respect to a given image* $\boldsymbol{x}^{(0)}$.

7.4. Walkthrough example

We have built a simple example to illustrate each step of Algorithm 7.1 (Fig. 7.1). In this example, the size of the image to be reconstructed is 3×3 and $D = \{0, 1\}$. We use 3 projection

angles: horizontal, vertical and diagonal from top left to bottom right. The projection model is defined below setting $\kappa = 3$.

The initial approximate image $\boldsymbol{x}^{(0)}$ is shown in Fig. 7.2(a) and the 11 line sums which form the projection data are: horizontal: $h_1 = x_1 + x_2 + x_3$, $h_2 = x_4 + x_5 + x_6$ and $h_3 = x_7 + x_8 + x_9$; vertical: $v_1 = x_1 + x_4 + x_7$, $v_2 = x_2 + x_5 + x_8$ and $v_3 = x_3 + x_6 + x_9$ and diagonal: $d_1 = x_7$, $d_2 = x_4 + x_8$, $d_3 = x_1 + x_5 + x_9$, $d_4 = x_2 + x_6$ and $d_5 = x_3$. We leave the computation of each line sum for each iteration for the reader.

$x_1^{(0)} = 0.5$	$x_2^{(0)} = 0.8$	$x_3^{(0)} = 0.5$
$x_4^{(0)} = 0.5$	$x_5^{(0)} = 0.6$	$x_6^{(0)} = 0.7$
$x_7^{(0)} = 0.5$	$x_8^{(0)} = 0.4$	$x_9^{(0)} = 0.5$

(a) initial approx. solution $\boldsymbol{x}^{(0)}$

1	-1	0
-1	0	1
0	1	-1

(b) ghost image $\boldsymbol{y}^{(1)}$

0.8	0.5	0.5
0.2	0.6	1
0.5	0.7	0.2

(c) approximate solution $\boldsymbol{x}^{(1)}$

Figure 7.2: *Ghost image* $\boldsymbol{y}^{(1)}$ *together with* $\boldsymbol{x}^{(0)}$ *produces* $\lambda^{(1)} = 0.3$*, generating* $\boldsymbol{x}^{(1)} = \boldsymbol{x}^{(0)} + \lambda^{(1)}\boldsymbol{y}^{(1)}$.

In order to simplify the notation, we use the same symbols of the line sums to represent the sets L_i which form $G^{(t)}$, where $G^{(1)} = \{h_1, h_2, h_3, v_1, v_2, v_3, d_3\}$. A ghost image $\boldsymbol{y}^{(1)}$ is shown in Fig. 7.2(b), which has zero line sums for each $L_i \in G^{(1)}$. Therefore, for each $L_i \in G^{(1)}$ the respective line sums of $\boldsymbol{x}^{(0)}$ and $\boldsymbol{x}^{(1)}$ are equal.

For iteration 2, we have $G^{(2)} = \{h_1, h_3, v_1, v_2, d_3\}$ from Def. 1 and $\boldsymbol{y}_6^{(2)} = 0$ from Def. 2. As a consequence, the line sums h_2 and v_3 of the ghost image $\boldsymbol{y}^{(2)}$ are no longer restricted to zero. A possible ghost image $\boldsymbol{y}^{(2)}$ and the resulting $\boldsymbol{x}^{(2)}$ are shown in Fig. 7.3.

1	-1	0
-1	0	0
0	1	-1

(a) ghost image $\boldsymbol{y}^{(2)}$

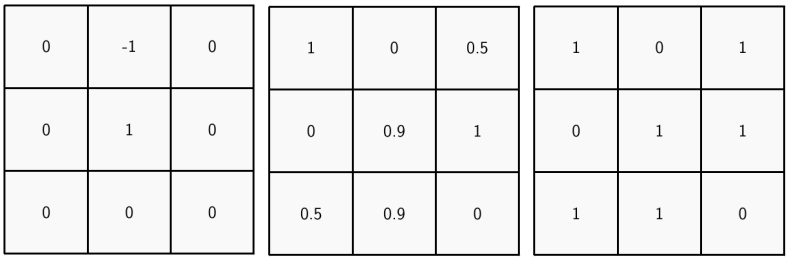
1	0.3	0.5
0	0.6	1
0.5	0.9	0

(b) approximate solution $\boldsymbol{x}^{(2)}$

Figure 7.3: *Ghost image* $\boldsymbol{y}^{(2)}$ *together with* $\boldsymbol{x}^{(1)}$ *produces* $\lambda^{(2)} = 0.2$*, generating* $\boldsymbol{x}^{(2)} = \boldsymbol{x}^{(1)} + \lambda^{(2)}\boldsymbol{y}^{(2)}$.

Notice that $I_D^{(3)} = \{1, 4, 6, 9\}$, based on $\boldsymbol{x}^{(2)}$. For iteration 3, we have $G^{(3)} = \{v_2\}$ and from Def. 2, the pixels $\boldsymbol{y}_2^{(3)}$, $\boldsymbol{y}_5^{(3)}$ and $\boldsymbol{y}_8^{(3)}$ are the only ones allowed to be non-zero. A ghost image $\boldsymbol{y}^{(3)}$ and the resulting image $\boldsymbol{x}^{(3)}$ are shown in Fig. 7.4.

As $G^{(4)} = \emptyset$, we have $\mathcal{N}(\boldsymbol{A}^{(4)}) = \{\boldsymbol{0}\}$ and Procedure 2 is used, obtaining $\bar{\boldsymbol{x}}$ by rounding the entries of $\boldsymbol{x}^{(3)}$ to the nearest element of D, after which the algorithm terminates. In this case,



(a) ghost image $\boldsymbol{y}^{(3)}$ (b) approximate solution $\boldsymbol{x}^{(3)}$ (c) approx. discrete solution $\bar{\boldsymbol{x}}$

Figure 7.4: *Ghost image $\boldsymbol{y}^{(3)}$ together with $\boldsymbol{x}^{(2)}$ produces $\lambda^{(3)} = 0.2$, generating $\boldsymbol{x}^{(3)} = \boldsymbol{x}^{(2)} + \lambda^{(3)}\boldsymbol{y}^{(3)}$.*

pixels having a value of 0.5 are rounded to 1. The final approximate solution $\bar{\boldsymbol{x}}$ is shown in Fig. 7.4(c).

7.5. The algorithm's proof

In this section we prove that the algorithm represented in the flowchart of Fig. 7.1 finds an image $\bar{\boldsymbol{x}} \in D^n$ such that the projection distance between the initial image $\boldsymbol{x}^{(0)}$ and $\bar{\boldsymbol{x}}$ is smaller than κd. First, we prove the requirements and implications of one iteration of Procedure 1.

Lemma 16. *Suppose that*

(H1) $\boldsymbol{x}^{(t-1)} \in [d_1, d_s]^n$ *is the reconstructed image that resulted from iteration* $t-1$*;*
(H2) $\boldsymbol{y}^{(t)} \neq \{\boldsymbol{0}\}$ *is a solution of* $\boldsymbol{A}^{(t)}\boldsymbol{y} = \boldsymbol{0}$*, for* $\boldsymbol{A}^{(t)}$ *given by Definition 2.*

Then after applying Procedure 1, the following statements are true:

(T1) $\left|I_D^{(t)}\right| > \left|I_D^{(t-1)}\right|$*;*
(T2) $x_j^{(t)} = x_j^{(t-1)} \quad$ *for all* $\quad x_j^{(t-1)} \in D$*;*
(T3) there is no $1 \leq j \leq n$ *such that there exists a* $d_i \in D$ *for which* $x_j^{(t-1)} < d_i < x_j^{(t)}$*;*
(T4) $\sum_{j \in L_i} w_{ij}(x_j^{(t)} - x_j^{(t-1)}) = 0 \quad$ *for all* $\quad L_i \in G^{(t)}$*;*
(T5) $\sum_{j \in L_i} w_{ij}\left|x_j^{(t)} - x_j^{(t-1)}\right| < \kappa d \quad$ *for all* $\quad L_i \notin G^{(t)}$*.*

Proof. Let $\boldsymbol{y}^{(t)} \neq \boldsymbol{0}$ such that $\boldsymbol{A}^{(t)}\boldsymbol{y}^{(t)} = \boldsymbol{0}$. Define $\boldsymbol{x}^{(t)} = \boldsymbol{x}^{(t-1)} + \lambda^{(t)}\boldsymbol{y}^{(t)}$ with $\lambda^{(t)}$ the smallest positive scalar λ for which $x_{\hat{j}}^{(t-1)} + \lambda y_{\hat{j}}^{(t)} \in D$ for some $\hat{j} \in \bar{I}_D^{(t)}$. This definition of $\boldsymbol{x}^{(t)}$ follows exactly the computation of Procedure 1. As $\boldsymbol{x}_{\hat{j}}^{(t)} \in D$ and none of the entries that were already in D are modified, statements (T1) and (T2) are true. By definition of $\lambda^{(t)}$, statement (T3) is also true.

Statement (T4) follows from the definition of $\boldsymbol{A}^{(t)}$ and statement (T2):

$$\sum_{j \in L_i} w_{ij}(x_j^{(t)} - x_j^{(t-1)}) = \lambda \sum_{j \in L_i \cap \bar{I}_D^{(t)}} w_{ij} y_j^{(t)} + 0 \sum_{j \in L_i \cap I_D^{(t)}} w_{ij} = 0 \quad \text{for all} \quad L_i \in G^{(t)}.$$

Also, (T3) implies $\left|x_j^{(t)} - x_j^{(t-1)}\right| < d$, for all $1 \le j \le n$, and since $\sum_{j\in L_i \cap \bar{I}_D^{(t)}} w_{ij} < \kappa$ for $L_i \notin G^{(t)}$ we have

$$\sum_{j\in L_i} w_{ij}\left|x_j^{(t)} - x_j^{(t-1)}\right| < d \sum_{j\in L_i \cap \bar{I}_D^{(t)}} w_{ij} + 0 \sum_{j\in L_i \cap I_D^{(t)}} w_{ij} < \kappa d \quad \text{for} \quad L_i \notin G^{(t)}$$

for all $i = 1,\ldots,m$. □

Whenever $\boldsymbol{y}^{(t)} = \boldsymbol{0}$ is the unique ghost image of iteration t, Procedure 1 has no effect. The next Lemma is an intermediate result for the case that Procedure 2 is required.

Lemma 17. *Suppose that $\mathcal{N}(\boldsymbol{A}^{(t)}) = \{\boldsymbol{0}\}$ for $\boldsymbol{A}^{(t)}$ given by Definition 2. Then*

$$\sum_{j\in L_i \cap \bar{I}_D^{(t)}} w_{ij} \le \kappa \qquad \textit{for} \qquad i = 1,\ldots,m.$$

Proof. As equations (7.3) already define $y_j^{(t)} = 0$ for a set of j's, let $\hat{\boldsymbol{A}}^{(t)} = (\hat{a}_{ij}^{(t)})$ be the matrix associated with equations (7.2). Since $\mathcal{N}(\hat{\boldsymbol{A}}^{(t)}) = \{\boldsymbol{0}\}$, the number of linear independent rows of $\hat{\boldsymbol{A}}^{(t)}$ is equal to the number of columns of $\hat{\boldsymbol{A}}^{(t)}$. We can eliminate rows which are linear combinations of others obtaining a square matrix, say of size $r \times r$. Supposing $r > 0$, the definition of $\kappa = \|\boldsymbol{W}\|_1$ yields

$$\sum_{i=1}^{r} \hat{a}_{ij}^{(t)} \le \kappa \quad \text{for } j = 1,\ldots,r, \quad \text{implying} \quad \sum_{j=1}^{r}\sum_{i=1}^{r} \hat{a}_{ij}^{(t)} \le r\kappa \tag{7.4}$$

and the definition of $G^{(t)}$ gives

$$\sum_{j=1}^{r} \hat{a}_{ij}^{(t)} \ge \kappa \quad \text{for } i = 1,\ldots,r, \quad \text{implying} \quad \sum_{i=1}^{r}\sum_{j=1}^{r} \hat{a}_{ij}^{(t)} \ge r\kappa. \tag{7.5}$$

From statements (7.4) and (7.5) we conclude that $\sum_{i=1}^{r}\sum_{j=1}^{r} \hat{a}_{ij}^{(t)} = r\kappa$ and $\sum_{j=1}^{r} \hat{a}_{ij}^{(t)} = \kappa$. Then

$$\sum_{j\in L_i \cap \bar{I}_D^{(t)}} w_{ij} = \sum_{j\in L_i \cap \bar{I}_D^{(t)}} \hat{a}_{ij}^{(t)} = \kappa \quad \text{for} \quad L_i \in G^{(t)}$$

for $i = 1,\ldots,m$.

