Delft University of Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft Institute of Applied Mathematics

Robust Algorithms for Discrete Tomography

A thesis submitted to the
Delft Institute of Applied Mathematics
in partial fulfilment of the requirements

for the degree

MASTER OF SCIENCE
in
APPLIED MATHEMATICS

by

FRANK TABAK

Delft, the Netherlands
June 2012

MSc THESIS APPLIED MATHEMATICS

"Robust Algorithms for Discrete Tomography"

FRANK TABAK

Delft University of Technology

Daily supervisor

Dr.ir. M.B. van Gijzen

Responsible professor

Prof.dr.ir. C. Vuik

Other thesis committee members

Prof.dr. K.J. Batenburg

Prof.dr.ir. G. Jongbloed

June 2012

Delft, the Netherlands

Preface

Since early in the 20th century tomography has been of major interest for it provides means to do non-invasive visualisation of the interior of objects such as the human body. Tomography methods concentrate on reconstructing objects from multiple projections that are obtained by sending, for example, X-rays through the object. Applications of these methods are, among others, radiology (CT-, MRI- and PET scans), geophysics and material science. The tomographic problems can be formulated as a system of linear equations. Unfortunately, these systems are not square and thus not symmetric or positive (semi)definite and in general rank deficient.

In material science one is often presented with very small objects (like crystals or nano-structures) that consist of one or a small number of different materials, each with its own density. Scanning these small objects can cause damage to the structure and thus one can only take a very limited amount of projections. Fortunately, one can use the prior knowledge about the object to arrive at a reconstruction of the original object. How to arrive at this reconstruction is studied by the field of *discrete tomography* (DT).

With every kind of tomography, and thus also with DT, one is faced with noisy data. Because of this noise the reconstruction process becomes more difficult since the system of linear equations becomes inconsistent. The DART (Discrete Algebraic Reconstruction Method) algorithm was developed by the All Scale Tomographic Reconstruction Antwerp (ASTRA) group from the university of Antwerp to solve DT problems. This algorithm deals with noise in a very heuristic method. The goal of this project is to investigate how the problem can be regularized such that it deals with the noise in a more efficient and robust manner.

The following document consists of five chapters. The first chapter is a general introduction in tomography which describes the history of the field and various kinds of tomography. Chapter 2 describes the analytical reconstruction method. This is an approach of reconstructing the original object from its projections using the Fourier transform. This technique is the method of choice in medical tomography for it provides reasonably accurate reconstructions in very short times. The downside of this approach is that the reconstructions of data acquired from a few number of projection angles is usually poor. The algebraic reconstruction methods (ARMs) discussed in Chapter 3 perform better in these kind of conditions. Four different ARMs will be described and investigated: ART, SIRT, SART and CGLS. Chapter 4 describes the DT problem formally and proposes four different methods of solving this problem. One approach is of particular interest since this is the approach DART uses. The DART algorithm will be explained in detail and the performance will be investigated by experiments. Finally, Chapter 5 presents the research questions that the upcoming research will hopefully answer. Test problems are presented which should aid with the validation of the answers.

This work is a report of the studied literature as part of the master thesis in Applied Mathematics, Delft University of Technology. The literature study forms the basis from which the subsequent research will be carried out. The project is commissioned by the Centrum Wiskunde & Informatica (CWI), a scientific research institute specialized in mathematics and computer science. The daily supervisor from Delft University of Technology of this master thesis is Martin van Gijzen and Kees-Joost Batenburg is the supervisor at the CWI. Both supervisors are greatly acknowledged for their involvement in this project.

Contents

Chapter 1

Tomography

The study of reconstructing (slices of an) object from its projections is called *tomography*. The projections are obtained from various angles via penetrating waves, e.g. X-rays. The word tomography has it origins in the two Greek words $\tau\acute{o}\mu o\sigma$ (tomos), meaning *slice* or *part*, and $\gamma\rho\acute{a}\phi\epsilon\iota\nu$ (graphein), meaning *to write*. Tomography was first considered after the invention of X-rays by Wilhelm Röntgen in 1895 although it began flourishing in the period before the second world war [26, p. 6]. Since reconstructions are usually done by computers the term *Computerised Tomography* (CT) is used.

Tomography is probably most famous for its applications in medicine for it provides non-invasive ways to see the internal structure of a (mostly human) body. Though the field of application is very broad, from nanoscopic scale where it is used to determine structure of certain nano-particles, to galactic-scale where it is used to reconstruct the X-ray structure of supernova remnants. Tomography is, among many other fields, studied in medicine, materials science, geo- and astrophysics. This motivates the need for developing better and faster reconstruction techniques.

There are many types of tomography. *Transmission tomography* for example is based on sending penetrating waves through an object and measuring the waves on the other end of the object with a detector. X-ray computed tomography (CT/CATScan) uses this principle. *Emission*

Figure 1.1: *'Hand mit Ringen', the first medical X-ray picture taken on December 22 1895. It shows the hand of Wilhelm Röntgen's wife.*

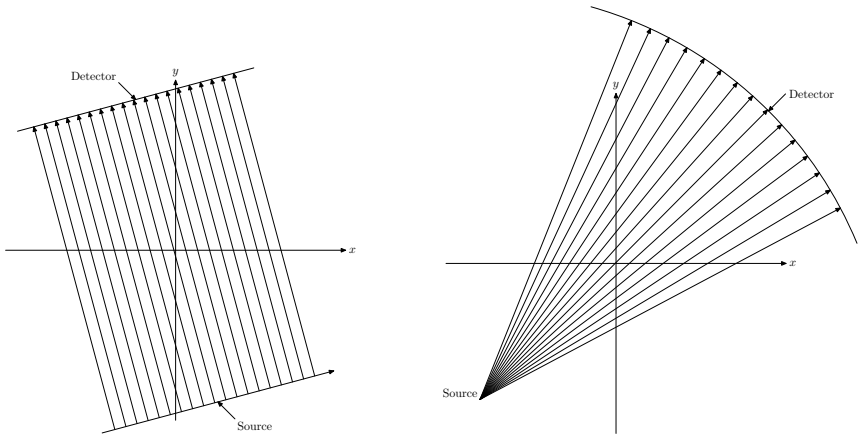


Figure 1.2: *Left: Parallel data acquisition geometry. Right: Fanbeam (non-parallel) data acquisition geometry.*

tomography on the other hand is based having the source inside the object, this source is usually a radioactive tracer which is injected into the blood stream. Two types of emission tomography can be distinguished, *Single Particle Emission Computed Tomography* (SPECT) which measures emitted particles along a half-line, and *Positron Emission Tomography* (PET) which is based on the conversion of a proton into a neutron. During this conversion a positron and neutrino are released, this positron will interact with an electron almost instantly creating two photons which travel in approximately opposite direction. These photons are measured and from the measurement one can determine the location where the annihilation took place, and hence the source was located. Another type of tomography is *reflection tomography*, typically sound waves are emitted towards an object, the sound is reflected and from that the object can be reconstructed.

All types of tomography reconstruct an object or function from its projections, or in mathematical sense, line (or hyperplane) integrals. The object has a certain internal distribution which one wants to reconstruct. In X-ray tomography for example the density of tissue attenuates the radiation. By measuring to which extent the rays are attenuated the internal distribution of the object, and thus the distribution of the tissue, can be reconstructed.

In what follows *rays* will be referred to as the lines a certain penetrating wave follows when travelling from the source to the detector. In X-ray CT this is simply the path from an X-ray tube (vacuum tube that produces X-rays) to the detector. There are many ways in which one can emit these rays, referred to as *data acquisition technique*. The rays can be emitted parallel to each other for each projection angle. This yields the most natural understanding of tomography. In practice, however, it is very hard to carry out such a parallel acquisition. Instead fan (in case of one-dimensional projections) or cone (two-dimensional projections) acquisition techniques are used in practice. Luckily, some correspondence exist between the various types of techniques which enables one to use results for parallel rays for non-parallel rays. Figure 1.2 shows two different data acquisition techniques for two-dimensional reconstruction.

For an extensive detailed report on the history of tomography and an account on the all various aspects associated with tomography one is referred to the book of Webb [26].

Chapter 2

Analytical Reconstruction Method

The problem of reconstructing an object from its projections can be solved using various methods. One of these methods is based on the Fourier transform of the object and its projections. This section will briefly discuss how this reconstruction is carried out and results will be shown using images. First the Radon transform is introduced which defines the relationship between the object and its projection. Then the Fourier slice theorem will be presented and proven, this theorem is fundamental to the Filtered Backprojection (FBP) method. This method reconstructs the object from its projections and will be considered for parallel-beams. Although in practice mostly fan- and cone-beam geometries are used, it is easier to consider and solve the problem for parallel beams. Data resulting from these other acquisition techniques can also be reconstructed, as will be show in Section 2.4.

2.1 Radon Transform

As was shown in Chapter 1 there are many kinds of tomography. All these methods are based upon the fact that a physical object alters the rays that are sent through that object. With X-rays for example the intensity will gradually drop as it moves through the object. This attenuation can be caused by tissue which absorbs energy of the X-ray as it passes through. The attenuation of an (two-dimensional) object will be represented by the function $f(x, y)$ which is assumed to be integrable. Since the attenuation of an object is assumed to be related to its shape and density, one can reconstruct the object if one knows $f(x, y)$. Hence $f(x, y)$ will often be referred to as the object or image.

Suppose parallel beams are used and that a ray passes through the object like in Figure 2.1, i.e. it makes an angle of θ with the y-axis and the distance from the ray to the origin is t_1. Note that the dashed line makes an angle of θ with the x-axis and is perpendicular to the ray, the source will move parallel to this line such that all the rays are parallel (hence *parallel* projections). The ray can be parametrized as $x\cos(\theta) + y\sin(\theta) = t_1$ and hence, in general, the projection $P_\theta(t)$ of a ray can be regarded as the line integral of $f(x, y)$ over the parametrization of the ray:

$$P_\theta(t) = \int_{x\cos(\theta)+y\sin(\theta)=t} f(x, y)\mathrm{d}s. \tag{2.1}$$

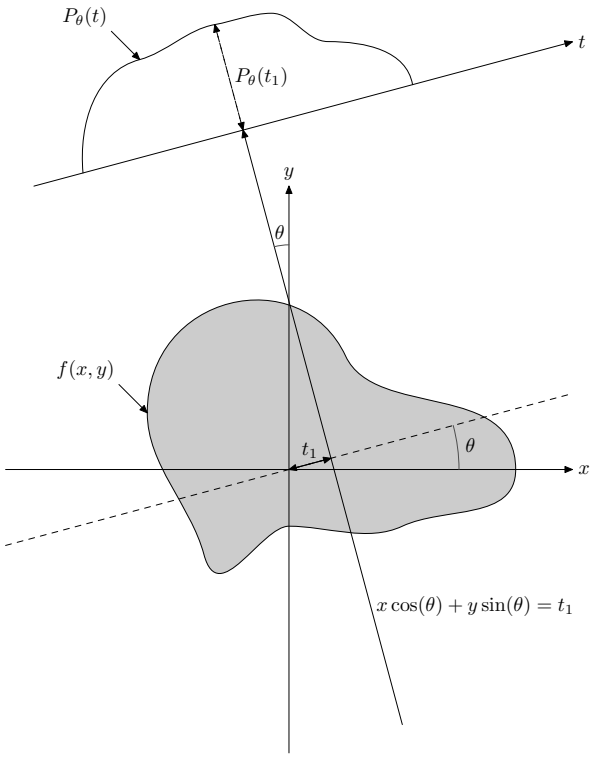


Figure 2.1: *A ray which passes through the object with attenuation $f(x, y)$, the t-axis represents the projection values of the object with projection angle θ.*

Definition 2.1. The Dirac-delta function $\delta(s)$ is defined by

$$\delta(s) = \begin{cases} +\infty & s = 0 \\ 0 & s \neq 0 \end{cases} \tag{2.2}$$

$$\int_{-\infty}^{\infty} \delta(s)\mathrm{d}s = 1. \tag{2.3}$$

Using this definition one can rewrite (2.1) as

$$P_\theta(t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(x\cos(\theta) + y\sin(\theta) - t)\mathrm{d}x\mathrm{d}y. \tag{2.4}$$

Definition 2.2. The function $P_\theta(t)$, as given in (2.4), is the Radon transform of $f(x, y)$.

The Radon transform is named after Johann Radon, an Austrian mathematician who proposed the formula in 1917. The transform is often denoted as $(\mathcal{R}f)(\theta, t)$.

For the moment only parallel projections of two-dimensional objects will be considered. Using this form of acquisition, the parallel rays for some fixed angle θ are collected for various t so that

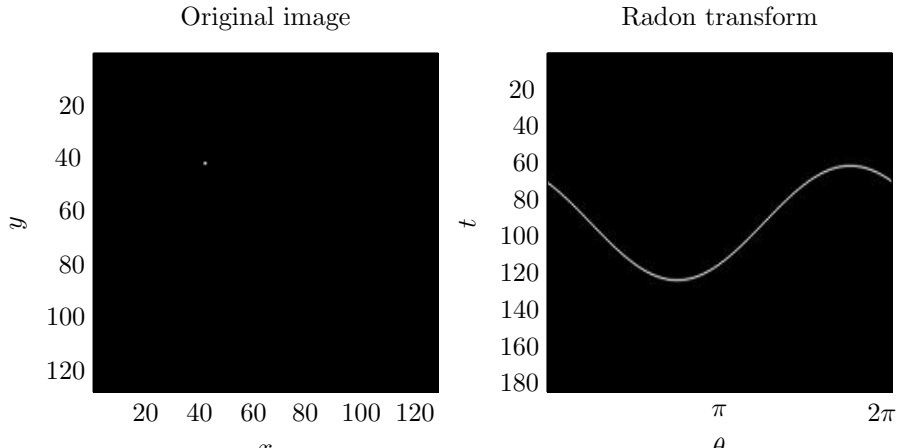


Figure 2.2: *Left: The original image. Right: The Radon transform of the image.*

one ends up with $P_\theta(t)$, the projection data corresponding to one angle (Figure 2.1 and the left figure of Figure 2.4). This is repeated for multiple angles to obtain a large set of projection data. The Radon transform defines the relation between the measured data and the object $f(x, y)$. Note that only angles $\theta \in [0, \pi)$ have to be considered since $P_\theta(t) = P_{\theta+\pi}(-t)$ and hence does not introduce any new information.

To get a feeling of what the Radon transform is, the transform is applied to an example image. Figure 2.2 shows the Radon transform in case the image is a single dot. The right figure is called the *sinogram* of the original image. Note that the relation $P_\theta(t) = P_{\theta+\pi}(-t)$ indeed holds. Furthermore since a Radon transform is a superposition of sinusoids, the Radon transform of a point is a sinusoid. Figure 2.3 shows the sinogram of the *Shepp-Logan head phantom*, this phantom will be used throughout for examples.

The goal of tomography is to reconstruct the original object or image from its projections (such as the Radon transform). When the original object is reconstructed exactly from the Radon transform one speaks of the inverse Radon transform. The FBP method, introduced later on, approximates the original solution and can thus be viewed as an approximation of the inverse Radon transform.

2.2 Fourier Slice Theorem

The filtered backprojection method is based on the result of the Fourier slice theorem. This theorem, informally, states that the two-dimensional Fourier transform of a parallel projection is equal to a slice of the two-dimension Fourier transform of the object. Thus using this result one can construct the two-dimensional Fourier transform of the object with the parallel projection data. Applying a two-dimensional inverse Fourier transform would then yield the original object.

First some definitions on Fourier theory are recalled.