If $r = 0$, we have $G^{(t)} = \emptyset$ and it follows from the definition of $G^{(t)}$ that $\sum_{j\in L_i \cap \bar{I}_D^{(t)}} w_{ij} < \kappa$ for $L_i \notin G^{(t)}$. □

Theorem 20. *Let $\boldsymbol{x}^{(0)} \in [d_1, d_s]^n$ be a given image such that $\boldsymbol{W}\boldsymbol{x}^{(0)} = \boldsymbol{p}$. Then, after termination of Algorithm 7.1 (Fig. 7.1), we obtain a vector $\bar{\boldsymbol{x}} \in D^n$ such that*

$$\|\boldsymbol{W}\bar{\boldsymbol{x}} - \boldsymbol{p}\|_\infty < \kappa d.$$

Proof. Procedure 2 is used as the last iteration of the algorithm generating $\bar{\boldsymbol{x}} \in D^n$. Hence we have to prove that the projection error accumulated during the iterations satisfies $\|\boldsymbol{W}\bar{\boldsymbol{x}} - \boldsymbol{p}\|_\infty < \kappa d$.

Lemma 16 ensures that between any two iterations $\hat{t}-1$ and $\hat{t}$ of Procedure 1 there is no $1 \le j \le n$ such that there exists a $d_i \in D$ for which $x_j^{(\hat{t}-1)} < d_i < x_j^{(\hat{t})}$, implying that there is no $1 \le j \le n$ such that there exists a $d_i \in D$ for which $x_j^{(0)} < d_i < x_j^{(t)}$.

Suppose that iteration $t+1$ uses Procedure 2: set $\bar{x}_j$ as the element from D which is nearest to $x_j^{(t)}$ for all $j \in \bar{I}_D^{(t+1)}$ and set $\bar{x}_j = x_j^{(t)}$ for all $j \in I_D^{(t+1)}$. Then we can conclude that $|\bar{x}_j - x_j^{(0)}| < d$ for $j = 1, \ldots, n$.

From Lemma 16, it is easy to check that after t iterations of Procedure 1 we have

$$\sum_{L_i} w_{ij}(x_j^{(t)} - x_j^{(0)}) = 0 \quad \text{for} \quad L_i \in G^{(t)}.$$

After applying Procedure 2, Lemma 17 provides $\sum_{j \in L_i \cap \bar{I}_D^{(t+1)}} w_{ij} \le \kappa$ for $i = 1, \ldots, m$, which yields

$$|\sum_{L_i} w_{ij}(\bar{x}_j - x_j^{(0)})| < \kappa d \quad \text{for} \quad L_i \in G^{(t)}. \tag{7.6}$$

Also from Lemma 16, for each $i = 1, \ldots, m$ such that, $L_i \notin G^{(t)}$, there exist disjoint sets $A, B \notin G^{(t)}$ such that $L_i = A \cup B$ with $\sum_{j \in A} w_{ij}(x_j^{(t)} - x_j^{(0)}) = 0$ and $\sum_{j \in B} w_{ij} < \kappa$. These properties are still valid after applying Procedure 2 yielding

$$|\sum_{L_i} w_{ij}(\bar{x} - x_j^{(0)})| = |\sum_{j \in B} w_{ij}(\bar{x} - x_j^{(0)})| < \kappa d \quad \text{for} \quad L_i \notin G^{(t)}. \tag{7.7}$$

Therefore, from Eqs. (7.6) and (7.7), it follows that $\bar{\boldsymbol{x}} \in D^n$ and $\|\boldsymbol{W}\bar{\boldsymbol{x}} - \boldsymbol{p}\|_\infty < \kappa d$.

□

7.6. A threshold variation

In some cases, after an iteration step, there are pixels with current values very close to a value in the set D. The use of a threshold parameter $\tau < d$ for rounding some pixels can speed up the algorithm, as it would need fewer iterations to make $\boldsymbol{x}^{(0)}$ converge to $\bar{\boldsymbol{x}} \in D^n$. As a drawback, the use of threshold may increase the difference in the projections. Moreover, the bound on the projection distance becomes dependent on the image size. Our variant of the reconstruction algorithm that uses such a threshold only requires a modification in Procedure 1. We denote the modified procedure as Procedure 3.

Theorem 21. *Let $\boldsymbol{x}^{(0)} \in [d_1, d_s]^n$ be a given image such that $\boldsymbol{W}\boldsymbol{x}^{(0)} = \boldsymbol{p}$ and $\tau < d$. Then, after termination of the Algorithm 7.1 (Fig. 7.1) using Procedure 3 instead of Procedure 1, we obtain a vector $\bar{\boldsymbol{x}} \in D^n$ such that*

$$\|\boldsymbol{W}\bar{\boldsymbol{x}} - \boldsymbol{p}\|_\infty < \kappa d + (\|\boldsymbol{W}\|_\infty - \kappa)\tau.$$

Procedure 3 *Calculate* $\boldsymbol{x}^{(t)}$

$\lambda_{\min} \leftarrow \infty$
for $j \in \bar{I}_D^{(t)}$ **do**
  Compute the smallest $\lambda > 0$ s.t. $x_j^{(t-1)} + \lambda y_j^{(t)} \in D$
  if $\lambda < \lambda_{\min}$ **then**
    $\lambda_{\min} \leftarrow \lambda$
  end if
end for
$\boldsymbol{x}^{(t)} \leftarrow \boldsymbol{x}^{(t-1)} + \lambda_{\min} \boldsymbol{y}^{(t)}$
for $j \in \{1, \ldots, n\}$ **do**
  if there exist $d_i \in D$ such that $|x_j^{(t)} - d_i| < \tau$ for $i = 1, \ldots, s$ **then**
    $x_j^{(t)} \leftarrow d_i$
  end if
end for

Proof. If $|x_j^{(t)} - d_l| \leq \tau$ for some $l = 1, \ldots, s$, then $|x_j^{(t+1)} - x_j^{(t)}| \leq \tau$. This implies that the projection difference of the i-th component increases by, at most, τ for each thresholded pixel in equation i, in one iteration step. Since $\tau < d$, the worst case happens when for an equation i, there exist sets of indices A and B such that $\sum_{j \in A} w_{ij} = \kappa$ and $\sum_{j \in B} w_{ij} = \|\boldsymbol{W}\|_\infty - \kappa$, and all the pixels defined by B are thresholded, but not the ones defined by A. After applying Procedure 2, we have $\sum_{j=1}^{n} w_{ij} |\bar{x}_j - x_j^{(0)}| < \kappa d + (\|\boldsymbol{W}\|_\infty - \kappa)\tau$. □

7.7. Computations

In this section, we will cover two important computational aspects of our approach: how to compute a suitable initial solution and how to compute a ghost image for the iteration steps. In both cases, we outline a particular method for computing an image. Many alternative methods exist for solving the corresponding mathematical problems, some of which may yield a more efficient algorithm.

7.7.1. Initial solution $\boldsymbol{x}^{(0)}$

The discrete reconstruction algorithm presented in this chapter needs an initial image $\boldsymbol{x}^{(0)} \in [d_1, d_s]^n$ which can be achieved, e.g., by an algorithm presented as a *Norm minimization over an interval* algorithm in [13]. Algorithm 4 is the version of the algorithm that we have used in the experiments. It minimizes $\frac{1}{2}\|\boldsymbol{z}\|^2$ subject to $\boldsymbol{W}\boldsymbol{z} = \boldsymbol{p}$ and $d_1 \leq z_j \leq d_s$ for all $j = 1, \ldots, n$, which guarantees $\boldsymbol{x}^{(0)} \in [d_1, d_s]^n$. This algorithm is a variation of ART [21] (ART is also known as Kaczmarz method [25, 38]).

Since we want to guarantee a small projection distance between $\bar{\boldsymbol{x}}$ and the unknown original object from which $\boldsymbol{p}$ was formed, it is necessary to have a small distance between $\boldsymbol{W}\boldsymbol{x}^{(0)}$ and $\boldsymbol{p}$. In order to do so, the stopping criteria of Algorithm 4 is given by $\|\boldsymbol{W}\boldsymbol{z} - \boldsymbol{p}\|_\infty < \varepsilon$, where ε is a positive constant close to zero.

Algorithm 4 *Compute* $\boldsymbol{x}^{(0)}$

$t \leftarrow 0$, $\boldsymbol{z} \leftarrow \boldsymbol{0}_n$, $\hat{\boldsymbol{z}} \leftarrow \boldsymbol{0}_n$
while $\|\boldsymbol{W}\boldsymbol{z}^{(t)} - \boldsymbol{p}\|_\infty > \varepsilon$ **do**
  $i \leftarrow (t \bmod m) + 1$
  $z_j \leftarrow z_j + \frac{p_i - \sum_{j=1}^{n} w_{ij}\hat{z}_j}{\sqrt{\sum_{j=1}^{n} w_{ij}^2}} w_{ij}$ for $j = 1, \ldots, n$
  for $j \in \{1, \ldots, n\}$ **do**

$$\hat{z}_j \leftarrow \begin{cases} d_1 & \text{if} \quad z_j < d_1 \\ z_j & \text{if} \quad d_1 \leq z_j \leq d_s \\ d_s & \text{if} \quad z_j > d_s \end{cases}$$

  end for
  $t \leftarrow t + 1$
end while
$\boldsymbol{x}^{(0)} \leftarrow \boldsymbol{z}$

7.7.2. Non-null ghost images

If $\mathcal{N}(\boldsymbol{A}^{(t)}) \neq \{\boldsymbol{0}\}$, Procedure 1 is executed. To this end, we must find a non-null solution of a homogeneous linear system

$$\boldsymbol{A}\boldsymbol{z} = \boldsymbol{0}. \tag{7.8}$$

To solve this problem we have used an iterative method called CGLS (Conjugate Gradient Least Squares) [39]. The CGLS algorithm requires an initial guess $\boldsymbol{z}^{(0)}$. Apart from numerical errors, applying the CGLS algorithm to the system (7.8) results, after convergence, in the computation of $\boldsymbol{z}_{LS} = (\boldsymbol{I} - \boldsymbol{A}^\dagger \boldsymbol{A})\boldsymbol{z}^{(0)}$, where $\boldsymbol{A}^\dagger$ is the *Moore-Penrose pseudo inverse* of $\boldsymbol{A}$, see [11, 12, 47]. The CGLS algorithm computes $\boldsymbol{z}_{LS}$ without explicitly computing $\boldsymbol{A}^\dagger$.

The matrix $(\boldsymbol{I} - \boldsymbol{A}^\dagger \boldsymbol{A})$ orthogonally projects $\boldsymbol{z}^{(0)}$ onto $\mathcal{N}(\boldsymbol{A})$. As a result, $\boldsymbol{z}_{LS} = \boldsymbol{0}$ if and only if $\boldsymbol{z}^{(0)} \perp \mathcal{N}(\boldsymbol{A})$. Hence, randomly selecting $\boldsymbol{z}^{(0)}$ will almost certainly yield a $\boldsymbol{z}_{LS} \neq \boldsymbol{0}$.

In practice, several algorithms use the relative residual as a parameter for the stopping criteria. The relative residual is given by the current iteration residual norm divided by the norm of the right-hand side. When the right-hand side is the null-vector, the relative residual is computed as infinity and this algorithm can never identify convergence. To overcome this problem, instead of applying the CGLS algorithm to (7.8), we apply it to the linear system

$$\boldsymbol{A}\boldsymbol{z} = \boldsymbol{b}, \tag{7.9}$$

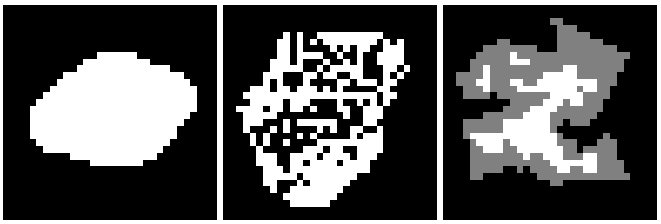
with $\boldsymbol{b} = -\boldsymbol{A}\boldsymbol{z}^{(0)}$. When applying CGLS to Eqs. (7.9), we select the null-vector as initial guess obtaining, therefore, $\boldsymbol{z}^* = \boldsymbol{A}^\dagger \boldsymbol{b}$, the minimum norm solution of system (7.9) [11, 12, 47]. Defining $\boldsymbol{y} = \boldsymbol{z}^{(0)} + \boldsymbol{z}^*$, it satisfies Eqs. (7.8): $\boldsymbol{A}\boldsymbol{y} = \boldsymbol{A}\boldsymbol{z}^{(0)} + \boldsymbol{A}\boldsymbol{z}^* = \boldsymbol{A}\boldsymbol{z}^{(0)} + \boldsymbol{b} = \boldsymbol{0}$.