Definition 2.3. Let $P : \mathbb{R} \to \mathbb{C}$ be a continuous integrable function. Then the Fourier transform of $P(t)$ is defined as

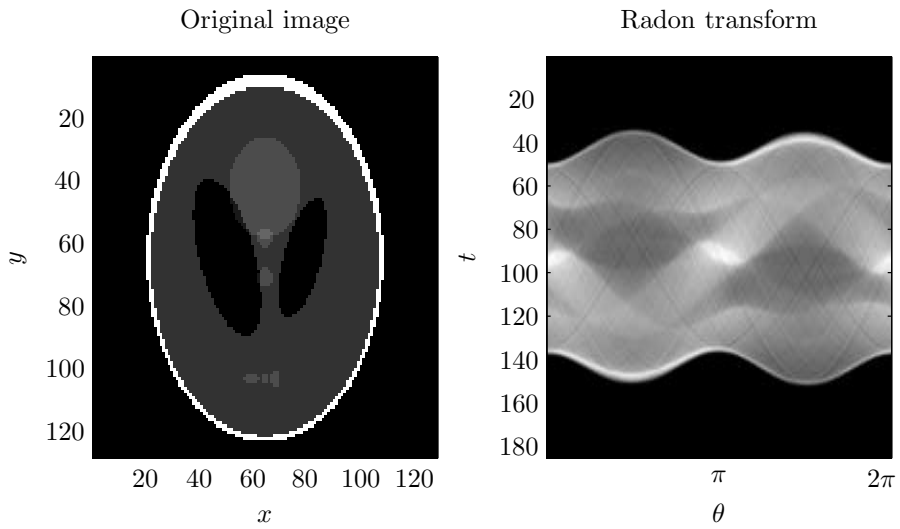


Figure 2.3: *Left: The original image. Right: The Radon transform of the image.*

$$\hat{P}(\omega) = \int_{-\infty}^{\infty} P(t) e^{-2\pi\omega t i} \mathrm{d}t \tag{2.5}$$

for every real value of ω. The inverse Fourier transform of $\hat{P}(\omega)$ is defined as

$$P(t) = \int_{-\infty}^{\infty} \hat{P}(\omega) e^{2\pi\omega t i} \mathrm{d}t. \tag{2.6}$$

The variable ω of the Fourier transform is often referred to as the *frequency*, $\hat{P}(\omega)$ is said to be defined in the frequency domain while $P(t)$ is defined in the spatial domain. Similarly the two-dimensional Fourier transform can be defined as:

Definition 2.4. Let $f : \mathbb{R}^2 \to \mathbb{C}$ be a continuous integrable function. Then the Fourier transform of $f(x, y)$ is defined as

$$\hat{f}(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi(ux+vy)i} \mathrm{d}x\mathrm{d}y \tag{2.7}$$

for every real value of u and v. The inverse Fourier transform of $\hat{f}(u, v)$ is defined as

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \hat{f}(u, v) e^{2\pi(ux+vy)i} \mathrm{d}u\mathrm{d}v. \tag{2.8}$$

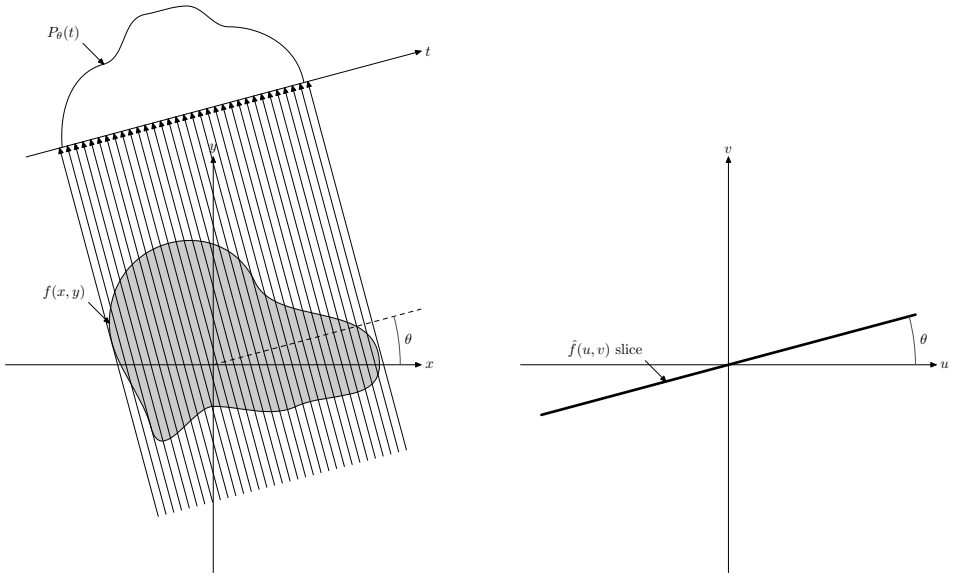


Figure 2.4: *The Fourier transform of the Projection $P_\theta(t)$ left corresponds to the line ($\hat{f}(u,v)$ slice) in the right figure.*

Now the actual Fourier slice theorem will be stated [17, p. 58].

Theorem 2.5. *The Fourier transform of a parallel projection of an image $f(x,y)$ taken at angle θ gives the values of the two-dimensional transform $\hat{f}(u,v)$ along a line subtending an angle θ with the u-axis in the frequency domain, i.e.*

$$\hat{P}_\theta(\omega) = \hat{f}(\omega\cos(\theta), \omega\sin(\theta)). \tag{2.9}$$

In Figure 2.4 the theorem is illustrated. It is clear that if one has projections from many angles in $[0,\pi)$ that then the two-dimensional Fourier transform of the object can be approximated. The proof of Theorem 2.5 will now be given and closely follows the proof given in [17, p. 58].

Proof. Assume an arbitrary angle $\theta \in [0,\pi)$ from which the projection is taken. The original Cartesian coordinate system (x,y) is rotated to a coordinate system (t,s) in such a way that the projections taken are along lines of constant t. The mapping from (x,y) to (t,s) is given by

$$\begin{pmatrix} t \\ s \end{pmatrix} = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{2.10}$$

The projections in the new coordinate system are given by

$$P_\theta(t) = \int_{-\infty}^{\infty} f(t,s)\mathrm{d}s. \tag{2.11}$$

Taking the Fourier transform of these projections gives

$$\hat{P}_\theta(\omega) = \int_{-\infty}^{\infty} P_\theta(t) e^{-2\pi\omega t i} \mathrm{d}t, \tag{2.12}$$

which, according to the theorem should equal a slice of the two dimensional Fourier transform of the object. To see this, substitute (2.11) into the equation above so one obtains

$$\hat{P}_\theta(\omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(t,s) e^{-2\pi\omega t i} \mathrm{d}s\mathrm{d}t. \tag{2.13}$$

Now transform the coordinate system back to (x, y). From (2.10) one has $t = x\cos(\theta) + y\sin(\theta)$ and $s = -x\sin(\theta) + y\cos(\theta)$. The Jacobian is given by $|\cos^2(\theta) + \sin^2(\theta)| = 1$ and thus $\mathrm{d}s\mathrm{d}t = \mathrm{d}y\mathrm{d}x$. Interchanging the integrals is allowed due to Fubini's theorem [20, p. 84], note that $f(x, y)$ is assumed to be integrable. Doing these substitutions yields

$$\hat{P}_\theta(\omega) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) e^{-2\pi\omega(x\cos(\theta)+y\sin(\theta))i} \mathrm{d}x\mathrm{d}y. \tag{2.14}$$

Now note that the right-hand side equals the two-dimensional Fourier transform $\hat{f}(u, v)$ for $u = \omega\cos(\theta)$ and $v = \omega\sin(\theta)$. Hence one is forced to conclude that the desired result is reached since $\hat{P}_\theta(\omega) = \hat{f}(\omega\cos(\theta), \omega\sin(\theta)) = \hat{f}(\omega, \theta)$. □

As noted before, the reconstruction can be acquired by approximating $\hat{f}(u, v)$ in the frequency domain and then taking the inverse Fourier transform of this approximation. In Figure 2.5 one can see the slices of known frequency if projections are taken every $\pi/12$ radians or 15 degrees. Note that the higher frequency components, these are (u, v) points further from the origin, are less well known than the low frequency components. Therefore some interpolation is required. One should note that in practice the function $f(x, y)$ is only defined on a finite domain and the discrete inverse Fourier transform should be used, hence the discrete Fourier coefficients $\hat{f}(u, v)$ should be known. Some of these coefficients might be known due to the slice theorem, others have to be approximated (interpolated) with those. How to tackle this discrete case is explained in more detail in [17, p. 58–59].

2.3 Filtered Backprojection for Parallel Beams

The filtered backprojection method (FBP) is based on the Fourier slice theorem and is a descretised version of the inverse Radon transform. As noted at the end of the previous section, the data in the frequency domain need to be transformed back to the spatial domain to obtain a reconstruction of the object. Since only a finite number of projections can be acquired, information is always missing and interpolation needs to be carried out. Rather than doing this interpolation in the frequency domain, as suggested in the previous section, the FBP method does the interpolation in the spatial domain, i.e., for each projection angle the Fourier transforms of the projections $P_\theta(t)$ are taken, these transformations are then multiplied by a weighting

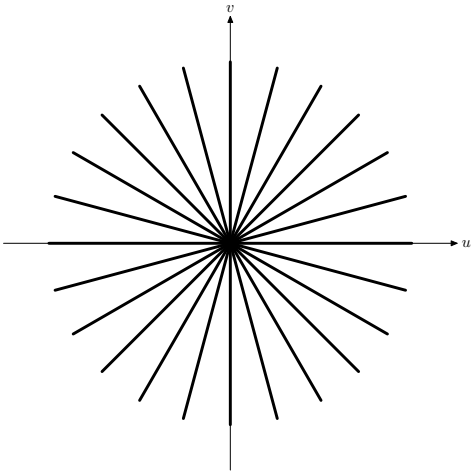


Figure 2.5: *The known frequency data from the Fourier transformed projections.*

function. The results for each angle, thus the slices in the frequency domain, are transformed back to the spatial domain and summed up. This approach has two advantages. First of all, the computations can be started as soon as the first projection data are available. Secondly, the interpolation of the missing information is now carried out in the spatial domain. Simple linear interpolation in the spatial domain often leads to reasonably accurate results while more complicated approaches are needed for the frequency domain [24]. One could also expect this since the unknown points in the frequency domain are not uniformly distributed and correspond mainly to the high frequency components. Close to the origin the slices are close together and nearest neighbours might lead to reasonable results, farther away, however, the nearest neighbour is actually quite distant (see Figure 2.5).

The FBP method naturally follows from a change of coordinates as will be shown. Until now the two-dimensional Fourier transform of the object was expressed in Cartesian coordinates, but since the transformed projections form a line through the origin subtending an angle θ in this domain it would be more natural to use polar coordinates instead of Cartesian. Recall the two-dimensional inverse Fourier transform (cf. (2.8)):

$$f(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \hat{f}(u,v)e^{2\pi(ux+vy)i}\mathrm{d}u\mathrm{d}v. \tag{2.15}$$

Let $u = \omega\cos(\theta)$ and $v = \omega\sin(\theta)$. For the Jacobian J of this coordinate change one finds

$$J = \det\left(\frac{\partial(u,v)}{\partial(\omega,\theta)}\right) = \begin{vmatrix} \cos(\theta) & \sin(\theta) \\ -\omega\sin(\theta) & \omega\cos(\theta) \end{vmatrix} = \omega\left(\cos^2(\theta)+\sin^2(\theta)\right) = \omega. \tag{2.16}$$

Hence the differentials $\mathrm{d}u\mathrm{d}v$ are replaced by $\omega\mathrm{d}\omega\mathrm{d}\theta$ so that (2.15) becomes

$$f(x,y) = \int_{0}^{2\pi}\int_{0}^{\infty} \hat{f}(\omega,\theta)\omega e^{2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta. \tag{2.17}$$

This integral can be split into two parts by considering $\theta \in [0, \pi)$ and $\theta \in [\pi, 2\pi)$. Doing the change of variables $\theta = \theta + \pi$ in the second integral yields

$$\begin{aligned} f(x,y) &= \int_0^{\pi}\int_0^{\infty} \hat{f}(\omega,\theta)\omega e^{2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta \\ &+ \int_0^{\pi}\int_0^{\infty} \hat{f}(\omega,\theta+\pi)\omega e^{2\pi\omega(x\cos(\theta+\pi)+y\sin(\theta+\pi))i}\mathrm{d}\omega\mathrm{d}\theta. \end{aligned} \tag{2.18}$$

By symmetry of the Fourier transform ones has $\hat{f}(\omega, \theta+\pi) = \hat{f}(-\omega, \theta)$ [5, p. 181]. Furthermore one has $\cos(\theta + \pi) = -\cos(\theta)$ and likewise $\sin(\theta + \pi) = -\sin(\theta)$. Applying this information gives

$$\begin{aligned} f(x,y) &= \int_0^{\pi}\int_0^{\infty} \hat{f}(\omega,\theta)\omega e^{2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta \\ &+ \int_0^{\pi}\int_0^{\infty} \hat{f}(-\omega,\theta)\omega e^{-2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta. \end{aligned} \tag{2.19}$$

Substituting $\omega = -\omega$ in the second integral results in

$$\begin{aligned} f(x,y) &= \int_0^{\pi}\int_0^{\infty} \hat{f}(\omega,\theta)\omega e^{2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta \\ &+ \int_0^{\pi}\int_{-\infty}^{0} \hat{f}(\omega,\theta)\omega e^{2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta \qquad (2.20) \\ &= \int_0^{\pi}\int_{-\infty}^{\infty} \hat{f}(\omega,\theta)|\omega| e^{2\pi\omega(x\cos(\theta)+y\sin(\theta))i}\mathrm{d}\omega\mathrm{d}\theta. \qquad (2.21) \end{aligned}$$

Write $t = x\cos(\theta) + y\sin(\theta)$ and recall that due to the Fourier slice Theorem $\hat{f}(\omega, \theta) = \hat{P}_\theta(\omega)$ to obtain the expression

$$f(x,y) = \int_0^{\pi}\int_{-\infty}^{\infty} \hat{P}_\theta(\omega)|\omega| e^{2\pi\omega t i}\mathrm{d}\omega\mathrm{d}\theta. \tag{2.22}$$

This expression shows the workings of the FBP method. Note that the transformed projection is multiplied by the weight function $|\omega|$ which followed from the change of coordinates. This

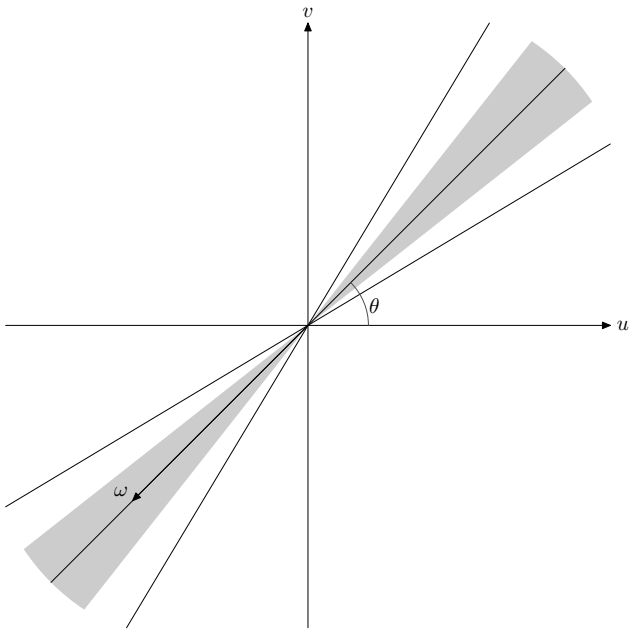


Figure 2.6: *The pie-shaped wedges represent the area of $\hat{f}(\omega, \theta)$ that is approximated by $|\omega|\hat{P}_\theta(\omega)$. Note that the weight increases as $|\omega|$ increases.*

weight function can be justified by the distribution of the unknowns in the frequency domain. As noted the slices are further apart for high frequency components, i.e. for $\hat{P}_\theta(\omega)$ with large $|\omega|$. Therefore these values are multiplied by $|\omega|$ as to make up for the missing information. In Figure 2.6 this is sketched. The middle line represents a slice, the two other lines are neighbouring slices. The grey area represents the area of the frequency domain that is approximated by $|\omega|\hat{P}_\theta(\omega)$.

Usually (2.22) is written as

$$Q_\theta(t) = \int_{-\infty}^{\infty} \hat{P}_\theta(\omega)|\omega| e^{2\pi\omega t i} \mathrm{d}\omega, \tag{2.23}$$

$$f(x,y) = \int_0^{\pi} Q_\theta(x\cos(\theta) + y\sin(\theta))\mathrm{d}\omega\mathrm{d}\theta. \tag{2.24}$$

This decouples the the original expression into two parts: (2.23) is the filtered backprojection of one slice, thus one projection angle, (2.24) is the summation of the backprojections over angles.

Filters are used to suppress or amplify certain frequencies. This is achieved by multiplying components of certain frequencies by a weight function. The foregoing reasoning then leads to the conclusion that the term $Q_\theta(t)$ represents a filtering operation. These filtered projections $Q_\theta(t)$ are then back projected, hence the name filtered backprojection. Often one also includes a window function in (2.23), this is a type of filter which typically suppresses high frequency components. One example of such a window is the Hamming window. Figure 2.7 shows the benefits of using a Hamming window.

One should note that for each angle this backprojection is carried out separately and the results

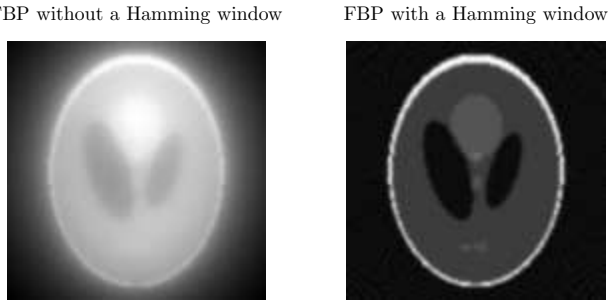


Figure 2.7: *The Shepp-Logaon head phantom reconstructed using the FBP without and with a Hamming window.*

are added in the spatial domain. For a certain projection angle θ all the points on the line $x\cos(\theta)+y\sin(\theta)=t$ get the same contribution from the backprojected data. The results of the backprojection are *smeared* back to the spatial domain. Figure 2.8 shows the results of the FBP method after a various number of projections. In this figure the smearing is clearly visible when a small number of projections is used. The results of 64 and 128 projections are very accurate. The original image is 128×128 pixels, the inner square is 43×43 pixels. The experiment was carried out with the `radon` and `iradon` functions in MATLAB. The latter function corresponds to the filtered backprojection and was used with nearest neighbour interpolation and a Hamming window.

When dealing with the implementation of this method one typically assumes that frequencies higher than a certain limit W are not present in the data. In that case the integral in (2.23) can be determined from $-W$ to W. Also the continuous Fourier transforms are replaced by discrete ones. For more details about how to implement the FBP the reader is referred to [17, p. 65–75].

2.4 Filtered Backprojection for Non-Parallel Beams

In practice mostly non-parallel acquisition techniques are used since the mechanics for parallel beams are complex and result in high data acquisition time which is unacceptable in most applications, such as medical CT. Therefore one would like to be able to use non-parallel beams. This section will list three different methods that allows one to use the FBP method when non-parallel acquisition techniques are used. Only the ideas behind the methods will be given.

2.4.1 Adapted FBP

When using non-parallel beams, such as a fan-beam, one can also deduce the FBP method for these beams. This requires that one parametrizes every ray for every projection angle separately. This parametrization was rather simple for the parallel case, every beam could be identified by t and θ as $t=x\cos(\theta)+y\sin(\theta)$. For non-parallel beams however this parametrization forms quite a challenge and results into far less elegant formulas. Moreover, for every type of data acquisition this adaptation is different, for example one can use fan-beams with a curved detector such that all the rays have even length or a linear detector (see Figure 2.9). This method is discussed in detail in [17, Ch. 3.4], [15, p. 177–181] and [5, Ch. 7.7.3–7.7.4].

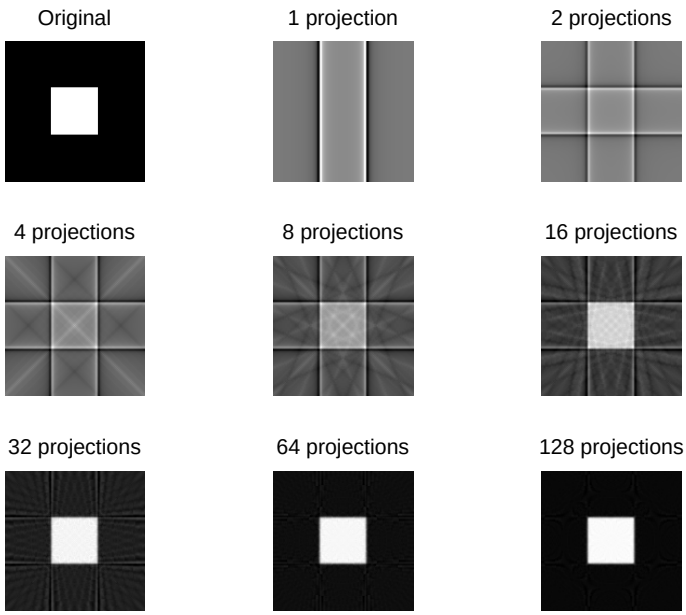


Figure 2.8: *The FBP method for a various number of projections.*

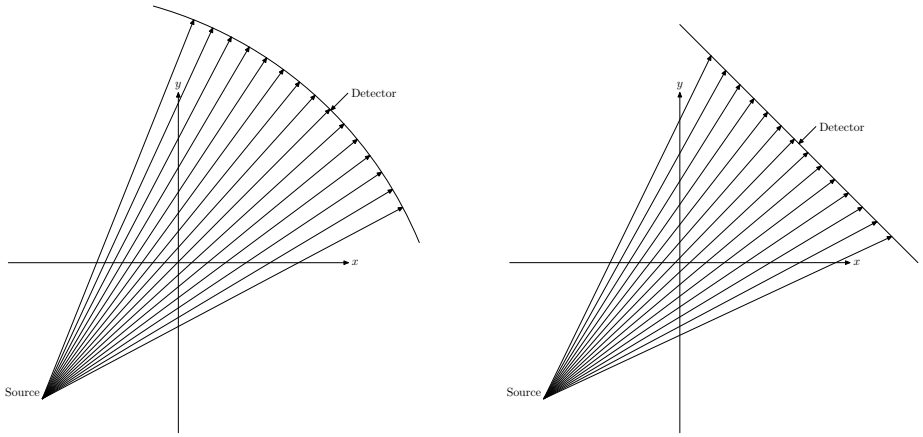


Figure 2.9: *Two types of fan-beams. Left: a curved detector. Right: a linear detector.*

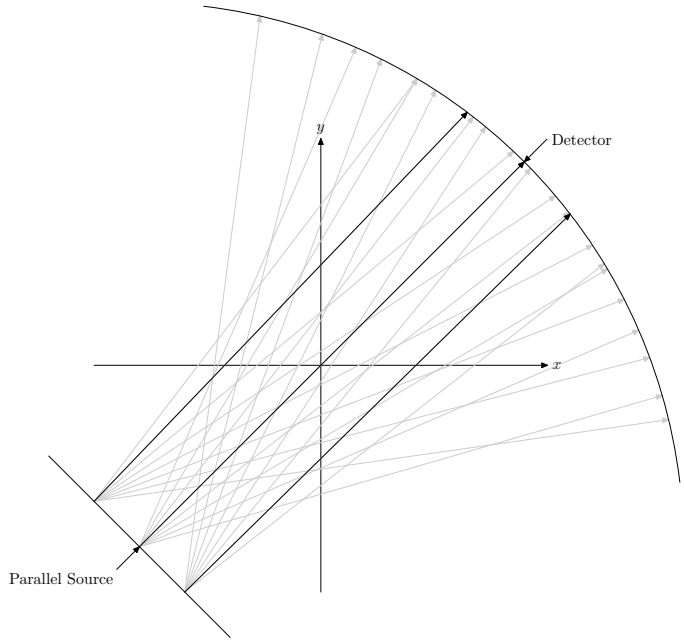


Figure 2.10: *Rebinning: the black rays are parallel and thus belong to the same bin.*

2.4.2 Rebinning

Another approach on reconstruction from non-parallel beams is based upon the idea that rays from different positions of the source can be parallel and thus can be thought of as rays from a parallel acquisition technique, see Figure 2.10. All parallel rays are binned together and the problem is then solved as if the data originated from parallel beams. Details of this method can be found in [5, Ch. 7.7.1–7.7.2] and [14, p. 172–174].

2.4.3 Feldkamp Algorithm

Until now only 2-dimensional acquisition techniques, such as parallel and fanbeam, have been discussed. In practice however cone-beams are used intensively. The Feldkamp algorithm, or also refereed to as the FDK cone-beam reconstruction algorithm after Feldkamp, Davis and Kress who proposed the algorithm in 1984, is an algorithm which approximates the exact reconstruction problem. The cone-beam projections are assumed to be 2-dimensional fan-beams in slices of the object which can then be reconstructed by using an adapted FBP. The resulting slices are back projected using a 3-dimensional cone-beam back projection to form the original 3-dimensional object. More details about this algorithm can be found in [7] and [5, Ch. 8.6.2–8.6.4].

Chapter 3

Algebraic Reconstruction Methods

This chapter will present a different approach to solving the reconstruction problem from Chapter 2. The analytical reconstruction methods assumed that the image or object is described by a certain function and Fourier analysis was used to derive this function from projections. Algebraic reconstructions methods (ARMs) employ the idea that the projections of an image or object can be written as a system of linear equations. Subsequently, the solution to this system is approximated by some iterative method. While the computational time of the algebraic methods is significantly higher than for the analytical methods, the results are also better when one uses a small number of projections. This is beneficial, for it is not always possible or desirable to take many projections. ARMs are also less prone to noise than the analytical methods [16].

Consider Figure 3.1. Suppose an image $f(x, y)$ is scanned using a total of M rays. A (square) grid is superimposed on $f(x, y)$ which divides the image in $n \times n = N$ square cells, which will also be referred to as pixels. Rays $p_i, i \in \{1, \ldots, M\}$, are modelled as stripes with width τ that run through $f(x, y)$. Each cell $f_j, j \in \{1, \ldots, N\}$ has area $\delta \times \delta$, it is assumed that in each cell $f(x, y)$ is constant. The reconstruction problem is written as

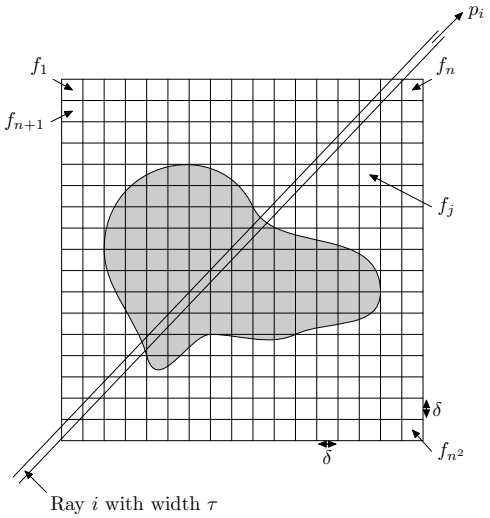


Figure 3.1: *The superimposed grid on $f(x, y)$.*

$$W\mathbf{f} = \mathbf{p}. \tag{3.1}$$

Here $\mathbf{p}$ is a vector containing all the (measured) projection values, hence the length of the vector is M. The vector $\mathbf{f}$ contains the (approximated) values of $f(x, y)$ per cell and thus is N long. The matrix W will be referred to as the weight matrix, this matrix contains per entry w_{ij} the weight cell j has corresponding to ray i, it is assumed that $w_{ij} \geq 0$. Each of the N cells will have M weights, hence $W \in \mathbb{R}^{M \times N}$. In tomography W is often too big to construct completely due to memory limitations. It is, however, possible to construct the w_{ij} on the fly during computations.

In practice $M \neq N$ and thus (3.1) is an under- or overdetermined system. In the case of an underdetermined system there are more unknowns than equations, hence if a solution exists, it is not unique. Overdetermined systems are systems with more equations than variables. An overdetermined system, if consistent and of full *rank* (see Section 3.2.1), yields a unique solution. It might even happen that an overdetermined system has infinitely many solutions. Note that the concepts of over- and underdetermination do not provide any information about the linear (in)dependency of the equations. Sometimes overdetermined systems are, erroneously, identified as inconsistent.

3.1 Kaczmarz's Method

Stefan Kaczmarz was a Polish mathematician who proposed a method for solving linear systems in 1937. This method, from now the *Kaczmarz's method*, was rediscovered and first introduced in the open literature by Gordon, Bender, and Herman in 1970 who used it to solve (3.1) [14, p. 204]. The method is based on viewing the solution $\mathbf{f}$ as a point in a N dimensional space which is the intersection of M hyperplanes. The hyperplanes are described by the equations of the linear system (3.1):

$$\begin{aligned} w_{11}f_1 + w_{12}f_2 + \cdots + w_{1N}f_N &= p_1 \\ w_{21}f_1 + w_{22}f_2 + \cdots + w_{2N}f_N &= p_2 \\ &\vdots \\ w_{M1}f_1 + w_{M2}f_2 + \cdots + w_{MN}f_N &= p_M. \end{aligned} \tag{3.2}$$

Note that if $M < N$ one is faced with an underdetermined problem. Then the intersection of these planes, if it exists, is not a point but a higher dimensional object, and thus infinitely many solutions are possible. Kaczmarz's method is iterative, thus a sequence of approximate solution vectors $(\mathbf{f}^0, \mathbf{f}^1, \ldots, \mathbf{f}^k)$ is generated such that $\lim_{k\to\infty} \mathbf{f}^k = \mathbf{f}$ assuming $\mathbf{f}$ is the unique solution to (3.1). Starting from an initial guess $\mathbf{f}^0$ the next approximation $\mathbf{f}^1$ is obtained by projecting $\mathbf{f}^0$ orthogonal onto the hyperplane described by the first equality in (3.2). This constitutes one iteration of the method. Subsequently $\mathbf{f}^2$ is obtained by projecting the vector $\mathbf{f}^1$ orthogonal onto the hyperplane given by the second equation in (3.2). After one has projected the subsequent approximate solutions onto all the hyperplanes the result $\mathbf{f}^M$ is again projected onto the first hyperplane of the system. The process is repeated until some stopping criterion is met, e.g. the

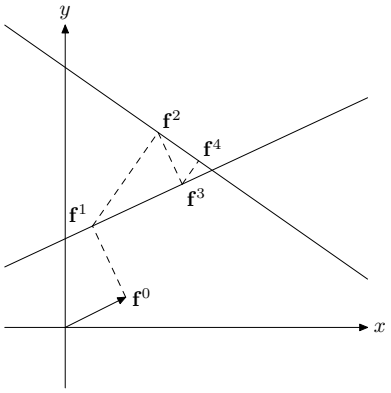


Figure 3.2: *The workings of the Kaczmarz's method in two dimensions.*

residual $\mathbf{r} = \mathbf{p} - W\mathbf{f}^k$ is smaller than some predetermined threshold. Figure 3.2 shows how the method works for a linear system with 2 unknowns and 2 equations.

Let $\mathbf{w}_{i,:} = (w_{i1}\ w_{i2}\ \ldots\ w_{iN})^T$ be the i-th row of W such that $W = \begin{pmatrix} \mathbf{w}_{1,:}^T \\ \mathbf{w}_{2,:}^T \\ \vdots \\ \mathbf{w}_{M,:}^T \end{pmatrix}$. The i-th iteration, $i \in \{1, \ldots, M\}$, of the Kaczmarz's method can be expressed as:

$$\mathbf{f}^i = \mathbf{f}^{i-1} + \frac{p_i - \left\langle \mathbf{f}^{i-1}, \mathbf{w}_{i,:} \right\rangle}{\left\langle \mathbf{w}_{i,:}, \mathbf{w}_{i,:} \right\rangle} \mathbf{w}_{i,:}. \tag{3.3}$$

Note that this expression is composed of two vectors. The vector $\mathbf{w}_{i,:}$ is orthogonal to the hyperplane $\mathbf{w}_{i,:}\mathbf{f} = p_i$ given by the i-th equation of (3.2). Hence the vectors described by $\mathbf{f}^{i-1} + \alpha\mathbf{w}_{i,:}$ lie on a line which is orthogonal to that hyperplane. The quantity $\alpha = \dfrac{p_i - \left\langle \mathbf{f}^{i-1}, \mathbf{w}_{i,:} \right\rangle}{\left\langle \mathbf{w}_{i,:}, \mathbf{w}_{i,:} \right\rangle}$ ensures that the sum of both vectors coincides with the hyperplane, thus (3.3) projects $\mathbf{f}^{i-1}$ orthogonal onto the hyperplane $\mathbf{w}_{i,:}\mathbf{f} = p_i$. For a full geometric description of how this quantity α comes about one is referred to [17, p. 278–280]. Note that $i = 1, 2, \ldots, M$ and $\mathbf{f}^k$ is N long, hence, it is possible that one update alters multiple entries of $\mathbf{f}^k$. The interpretation is that the update is computed according to the projection data, thus for every ray. All the cells j which affect this ray i have some weight $w_{ij} \neq 0$ and thus all those entries j in $\mathbf{f}^k$ are altered.