7.8. Numerical experiments

A series of experiments was performed to investigate the quality of images reconstructed by the algorithm presented and the resulting bounds on the projection distance, for a range of

projection angles and images. The experiments are all based on simulated projection data obtained by computing the projections of the test images (so-called *phantoms*) in Fig. 7.5:

(a) Phantom 1, 128×128 (b) Phantom 2, 128×128 (c) Phantom 3, 128×128



(d) Phantom 1, 32×32 (e) Phantom 2, 32×32 (f) Phantom 3, 32×32

Figure 7.5: *Original phantom images used for the experiments.*

Phantom 1 represents a very simple, nearly convex shaped object;

Phantom 2 was constructed from a micro-CT image of a rat bone acquired with a SkyScan 1072 cone-beam micro-CT scanner.

Phantom 3 represents an object with three grey levels and fairly complex boundaries.

As a projection model, we focus on a model for a *continuous* representation of the object, the *strip model* [33, section 7.4.1]. In the strip model, a projection is computed by considering a set of parallel strips in a given direction. For each strip we compute the weighted sum of all the pixels which intersect that strip. Each weight w_{ij} equals the intersection area of the strip and the pixel, see Fig. 7.6.

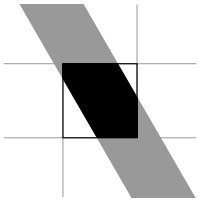


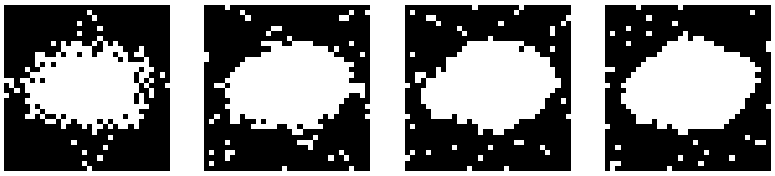
Figure 7.6: *The contribution weight of each pixel to each projection component is given by the intersection area between the pixel and the respective strip.*

7.8.1. Reconstruction comparison

In this section we compare two variations of the reconstruction algorithm: (i) Algorithm 7.1 (Fig. 7.1), (ii) the thresholded version of Alg. 7.1 presented in Section 7.6. For the two algorithms mentioned above, we compare quality of the images reconstructed as well as reconstruction time required for each one of them. Also, the quality of the initial solution $\boldsymbol{x}^{(0)}$ directly affects the final reconstruction. We tested different approximate solutions $\boldsymbol{x}^{(0)}$.

The quality of the initial approximate solution $\boldsymbol{x}^{(0)}$

The quality of the image reconstructed by the presented algorithms depends on the quality of the initial solution $\boldsymbol{x}^{(0)}$ which was computed by the algorithm presented in section 7.7.1. Fig. 7.7 presents reconstructions of phantom image 7.5(d), using the standard algorithm (Fig. 7.1), for different number of projection angles. For $\boldsymbol{p}$ being the vector of projections of phantom 7.5(d), Fig. 7.7 shows reconstructions in which the initial approximate solution satisfies $\|\boldsymbol{W}\boldsymbol{x}^{(0)} - \boldsymbol{p}\|_\infty \leq \varepsilon$ for $\varepsilon \in \{10^{-1}, 5\times 10^{-1}\}$. Since $\boldsymbol{W}\boldsymbol{x}^{(0)} \neq \boldsymbol{p}$, the reconstruction algorithm 7.1 can only guarantee $\|\boldsymbol{W}\bar{\boldsymbol{x}} - \boldsymbol{p}\|_\infty < \kappa d + \varepsilon$. If a threshold parameter τ is included, then $\|\boldsymbol{W}\bar{\boldsymbol{x}} - \boldsymbol{p}\|_\infty < \kappa d + (\|\boldsymbol{W}\|_\infty - \kappa)\tau + \varepsilon$.



(a) $\varepsilon = 5\times 10^{-1}$ and number of projection angles: left: 2; middle-left: 6; middle-right: 8; right: 12.

(b) $\varepsilon = 10^{-1}$ and number of projection angles: left: 2; middle-left: 6; middle-right: 8; right: 12.

Figure 7.7: *Reconstructions of phantom 7.5(d) of dimensions* 32×32 *using Algorithm 7.1.*

In Fig. 7.7, it is possible to see white pixels (pixels assigned with a value larger than zero) in regions for which a projected strip that intersects these pixels has a projection value of 0. These line sums accumulate detectable errors and since the error per projection is limited to κd, the error in a region where it is not detectable will diminish. After the reconstruction algorithm, post-processing of the reconstructed image may correct these easily identifiable wrong pixels.

The quality of the reconstruction $\bar{x}$

Reconstructed images from the thresholded version of Algorithm 7.1 for phantoms 7.5(e) and 7.5(f) are displayed in Figs. 7.8 and 7.9 for $\varepsilon = 10^{-1}$ and $\tau \in \{0, \frac{1}{32}, \frac{1}{\sqrt{32}}\}$. For phantoms 7.5(b) and 7.5(c), the reconstructed images from the thresholded version of Algorithm 7.1 with $\tau \in \{\frac{1}{128}, \frac{1}{\sqrt{128}}\}$, are displayed in Fig. 7.10.

We have measured the projection distance (Pd) of the previously presented reconstructions and compared these with their respective bound (B) as can be seen in Figs. 7.11 and 7.12. Also, Figs. 7.11 and 7.12 present the *Image distance* (Id) defined as the Euclidean distance between the reconstructed image and the respective phantom image. Notice that despite the bound $\kappa d + (\|\boldsymbol{W}\|_\infty - \kappa)\tau + \varepsilon$ might increase with the number of projection angles, the quality of the reconstructed images improves, in general. It also means that the projection distance of the reconstructed images decreases with increasing number of projections angles.

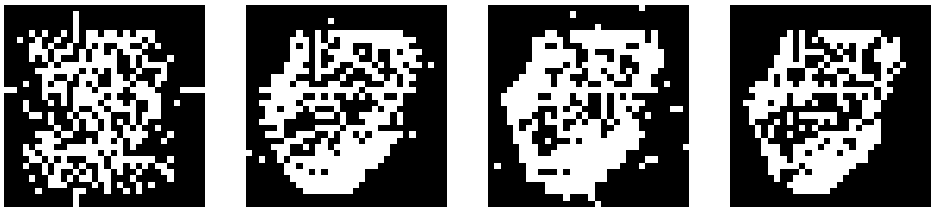
We remark that the key advantage of the proposed algorithm is the guaranteed error bound of the resulting reconstruction. Alternative algorithms, which do not yield such a bound, can yield more accurate reconstructions in practice, using less computation time. As an example, we refer to the DART algorithm in [9], where similar phantoms are used in the experiments. It can be seen in Fig. 7 of [9] that Phantom 2 and 3 (corresponding to Phantom 5 and 7 in [9]) are reconstructed by DART far more accurately from few projections compared to our proposed method, yet without any guarantee on the reconstruction error.

Running time

The algorithm variants with different threshold parameters compared in this section have different running times, which can vary from 0.1 seconds for the fastest run up to 10's of minutes for the slowest runs. All experiments were run on a workstation PC using a single core of an Intel Core-i5 CPU at 2.8GHz. As the computation time is machine dependent, we display a *relative time*, which is computed by dividing the time needed for a reconstruction by the time of the fastest reconstruction among all of the runs presented in the experiments. The relative time comparison between the reconstruction algorithms can be seen in Table 7.1. For the phantoms in Fig. 7.5(a), 7.5(b), and 7.5(c), the running time was prohibitively large for the basic algorithm, which does not use a threshold. The timing results for these cases are not included in the table.

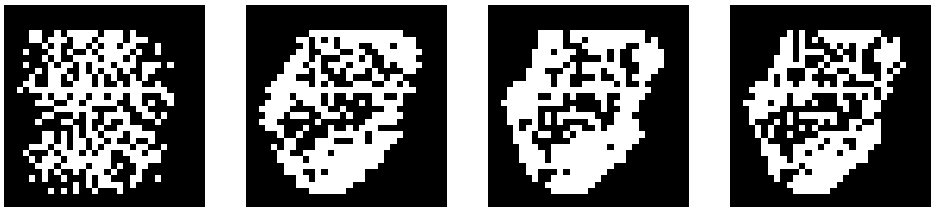
We have measured the number of iterations of the algorithm in each experiment and, in general, it decreases with increasing number of projection angles. In the experiments, the maximum number of iterations required was less than 60% of the number of image pixels.

The key bottleneck in the computation complexity of the algorithm is the fact that a large system of equations (up to the same order as the tomography problem itself) must be solved to obtain a ghost image, and that this computation will have to be performed many times (at most once for each pixel). As there is a stochastic component in the computation of the ghost images, and as the particular computational route followed by the algorithm depends strongly on the input projections, a meaningful (somewhat sharp) complexity analysis of the algorithm is highly challenging. We consider the algorithm proposed here as



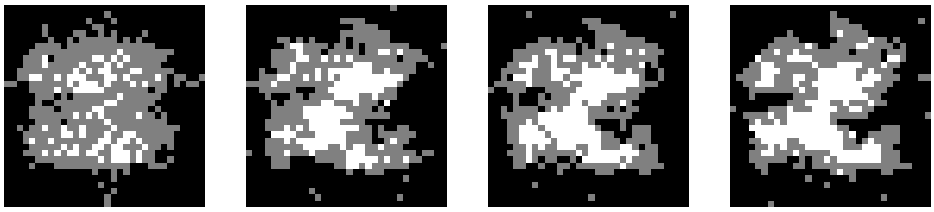
(a) Algorithm 7.1 with $\tau = 0$. Number of projection angles: left: 2; mid-left: 6; mid-right: 10; right: 14.

(b) Algorithm 7.1 with $\tau = \frac{1}{32}$. Number of projection angles: left: 2; mid-left: 6; mid-right: 10; right: 14.

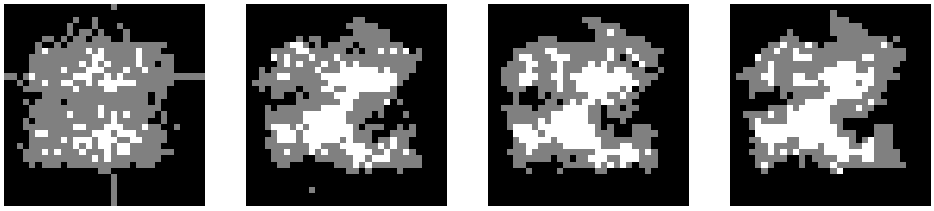


(c) Algorithm 7.1 with $\tau = \frac{1}{\sqrt{32}}$. Number of projection angles: left: 2; mid-left: 6; mid-right: 10; right: 14.

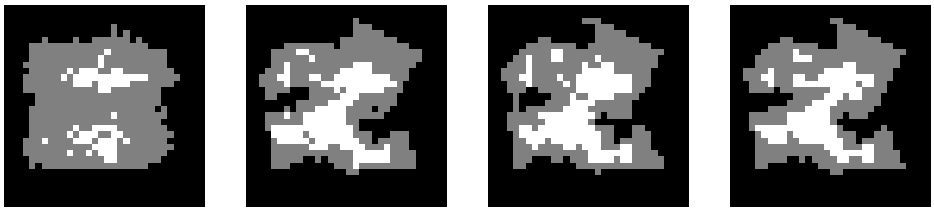
Figure 7.8: *Reconstructions of phantom 7.5(e) of dimensions* 32×32 *and* $\varepsilon = 10^{-1}$.



(a) Algorithm 7.1 with $\tau = 0$. Number of projection angles: left: 2; mid-left: 6; mid-right: 8; right: 12.



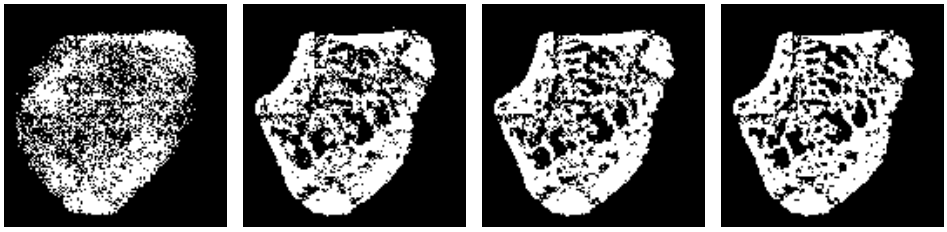
(b) Algorithm 7.1 with $\tau = \frac{1}{32}$. Number of projection angles: left: 2; mid-left: 6; mid-right: 8; right: 12.