Note that in (3.3) $i \in \{1, \ldots, M\}$ for clarity, one can easily adapt the expression to allow for iterations with iteration number $i > M$. Set $\tilde{i} = (i-1) \mod (M) + 1$, then

$$\mathbf{f}^i = \mathbf{f}^{i-1} + \frac{p_{\tilde{i}} - \left\langle \mathbf{f}^{i-1}, \mathbf{w}_{\tilde{i},:} \right\rangle}{\left\langle \mathbf{w}_{\tilde{i},:}, \mathbf{w}_{\tilde{i},:} \right\rangle} \mathbf{w}_{\tilde{i},:}. \tag{3.4}$$

The denominator of (3.3) (and consequently (3.4)) can be assumed to be unequal to zero. Indeed, if $\langle \mathbf{w}_{i,:}, \mathbf{w}_{i,:} \rangle = \sum_{j=1}^{N} w_{ij}^2 = 0$ this would mean that $w_{ij} = 0$ for every j and thus equation j in (3.2) would be $\mathbf{0f} = p_j$, with $\mathbf{0}$ the zero-vector. But this in turn represents a ray hitting no cells,

consequently one can thus safely conclude that $p_j = 0$. And hence removing this equation from (3.2) would not result in any loss of information.

This method only converges to a solution of $W\mathbf{f} = \mathbf{p}$ if the system has an exact solution [25, p. 59]. Unfortunately the projections obtained in tomography often contain noise which causes $\mathbf{p}$ not to lie in the range of W and thus no solution exists. In this case the sequence of approximate solutions will keep fluctuating. And even if convergence is guaranteed, it does not mean that a solution is found quickly. Consider for example a system with two unknowns (cf. Figure 3.2) where the two hyperplanes (lines) differ by a very small angle. The Kaczmarz's method would then convergence very slowly to the solution. If the two lines were perpendicular then convergence would be reached within two iterations. One can imagine that hyperplanes arising from two adjacent rays will most likely be nearly parallel because of high correlation between information, hence it would be better to project onto the hyperplanes in a different order, for example random, than in ascending order.

3.2 Ill-conditioned Systems

As noted, problems originating from tomography result in the system $W\mathbf{f} = \mathbf{p}$, $W \in \mathbb{R}^{M\times N}$ and generally $M \neq N$. The problems that will be considered in this work will mostly be underdetermined, i.e. $M < N$. Thus the number of rays is smaller than the number of cells. Moreover, since the matrix W will be such that there usually is no unique solution to the problem, the challenge is then find the best approximate solution. To analyse and motivate the methods that will be presented in this chapter some linear algebra is revised.

3.2.1 Linear Algebra

This section will list some useful definitions and properties from the linear algebra that will be used to analyse the given problems. Throughout this section let $W \in \mathbb{R}^{M\times N}, \mathbf{f} \in \mathbb{R}^N$ and $\mathbf{p} \in \mathbb{R}^M$ with $M < N$ as before.

Definition 3.1. The *row rank* of a matrix W is the number of linearly independent row vectors in W. The *column rank* of W is the number of linearly independent column vectors in W.

One fundamental property of linear algebra is that the row rank and the column rank are equal.

It is assumed that one is familiar with the definitions of a *inner product* and a *norm*. For completeness, the inner product of two vectors $\mathbf{x}$ and $\mathbf{y}$ is denoted $\langle \mathbf{x}, \mathbf{y}\rangle = \mathbf{x}^T\mathbf{y}$. The Euclidean norm of $\mathbf{x}$ is denoted $\|\mathbf{x}\|_2 = \sqrt{\sum_i x_i^2} = \sqrt{\langle x, x\rangle}$.

Definition 3.2. Two vectors $\mathbf{x}$ and $\mathbf{y}$ are *orthogonal* if $\langle \mathbf{x}, \mathbf{y}\rangle = 0$. This is sometimes denoted as $\mathbf{x} \perp \mathbf{y}$.

Definition 3.3. The subspace of all vectors $\mathbf{p} \in \mathbb{R}^M$ with $W\mathbf{f} = \mathbf{p}$, $\mathbf{f} \in \mathbb{R}^N$ is called the *range* of W and denoted $\mathcal{R}(W)$.

The dimension of the range of a matrix is the row rank of the matrix. One often uses the term *rank* for the row rank.

Definition 3.4. The subspace of all vectors $\mathbf{f} \in \mathbb{R}^N$ with $W\mathbf{f} = \mathbf{0}$ is called the *nullspace* of W and denoted $\mathcal{N}(W)$.

The two concepts of range and nullspace are related by the following theorem.

Theorem 3.5. *Let $W \in \mathbb{R}^{M\times N}$, then the following relationship holds*

$$M = \dim \mathcal{R}(W) + \dim \mathcal{N}(W) \tag{3.5}$$

where dim *denotes the dimension of the corresponding subspace.*

3.2.2 Least Squares Solutions

The range of W are all the projections arising from scanning an object with the operator W. One can easily imagine that it is possible that multiple objects result in the same projection when scanned. Imagine $\mathcal{R}(W)$ as a plane in some three-dimensional space. The operator W, arising from the discretization of projection process, can only construct the projections in $\mathcal{R}(W)$. The measured projection $\mathbf{p}$, however, might not be an element of this range. In that case one cannot find an object $\mathbf{f}$ for which $W\mathbf{f} = \mathbf{p}$, i.e. the problem has no exact solution. One then often resorts to finding a *least squares solution*. This solution is a vector for which the residual has minimal norm, i.e. $\|\mathbf{p} - W\mathbf{f}\|_2$ is minimal, see Figure 3.3. In the problems considered in this work one generally also has that W is not of full column rank, in that case the least squares solution is not unique. There is, however, a unique *minimal norm least squares solution*. This particular vector is the solution one is interested in.

Now some properties of the least squares solution will be given assuming that W is not of full rank. From Figure 3.3 it is clear that the vector $W\mathbf{f}' - \mathbf{p}$ is orthogonal to $\mathcal{R}(W)$ and, indeed, if $\mathbf{f}$ is a least squares solution of (3.1) then $W\mathbf{f} - \mathbf{p} \perp \mathcal{R}(W)$. From this one can see that $W\mathbf{f}$ is the same for all least squares solutions $\mathbf{f}$. The expression $W\mathbf{f} - \mathbf{p} \perp \mathcal{R}(W)$ can be rewritten as

$$\begin{aligned} W\mathbf{f} - \mathbf{p} \perp \mathcal{R}(W) &\Leftrightarrow \langle W\mathbf{f} - \mathbf{p}, W\mathbf{q}\rangle = 0 \quad \forall \mathbf{q} \in \mathbb{R}^N \\ &\Leftrightarrow \left\langle W^T(W\mathbf{f} - \mathbf{p}), \mathbf{q}\right\rangle = 0 \quad \forall \mathbf{q} \in \mathbb{R}^N \\ &\Leftrightarrow W^T(W\mathbf{f} - \mathbf{p}) = 0 \\ &\Leftrightarrow W^TW\mathbf{f} = W^T\mathbf{p}. \end{aligned} \tag{3.6}$$

The third line is due to the fact that an inner product is positive definite. From (3.6) it follows that $\mathbf{f}$ is a least squares solution if and only if $W^TW\mathbf{f} = W^T\mathbf{p}$. This system of equations is known as the *normal equations* and will be properly introduced in Section 3.2.4.

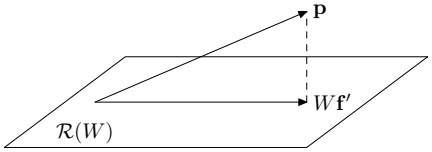


Figure 3.3: *The vector $\mathbf{f}'$ is a least squares solution for the problem $W\mathbf{f} = \mathbf{p}$.*

Note that if $\mathbf{f}$ is a least squares solution of $W\mathbf{f} = \mathbf{p}$ that then all other least squares solutions can be written as $\mathbf{f} + \mathbf{q}$ with $\mathbf{q} \in \mathcal{N}(W)$ since $W(\mathbf{f} + \mathbf{q}) = W\mathbf{f} + W\mathbf{q} = W\mathbf{f}$. From this one can conclude that $\mathbf{f}$ is the *minimum norm* least squares solution of (3.1) if and only if $\mathbf{f} \perp \mathcal{N}(W)$ which is equivalent to $\mathbf{f} \in \mathcal{R}(W^T)$.

3.2.3 Matrix Properties

This section will list some useful matrix properties. Knowledge about eigenvalues and eigenvectors is assumed.

Definition 3.6. A matrix $A \in \mathbb{R}^{n\times n}$ is said to be *positive definite* if $\langle A\mathbf{x}, \mathbf{x}\rangle > 0$ for all $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{x} \neq 0$.

Note that if $\mathbf{x}$ is an eigenvector of A with corresponding eigenvalue λ then $A\mathbf{x} = \lambda\mathbf{x}$ and thus $\mathbf{x}^T A\mathbf{x} = \lambda\|x\|_2^2 > 0$. From this it follows that $\lambda > 0$ and thus all eigenvalues of A are positive.

Definition 3.7. A matrix $A \in \mathbb{R}^{n\times n}$ is said to be *positive semidefinite* if $\langle A\mathbf{x}, \mathbf{x}\rangle \geq 0$ for all $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{x} \neq 0$.

By similar reasoning concerning the eigenvalues of positive definite matrices one can conclude that the eigenvalues λ of A are non-negative.

Definition 3.8. The *spectral radius* of $A \in \mathbb{R}^{n\times n}$, denoted $\rho(A)$, is defined as

$$\rho(A) = \max_{\lambda_i, i=1,\ldots,n} |\lambda_i| \tag{3.7}$$

where λ_i are the eigenvalues of A.

For the spectral radius one has the property $\rho(A) \leq \|A\|$, where $\|\cdot\|$ is any operator norm. For the proof of this claim see [6, p. 279].

3.2.4 Normal Equations

In tomography one generally faces the linear system $W\mathbf{f} = \mathbf{p}$ where the matrix W is not square, let alone symmetric. To be able to solve this system one can consider a transformation of the original system.

Definition 3.9. Let the linear system $W\mathbf{f} = \mathbf{p}$, $W \in \mathbb{R}^{M\times N}, \mathbf{f} \in \mathbb{R}^N$ and $\mathbf{p} \in \mathbb{R}^M$ be given, then the least squares solution to this system is the solution to the system

$$W^T W\mathbf{f} = W^T\mathbf{b}. \tag{3.8}$$

The linear system (3.8) is referred to as the system of *normal equations* associated with the least-squares problem

$$\min_{\mathbf{f}\in\mathbb{R}^N} \|\mathbf{p} - W\mathbf{f}\|_2. \tag{3.9}$$

Note that the matrix W^TW is symmetric. If W is of full rank then W^TW is also positive definite since $\langle W^TW\mathbf{f}, \mathbf{f}\rangle = \langle W\mathbf{f}, W\mathbf{f}\rangle > 0$ for all $\mathbf{f} \neq 0$. For rank deficient W the normal equations yield a symmetric positive semidefinite system. The matrix W^TW is thus symmetric positive definite (SPD) or symmetric positive semidefinite (SPSD) which are very attractive properties for solving linear systems. The system (3.8) is typically used to solve problem (3.9) for an overdetermined system W [22, p. 245]. An alternative to the normal equations, from now on the *alternative normal equations* is obtained by setting $\mathbf{f} = W^T\mathbf{u}$. Solving the system

$$WW^T\mathbf{u} = \mathbf{p} \tag{3.10}$$

for $\mathbf{u}$ then yields the solution $\mathbf{f} = W^T\mathbf{u}$ to the original system. This method only works if system 3.10 is consistent. This alternative is used to solve underdetermined systems [22, p. 246]. Unlike the normal equations, which solve (3.9) and hence minimizes the residual, this alternative minimizes the error $\mathbf{e}^k = \mathbf{f} - W^T\mathbf{u}^k = \mathbf{f} - \mathbf{f}^k$ of the system, where $\mathbf{f}$ is the exact solution to the system. Note that this exact solution is only unique if the matrix W is of full rank. Hence, given W is of full rank, it solves

$$\min_{\mathbf{u}\in\mathbb{R}^m} \|\mathbf{f} - W^T\mathbf{u}\|_2. \tag{3.11}$$

Methods derived from (3.8) are often labelled NR since they employ the "Normal" equations to minimize the "Residual". On the other hand a method derived from (3.10) is labelled NE since it uses the "Normal" equations to minimize the "Error". Note that (3.11) only holds in case W is of full rank, something one in tomography generally does not have.

3.2.5 Singular Value Decomposition

Lastly, the concept of *singular value decompositions* (SVD) is introduced. Again it is assumed that the reader is familiar with eigenvalues and eigenvectors. The SVD of a matrix decomposes the matrix operation in three simpler matrices (operators). The singular value decomposition of W is given as

$$W = U\Sigma V^T, \tag{3.12}$$

with $U \in \mathbb{R}^{M\times M}$ an orthogonal matrix, $V \in \mathbb{R}^{N\times N}$ an orthogonal matrix and $\Sigma \in \mathbb{R}^{M\times N}$ a diagonal matrix with diagonal elements $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_M$. Note that it is assumed that $M < N$. Schematically the SVD of W is given as

$$\begin{bmatrix} \quad W \quad \end{bmatrix} = \begin{bmatrix} \ U \ \end{bmatrix} \begin{bmatrix} \quad \Sigma \quad \end{bmatrix} \begin{bmatrix} \quad V^T \quad \\ \\ \end{bmatrix}. \tag{3.13}$$

The columns of U respectively V are the left and right *singular vectors*, respectively, of W, moreover they are the eigenvectors of WW^T and W^TW, respectively. The σ_j are the *singular values* of W and the systems WW^T and W^TW have eigenvalues σ_j^2.

Definition 3.10. The *condition number* of a matrix W, denoted $\kappa(W)$, is defined as the ratio between the largest singular value σ_1 and the smallest σ_M, i.e.

$$\kappa(W) = \frac{\sigma_1}{\sigma_M}. \tag{3.14}$$

A system $W\mathbf{f} = \mathbf{p}$ is said to be *ill-conditioned* if $\kappa(W)$ is large and *well-conditioned* if $\kappa(W)$ is small. The condition number can be thought of as giving a measure for the linear dependency of the equations in the corresponding system. A large condition number implies that some (or all) equations of the system are almost linearly dependent.

One has that the rank of W is equal to $n \leq M$ if and only if $\sigma_n \neq 0$ and $\sigma_j = 0$ for all $n+1 \leq j \leq M$. When $\sigma_{n+1}, \ldots, \sigma_M$ are very small with respect to $\sigma_1, \ldots, \sigma_n$ one says that W is *effectively* of rank n since W is hardly distinguishable from an operator of rank n for computational purposes [25, p. 53]. If W is (effectively) of rank $n < M$ then $\kappa(W)$ would most likely be large (since σ_M is either zero or very small compared to σ_1) and thus the corresponding system is ill-conditioned.

The SVD of a matrix is a powerful tool which can be used to solve both theoretical and practical problems. Later on the SVD of W will be used to analyse the effects of noise and motivate the use of regularization.

3.3 Ambiguous Terminology

The ARMs ART, SIRT and SART that will be considered in what follows originate from various research fields that are involved with tomography. As a result the methods are not unambiguously defined. Some methods which are clearly different are referred to with the same name which can cause a lot of confusion. When dealing with these methods one should always keep this in mind.

Another thing one should be aware of is that in the literature ARMs are often called algebraic reconstruction techniques. This can be confusing since ART is an abbreviation of Algebraic Reconstruction Technique. To avoid ambiguity the class of methods is denoted ARMs while ART, SIRT and SART are actual methods.

The ART method is nearly always defined as Kaczmarz's method. This is also the definition which will be used in this work. For details one is referred to [17, p. 283–284].

The SIRT method is a method for which one can find numerous definitions. One always has to check which definition is used since convergence proofs of one method may (and in general will) not hold for another method. Van der Sluis and van der Vorst introduced SIRT as a family of methods and proved the convergence for this family [25]. Another definition of a SIRT family is from Hansen [13]. Both definitions have some similarities but are not the same. Gregor and Benson [10] present SIRT as an instance which is in both the Hansen as the van der Vorst family this work will use that definition.

The final method, SART, was first defined in 1984 by Andersen and Kak. Jiang and Wang claimed to be the first to have proven the convergence of SART in 2003 [16]. Their definition of

SART, however, is equivalent to Gregor's SIRT in [10] and in turn equivalent to van der Vorst's SIRT with $\alpha = 1$, [25, p. 59] and for that family convergence was already established in 1987. The definition of SART given later on is different, SART is then viewed as a hybrid between ART and SIRT. This definition coincides with the definition that Andersen and Kak proposed in [1, p. 87].

3.4 Basic Iterative Methods

This section will present three iterative methods which exploit Kaczmarz's method. The Algebraic Reconstruction Technique (ART) is the same as Kaczmarz's method. The Simultaneous Iterative Reconstruction Technique (SIRT) is a method for which convergence will be proven. The Simultaneous Algebraic Reconstruction Technique (SART) combines ART and SIRT. Before these ARMs are introduced the Jacobi and Gauss-Seidel methods are revised since they are closely related to ART, SIRT and SART.

3.4.1 Jacobi and Gauss-Seidel method

The Jacobi and Gauss-Seidel method are iterative methods which can be used to solve the system $A\mathbf{x} = \mathbf{b}$ for $A \in \mathbb{R}^{n\times n}$ and $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$. Every square matrix A can be decomposed as $A = D - E - F$ where D is a diagonal matrix formed with the diagonal elements of A, it is assumed that these elements are all nonzero. $-E$ is the strict lower part of A and $-F$ is the strict upper part of A. Thus:

$$A = \begin{pmatrix} \ddots & & -F \\ & D & \\ -E & & \ddots \end{pmatrix}. \tag{3.15}$$

The Jacobi method chooses the i-th component of the next approximation $x_i^k, k \geq 1$, such that the i-th component of the next residual vector is equal to zero, thus $(\mathbf{r}^k)_i = (\mathbf{b} - A\mathbf{x}^k)_i = 0$. This can be written as

$$a_{ii}x_i^k = -\sum_{j=1,\ j\neq i}^{n} a_{ij}x_i^{k-1} + b_i \Rightarrow x_i^k = \frac{1}{a_{ii}}\left(-\sum_{j=1,\ j\neq i}^{n} a_{ij}x_i^{k-1} + b_i\right), \tag{3.16}$$

where a_{ij} is the element in the i-th row and j-th column of A. Or equivalently, in matrix notation:

$$\mathbf{x}^k = D^{-1}(E + F)\mathbf{x}^{k-1} + D^{-1}\mathbf{b}. \tag{3.17}$$

The Gauss-Seidel method is similar to the Jacobi method but uses the newly obtained values of $\mathbf{x}^k$ whereas Jacobi only uses values of $\mathbf{x}^{k-1}$. Assuming the updates are carried out in ascending order of i this results into the following expression

$$x_i^k = \frac{1}{a_{ii}} \left(-\sum_{j=1}^{i-1} a_{ij} x_i^k - \sum_{j=i+1}^{n} a_{ij} x_i^{k-1} + b_i \right), \tag{3.18}$$

or in matrix notation:

$$\mathbf{x}^k = (D+E)^{-1} F \mathbf{x}^{k-1} + (D+E)^{-1} \mathbf{b}. \tag{3.19}$$

If the updates were performed in descending order this method is referred to as backward Gauss-Seidel. The symmetric Gauss-Seidel method consists of executing a forward sweep followed by a backward sweep.

In general one can write this class of iterative methods as

$$M\mathbf{x}^k = N\mathbf{x}^{k-1} + b \Rightarrow \mathbf{x}^k = M^{-1}N\mathbf{x}^{k-1} + M^{-1}b, \tag{3.20}$$

with *splitting* $A = M - N$.

Shortly it will be shown that ART and SIRT are actually a Gauss-Seidel and Jacobi method, respectively, applied to the system of the alternative normal equations.

3.4.2 ART

As stated earlier: ART is the same as Kaczmarz's method. Here the ART algorithm will be derived from another point of view and it will be shown that ART is actually the Gauss-Seidel method applied to system (3.10) of the alternative normal equations. This derivation is due to [22, p.247–248] and will closely follow the reasoning in said source. Intuitively one can see that ART is similar to Gauss-Seidel in that it uses all the newly obtained information for the update of the i-th component.

Solving (3.10) with Gauss-Seidel yields for i-th update $\mathbf{u}^k = \mathbf{u}^{k-1} + \delta_i \mathbf{e}_i$, where $\mathbf{e}_i$ is the i-th unit vector and $i = (k-1) \mod (M) + 1$ (cf. $\tilde{i}$ in (3.4)). The scaler δ_i is chosen such that the i-th component of the residual vector $\mathbf{r}^k = \mathbf{p} - WW^T\mathbf{u}^k = 0$, or equivalently

$$\begin{aligned}
\left(\mathbf{p} - WW^T(\mathbf{u}^{k-1} + \delta_i \mathbf{e}_i)\right)_i &= 0 && \Rightarrow \\
\left\langle \mathbf{p} - WW^T(\mathbf{u}^{k-1} + \delta_i \mathbf{e}_i), \mathbf{e}_i \right\rangle &= 0 && \Rightarrow \\
\left\langle WW^T \delta_i \mathbf{e}_i, \mathbf{e}_i \right\rangle &= \left\langle \mathbf{r}^{k-1}, \mathbf{e}_i \right\rangle && \Rightarrow \\
\delta_i \left\langle W^T\mathbf{e}_i, W^T\mathbf{e}_i \right\rangle &= \left\langle \mathbf{r}^{k-1}, \mathbf{e}_i \right\rangle && \Rightarrow \\
\delta_i &= \frac{\left\langle \mathbf{r}^{k-1}, \mathbf{e}_i \right\rangle}{\|W^T\mathbf{e}_i\|_2^2}
\end{aligned}$$

To analyse this re-substitute the definition of $\mathbf{r}^{k-1}$, this gives

$$\delta_i = \frac{p_i - \langle W^T \mathbf{u}^{k-1}, W^T \mathbf{e}_i \rangle}{\|W^T \mathbf{e}_i\|_2^2}. \tag{3.21}$$

Since $\mathbf{f} = W^T \mathbf{u}$ one finds $\mathbf{f}^k = \mathbf{f}^{k-1} + \delta_i W^T \mathbf{e}_i$:

$$\mathbf{f}^k = \mathbf{f}^{k-1} + \frac{p_i - \langle \mathbf{f}^{k-1}, W^T \mathbf{e}_i \rangle}{\|W^T \mathbf{e}_i\|_2^2} W^T \mathbf{e}_i. \tag{3.22}$$

Note that the updates are performed in ascending order $i = 1, 2, \ldots, M$ and are immediately stored in the new vector $\mathbf{f}^k = W^T \mathbf{u}^k$, hence this yields a Gauss-Seidel method which is applied to the system $WW^T \mathbf{u} = \mathbf{p}$. Inspection of (3.22) confirms that this expression is actually equivalent to (3.3). Simply observe that $W^T \mathbf{e}_i = \mathbf{w}_{i,:}$, $\langle \mathbf{f}^{k-1}, W^T \mathbf{e}_i \rangle = \langle \mathbf{f}^{k-1}, \mathbf{w}_{i,:} \rangle$ and $\|W^T \mathbf{e}_i\|_2^2 = \langle \mathbf{w}_{i,:}, \mathbf{w}_{i,:} \rangle$. The ART algorithm, in pseudo code, is listed below.

ART

Choose $\mathbf{f}^0$
$k \leftarrow 1$
while termination criterion is not met **do**
  for $i = 1, 2, \ldots, M$ **do**
    $\delta_i = \left(p_i - \langle \mathbf{f}^{k-1}, \mathbf{w}_{i,:} \rangle\right) / \langle \mathbf{w}_{i,:}, \mathbf{w}_{i,:} \rangle$
    $\mathbf{f}^k \leftarrow \mathbf{f}^{k-1} + \delta_i \mathbf{w}_{i,:}$
    $k \leftarrow k + 1$
  end for
end while

Note that it is unclear what one iteration of the ART algorithm constitutes, from now on one iteration of the ART algorithm will be regarded as executing the while-loop once (thus updating all the components). Recall from the discussion at the end of Section 3.1 that updating the components in random order order rather than ascending order, as in the algorithm above, will yield faster convergence. This adaptation is easily implemented, one only has to change to order in which the for-loop traverses the various rays.

As noted in Section 3.1 Kaczmarz's method, and thus ART, only converges if the system $W\mathbf{f} = \mathbf{p}$ has a unique solution. This is only the case when one has a consistent system of full rank. In this case, due to that fact one uses the alternative normal equations (cf. Section 3.2.4), problem (3.11) is solved and hence the error is minimized. In the case the system is not of full rank convergence is not guaranteed.

3.4.3 SIRT

SIRT is differs from ART in that it updates $\mathbf{f}^{k-1}$ to $\mathbf{f}^k$ only after all the N new components have been computed, this suggests a Jacobi method. Indeed, SIRT actually stems from Jacobi applied to the alternative normal equations (3.10) $WW^T \mathbf{u} = \mathbf{p}$.

In the previous section it was established that in ART the update for the j-th component was

$$\delta_i = \frac{\langle \mathbf{r}^{k-1}, \mathbf{e}_i \rangle}{\|W^T \mathbf{e}_i\|_2^2}. \tag{3.23}$$

Instead of the Euclidean norm in the denominator, the 1-norm (or also taxicab norm) is used for SIRT, i.e. $\|\mathbf{x}\|_1 = \sum_i |x_i|$. This yields:

$$\delta_i = \frac{\langle \mathbf{r}^{k-1}, \mathbf{e}_i \rangle}{\|W^T \mathbf{e}_i\|_1^2}. \tag{3.24}$$

Since one deals with a Jacobi iteration the actual update is performed after all the other updates have been computed, thus for every update the same residual is used. After the updates have been computed for $i = 1, \ldots, M$ one finds the new approximation as

$$\mathbf{d} = \sum_{i=1}^{M} \frac{\langle \mathbf{r}^{k-1}, \mathbf{e}_i \rangle}{\sum_{h=1}^{N} |w_{ih}|} \mathbf{w}_{i,:} \tag{3.25}$$

$$\mathbf{f}^{k+1} = \mathbf{f}^k + C\mathbf{d} \tag{3.26}$$

Recall that $w_{ij} \geq 0$ was assumed. Note that in this case every component (pixel) j, will receive updates from multiple rays i, these are precisely the rays which are affected by that pixel. At the end of the cycle SIRT avarages over these updates proportional to total weight the cell has, thus proportional to the j-th column sum of W, i.e. C in 3.26 is a diagonal matrix containing the inverse of these sums on its diagonal: $c_j j = 1/\sum_{i=1}^{M}$. Clearly, if such a column sum is equal to zero one would have that no ray passes through cell j and thus it would make no sense to have this variable in the system since the cell is not part of the scanned image. And indeed, if one were to incorporate this cell (and thus the zero column) into the system then one would never find a unique solution, for if the system without this variable has a unique solution, the system with this variable has infinitely many solutions, namely the unique solution found earlier where one can freely choose the extra variable. The new averaged update for component j is given by

$$f_j^k = f_j^{k-1} + \frac{1}{\sum_{i=1}^{M} w_{ij}} \sum_{i=1}^{M} \frac{w_{ij} r_i^{k-1}}{\sum_{h=1}^{N} w_{ih}}. \tag{3.27}$$

Although this expression looks messy, it is actually quite elegant. Moreover, one SIRT iteration, i.e. computing the increments for all the components and obtaining the new approximation $\mathbf{f}^k$, can be conveniently written in matrix-vector notation. Let $R \in \mathbb{R}^M, C \in \mathbb{R}^N$ be a diagonal matrices with $r_{ii} = 1/\sum_{j=1}^{N} w_{ij}$ and $c_{jj} = 1/\sum_{i=1}^{M} w_{ij}$. Note that $\sum_{i=1}^{M} w_{ij} > 0$ and $\sum_{j=1}^{N} w_{ij} > 0$. With these matrices one SIRT iteration can be written as

$$\mathbf{f}^k = \mathbf{f}^{k-1} + CW^T R\mathbf{r}^{k-1}. \tag{3.28}$$

And hence the algorithm can be formulated as

SIRT

Choose $\mathbf{f}^0$
$\mathbf{r}^0 \leftarrow \mathbf{p} - W\mathbf{f}^0$
$k \leftarrow 1$
while termination criterion is not met **do**
  $\mathbf{f}^k \leftarrow \mathbf{f}^{k-1} + CW^T R\mathbf{r}^{k-1}$
  $k \leftarrow k + 1$
end while

Now the convergence of SIRT will investigated, the following reasoning follows the lines of the prove given in [10]. One can rewrite SIRT as

$$\begin{aligned}\mathbf{f}^k &= \mathbf{f}^{k-1} + CW^T R\mathbf{r}^{k-1} \\ &= \mathbf{f}^{k-1} + CW^T R\left(\mathbf{p} - W\mathbf{f}^{k-1}\right) \\ &= (I - CW^T RW)\mathbf{f}^{k-1} + CW^T R\mathbf{p}. \end{aligned} \tag{3.29}$$

Recall (3.20) to observe that in (3.29) SIRT is written as an iterative method with $M = I$ and $N = (I - CW^T RW)$. The matrix $(I - CW^T RW)$ is called the *iteration matrix*. This iterative procedure converges for any $\mathbf{p}$ and $\mathbf{f}^0$ if $\rho\left(I - CW^T RW\right) < 1$ [22, p. 112].

Let Λ be the set of all the eigenvalues of $CW^T RW$. Suppose μ is an eigenvalue of $(I - CW^T RW)$ then

$$\begin{aligned}(I - CW^T RW)\mathbf{f} = \mu\mathbf{f} \quad &\Leftrightarrow \\ -CW^T RW\mathbf{f} = \mu\mathbf{f} - I\mathbf{f} \quad &\Leftrightarrow \\ CW^T RW\mathbf{f} = (1-\mu)\mathbf{f}\end{aligned}$$

and hence $(1 - \mu) \in \Lambda$. Thus for the spectral radius of the iteration matrix one finds

$$\rho\left(I - CW^T RW\right) = \max_{\lambda\in\Lambda} |1 - \lambda|. \tag{3.30}$$

So if one were to show that $-1 < |1 - \lambda| < 1 \Rightarrow 0 < \lambda < 2$ then one has proven the convergence of (3.29) and thus SIRT. To that extent bounds for λ are deduced. First it is shown that $\lambda \geq 0$.

In the case the spectral radius of the iteration matrix is equal to 1 the iteration process may *stagnate*, i.e. subsequent errors do not change. This is a result of the fact that W is not of full rank. One has that of this $W^T W$ is SPSD and so is $W^T RW$. Fix $\lambda \in \Lambda$, one then finds

$$\begin{aligned} CW^T RW\mathbf{f} &= \lambda\mathbf{f} && \Leftrightarrow \\ C^{1/2}W^T RW\mathbf{f} &= \lambda C^{-1/2}\mathbf{f} && \Leftrightarrow \\ C^{1/2}W^T RWC^{1/2}\mathbf{h} &= \lambda\mathbf{h}. \end{aligned}$$

This matrix $C^{1/2}W^T RWC^{1/2}$ is positive semidefinite since

$$\left\langle C^{1/2}W^T RWC^{1/2}\mathbf{h}, \mathbf{h} \right\rangle = \left\langle R^{1/2}WC^{1/2}\mathbf{h}, R^{1/2}WC^{1/2}\mathbf{h} \right\rangle \geq 0. \tag{3.31}$$

And hence one finds $\lambda \geq 0$ for all $\lambda \in \Lambda$.

Now it will be shown that $\lambda \leq 1$. The spectral radius of a matrix is less than or equal to any operator norm (cf. Section 3.2.3), this yields

$$\rho(CW^T RW) \leq \|CW^T RW\|_\infty \leq \|CW^T\|_\infty \|RW\|_\infty. \tag{3.32}$$

The ∞-norm is given by $\|A\|_\infty = \max_j \sum_j |a_{ij}|$. In this case, since all entries of the matrices are positive, it is simply the largest row sum. Recall that C is the diagonal matrix with inverse column sums of W, or equivalently the inverse row sums of W^T and thus $\|CW^T\|_\infty = 1$. By similar reasoning one finds $\|RW\|_\infty = 1$ and thus $\lambda \leq 1$.

Since $0 \leq \lambda \leq 1$ is shown $0 \leq \lambda < 2$ follows. Thus it is proven that $\rho(I - CW^T RW) \leq 1$. Note that convergence is not guaranteed since one can still have stagnation of the iterative process.