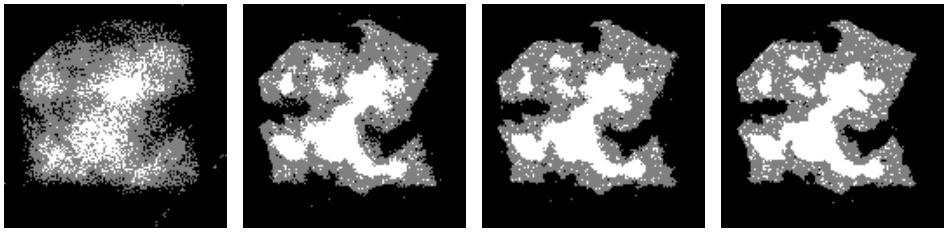
(c) Algorithm 7.1 with $\tau = \frac{1}{\sqrt{32}}$. Number of projection angles: left: 2; mid-left: 6; mid-right: 8; right: 12.

Figure 7.9: *Reconstructions of phantom 7.5(f) of dimensions* 32×32 *and* $\varepsilon = 10^{-1}$.

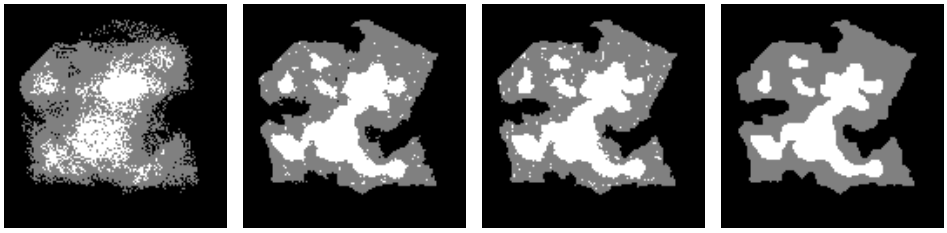
(a) Phantom 7.5(b). Alg. 7.1 with $\tau = \frac{1}{128}$. N. proj. angles: left: 4; mid-left: 16; mid-right: 20; right: 24.



(b) Phantom 7.5(b). Alg. 7.1 with $\tau = \frac{1}{\sqrt{128}}$. N. proj. angles: left: 4; mid-left: 16; mid-right: 20; right: 24.

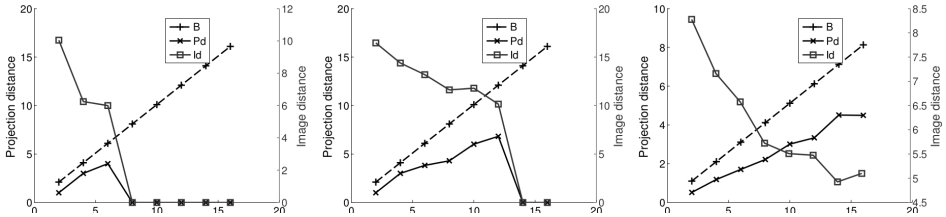


(c) Phantom 7.5(c). Alg. 7.1 with $\tau = \frac{1}{128}$. N. proj. angles: left: 4; mid-left: 16; mid-right: 20; right: 24.

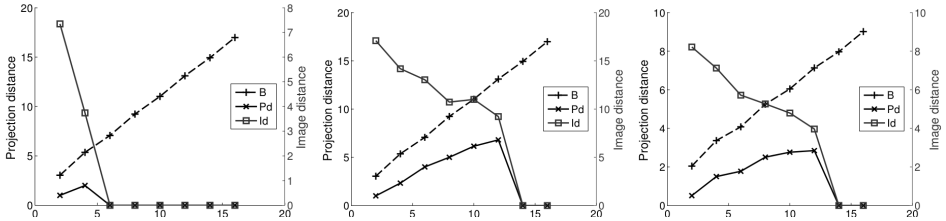


(d) Phantom 7.5(c). Alg. 7.1 with $\tau = \frac{1}{\sqrt{128}}$. N. proj. angles: left: 4; mid-left: 16; mid-right: 20; right: 24.

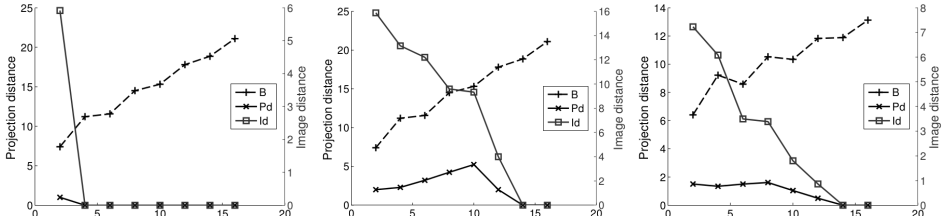
Figure 7.10: *Reconstructions of phantoms 7.5(b) and 7.5(c) of dimensions* 128×128 *and* $\varepsilon = 10^{-1}$.



(a) Algorithm 7.1.

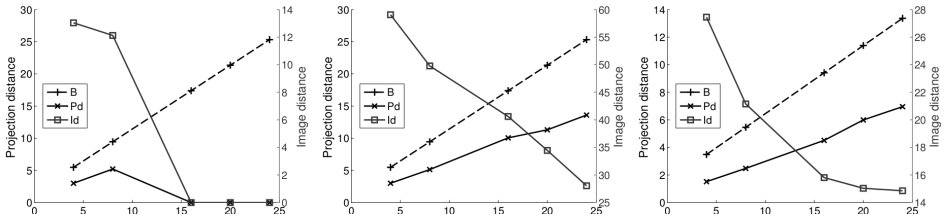


(b) Algorithm 7.1 with threshold $\tau = \frac{1}{32}$.

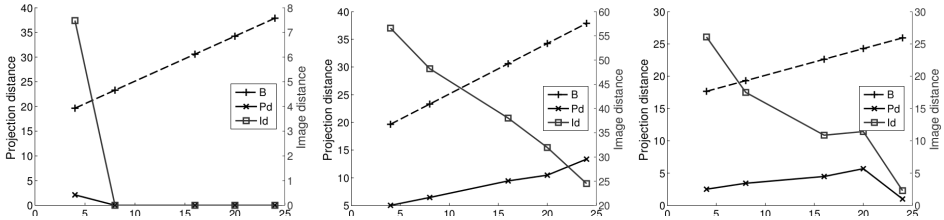


(c) Algorithm 7.1 with threshold $\tau = \frac{1}{\sqrt{32}}$.

Figure 7.11: *Comparison between the projection distance bound (B) and actual projection distance (Pd) measured in the left vertical axis for increasing number of projection angles. The Image distance (Id) uses the right vertical axis. From left to right: Phantom images 7.5(d), 7.5(e) and 7.5(f).*



(a) Algorithm 7.1 with threshold $\tau = \frac{1}{128}$.



(b) Algorithm 7.1 with threshold $\tau = \frac{1}{\sqrt{128}}$.

Figure 7.12: *Comparison between the projection distance bound (B) and actual projection distance (Pd) measured in the left vertical axis for increasing number of projection angles. The Image distance (Id) uses the right vertical axis. From left to right: Phantom images 7.5(a), 7.5(b) and 7.5(c).*

a proof-of-concept method that requires further optimization and analysis to be useful for reconstructing large images.

7.9. Outlook and conclusions

In this chapter, we have presented a reconstruction algorithm which computes an image that has only pixel values from a given finite set, and for which the projection distance to the unknown ground truth object is bounded. Contrary to alternative methods, which often perform well in practice but do not come with guarantees, our approach is specifically designed to yield a reconstruction for which the projections are provably close to the given projection data. Our experimental results for simulated phantom images demonstrate that the algorithm not only computes images that approximately match the given projections, but also yields reconstructions that resemble the unknown original image, even if only a small number of projections are used. The techniques of using ghost images and thresholding employed in our algorithm are quite generic and leave a large degree of freedom in parameter selection. For example, the particular switching element considered in a step of Procedure 1 is currently determined randomly, while it could also be chosen depending on features of the current reconstruction.

In its current form, the proposed algorithm is neither optimized for speed of computation, nor for quality of the reconstructed image. In future work, we intend to optimize the

N. proj.	4	16	24
Alg. 7.1, $\tau = \frac{1}{128}$	5387	34.8	34.7
Alg. 7.1, $\tau = \frac{1}{\sqrt{128}}$	817	35.6	35.6

(a) Phantom 7.5(a).

N. proj.	4	16	24
Alg. 7.1, $\tau = \frac{1}{128}$	5×10^5	2×10^5	7×10^4
Alg. 7.1, $\tau = \frac{1}{\sqrt{128}}$	3765	1881	423

(b) Phantom 7.5(b)

N. proj.	4	16	24
Alg. 7.1, $\tau = \frac{1}{128}$	3×10^5	1×10^5	1×10^5
Alg. 7.1, $\tau = \frac{1}{\sqrt{128}}$	1361	428	79.1

(c) Phantom 7.5(c).

N. projections	2	10	16
Alg. 7.1, $\tau = 0$	37.7	3.25	1
Alg. 7.1, $\tau = \frac{1}{32}$	12.2	1	1
Alg. 7.1, $\tau = \frac{1}{\sqrt{32}}$	3.7	1	1

(d) Phantom 7.5(d).

N. projections	2	10	16
Alg. 7.1, $\tau = 0$	83.7	59.8	9.5
Alg. 7.1, $\tau = \frac{1}{32}$	36.8	15.7	1
Alg. 7.1, $\tau = \frac{1}{\sqrt{32}}$	5.2	1.8	1

(e) Phantom 7.5(e).

N. projections	2	10	16
Alg. 7.1, $\tau = 0$	82.7	91	61.1
Alg. 7.1, $\tau = \frac{1}{32}$	18.7	10	1.6
Alg. 7.1, $\tau = \frac{1}{\sqrt{32}}$	1.8	1	1

(f) Phantom 7.5(f).

Table 7.1: *Relative reconstruction time for Alg. 7.1 with different thresholds.* $\varepsilon = 10^{-1}$

algorithm, and make it more robust, to the extent where it can be applied to larger images, and to experimental data.

Chapter 8

A method for feature detection in binary tomography

This chapter (with minor modifications) has been published as: W. Fortes and K. J. Batenburg. A method for feature detection in binary tomography. In Gonzalez Diaz, Rocio; Jimenez, Maria Jose; Medrano, Belen (eds.) DGCI 2013, volume 7749 of LNCS, pages 372-382. Heidelberg, 2013. Springer.

8.1. Introduction

Binary tomography deals with the problem of reconstructing a binary image from its projections. While accurate image reconstruction requires availability of a large number of projections for general grey scale images, knowledge about the fact the unknown original image is binary can drastically reduce the number of projection angles needed for a detailed reconstruction in some cases.

A range of algorithms have been proposed for binary tomography [6,9,25,40]. Although each of these methods has demonstrated the ability to compute accurate reconstructions from a small number of projections in certain cases, none of these methods offer a *guarantee* that the reconstructed image is identical, or even similar to the unknown original image. In fact, one can state that giving such a guarantee will be impossible in general, as the reconstruction problem in binary tomography is known to be inherently *unstable*: a small change in the projection data can lead to a dramatic change in the (unique) reconstruction [3, 4, 46]. Moreover, several constructions are known for so-called *switching components*: binary images in which a selected set of zeros and ones can be interchanged, leading to a different image having the same projections [22, 27].

Even in cases when insufficient information is available to compute an accurate reconstruction of the complete image, it may still be possible to answer certain *questions* about the original image, or to determine certain *features* of it. In [19], it was shown that connectivity and convexity properties can be derived – to some extent – directly from the projection data.

It can also be desirable to know whether a certain boundary or homogeneous region can possibly exist in the unknown image, or not.

Even though finding a binary solution of the reconstruction problem is typically hard, it is often easier to prove that a solution *cannot* exist. For example, if the projections do not satisfy certain consistency conditions, a solution will certainly not exist. General consistency conditions for the Radon transform are presented in [24], while a detailed analysis of consistency conditions for the grid model in discrete tomography can be found in [43]. A particular condition for the existence of binary solutions is given in Chapter 2, which will be used and extended throughout the present chapter.

In this chapter, we extend the general idea of consistency to the detection whether or not certain *substructures* can exist in the original image. We present a computational technique for discovering the possible presence of certain features (e.g., blobs, edges). For each feature, a *probe* structure is defined, which can detect that particular feature. Based on an analysis of the existence of binary solutions of the reconstruction problem, our technique can prove, in certain cases, if the probed feature *cannot* exist in a given region of the original image. Our approach is independent of a particular reconstructed image or reconstruction method.

This chapter is structured as follows. In Section 8.2, the basic model and notation are introduced. In Section 8.3, the basic idea of a *probe image* is presented and formally defined. Section 8.4 covers various algorithms that can be used to prove – in certain cases – that a given probe image cannot be present in the unknown original image. Section 8.5 presents a series of simulation experiments that was performed to obtain a first assessment of the capabilities of the proposed method. Conclusions are drawn in Section 8.6.