3.4.4 SART

The Simultaneous Algebraic Reconstruction Technique is combination of ART and SIRT. In ART the update of $\mathbf{f}^k$ was determined by only looking at the contribution of the i-th ray for $i = 1, 2, \ldots, M$. SIRT computed the average contribution of all the rays i, $i = 1, 2, \ldots, M$, first, using the previous approximation $\mathbf{f}^{k-1}$, and then did an update to $\mathbf{f}^k$. SART is a combination of ART and SIRT in that the update of $\mathbf{f}^k$ is carried out per projection angle. Suppose the object is scanned using the angles $P = \{\theta_1, \theta_2, \ldots, \theta_p\}$ with R rays per angle such that $p \cdot R = M$. Then the update for $\mathbf{f}^k$ is computed for all rays per projection angle, thus the updates are performed per block of R projections. Hence this method is also referred to as a *block iterative method*. For the l-th block this results into the following expression for the j-th component of the next approximation update:

$$f_j^k = f_j^{k-1} + \frac{1}{\sum_{i=R\cdot(l-1)+1}^{R\cdot l} w_{ij}} \sum_{i=R\cdot(l-1)+1}^{R\cdot l} \frac{r_i^{k-1} w_{ij}}{\sum_{h=1}^{N} w_{ih}}. \tag{3.33}$$

Let C be a diagonal matrix with diagonal elements $c_{jj} = 1/\sum_{i=R\cdot(l-1)+1}^{R\cdot l} w_{ij}$. The following algorithm results

SART

Choose $\mathbf{f}^0$
$\mathbf{r}^0 \leftarrow \mathbf{p} - W\mathbf{f}^0$
$k \leftarrow 1$
while termination criterion is not met **do**
  for $l = 1, 2, \ldots, p$ **do**
    for $i = R \cdot (l-1) + 1, R \cdot (l-1) + 2, \ldots, R \cdot l$ **do**
      $\mathbf{f}^k \leftarrow \mathbf{f}^{k-1} + C \dfrac{r_i^{k-1}}{\sum_{h=1}^{N} w_{ih}} \mathbf{w}_{i,:}$
    end for
    $k \leftarrow k + 1$
  end for
end while

Note that as in with the ART algorithm it is unclear what constitutes one iteration. Also in this case one iteration is defined as executing the while-loop once, thus after all p blocks have been used.

3.4.5 Relaxation

Iterative methods may convergence faster when some relaxation parameter ω is introduced. The method is then given by

$$\mathbf{f}^k = \mathbf{f}^{k-1} + \omega \mathbf{b}^k, \tag{3.34}$$

where $\mathbf{b}^k$ is the update for the k-th approximation. Typically one has $\omega \in (0, 2)$. Faster convergence is not guaranteed, it may occur that some ω will lead to worse convergence and in general there is no easy way to determine the optimal ω.

3.5 Advanced Iterative Methods

In this section Krylov subspaces will be introduced. These subspaces are used by various iterative methods to solve sparse linear systems. One of those methods is the conjugate gradient (CG) method which is arguably one of the best known iterative techniques for solving sparse SPD linear systems. The system matrix W of the considered tomography problem is not SPD, but the normal equations $W^T W\mathbf{f} = W^T\mathbf{p}$ do yield an SPD linear system. Therefore it is reasonable to attempt to solve the problem using the CG method applied to the normal equations.

3.5.1 Krylov Subspaces

Kaczmarz's method, considered in Section 3.4, employed subsequent projections onto subspaces to approximate the solution to (3.1). Krylov subspace methods are essentially the same but one projects onto a so called *Krylov subspace*:

Definition 3.11. Let $A \in \mathbb{R}^{n \times n}$ and $\mathbf{v} \in \mathbb{R}^n$. The Krylov subspace $\mathcal{K}_r(A, \mathbf{v})$ of order r is given by

$$\mathcal{K}_r(A, \mathbf{v}) = \text{span}\{\mathbf{v}, A\mathbf{v}, A^2\mathbf{v}, \ldots, A^{r-1}\mathbf{v}\}. \tag{3.35}$$

The subspaces are named after the Russian naval engineer and mathematician Aleksey Krylov who published about this concept and used it to determine the characteristic polynomial of A in 1931.

When no confusion is possible $\mathcal{K}_r(A, \mathbf{v})$ is written as $\mathcal{K}_r$ for brevity. Note that $\mathcal{K}_m$ is the subspace of $\mathbb{R}^n$ in which every vector $\mathbf{x} \in \mathcal{K}_r$ can be written as $\mathbf{x} = p(A)\mathbf{v}$ where $p(A)$ is some polynomial of degree at most $r-1$, from this one can easily see that $\mathcal{K}_r \subseteq \mathcal{K}_p$ for $r \leq p$.

3.5.2 CGLS method

The conjugate gradient method is widely used to solve sparse SPD linear systems. In essence CG uses the (shifted) Krylov subspace $\mathbf{x}^0 + \mathcal{K}_r(A, \mathbf{r}^0)$ to construct subsequent approximations, with $A \in \mathbb{R}^{n \times n}$ SPD. Here $\mathbf{r}^0 = \mathbf{b} - A\mathbf{x}^0$, the residual corresponding to the initial guess $\mathbf{x}^0$. The method was first proposed by M. R. Hestenes and E. Stiefel in 1952 to solve SPD systems. The CG algorithm will now be presented without any further introduction or derivation. For more details about the algorithm the reader is referred to [22, Ch. 6.7].

CG

Choose $\mathbf{x}^0$
$\mathbf{r}^0 \leftarrow \mathbf{b} - A\mathbf{x}^0$
$\mathbf{d}^0 \leftarrow \mathbf{r}^0$
for $j = 1, 2, \ldots$ until convergence **do**
  $\alpha_{j-1} \leftarrow \langle \mathbf{r}^{j-1}, \mathbf{r}^{j-1} \rangle / \langle A\mathbf{d}^{j-1}, \mathbf{d}^{j-1} \rangle$
  $\mathbf{x}^j \leftarrow \mathbf{x}^{j-1} + \alpha_{j-1}\mathbf{d}^{j-1}$
  $\mathbf{r}^j \leftarrow \mathbf{r}^{j-1} - \alpha_{j-1}A\mathbf{d}^{j-1}$
  $\beta_{j-1} \leftarrow \langle \mathbf{r}^j, \mathbf{r}^j \rangle / \langle \mathbf{r}^{j-1}, \mathbf{r}^{j-1} \rangle$
  $\mathbf{d}^j \leftarrow \mathbf{r}^j + \beta_{j-1}\mathbf{d}^j$
end for

CG chooses α_j and β_j in such a way the the subsequent residuals $\mathbf{r}^0, \mathbf{r}^1, \ldots$ are mutually orthogonal and hence the CG method is guaranteed to produce the exact solution to the system after at most n iterations (where n is the size of the problem). Unfortunately, due to round-off errors this may, and often will, fail in practice. Fortunately, however, it can be shown that the approximate solution $\mathbf{x}^j$ minimizes the so called *energy norm* $\|\mathbf{x}^* - \mathbf{x}^j\|_A = \sqrt{\langle A(\mathbf{x}^* - \mathbf{x}^j), \mathbf{x}^* - \mathbf{x}^j \rangle}$ where $\mathbf{x}^*$ is the exact solution [22, Ch. 6.11.3]. CG thus produces a sequence of monotonically improving solutions. In practice n is very big and it is very costly to perform that many iterations, but, luckily, the results after $k << n$ are often accurate enough.

As mentioned, the normal equations applied to ill-conditioned system $W\mathbf{f} = \mathbf{p}$ yield $W^TW\mathbf{f} = W^T\mathbf{p}$, which is an SPD system and hence the CG method can be used to find a, possibly approximate, solution to the problem. This means that the approximate solution minimizes $\| \left(\mathbf{f}^* - \mathbf{f}^j\right) \|_{W^TW}$ where $\mathbf{f}^*$ is an exact solution. This term might seem puzzling since one can have multiple exact (i.e. least square) solutions to the original system. However, for each of these $\mathbf{f}^*$ one has the same energy norm $\|\mathbf{f}^*\|_{W^TW}$ and thus the term yields the same value for all the exact solutions. Analysis of this term results in

$$\begin{aligned}\|\mathbf{f}^* - \mathbf{f}^j\|^2_{W^TW} &= \left\langle W^TW\left(\mathbf{f}^* - \mathbf{f}^j\right), \mathbf{f}^* - \mathbf{f}^j\right\rangle \\ &= \left\langle W\left(\mathbf{f}^* - \mathbf{f}^j\right), W\mathbf{f}^* - \mathbf{f}^j\right\rangle \\ &= \left\langle \mathbf{p} - W\mathbf{f}^j, \mathbf{p} - W\mathbf{f}^j\right\rangle \\ &= \|\mathbf{p} - W\mathbf{f}^j\|_2^2.\end{aligned}$$

Thus CG applied to the normal equations of (3.1) minimizes the least squares residual of the original problem.

The corresponding algorithm is virtually the same but the residual is replaced by $\mathbf{z}^j = W^T\mathbf{r}^j$. This yields:

CGNR

Choose $\mathbf{x}^0$
$\mathbf{r}^0 \leftarrow \mathbf{p} - W\mathbf{f}^0$
$\mathbf{z}^0 \leftarrow W^T\mathbf{r}^0$
$\mathbf{d}^0 \leftarrow \mathbf{z}^0$
for $j = 1, \ldots$ until convergence **do**
  $\alpha_{j-1} \leftarrow \left\langle \mathbf{z}^{j-1}, \mathbf{z}^{j-1}\right\rangle / \left\langle W^TW\mathbf{d}^{j-1}, \mathbf{d}^{j-1}\right\rangle$
  $\mathbf{f}^j \leftarrow \mathbf{f}^{j-1} + \alpha_{j-1}\mathbf{d}^{j-1}$
  $\mathbf{r}^j \leftarrow \mathbf{r}^{j-1} - \alpha_{j-1}W\mathbf{d}^{j-1}$
  $\mathbf{z}^j \leftarrow W^T\mathbf{r}^{j-1}$
  $\beta_{j-1} \leftarrow \left\langle \mathbf{z}^j, \mathbf{z}^j\right\rangle / \left\langle \mathbf{z}^{j-1}, \mathbf{z}^{j-1}\right\rangle$
  $\mathbf{d}^j \leftarrow \mathbf{z}^j + \beta_{j-1}\mathbf{d}^j$
end for

Note that in the first line of the for-loop one has to do two matrix-vector multiplications which are costly procedures, it is better to compute this line as $\dfrac{\left\langle \mathbf{z}^{j-1}, \mathbf{z}^{j-1}\right\rangle}{\left\langle W\mathbf{d}^{j-1}, W\mathbf{d}^{j-1}\right\rangle}$. This scheme also leads to more numerically stable results according to Björk and Elfving [25, p. 72]. The CGNR algorithm will from now on be referred to as CGLS for it finds the least squares (LS) solution to (3.9).

Similarly as with the normal equations one can apply CG to the system of the alternative normal equations $WW^T\mathbf{u} = \mathbf{p}$. This results into a error-minimizing method known as *Craig's Method*. A major disadvantage of error-minimizing methods is that they are particularly susceptible to noise in the right-hand-side, which problems originating from tomography generally have.

3.6 Noise

As mentioned throughout, problems originating from tomography often contain noise. It is assumed that the source emits a uniform continuous ray. In practice, however, sources as for example a X-ray tube emit photons not uniformly but rather according to some Poisson distribution. From a mathematical point of view one can model the noise as some perturbation $\boldsymbol{\epsilon}$ in the measured projection data $\mathbf{p}$, thus if the ideal projection is $\tilde{\mathbf{p}}$ then $\mathbf{p} = \tilde{\mathbf{p}} + \boldsymbol{\epsilon}$. Due to

this noise it is possible that the system $W\mathbf{f} = \mathbf{p}$ has no solution and thus one wants to find the minimum norm least squares solution (see Section 3.2.1).

3.6.1 Singular Values and Data Errors

First the effects of the noise, data errors, will be investigated. The following reasoning is due to [25, p. 53–54]. Let W be of rank p with $p < M$, hence W is not of full rank, then from Section 3.2.5 it is known that $\sigma_{p+1} = \cdots = \sigma_M = 0$. Let the SVD of W be $W = U\Sigma V^T$, write $\mathbf{p} = U\mathbf{g}$ and $\mathbf{f} = V\mathbf{z}$, thus $\mathbf{p}$ is expressed as a linear combination of the columns of U and similarly with $\mathbf{f}$ and V. Note that for an orthogonal matrix U one has $U^{-1} = U^T$ and hence $\mathbf{g} = U^T\mathbf{f}$. Substituting this into the original equations yields:

$$\begin{aligned} W\mathbf{f} &= \mathbf{p} && \Rightarrow \\ U\Sigma V^T V\mathbf{z} &= U\mathbf{g} && \Rightarrow \\ \Sigma\mathbf{z} &= \mathbf{g} && \Rightarrow \\ \mathbf{z} &= \Sigma^{-1}\mathbf{g}. \end{aligned} \tag{3.36}$$

Thus one finds that $\mathbf{f}$ is a least squares solution if and only if

$$z_j = \frac{g_j}{\sigma_j} = g_j\phi(\sigma_j) \tag{3.37}$$

for $j \leq p$. The function $\phi(t) = 1/t$ is the *response function* for the system, it describes how the various components of $\mathbf{p}(= U\mathbf{g})$ affect the solution $\mathbf{f} = (V\mathbf{z})$. This solution is also of minimal norm if $z_{p+1} = \cdots = z_M = 0$.

Suppose the data in the measured vector $\mathbf{p}$ is perturbed from the ideal projection data $\tilde{\mathbf{p}}$, thus $\mathbf{p} = \tilde{\mathbf{p}} + \boldsymbol{\epsilon}$. This perturbation vector $\boldsymbol{\epsilon}$ represents the noise in the system and it is assumed that its components have expectation 0, variance σ^2 and are uncorrelated. It needs to be noted that this σ is not related to the σ_j of the singular values, these two variables are unfortunately just the standard notations for these notions. Let $\tilde{\mathbf{f}}$ be such that $W\tilde{\mathbf{f}} = \tilde{\mathbf{p}}$. The error $\Delta\mathbf{f}$ is then given by $\mathbf{f} - \tilde{\mathbf{f}}$. One can show that the variance of this error is given by:

$$\operatorname{var}(\|\Delta\mathbf{f}\|^2) = \sigma^2 \sum_{j=1}^{M} \frac{1}{\sigma_j^2} \tag{3.38}$$

(see [25, p. 54]). This suggests that small singular values will result into large errors in the approximated solution $\mathbf{f}$. Thus a perturbation (noise) in $\mathbf{p}$, however small, can have disastrous results. This is precisely why systems with a small condition number, and thus most likely small singular values, are called ill-conditioned.

3.6.2 Regularization

Small singular values of the system can cause the components of the approximate solution to have large errors as can be seen in (3.38). Regularization is a concept to circumvent this. There are many types of regularization and none of these methods are superior to all the other methods.

Rather, each method has its advantages, depending on the properties of the system it is applied to [12, p. 2]. The main idea of regularization is to incorporate additional information about the solution in order to arrive at a better approximation (and thus a smaller error).

In Section 3.2.5 the concept of ill-conditioned matrices was introduced. Ill-conditioned problems can roughly be categorized into two classes [12, p. 2]: rank-deficient problems and discrete ill-posed problems. The former class is characterized by a system matrix W which has a cluster of small singular values and there is a well-determined gap between the larger and small singular values (recall the concept of effectively of rank n, Section 3.2.5). The latter has singular values which, on average, decay gradually to zero. Both classes require other types of regularization. Tomographic problems typically are rank-deficient.

Now two methods of regularization will be introduced. Both methods employ the idea to reduce the influence of the small singular values on the solution. The first method is called *Tikhonov regularization* and was independently proposed by Phillips (1962) and Tikhonov (1963). Instead of solving the system $W\mathbf{f} = \mathbf{p}$ the system

$$\begin{pmatrix} W \\ \lambda I \end{pmatrix} \mathbf{f} = \begin{pmatrix} \mathbf{p} \\ \mathbf{0} \end{pmatrix} \tag{3.39}$$

is solved for a certain λ. Recall from Section 3.2.2 that $\mathbf{f}^*$ is a least squares solution to (3.39) if and only if

$$\begin{pmatrix} W & \lambda I \end{pmatrix} \begin{pmatrix} W \\ \lambda I \end{pmatrix} \mathbf{f}^* = \begin{pmatrix} W & \lambda I \end{pmatrix} \begin{pmatrix} \mathbf{p} \\ \mathbf{0} \end{pmatrix} \Rightarrow \mathbf{f}^* = \left(W^T W + \lambda^2 I\right)^{-1} W^T \mathbf{p}. \tag{3.40}$$

Let $W = U\Sigma V^T$ be the SVD of W, let $\mathbf{f}^* = V\mathbf{z}^*$ and $\mathbf{p} = U\mathbf{g}$. Then the conclusion of (3.40) reduces to

$$\mathbf{z}^* = \left(\Sigma^T \Sigma + \lambda^2 I\right)^{-1} \Sigma^T \mathbf{g} \tag{3.41}$$

and thus

$$z_j^* = \frac{g_j \sigma_j}{\sigma_j^2 + \lambda^2} = g_j \phi^*(\sigma_j). \tag{3.42}$$

Recall that for the non-regularized case the response function $\phi(t) = 1/t$ was found. Figure 3.4 shows both response functions. One should note that the response $\phi(t)$ for small t (singular values) is much higher (and tends to infinity as $t \downarrow 0$) than for the regularized response function $\phi^*(t)$ and hence the effects of noise in $\mathbf{p}$ are much smaller so that ultimately the approximate solution $\mathbf{f}$ is more accurate.

The second method of regularization is less sophisticated than Tikhonov regularization and will be referred to as *truncated SVD*. The idea is to determine the singular values of W, e.g. from SVD, and setting $\sigma_j = 0$ when the value is below some threshold η. This regularization method results in the response function $\hat{\phi}(\sigma_j) = 1/\sigma_j$ if $\sigma_j \geq \eta$ and $\hat{\phi}(\sigma_j) = 0$ otherwise. The major problem with this method is that one has to compute the SVD of W and this is a costly process which one generally cannot afford.

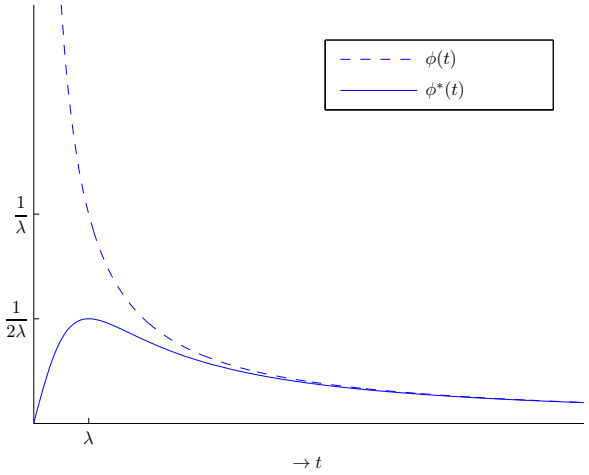


Figure 3.4: *The response functions $\phi(t)$ and $\phi^*(t)$.*

For more information about these two and other regularization methods one is referred to [12, Ch 5,6].

3.6.3 Semi-Convergence

When solving an ill-conditioned linear system of equations one is interested in finding the best approximate solution. Perturbations, such as noise, in the right-hand side might cause the error of the approximation to explode (cf. (3.38)). By means of regularization this can be circumvented, but one still obtains an approximate solution $\mathbf{f}$ of $W\mathbf{f} = \mathbf{p}$. Suppose $\tilde{\mathbf{f}}$ is the solution to the unperturbed system $W\tilde{\mathbf{f}} = \tilde{\mathbf{p}}$ and $\boldsymbol{\epsilon}$ is the perturbation such that $\mathbf{p} = \tilde{\mathbf{p}} + \boldsymbol{\epsilon}$. Assume that every component of the noise vector $\boldsymbol{\epsilon}$ has expectation 0, variance σ^2 and are uncorrelated. The error can then be decomposed as follows:

$$\Delta\mathbf{f} = \Delta\mathbf{f}^{appr} + \Delta\mathbf{f}^{pert}. \tag{3.43}$$

The first term $\Delta\mathbf{f}^{appr}$ represents the *approximation error* and is ever-present, even if there is no perturbation in the data. This error depends on how the solution is obtained and thus also on the regularization method that is used. The second term $\Delta\mathbf{f}^{pert}$ is the *perturbation error* and is a result of the noise that was present in the system.

Typically, when solving a perturbed ill-posed problem with an iterative method, the approximation error dominates during the initial iterations and thus the perturbation error is negligible. But eventually the approximation becomes more accurate, thus the approximation error decreases and the perturbation error will dominate and even increases [13]. The norm of the error $\|\Delta\mathbf{f}\|_2$ will thus decrease for the initial iterations but will increase after a certain number of iterations. In Figure 3.5 one can see a sketch of the development of the error norm. This phenomenon is coined *semi-convergence* by Natterer in 1986.

For a more thorough analysis one is referred to [25, Ch 2.7].

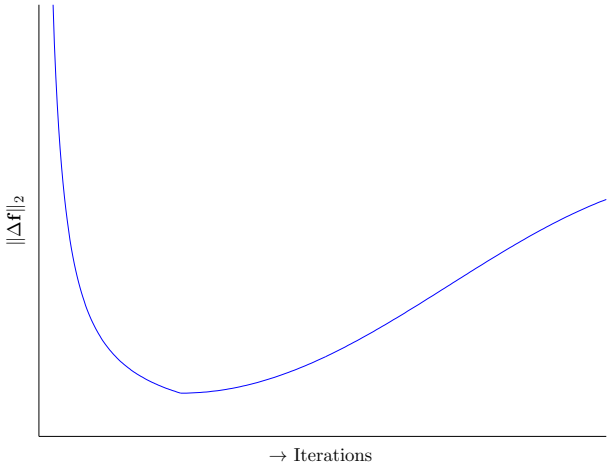


Figure 3.5: *The development of the error norm for an increasing number of iterations.*

3.7 Experiments

This section compares the four proposed methods: ART, SIRT, SART and CGLS to each other as well as to FBP with a Hamming window. The performance will be investigated with and without noise. The Shepp-Logan head phantom is used for the experiments. The image of 128 by 128 pixels was scanned using 32 angles and the number of rays per angle is 192 yielding a total number 6144 rays. The IBBT-Visionlab, Dept. of Physics, University of Antwerp, implementeded all the discussed algorithms as part of the *All Scale Tomographic Reconstruction Antwerp toolbox* or ASTRA-toolbox. The noise is added to the sinogram of the image. Firstly, the performance of the methods is investigated without noise.

In Figure 3.6 ART reconstructions of the Shepp-Logan phantom are shown. From the figure one can see that the reconstruction after 1 iteration is quite good and more iterations only show minor improvements, if any. Figure 3.7 shows the reconstruction of the phantom using SIRT after a different number of iterations. The first couple of SIRT iterations seem to produce inferior results to ART with the same number of iterations, thus the convergence of SIRT seems to be slower than that of ART, if convergence is to be expected, that is. In Figure 3.8 the SART reconstructions of the head phantom after a different number of iterations are shown. Figure 3.9 shows the reconstruction using CGLS. Since the image was scanned using a total of 6144 rays CGLS should yield convergence after 6144 iterations, but from the figure one can conclude that the quality of the reconstruction does not improve significantly after (only) 25 iterations, this motivates the earlier claim that one only needs $k << n$ iterations.

Figure 3.10 shows the relative error $\frac{\|\Delta \mathbf{f}\|_2}{\|\mathbf{f}^*\|_2}$, where $\mathbf{f}^*$ was the original image. The constant dotted line represents the relative error of FBP, which is obviously not an iterative method. The asterisk on the lines represent the number of iterations for which the relative errors is minimal. It seems that ART, SART and CGLS converge very fast, but after about 10-15 iterations the relative error stays more or less constant. SIRT seems to converge much slower. The minimal relative error for all the methods except ART is after 200 iterations, ART has a reconstruction with minimal error after 5 iterations. Both SIRT and SART have monotonically decreasing errors, i.e. the error of the next iteration is less or the same as the current iteration. CGLS and ART do not have this property. For SIRT this was to be expected since the convergence (or,

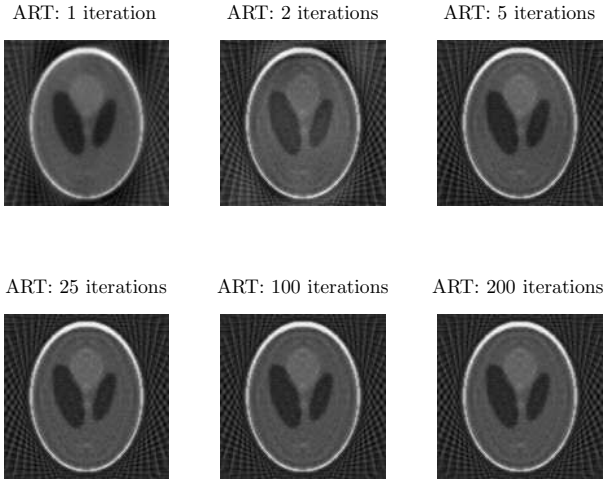


Figure 3.6: *The reconstructed images after a different number of ART iterations.*

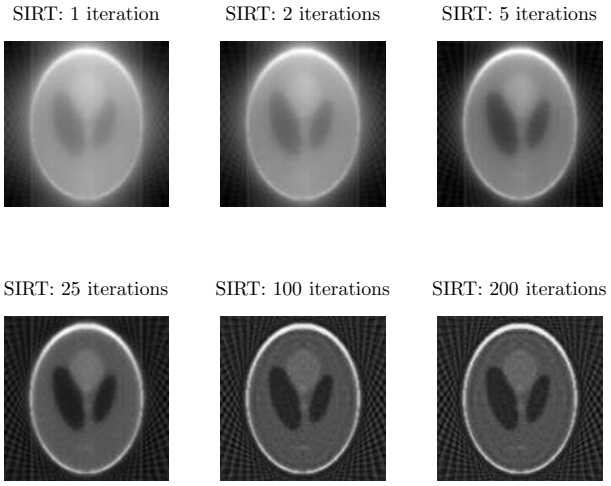


Figure 3.7: *The reconstructed images after a different number of SIRT iterations.*

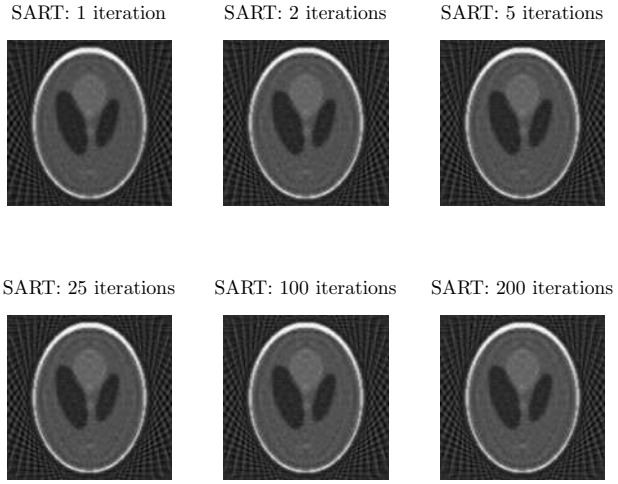


Figure 3.8: *The reconstructed images after a different number of SART iterations.*

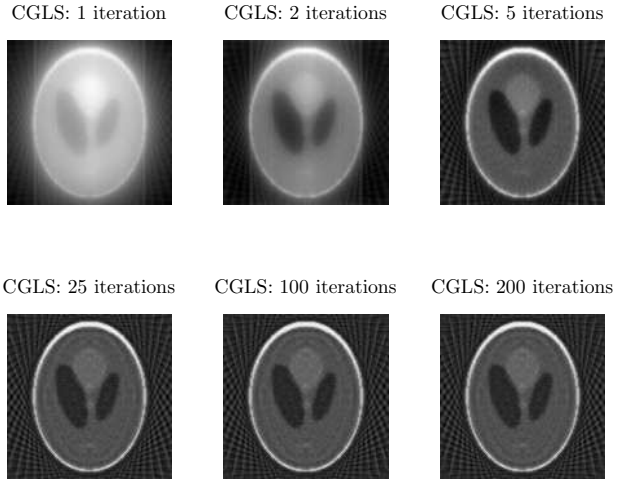


Figure 3.9: *The reconstructed images after a different number of CGLS iterations.*

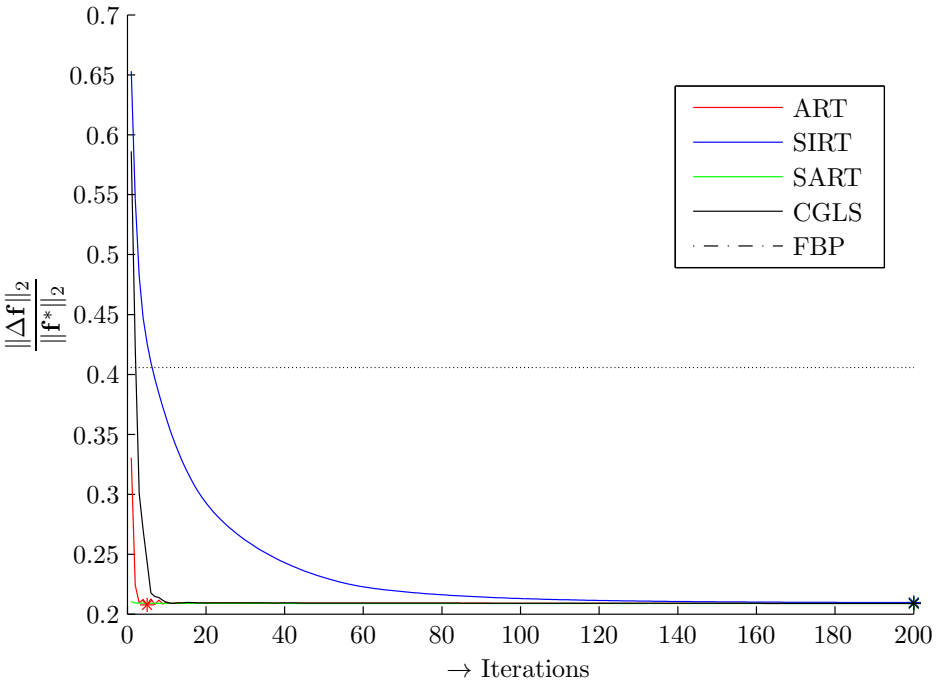


Figure 3.10: *The relative error of the various ARM's when no noise is present.*

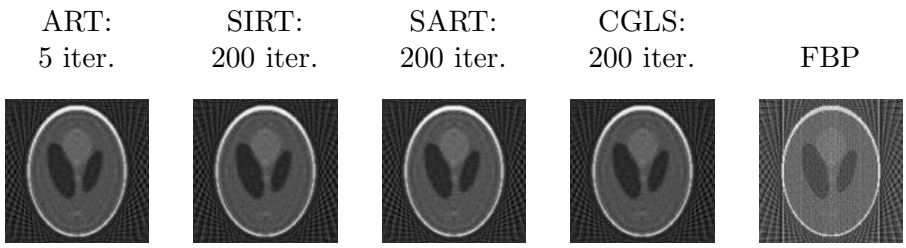


Figure 3.11: *The reconstruction without noise of the ARM's when the relative error is minimal.*

better, non-divergence) was proven in Section 3.4.3. For SART this may and will not be true in general. Figure 3.11 shows the results of the reconstruction of the ARMs when the relative error is minimal and the reconstruction obtained using FBP. None of the ARMs seem to produce significantly superior results but all perform slightly better than FBP. The following table shows the execution times of the various algorithms.

Algorithm	ART	SIRT	SART	CGLS	FBP
Executiontime	2.38s	2.80s	24.34s	2.66s	0.02s

ART, SIRT and CGLS seem to take about the same time for 200 iterations while SART takes significantly longer. FBP is the fastest algorithm, as expected, but note that this is not an iterative method.