8.2. Basic notation and model

Throughout the discrete tomography literature, several imaging models have been considered: the grid model, the strip model, the line model, etc. [33, section 7.4.1]. In this chapter we focus on the strip model, but our approach can be used for other projection models as well.

In the strip model, a projection is computed by considering a set of parallel strips in a given direction and for each strip computing the weighted sum of all the pixels which intersect that strip with a weight equal to the intersection area of the strip and the pixel (see Fig. 8.1).

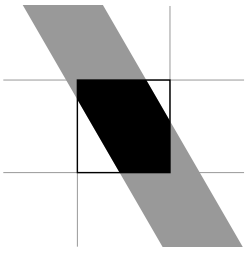


Figure 8.1: *The strip model.*

We now define some general notation. An *image* is represented by a vector $\boldsymbol{x} = (x_i) \in \mathbb{R}^n$. We refer to the entries of $\boldsymbol{x}$ as *pixels*, which correspond to unit squares in the strip model. Throughout this chapter we assume that all images are square, consisting of c rows and c columns, where $n = c^2$. A *binary image* corresponds with a vector $\bar{\boldsymbol{x}} \in \{0,1\}^n$.

For a given set of k projection directions, the *projection map* maps an image $\boldsymbol{x}$ to a vector $\boldsymbol{p} \in \mathbb{R}^m$ of *projection data*, where m denotes the total number of line measurements. As the projection map is a linear transformation, it can be represented by a matrix $\boldsymbol{W} = (w_{ij}) \in \mathbb{R}^{m \times n}$,

Figure 8.2: *Two probe images. The pixels are colored as follows: black (0), grey (?), white (1). Left: homogeneous white region; right: horizontal edge at the bottom of a white region.*

called the *projection matrix*. Entry w_{ij} represents the weight of the contribution of x_j to projected line i. Note that for the strip model its entries are real values in $[0,1]$. From this point on, we assume that the projection matrix has the property that $\sum_{i=1}^{m} w_{ij} = k$ for all $j = 1,\ldots,n$. This property is satisfied for the strip projection model, as the total pixel weight for each projection angle is equal to the area of a pixel, which is 1.

The *general reconstruction problem* consists of finding a solution of the system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ for given projection data $\boldsymbol{p}$, i.e., to find an image that has the given projections. In *binary tomography*, one seeks a binary solution of the system. For a given projection matrix $\boldsymbol{W}$ and given projection data $\boldsymbol{p}$, let $S_{\boldsymbol{W}}(\boldsymbol{p}) = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}\}$, the set of all real-valued solutions corresponding with the projection data, and let $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p}) = S_{\boldsymbol{W}}(\boldsymbol{p}) \cap \{0,1\}^n$, the set of *binary solutions* of the system. As the main goal of incorporating prior knowledge of the binary grey levels in the reconstruction is to reduce the number of required projections, we focus on the case where m is small with respect to n, such that the real-valued reconstruction problem is severely underdetermined.

8.3. Probe structure

We now introduce the concept of a *probe image*. A probe image is represented by a vector $\boldsymbol{v} = (v_i) \in \{0,1,?\}^n$. We say that a binary image $\bar{\boldsymbol{x}}$ *satisfies* the probe image $\boldsymbol{v}$ iff $\bar{x}_i = v_i$ whenever $v_i \in \{0,1\}$. This relation is denoted by the predicate $F(\boldsymbol{v},\bar{\boldsymbol{x}})$. In other words, the zeros and ones in the probe image prescribe the values of the corresponding pixels in $\bar{\boldsymbol{x}}$, while a pixel value of '?' in the probe image allow any pixel value in the corresponding pixel of $\bar{\boldsymbol{x}}$. We denote the set of all images satisfying a probe image $\boldsymbol{v}$ by $\mathcal{F}(\boldsymbol{v}) = \{\bar{\boldsymbol{x}} \in \{0,1\}^n : F(\boldsymbol{v},\bar{\boldsymbol{x}})\}$. For any probe image $\boldsymbol{v}$, define $s(\boldsymbol{v}) = \#\{1 \le i \le n : v_i \neq '?'\}$, the total number of 0's and 1's in the probe image.

Suppose that we want to know if the unknown original image may contain a certain homogeneous region of 1's (i.e., white pixels). We then define a probe image $\bar{\boldsymbol{v}}$ that has such a homogeneous region, and contains the '?' symbol in all pixels that are not in this region (see Fig. 8.2a). The question whether there exists a binary solution of the tomography problem that has such a region can then be rephrased as a check whether the set $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p}) \cap \mathcal{F}(\boldsymbol{v})$ is empty or not. Similarly, one can define an *edge detection* probe image such as shown in Fig. 8.2b. Any image that has a horizontal edge at the indicated position, consisting of a black strip of background pixels and a white strip of foreground pixels (i.e., an edge at the bottom

of a white region), will be part of the set $\mathcal{F}(\boldsymbol{v})$ for this probe image $\boldsymbol{v}$. This brings us to the central problem considered in this chapter:

Problem 1. *(Probe problem). Let $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ be a given projection matrix and let $\boldsymbol{v} \in \{0,1,?\}^n$ be a given probe image. Determine if $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p}) \cap \mathcal{F}(\boldsymbol{v}) = \emptyset$.*

If the intersection between the solution set of the tomography problem and the set of images that satisfy the probe image is not empty, we cannot conclude if the unknown original image satisfies the probe image. However, if the intersection between both sets is empty, we can conclude that no binary solution exists that has the probed feature. As we will see in the next sections, one can often prove that the answer to Problem 1 is "yes", even without enumerating the set $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ of binary solutions of the reconstruction problem.

Now consider the system of equations

$$\begin{pmatrix} | & & | \\ \boldsymbol{w}_1 & \cdots & \boldsymbol{w}_n \\ | & & | \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \boldsymbol{p}, \tag{8.1}$$

where $\boldsymbol{w}_i$ denotes the ith column vector of $\boldsymbol{W}$. We now define the operation of *fixing* a pixel x_i at value $v_i \in \mathbb{R}$. It transforms the system (8.1) into the new system

$$\begin{pmatrix} | & & | & | & & | \\ \boldsymbol{w}_1 & \cdots & \boldsymbol{w}_{i-1} & \boldsymbol{w}_{i+1} & \cdots & \boldsymbol{w}_n \\ | & & | & | & & | \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_{i-1} \\ x_{i+1} \\ \vdots \\ x_n \end{pmatrix} = \boldsymbol{p} - v_i \boldsymbol{w}_i. \tag{8.2}$$

The new system has the same number of equations as the original system, whereas the number of variables is decreased by one. The *fixing* operation can be performed for more than one pixel at time.

Proposition 2. *Let $\boldsymbol{W} \in \mathbb{R}^{m\times n}$ be a given projection matrix and let $\boldsymbol{v} \in \{0,1,?\}^n$ be a given probe image. Let $\boldsymbol{R}\boldsymbol{y} = \boldsymbol{q}$ be the linear system that is obtained by fixing all pixels x_i to value v_i whenever $v_i \in \{0,1\}$. Then solving Problem 1 is equivalent to checking whether $\bar{S}_{\boldsymbol{R}}(\boldsymbol{q}) = \emptyset$.*

We call the linear system formed in Prop. 2 the *reduced linear system* corresponding to the probe image $\boldsymbol{v}$.

8.4. Partially solving the probe problem

As noted in the previous section, the probe problem can be rephrased as the question whether or not the reduced linear system has a binary solution. In Chapter 2, we present a sufficient condition for the existence of binary solutions of a given linear system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ which satisfies $\sum_{i=1}^{m} w_{ij} = k$ for all $j = 1,\ldots,n$. In summary, it is proved that all binary solutions of this

linear system lie on a hypersphere centered in the minimum norm solution $\boldsymbol{x}^*$ and having radius $\mathcal{R}(\boldsymbol{p},\boldsymbol{x^*}) = \sqrt{\frac{\sum_{i=1}^{m} p_i}{k} - \|\boldsymbol{x}^*\|_2^2}$. If the binary vector closest to $\boldsymbol{x}^*$ is outside this hypersphere then the given linear system contains no binary solutions:

Theorem 22. *Let $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$, where $\boldsymbol{W}^\dagger$ denotes the Moore-Penrose inverse of $\boldsymbol{W}$ [12]. For $\alpha \in \mathbb{R}$, let $\rho(\alpha) = min(|\alpha|, |1-\alpha|)$ and put $T(\boldsymbol{x}^*) = \sqrt{\sum_{i=1}^{n} \rho^2(x_i^*)}$. If $\mathcal{R}(\boldsymbol{p},\boldsymbol{x^*}) < T(\boldsymbol{x^*})$, then $\bar{S}_{\boldsymbol{W}}(\boldsymbol{p}) = \emptyset$.*

Proof. See Chapter 2. □

In the remainder of this section we present two related techniques for proving that the unknown original image *does not* satisfy a given probe image $\boldsymbol{v}$. Both methods use variants of Theorem 22.

8.4.1. Probing by analyzing the binary solutions of the reduced linear system

Let $\boldsymbol{v}$ be a given probe image. We now analyze the reduced linear system $\boldsymbol{Ry} = \boldsymbol{q}$ corresponding to $\boldsymbol{v}$, in terms of the existence of binary solutions, following the idea of Theorem 22.

Let $\boldsymbol{y}^* = \boldsymbol{R}^\dagger \boldsymbol{q}$ and $\rho(\alpha) = \min(|\alpha|, |1-\alpha|)$. Put $T(\boldsymbol{y}^*) = \sqrt{\sum_{i=1}^{n} \rho^2(y_i^*)}$ and define the set $\mathcal{T}(\boldsymbol{y}^*) = \{\bar{\boldsymbol{r}} \in \{0,1\}^{n-s(\boldsymbol{v})} : \|\bar{\boldsymbol{r}} - \boldsymbol{y}^*\|_2 = T(\boldsymbol{y}^*)\}$. Also, let $\bar{\boldsymbol{r}} \in \mathcal{T}(\boldsymbol{y}^*)$, i.e., $\bar{\boldsymbol{r}}$ is among the binary vectors that are nearest to $\boldsymbol{y}^*$ in the Euclidean sense. Vector $\bar{\boldsymbol{r}}$ can be easily computed by rounding the entries of $\boldsymbol{y}^*$ to their nearest value in the set $\{0,1\}$. Despite $\bar{\boldsymbol{r}}$ may not be unique, any choice of $\bar{\boldsymbol{r}}$ yields the same results in this context.

Rewriting Theorem 22 in the framework of identifying the existence of binary images satisfying a given probe image $\boldsymbol{v}$, we have:

Theorem 23. *Let $\boldsymbol{y}^* = \boldsymbol{R}^\dagger \boldsymbol{q}$ and $\bar{\boldsymbol{r}} \in \mathcal{T}(\boldsymbol{y}^*)$. If $\|\bar{\boldsymbol{r}} - \boldsymbol{y}^*\|_2 > \sqrt{\frac{\sum_{i=1}^{m} q_i}{k} - \|\boldsymbol{y}^*\|_2^2}$ then the original system $\boldsymbol{Wx} = \boldsymbol{p}$ does not have a binary solution which satisfies $\boldsymbol{v}$.*

Proof. From Theorem 22, we know that if $\|\bar{\boldsymbol{r}} - \boldsymbol{y}^*\|_2 > \sqrt{\frac{\sum_{i=1}^{m} q_i}{k} - \|\boldsymbol{y}^*\|_2^2}$ then there is no binary vector satisfying $\boldsymbol{Ry} = \boldsymbol{p}$. Hence, there is no $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$ that satisfies $\boldsymbol{v}$. □

8.4.2. Probing by analyzing the binary solutions of the original linear system

Using an idea similar to what was used in the previous subsection, we now analyze the consistency of the original linear system with respect to binary solutions. However, instead of using $\bar{\boldsymbol{r}}$, the binary vector closest to the minimum norm solution $\boldsymbol{x}^*$, we define $\tilde{\boldsymbol{r}}$ as the binary vector, satisfying the probe image $\boldsymbol{v}$, which is closest to $\boldsymbol{x}^*$.

Theorem 24. *Let $\boldsymbol{x}^* = \boldsymbol{W}^\dagger \boldsymbol{p}$ and $\bar{\boldsymbol{r}} \in \mathcal{T}(\boldsymbol{x}^*)$. For $i = 1, \ldots, n$, define $\tilde{r}_i = \bar{v}_i$ if $v_i \neq '?'$ and $\tilde{r}_i = \bar{r}_i$ otherwise. If $\|\tilde{\boldsymbol{r}} - \boldsymbol{x}^*\|_2 > \sqrt{\frac{\sum_{i=1}^{m} p_i}{k} - \|\boldsymbol{x}^*\|_2^2}$, then $\boldsymbol{v}$ is not satisfied by any $\bar{\boldsymbol{x}} \in \bar{S}_{\boldsymbol{W}}(\boldsymbol{p})$.*

Proof. The vector $\tilde{\boldsymbol{r}}$ is the binary image which contains the structure of the probe image that is closest to $\boldsymbol{x}^*$. If $\tilde{\boldsymbol{r}}$ is out of the hypersphere containing all binary solutions of $\boldsymbol{Wx} = \boldsymbol{p}$ (see Chapter 2), then there is no binary image satisfying the probe image $\boldsymbol{v}$ that is *on* this hypersphere. Therefore $\boldsymbol{v}$ is not satisfied by any binary solution of $\boldsymbol{Wx} = \boldsymbol{p}$. □

8.5. Numerical experiments

Although the two techniques from Section 8.4 can detect sufficient conditions for the non-existence of binary solutions of the reconstruction problem that satisfy the given probe image, an empirical study is needed to determine the usefulness of the proposed methods for actual tomography data. A series of experiments was performed to investigate the presented method, for three different phantom images using a variable number of projections. The experiments are all based on simulated projection data obtained by computing the projections of the test images (so-called *phantoms*) in Fig. 8.3:

(a) Phantom 1, 64×64 (b) Phantom 2, 64×64 (c) Phantom 3, 64×64

Figure 8.3: *Original phantom images used for the experiments.*

For the experiments, we have used probe images that only consider 0's and 1's inside a square sub-image of size 8×8 pixels. This subregion is then moved across the full image region, scanning the possible presence of the probe structure throughout the image of size 64×64 pixels.

For each experiment, both techniques from Section 8.4 were used, checking whether any of these two methods can prove that the probe structure cannot occur in a particular region of the unknown original image.

To compute the shortest least-squares solutions of the linear systems involved in the methods of Section 8.4, the CGLS algorithm was used. We refer to Chapter 2 for details.

In the following subsections, we consider experiments for two different probe structures, for detecting homogeneous regions and horizontal edges, respectively.

8.5.1. Homogeneous regions

In this section, we focus on the identification of square homogeneous regions in the unknown original image. Two types of probe images were defined: a square 8×8 region of 1's

(white pixels) surrounded by '?' pixels, and a square 8×8 region of black pixels, also surrounded by '?' pixels. These square regions were then moved across the full 64×64 image region to determine at each location whether such a homogeneous black or white square can possibly occur in the binary solution set of the tomography problem.

For each probe image we are able to define a *status* based on the results obtained by applying the presented methods with the two different types of probe images. We define the status **forbidden** for a probe image which, according to the methods, have no binary solution satisfying it. We also define the status **allowed** for a probe image in which the methods could not determine whether there exists a binary solution satisfying this probe image.

The results for a given phantom image leads to a new 2D greyscale image, which represents – for each position of the probe region – the outcome for both probe types, as follows:

- If the black region is **allowed** and the white region is **forbidden** then associate a *black* color;
- If the black region is **forbidden** and the white region is **allowed** then associate a *white* color;
- If the black region is **allowed** and the white region is **allowed** then associate a *light grey* color;
- If the black region is **forbidden** and the white region is **forbidden** then associate a *dark grey* color;

The resulting greyscale images are shown in Fig. 8.4 for Phantoms 1 and 2, depicting results for an increasing number of projection angles. It can be observed that as the number of angles grows, the results of the probe experiments provide an increasingly accurate view of the true presence of homogeneous regions in the phantom image.

8.5.2. Horizontal edges

The goal of this section is to identify straight horizontal edges which could be present in the original image. We use the term horizontal edges to indicate horizontally adjacent pixels with intensity 1 (white) which are vertically adjacent to the same number of horizontally adjacent pixels with intensity 0 (black color). So, we define a square probe structure of size 8×8 such that the pixels in the upper half of the square are set to 0 and the pixels in the lower half of the square are set to 1. The vertically mirrored version of this probe structure was also used to detect edges at the bottom of an object.

Similar to the previous section, the results for this probe structure give rise to a new greyscale image, defined as follows. Starting from a completely black image, if at a certain position for the probe structure no unsatisfiability is detected, the “white" part of the edge (corresponding to the interior of the object) is colored white in the output image if it is also white in the original image and dark grey if it is black in the original image. The “black" part of the edge (corresponding to the outside of the object) is colored black in the output image if it is also black in the original image and light grey if it is white in the original image. Also, if at a certain position for the probe structure the unsatisfiability is detected but there are

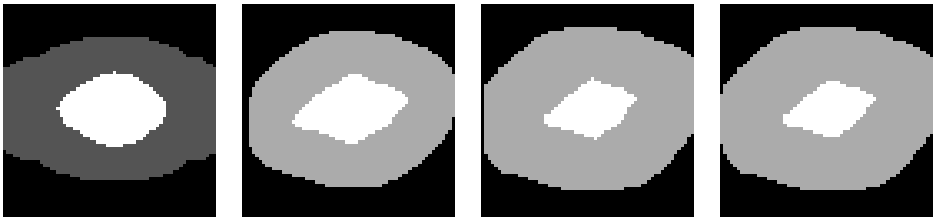
white pixels in this region in the original image, then those pixels are colored as light grey. The results of this procedure are shown in Fig. 8.5, which identifies the regions that could be edges according to our results, for a varying number of projection angles. Again, we see that as the number of projections increases, the results of the probe experiments provide an increasingly accurate view of the true presence of horizontal edges in the phantom image.

8.6. Outlook and conclusion

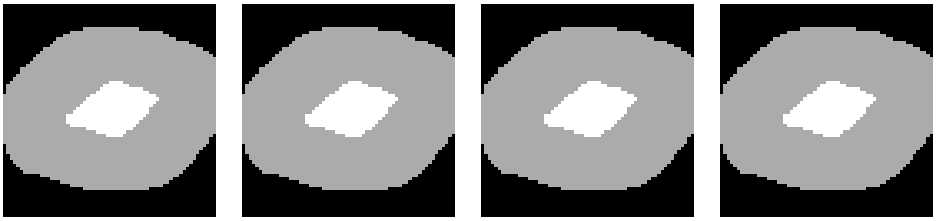
In this chapter we have proposed a novel approach for obtaining information about an object from a small number of its projections. By using necessary conditions for the existence of binary solutions of the tomography problem, and combining these with probe images for particular substructures of the image, it can be determined whether such a substructure can possibly occur, or whether it can certainly not occur in the unknown original image.

The experimental results for a limited set of simulation experiments show that this approach can indeed lead to the recovery of substantial information about the original image, without resorting to a particular, possibly non-unique reconstruction.

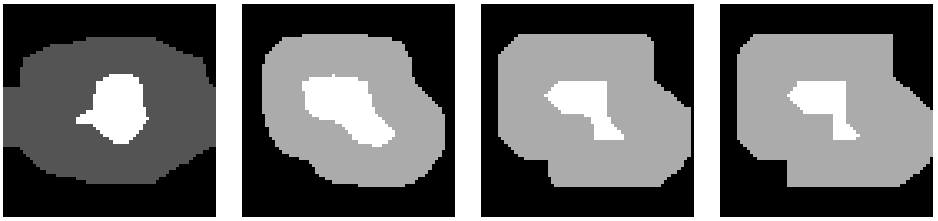
More research in this direction will be necessary to determine what the limitations are of the proposed method, and how it compares to image analysis algorithms that try to find the structure directly in a reconstructed image.



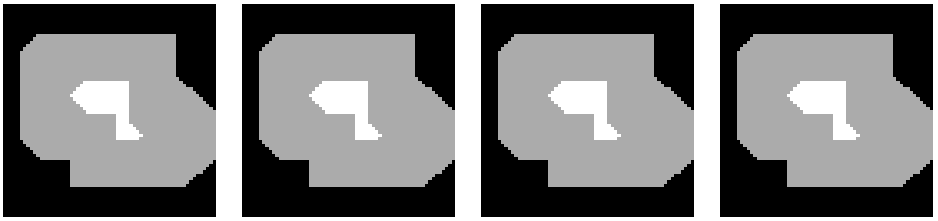
(a) Phantom 1. Number of projection angles: 2, 4, 8 and 12



(b) Phantom 1. Number of projection angles: 16, 20, 24, and 28

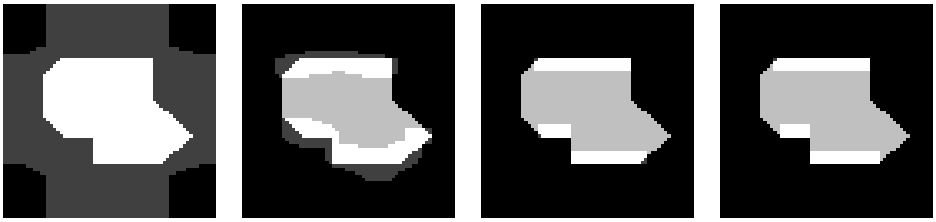


(c) Phantom 2. Number of projection angles: 2, 4, 8 and 12

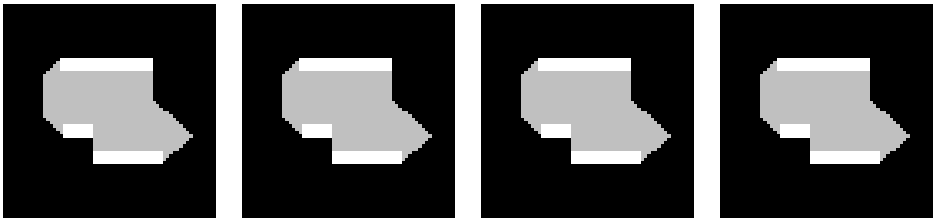


(d) Phantom 2. Number of projection angles: 16, 20, 24, and 28

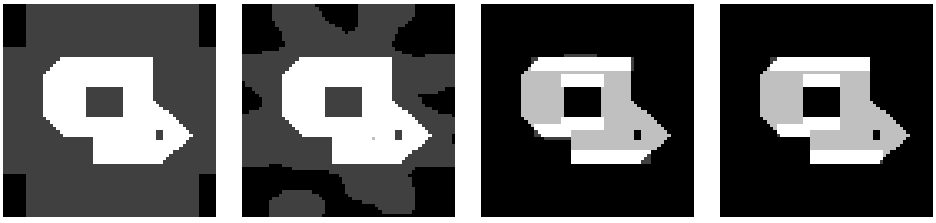
Figure 8.4: *Homogeneous region status for the phantom images of dimension* 64×64.



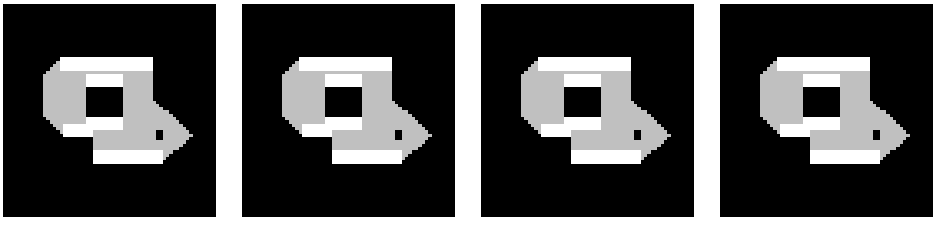
(a) Phantom 2. Number of projection angles: 2, 4, 8 and 12



(b) Phantom 2. Number of projection angles: 16, 20, 24, and 28



(c) Phantom 3. Number of projection angles: 2, 4, 8 and 12



(d) Phantom 3. Number of projection angles: 16, 20, 24, and 28

Figure 8.5: *Possible edges for the phantom images of dimension* 64×64.

Bibliography

[1] S. Van Aert, K.J. Batenburg, M. Rossell, R. Erni, and G. Van Tendeloo. Three-dimensional atomic imaging of crystalline nanoparticles. *Nature*, 470:374–377, 2011.

[2] A. Alpers. *Instability and Stability in Discrete Tomography*. PhD thesis, Technische Universität München, 2003. Shaker Verlag, ISBN 3-8322-2355-X.

[3] A. Alpers and S. Brunetti. Stability results for the reconstruction of binary pictures from two projections. *Image and Vision Computing*, 25(10):1599–1608, 2007.

[4] A. Alpers and P. Gritzmann. On stability, error correction, and noise compensation in discrete tomography. *SIAM Journal on Discrete Mathematics*, 20(1):227–239, 2006.

[5] K. J. Batenburg. Analysis and optimization of an algorithm for discrete tomography. *Electr. Notes Discrete Math.*, 12:35–46, 2003.

[6] K. J. Batenburg. A network flow algorithm for reconstructing binary images from continuous X-rays. *J. Math. Im. Vision*, 30(3):231–248, 2008.

[7] K. J. Batenburg, S. Bals, J. Sijbers, C. Kübel, P. A. Midgley, J. C. Hernandez, U. Kaiser, E. R. Encina, E. A. Coronado, and G. Van Tendeloo. 3D imaging of nanomaterials by discrete tomography. *Ultramicroscopy*, 109(6):730–740, 2009.

[8] K. J. Batenburg and J. Sijbers. Generic iterative subset algorithms for discrete tomography. *Discrete Appl. Math.*, 157(3):438–451, 2009.

[9] K. J. Batenburg and J. Sijbers. Dart: a practical reconstruction algorithm for discrete tomography. *IEEE Trans. Image Processing*, 20(9):2542–2553, 2011.

[10] J. Beck and T. Fiala. Integer-making theorems. *Discr. Appl. Math.*, 3:1–8, 1981.

[11] A. Ben-Israel and T. N. E. Greville. *Generalized inverses: Theory and Applications*. Canadian Math. Soc., 2002.

[12] Å Björck. *Numerical Methods for Least Square Problems*. SIAM, Linköping University, Sweden, 1996.

[13] Y. Censor. Row-action methods for huge and sparse systems and their applications. *SIAM review*, 23(4):444–466, 1981.

[14] G.-H. Chen, J. Tang, and S. Leng. Prior image constrained compressed sensing (PICCS): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets. *Medical Physics*, 35(2):660Ű–663, 2008.

[15] R. A. Crowther, D. J. DeRosier, and A. Klug. The reconstruction of a three-dimensional structure from projections and its application to electron microscopy. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 317(1530):319–340, 1970.

[16] D. Donoho. Compressed sensing. *IEEE Trans. Information Theory*, 52(4):1289–1306, 2006.

[17] A. Frosini and M. Nivat. Binary matrices under the microscope: A tomographical problem. *Theoretical Computer Science*, 370:201–217, 2007.

[18] A. Frosini, M. Nivat, and S. Rinaldi. Scanning integer matrices by means of two rectangular windows. *Theoretical Computer Science*, 406:90–96, 2008.

[19] M. Gara, T. Samuel Tasi, and P. Balazs. Learning connectedness and convexity of binary images from their projections. *Pure Math. Appl.*, 20(1-2):27–48, 2009.

[20] R. J. Gardner, P. Gritzmann, and D. Prangenberg. On the computational complexity of reconstructing lattice sets from their X-rays. *Discrete Math.*, 202:45–71, 1999.

[21] R. Gordon, R. Bender, and G. T. Herman. Algebraic reconstruction techniques (art) for three-dimensional electron microscopy and x-ray photography. *J. Theor. Biol.*, 29(3):471–481, 1970.

[22] L. Hajdu and R. Tijdeman. Algebraic aspects of discrete tomography. *J. Reine Angew. Math.*, 534:119–128, 2001.

[23] Lajos Hajdu and Robert Tijdeman. Bounds for approximate discrete tomography solutions. *SIAM J. Discrete Math.*, 27(2):1055–1066, 2013.

[24] S. Helgason. *The Radon transform*. Birkhäuser, Boston, 1980.

[25] G. T. Herman. *Fundamentals of Computerized Tomography: Image reconstruction from projections*. Springer, 2009.

[26] G. T. Herman. *Fundamentals of Computerized Tomography*. Springer, second edition, 2010.

[27] G. T. Herman and A. Kuba, editors. *Discrete Tomography: Foundations, Algorithms and Applications*. Birkhäuser, Boston, 1999.

[28] G. T. Herman and A. Kuba, editors. *Advances in Discrete Tomography and its Applications*. Birkhäuser, Boston, 2007.

[29] G.T. Herman and A. Kuba. Discrete tomography in medical imaging. In *Proc. of IEEE 91*, pages 1612–1626, 2003.

[30] Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. pub-SIAM, pub-SIAM:adr, second edition, 2002.

[31] J. R. Jinschek, K. J. Batenburg, H. A. Calderon, R. Kilaas, V. Radmilovic, and C. Kisielowski. 3-D reconstruction of the atomic positions in a simulated gold nanocrystal based on discrete tomography. *Ultramicroscopy*, 108(6):589–604, 2007.

[32] P. M. Joseph. An improved algorithm for reprojecting rays through pixel images. *IEEE Trans. Med. Imag.*, MI-1(3):192–196, 1982.

[33] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging*. SIAM, 2001.

[34] Li. Li, V. Stankovic, L. Stankovic, Lijie. Li, S. Cheng, and D. Uttamchandani. Single pixel optical imaging using a scanning mems mirror. *J. Micromech. Microeng.*, 21:025022, 2011.

[35] A. K. Louis. Nonuniqueness in inverse radon problems: The frequency distribution of the ghosts. *Mathematische Zeitschrift*, 185(3):429–440, 1984.

[36] P. A. Midgley and R. E. Dunin-Borkowski. Electron tomography and holography in materials science. *Nature Materials*, 8(4):271–280, 2009.

[37] K. Mueller. *Fast and accurate three-dimensional reconstruction from cone-beam projection data using algebraic methods*. PhD thesis, The Ohio State University, 1998.

[38] F. Natterer. *The mathematics of computerized tomography*. John Wiley & Sons, 1986.

[39] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, PA, USA, 2003.

[40] Th. Schüle, C. Schnörr, S. Weber, and J. Hornegger. Discrete tomography by convex-concave regularization and D.C. programming. *Discr. Appl. Math*, 151:229–243, 2005.

[41] E. Y. Sidky, M. A. Anastasio, and X. Pan. Image reconstruction exploiting object sparsity in boundary-enhanced X-ray phase-contrast tomography. *Optics Express*, 18(10):10404–10422, 2010.

[42] E. Y. Sidky, C. M. Kao, and X. Pan. Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT. *J. X-ray Sci. Tech.*, 14:119Ű–139, 2006.

[43] A. Stolk and K. J. Batenburg. An algebraic framework for discrete tomography: Revealing the structure of dependencies. *SIAM J. Discrete Math.*, 24(3):1056–1079, 2010.

[44] B. Van Dalen. On the difference between solutions of discrete tomography problems. *Journal of Combinatorics and Number Theory*, 1:15–29, 2009.

[45] B. Van Dalen. On the difference between solutions of discrete tomography problems II. *Pure Mathematics and Applications*, 20:103–112, 2009.

[46] B. Van Dalen. Stability results for uniquely determined sets from two directions in discrete tomography. *Discrete Mathematics*, 309:3905–3916, 2009.

[47] A. van der Sluis and H. A. van der Vorst. SIRT and CG-type methods for the iterative solution of sparse linear least-squares problems. *Linear Algebra Appl.*, 130:257–Ű302, 1990.

[48] R. von Mises and H. Pollaczek-Geiringer. Praktische verfahren der gleichungsauflösung. *ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik.*, 9:152–164, 1929.

[49] M. Wakin, J. Laska, M. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. Kelly, and R. Baraniuk. An architecture for compressive imaging. In *Proceedings of the International Conference on Image Processing (ICIP)*, pages 1273–1276, 2006.

[50] J. Zhua, X. Li, Y. Ye, and G. Wang. Analysis on the strip-based projection model for discrete tomography. *Discrete Appl. Math.*, 156(12):2359–2367, 2008.

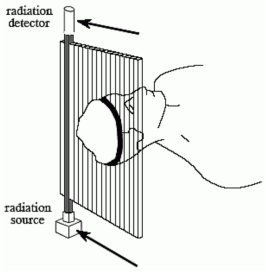
[51] D. A. Zolotov, A. V. Buzmakov, A. A. Shiryaev, and V. E. Asadchikov. X-ray computer tomography of natural fibrous diamonds and ballas. *Journal of Surface Investigation.*, 3(5):659–664, 2009.

Summary

The research in this thesis is devoted to the development of quantitative guarantees on the quality of reconstructions in discrete tomography.

Tomography refers to imaging by sections through the use of any kind of penetrating beam. The word tomography is derived from the Greek tomē (“cut") or tomos (“part" or “section") and graphein (“to write"). The goal in tomography is to obtain information from the interior of an object without having to open or destroy it. An example is the Computerized Tomography (CT) scan which is often used to scan medical patients in order to detect tumors or other irregularities in the tissues of internal organs; see Fig. 8.6(a). The scanning device used in tomography is called a *tomograph*. A source, emitting a penetrating beam, and a detector are positioned on opposite sides of the object. The tomograph records the intensity profile of the beam after it has passed through the object (or patient), acquiring a *projection* of the object at a particular angle, as in Fig. 8.6(b). From projection data obtained from different angles, a mathematical method produces a *tomogram* (also called *reconstruction*): an image of the interior of the object. A 2-dimensional reconstructed image resembles a cut of the object, as in Fig. 8.6(c).

(a) A medical CT scanner[1]



(b) Projection acquisition[2]

(c) A brain tomogram[3]

Figure 8.6: *After the scanning process, a reconstruction algorithm generates a tomogram*

[1]Thanks to Dan Dry and the University of Chicago Medicine
[2]Image from the book “The Scientist and Engineer’s Guide to Digital Signal Processing" by Steven W. Smith
[3]By Mikael HÃd’ggstrÃűm [Public domain], via Wikimedia Commons

In some cases, the judgment of a medical doctor relies on the quality of the reconstructed images (the tomograms). Tomographs often make use of harmful radiation, such as X-rays, and the dosage must be controlled in order not to harm the patient. Dose restriction is one of the limitations that affects the quality of reconstructions. Unfortunately, with the current technology and techniques available, reconstructed images are far from perfect. If doctors encounter a damaged organ or a strange object inside a patient by observing the tomogram, they would be glad to know how much they can rely on that image before sending the patient to emergency surgery. Tomography is not restricted to medical applications and can be found, for example, in industrial production lines to identify defects inside the produced products for quality control.

The seek for guarantees on the quality of reconstructions is inspired by the need of accuracy in tomography. Providing theoretical quality bounds for tomographic reconstructions is in general an inherently very difficult problem due to the ill-posedness of the mathematical problem. Firstly, the modeled problem of reconstructing images from a set of projections is typically underdetermined, having an infinite number of solutions. Finding one solution means reconstructing an image that may – or may not – be equal to the original object. Obtaining a solution is not a trivial mathematical procedure. Obtaining a solution among an infinite number of other solutions, such that it is equal to the original object, is in practice often impossible. Secondly, image reconstruction algorithms usually do not find an exact solution, but rather an approximate solution, i.e, an image with projections that approximate the given set of projections. As a consequence, one should not expect to "know" the inside of the scanned object but rather something that resembles it. Thirdly, if the projection set is acquired by a real scanning device, the projection data is corrupted by noise. The problem of finding an image which corresponds perfectly to the noisy projection data can typically not be solved exactly, because the noise perturbs the data in such a way that the reconstruction problem has contradicting information. An attempt to reconstruct an image from noisy projection data propagates the noise into the reconstructed image. Furthermore, in several applications, there are limitations on the number of projections or on the angular range for which projections can be acquired, which directly affects the quality of reconstructions. In summary, in tomography the reconstructed image might substantially differ from the true object that has been scanned.

In the process of modelling the reconstruction problem, the interaction between the beam and the object must be defined mathematically in a *projection model*. In this thesis we model the unknown object by representing it on a grid of square pixels. Some projection models have the property that the sum of the intensities of the pixels is equal to the sum of the intensities of the projection for each projection angle. For these cases, we have developed computable error bounds for *binary tomography* (where the unkown image is known to be black and white), which provide guarantees on the number of pixel errors between a reconstructed image of the object and the true object. Although it is not possible to measure the error of the reconstructed image without having access to all the information of the original object, we can guarantee a bound, a measure that is equal or larger than the actual error. With this information, one may be able to evaluate whether the reconstruction is good enough or not. A key idea in deriving these bounds is an observation first made by Hajdu and Tijdeman in [22], concerning the fact that all binary solutions have the same number of white pixels and lie on a hypersphere of which the center and the radius can be computed.

With this information, we compute bounds on the difference between binary solutions of the reconstruction problem, which also bounds the difference between any binary solution and the original binary object. In addition, we have presented bounds on the difference between any given binary image and all binary solutions.

We have developed similar bounds for the problem of high resolution binary image reconstruction from lower resolution scans. In this case, a scanning device (a video camera, for example) scans an image but due to the low resolution of the device, certain characteristics of the image cannot be identified or differentiated. A superresolution technique is used to obtain a higher resolution image from which those characteristics are now better defined. However, this new image is not guaranteed to provide an accurate representation of the true object that was scanned. This problem calls for a technique which can provide error guarantees.

When studying the problem of generating error bounds for binary tomography, we obtained a sufficient condition for the existence of binary solutions for the reconstruction problem. If one searches for a binary solution with certain specific characteristics, these characteristics can be enforced into the modeled reconstruction problem, generating a modified problem. Checking whether the modified reconstruction problem satisfies the condition of existence of binary solutions, we can determine if the feature of interest may be part of a binary solution or if it can certainly not be part of a binary solution.

Expanding the research on bounding errors in binary tomography, we developed another approach with the goal of bounding the difference between binary solutions of any underdetermined algebraic linear system of equations. It includes the case of fan beam tomography and the use of any projection model. Also, we have studied the case of binary tomography with noisy projection data. In this case, it is likely that no binary image has projections that equal the given set of projections. We assume that a noiseless projection set exists (yet not known) and take the noise level into account to compute approximate error bounds on binary solutions of the noiseless reconstruction problem (the problem for which has the true scanned object as a solution).

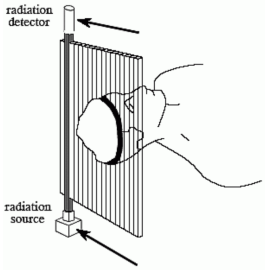
Discrete tomography does not only deal with black and white images, but also with more than two grey levels (say, 3 or 4). In some cases in discrete tomography, we may assume that the grey levels of the image to be reconstructed are known in advance. This prior knowledge can help to improve the quality of reconstructions. However, reconstructing discrete images from few projections is still a highly challenging problem. Many discrete reconstruction algorithms obtain accurate reconstructions in practice but do not offer mathematical guarantees. We have developed a discrete reconstruction algorithm which guarantees that the projections of the reconstructed discrete image are close to the given set of projections.

Samenvatting

Het onderzoek in dit proefschrift is gewijd aan de ontwikkeling van kwantitatieve zekerheden over de kwaliteit van reconstructies in discrete tomografie.

Tomografie gaat over het verkrijgen van beelden van dwarsdoorsneden door gebruik te maken van verschillende vormen van doordringende straling. Het woord tomografie is afgeleid van de Griekse woorden tomē (“snijden") of tomos (“snede") en graphein (“schrijven"). Het doel van tomografie is het verkrijgen van informatie over het binnenste van een object, zonder het te hoeven openen of kapot te maken. Een voorbeeld is een Computertomografie (CT) scan, welke vaak gebruikt wordt om bij medische patienten tumoren of andere onregelmatigheden te detecteren in het weefsel van interne organen; zie Fig. 8.7(a). Een apparaat dat tomografische scans maakt wordt een *tomograaf* genoemd. Een bron van doordringende straling en een detector van de straling worden aan weerskanten van het object geplaatst. De tomograaf meet het intensiteits-profiel van de straling nadat het door het object (of de patient) is gegaan, waardoor een *projectie* van het object onder een bepaalde hoek wordt verkregen, zoals te zien in Fig. 8.7(b). Een wiskundige methode maakt van de projecties verkregen onder verschillende hoeken een *tomogram* (ook wel *reconstructie* genoemd): een beeld van het binnenste van het object. Een 2-dimensionaal gereconstrueerd beeld lijkt op een dwarsdoorsnede van het object, zoals te zien in Fig. 8.7(c).

(a) Een medische CT-scanner[1]



(b) Opname van projecties[2]

(c) Een tomogram van de hersenen[3]

Figure 8.7: *Na het scannen maakt een reconstructie algoritme een tomogram*

[1] Met dank aan Dan Dry en de University of Chicago Medicine

[2] Figuur uit “The Scientist and Engineer’s Guide to Digital Signal Processing"door Steven W. Smith

[3] Door Mikael Häggström [Public domain], via Wikimedia Commons

In sommige gevallen zal het oordeel van een medische dokter afhangen van de kwaliteit van de gereconstrueerde beelden (de tomograms). Tomografen maken vaak gebruik van schadelijke straling, zoals Röntgen straling, en de dosis moet worden beperkt om de patient niet te schaden. Deze beperking van stralingsdosis is één van de beperkingen die de kwaliteit van de reconstructies kan beïnvloeden. Met de huidige technologie en technieken zijn de gereconstrueerde beelden helaas ver van perfect. Als dokters een beschadigd orgaan of een vreemd object in de patient opmerken door naar een tomogram te kijken, zouden ze graag willen weten hoeveel ze kunnen vertrouwen op de kwaliteit van het beeld, voordat ze de patient naar de operatiekamer sturen. Tomografie is niet beperkt tot medische toepassingen, maar wordt bijvoorbeeld ook gebruikt in industriële kwaliteitscontrole, om defecten te identificeren binnenin geproduceerde voorwerpen.

Het zoeken naar zekerheden over de kwaliteit van reconstructies is geïnspireerd door de noodzaak van nauwkeurigheid in tomografie. Het geven van theoretische kwaliteitsgrenzen voor tomografische reconstructies is een inherent erg moeilijk probleem doordat het onderliggende wiskunde probleem slecht bepaald is. Ten eerste, het gemodeleerd probleem van het reconstrueren van beelden van een verzameling projecties is vaak onderbepaald, en heeft daarom een oneindig aantal oplossingen. Een gevonden oplossing voor het probleem hoeft dan ook niet gelijk te zijn aan het werkelijke object. Het vinden van een oplossing zelf is, wiskundig gezien, al niet triviaal. Het vinden van een oplossing tussen een oneindig aantal oplossing, zodat de oplossing gelijk is aan het werkelijke object, is in de praktijk vaak onmogelijk. Ten tweede, reconstructiemethodes vinden normaal gesproken niet een exacte oplossing, maar een benadering. Dat wil zeggen, ze vinden een beeld met projecties die de gemeten projecties benaderen. Daarom kunnen we ook niet verwachten om het exacte binnenste van een object te vinden, maar vinden waarschijnlijk een benadering daarvan. Ten derde, als de projecties gemeten zijn door een echt apparaat is de data vaak vervuild met ruis. Het probleem van het vinden van een beeld dat perfect voldoet aan de ruizige projectie data kan gewoonlijk niet exact opgelost worden, omdat de ruis de data zo verstoort, dat het reconstructie probleem conflicterende informatie bevat. Een poging om een beeld te reconstrueren van ruizige projectie data verplaatst de ruis naar het gereconstrueerde beeld. Bovendien zijn er in sommige toepassingen beperkingen op het aantal projecties of het hoekbereik waarvoor projecties kunnen worden gemeten, wat een directe invloed heeft op de kwaliteit van de reconstructies. Samengevat kunnen we stellen dat bij tomografie een gereconstrueerd beeld aanzienlijk kan afwijken van het werkelijke object dat is gescand.

Tijdens het modelleren van het reconstructie probleem moet de interactie tussen de straling en het object wiskundig gedefinieerd worden in een *projectiemodel*. In dit proefschrift modelleren we het onbekende object door het te representeren op een raster van vierkante pixels. In sommige projectiemodellen is de som van de waarden van de pixels gelijk aan de som van de waarden van de projectie voor elke projectiehoek. Voor deze gevallen hebben we berekenbare foutgrenzen ontwikkeld voor *binaire tomografie* (waar het onbekende beeld alleen zwart of wit kan zijn), wat zorgt voor zekerheden over het aantal pixels die anders zijn in het gereconstrueerde beeld dan in het werkelijke object. Hoewel het niet mogelijk is om het verschil direct te berekenen zonder het werkelijke object te kennen, kunnen we wel een grens garanderen, welke gelijk of groter is dan het werkelijke verschil. Met deze informatie kan men bepalen of de reconstructie goed genoeg is of niet. Een belangrijk idee in het afleiden van deze grenzen is een observatie die als eerst gemaakt is door Hajdu en

Tijdeman in [22], met betrekking tot het feit dat alle binaire oplossingen evenveel witte pixels bevatten en allen op een hypersfeer liggen waarvan het midden en de straal berekend kunnen worden. Met deze informatie kunnen we grenzen berekenen aan het verschil tussen binaire oplossingen van het reconstructie probleem, wat ook grenzen geeft tussen elke binaire oplossing en het werkelijke binaire object. Daarnaast hebben we grenzen voorgesteld aan het verschil tussen een gegeven binair beeld en alle binaire oplossingen.

We hebben soortgelijke grenzen ontwikkeld voor het probleem van het verkrijgen van hoge resolutie binaire reconstructies van lagere resolutie scans. In dit geval scant een apparaat (zoals een video camera) een beeld, maar door de lage resolutie van het apparaat kunnen sommige kenmerken van het beeld niet worden geidentificeerd. Een superresolutie techniek kan worden gebruikt om een hoger resolutie beeld te krijgen, waarin de kenmerken beter te zien zijn. Dit nieuwe beeld hoeft echter niet een accurate weergave te zijn van het werkelijke gescande object. Dit probleem vraagt om een methode die zekerheden kan geven over de gemaakte fouten.

Tijdens het zoeken naar foutgrenzen voor binaire tomografie, hebben we een voldoende voorwaarde gevonden voor het bestaan van binaire oplossingen voor het reconstructie probleem. Als gezocht wordt naar een binaire oplossing met bepaalde eigenschappen, kunnen deze eigenschappen opgelegd worden aan het gemodeleerde reconstructie probleem, wat leidt tot een gemodificeerd probleem. Door te controleren of het gemodificeerde probleem voldoet aan de voorwaarde voor het bestaan van binaire oplossingen kunnen we bepalen of de opgelegde kenmerken deel kunnen zijn van een binaire oplossing, of zeker niet deel zijn van een binaire oplossing.

Om de kennis van foutgrenzen voor binaire tomografie verder uit te breiden hebben we een andere aanpak ontwikkeld met het doel om grenzen te bepalen voor het verschil tussen twee binaire oplossingen van een onderbepaald algebraïsch linear systeem van vergelijkingen. Dit kan bijvoorbeeld worden toegepast op verschillende projectiemodellen van tomografie. We hebben bovendien het probleem onderzocht van binaire tomografie met ruis in de projectie data. In dit geval is het waarschijnlijk dat er geen enkele binaire oplossing te vinden is met projecties gelijk aan de gemeten projecties. We veronderstellen dat er een ruisvrije set van projecties bestaat (die nog onbekend is), en houden rekening met de hoeveelheid ruis om geschatte foutgrenzen voor binaire oplossingen van het ruisvrije reconstructie probleem te berekenen.

In discrete tomografie hebben de oplossingen niet alleen zwarte of witte pixels, maar bestaat het beeld uit een aantal verschillende grijswaarden (zeg, 3 of 4). In sommige gevallen kunnen we veronderstellen dat de mogelijke grijswaarden van het uiteindelijke beeld van te voren bekend zijn. Deze voorkennis kan gebruikt worden om de kwaliteit van de reconstructies te verhogen. Het is echter nog steeds een erg moeilijk probleem om discrete beelden te reconstrueren van een klein aantal projecties. In de praktijk blijkt dat veel reconstructie algoritmes voor discrete tomografie nauwkeurige reconstructies kunnen berekenen, maar de algoritmes kunnen geen wiskundige garanties geven. Wij hebben een discreet reconstructie algoritme ontwikkeld die kan garanderen dat de projecties van het gereconstrueerde beeld weinig verschillen van de gemeten projecties.

Curriculum Vitae

On the twentieth of May 1983, Wagner Fortes was born in Rio de Janeiro, Brazil. He was also raised in Rio de Janeiro, attending Colégio Sagres from 1986 until 2000, where he completed the high school education. He studied Mathematics at the Rio de Janeiro State University from 2002 until 2006. The bachelor's diploma was granted to him after presenting a dissertation on mathematics applied to quantum computation. In 2008 he obtained the degree of Master of Sciences in Mechanical Engineering from the Rio de Janeiro State University, with a specialization in Transport Phenomena.

His master's thesis entitled "Pre-conditioners and solvers for algebraic linear systems obtained from the simulation of reservoir flooding" was written as part of a project on the development of software to aid in the decision making process when creating hydroelectric reservoirs while considering possible environmental impacts. In parallel to his studies, from the beginning of 2007 until the middle of 2008, Wagner Fortes also worked as a teacher at the department of applied mathematics of Rio de Janeiro State University.

In late 2008, he started his PhD studies in optimization at the Federal University of Rio de Janeiro. Having decided to pursue a PhD abroad, he moved to The Netherlands in August 2009. After being granted an Erasmus Mundus Scholarship for a PhD position, he started working at the Centrum Wiskunde & Informatica (CWI) under the supervision of Prof. dr. Kees Joost Batenburg and Prof. dr. ir. Barry Koren. During the four years of this PhD research he has been seconded at Leiden University.