Now some noise is added to the sinogram and the performance is investigated. The relative error as a function of the number of iterations can be found in Figure 3.12. This figure clearly shows semi-convergence of ART. Also SART and CGLS seem to suffer from this phenomenon, albeit

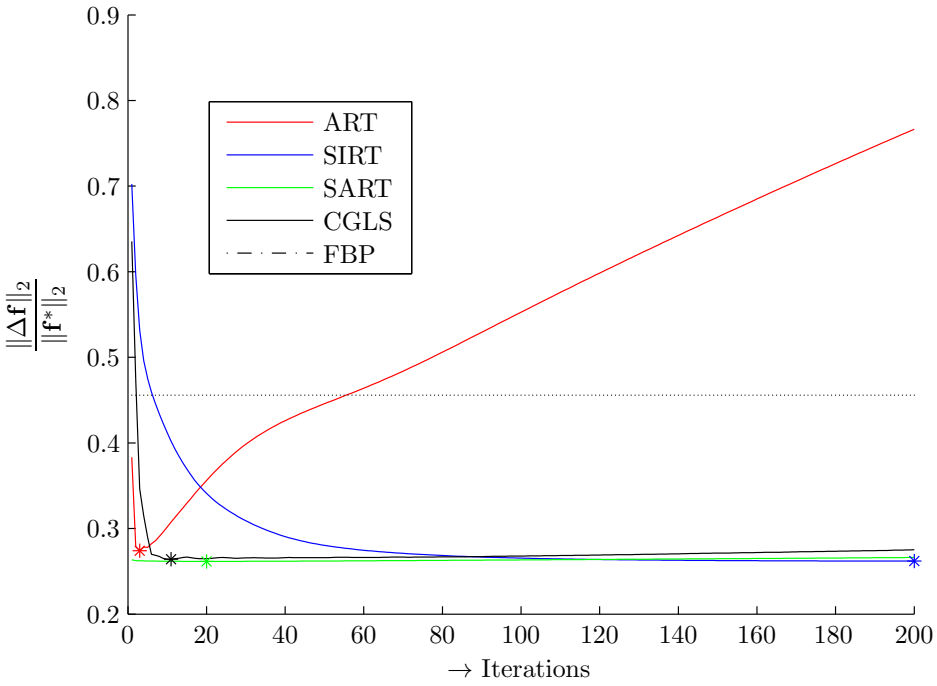


Figure 3.12: *The relative error of the various ARM's when noise is present.*

to a far lesser degree than ART. Again SIRT has a monotonically decreasing error. Figure 3.13 shows the reconstruction of the ARMs when the relative error is minimal and after 200 iterations. The reconstruction of ART after 3 iterations was the 'best' (as in minimal relative error), but still it suffers from the noise, after 200 iterations the original image is barely recognisable. The best reconstructions of the other three ARMs are similar but one should note that SART and CGLS arrive at this stage a lot sooner (iteration wise) than SIRT. FPB seems to produce a very noisy result, but the original image can still be recognized, the reconstruction is, however, far inferior to that of SIRT, SART and CGLS.

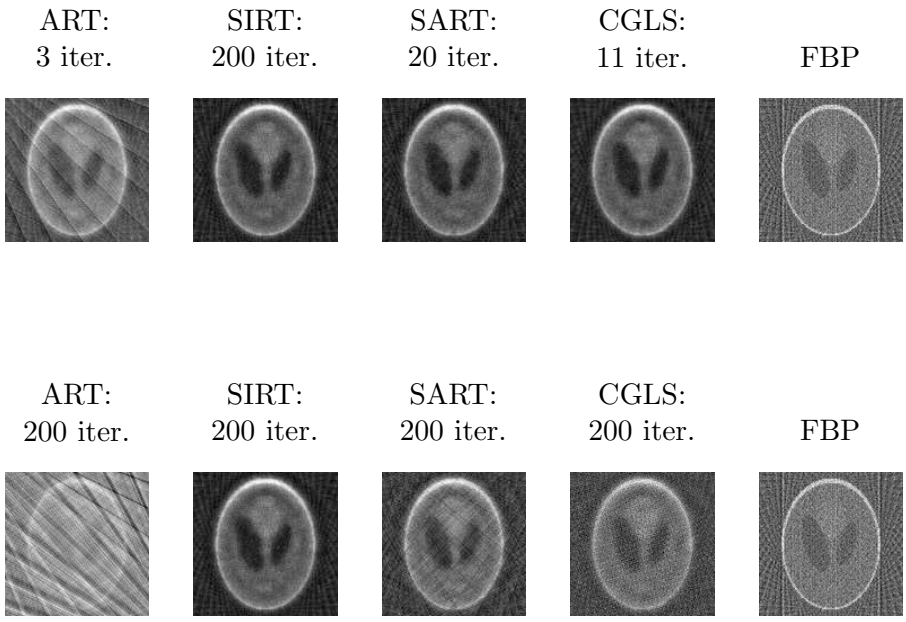


Figure 3.13: *The reconstruction without noise of the ARM's when the relative error is minimal and after 200 iterations.*

Chapter 4

Discrete Tomography

This chapter focusses on the *discrete tomography* (DT) problem. Unlike the previously considered problems, discrete tomography assumes certain properties about the image or object. Furthermore, the number of projections used in DT is far fewer than in classical (continuous) tomography. The resulting problem is far more complex than for the continuous case. Four different solution strategies will be briefly discussed, one of these strategies will be of particular interest.

4.1 Description

The term discrete tomography was first coined by Larry Shepp who organised the first meeting devoted to this topic in 1994. Similar problems and a range of results were already known from earlier research. For a more detailed account on the history of this field one is referred to [18, Ch. 1].

In the foregoing chapters various methods were investigated to reconstruct a certain two-dimensional image $\mathbf{f} \in \mathbb{R}^N$ from its projections. This image could be viewed as a function $f : \mathbb{R}^2 \to \mathbb{R}$. Thus every point in the image has a corresponding function value, or grey value. Discrete tomography solves a similar problem with some extra constraints.

Roughly two kinds of discrete tomography can be identified. The first one assumes that the scanned image is defined on a lattice, i.e. the object or image is physically made up out of 'pixels'. All these pixels are assumed to have some uniform density and the density can only be one of a finite set of allowed densities. One can for example imagine that this model is suitable for scanning a diamond on nanoscopic scale. All the atoms (pixels) lie in a neat lattice and have uniform density. Hence a pixel can either represent an atom, in this case the corresponding grey value would be 1 (for simplicity) or the pixel is just void, in which case the grey value is 0. In this variant of DT one has a function with discrete domain and the range is a finite set, thus $f : \mathcal{L} \to \{\rho_1, \rho_2, \ldots, \rho_l\}$ with $\mathcal{L}$ a lattice. A lattice $\mathcal{L}$ in some d-dimensional domain is isomorphic to the integer lattice $\mathbb{Z}^d$, hence one can assume that $\mathcal{L} = \mathbb{Z}^d$. The grey values of each of these lattice points must be one of $\{\rho_1, \rho_2, \ldots, \rho_l\}$. Usually these assumed densities follow from prior knowledge about the image or object. In many cases there are only two allowed densities, then, without loss of generality, the set is denoted as $\{0, 1\}$ and one speaks of a *binary image*.

In the second kind of discrete tomography the assumption that the image is only defined on a lattice is dropped. Hence one has $f : \mathbb{R}^2 \to \{\rho_1, \rho_2, \ldots, \rho_l\}$, a function with continuous domain and a finite set as range. Also for these problems it is common to only have two density values. The upcoming work will mainly focus on this second kind of DT.

One should note that the former DT variant is actually a special case of the latter variant. Both problems can be written, like in the continuous case, as $W\mathbf{f} = \mathbf{p}$. The weight matrix of the DT lattice problem will be a $(0, 1)$-matrix. For the latter variant the weights usually represent the length of the intersection of a ray with a cell. In continuous tomography the number of projections is often quite large compared to that in DT. On top of that, the few projections obtained in DT can originate from a small angular range. All previously proposed methods fail in solving this problem. This calls for better reconstruction techniques that incorporate the extra information known about the image.

4.2 Solution Strategies

This section will present four different strategies on how to solve the DT problem. The first approach will be to rewrite the problem as a combinatorial problem. The second strategy is statistical in nature. The two remaining techniques both solve the continuous problem. One of these approaches obtains a reconstruction using optimisation techniques, this reconstructed image may have different values than the allowed densities but this is penalised. The other method uses a discretisation step on the initial reconstruction, that was obtained via a continuous reconstruction method. This is also the strategy the Discrete Algebraic Reconstruction Technique (DART) algorithm uses. This algorithm will be studied extensively in this work.

4.2.1 Combinatorial

Consider the problem of reconstructing a $(0, 1)$-matrix from its row and column sums. One can easily see that this is equivalent to the reconstruction of a binary lattice image or object from two orthogonal projections. The $(0, 1)$-reconstruction was studied by Ryser in 1957 who also proposed an algorithm for solving the corresponding problem [21]. Unfortunately, one does not have an unique solution for this problem in general. Consider for example the following two $(0, 1)$-matrices:

$$\begin{array}{cccc|c} 1 & 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 1 & 2 \\ 1 & 1 & 1 & 0 & 3 \\ 1 & 0 & 1 & 1 & 3 \\ \hline 3 & 1 & 4 & 2 & \end{array} \qquad \begin{array}{cccc|c} 0 & 1 & 1 & 0 & 2 \\ 1 & 0 & 1 & 0 & 2 \\ 1 & 0 & 1 & 1 & 3 \\ 1 & 0 & 1 & 1 & 3 \\ \hline 3 & 1 & 4 & 2 & \end{array} \tag{4.1}$$

Both matrices have the same column and row sums but are fairly different. Hence one cannot expect to find the correct solution in this case. More information about the reconstruction of $(0, 1)$-matrices can be found in [18, Ch. 1.2.2]. Gale modelled this reconstruction problem, also in 1957, as a network flow problem [8]. The Ford-Fulkerson algorithm for maximum flows can then be used to find the reconstruction (if a unique solution exists). The solution strategies for this problem proposed by Ryser and Gale are particularly computationally efficient. If one wants to solve the problem for more than 2 projection angles, however, the corresponding problem

becomes NP-hard [9]. Batenburg employed the idea of Gale to propose an algorithm which is able to reconstruct a *continuous* image (not defined on a lattice) [2]. The idea behind Batenburg's algorithm is to iteratively reconstruct the next image by using the previous reconstruction and two projection angles.

Another combinatorial approach for solving the DT lattice problem is from Gritzmann et al. [11]. They propose to solve two complementary problems: Best-Inner-Fit [BIF] and Best-Outer-Fit [BOF]. The reconstruction is found using optimisation techniques.

4.2.2 Statistical

The statistical reconstruction method uses a *maximum a posteriori probability estimate* (MAP estimate) to approximate an optimal reconstruction based on the measured projection data. The admitted grey values are assumed to be distributed among the various pixels/cells according to some probability distribution. In 2005 Liao and Herman proposed to reconstruct discrete images using a Gibbs distribution [19].

4.2.3 Continuous Optimisation

The DT problem can also be solved using continuous optimisation. In this approach the standard continuous formulation of the problem is actually solved, i.e. the problem is relaxed so that $\mathbf{f}$ can take any real value, but function values outside the set of admitted grey values are penalised. Schüle et al. proposed a convex-concave regularization approach for the binary problem in 2003 [23]. The idea behind this approach is to use primal-dual optimisation to arrive at a solution which is steered towards a binary solution.

4.2.4 Continuous with Discretisation Step

Perhaps this is the most natural approaches to solve the DT problem. The problem is first solved as if it were a continuous tomographic problem, thus the values of the image can attain any real value. Subsequently the reconstructed values are discretised, i.e. set to one of the admitted densities. The DART algorithm uses this approach. The algorithm itself is explained in detail in Section 4.3.

4.3 DART

The Discrete Algebraic Reconstruction Technique (DART) was first proposed by Batenburg et al. in 2007 [3]. In 2011 Batenburg and Sijbers presented a more elaborate paper [4] giving a description of the DART algorithm along with experiments which validate the results of the algorithm. For this section the proceedings in said paper are used to describe the DART algorithm. The performance of DART will also be investigated.

4.3.1 The DART algorithm

DART is an iterative method of the type described in Section 4.2.4. It consists of reconstructing the image using a continuous update step followed by a discretisation step. The continuous step is just the reconstruction of the image using some ARM, the discretisation uses the prior knowledge, i.e. the allowed grey values $\{\rho_1, \rho_2, \ldots, \rho_l\}$. The number of allowed grey values should

not be too large, according to Batenburg et al. [4] the algorithm in general performs well for five or fewer grey values.

Initially a continuous reconstruction is computed by performing a fixed number of ARM iterations, this serves as a starting point for the DART algorithm. The reconstructed image is then *segmented* to obtain an image that consists of only the admitted grey values. The segmentation can be obtained by simply rounding the pixel values to the nearest allowed grey values. Or formally, define the threshold τ_i by

$$\tau_i = \frac{\rho_i + \rho_{i+1}}{2}, \tag{4.2}$$

for $i = 1, 2, \ldots, l-1$. Then the threshold function $r : \mathbb{R} \to \{\rho_1, \rho_2, \ldots, \rho_l\}$ is given by

$$r(v) = \begin{cases} \rho_1, & (v < \tau_1) \\ \rho_2, & (\tau_1 \leq v < \tau_2) \\ \vdots & \\ \rho_l, & (\tau_{l-1} \leq v) \end{cases}. \tag{4.3}$$

Other more intricate segmentation methods can be used that may lead to better convergence or more accurate reconstructions.

Next this segmented image is subdivided into two groups of pixels, the *free pixels* U and the *fixed pixels* F. The set U are all pixels that are adjacent to at least one pixel with a different grey value. Here one can choose the adjacent pixels of pixel j as the pixels North, East, South and West of the considered pixel, i.e. the 4-connected neighbourhood of pixel j. DART uses the 8-connected neighbourhood of pixel j which is simply the 4-connected neighbourhood augmented with the diagonally neighbouring pixels. The set F are all the pixels that are not free, hence fixed. Note that $U \cap F = \emptyset$ and $U \cup F$ are all the pixels of the image. The set of free pixels represents the edges of the object, where an edge is the transition of one density into another. It may happen that there are edges which are not covered by the free pixels, holes in the image may for example be overlooked. The set U is therefore supplemented with some random pixels from F. Define $0 < p \leq 1$ as the *fix probability*. Each fixed pixel from F is freed with probability $1 - p$, independently of neighbouring pixels.

Subsequently DART performs some fixed number of ARM iterations on the free pixels while, obviously, keeping the fixed pixels at their respective grey values from the set $\{\rho_1, \rho_2, ..., \rho_l\}$. The initial guess for the ARM iterations are the grey values of the free pixels from *before* the segmentation. Fixing some pixels results into a system with less variables but the same number of equations as initially, as will be shown later on. This, in combination with noise, causes heavy fluctuations in the values of the free pixels after the ARM iterations. Therefore, a *smoothing operation* is carried out. Smoothing evens out the function values and thus reduces the influence of the noise. The smoothing in DART is carried out by applying a Gaussian smoothing filter with radius 1 to the free pixels.

The process of segmenting the image, dividing it into free and fixed pixels and applying some ARM iterations to the free pixels constitute one DART iteration. DART can either terminate after the total projection error is below some threshold ε, i.e.

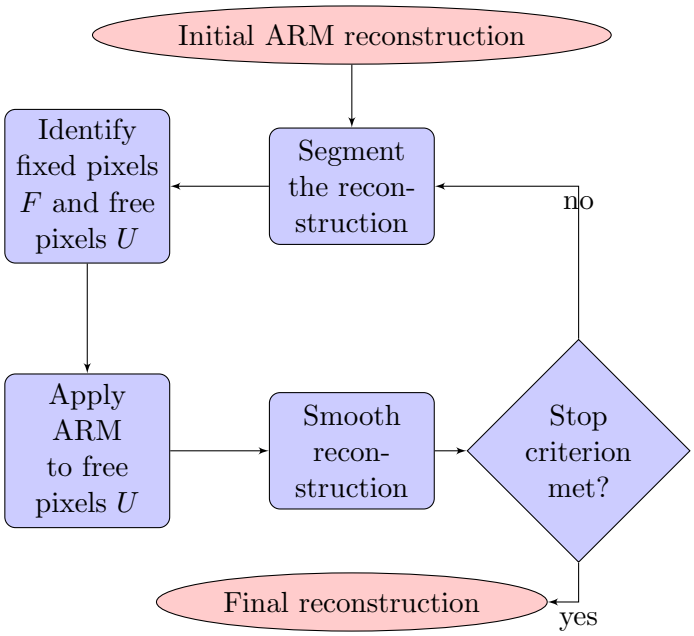


Figure 4.1: *Flowchart of the DART algorithm*

$$\|W\mathbf{f} - \mathbf{p}\|_2 \leq \varepsilon, \tag{4.4}$$

or simply after a fixed number of iterations. After the last iteration the reconstruction is again segmented to obtain the final reconstruction which only contains pixels with admitted grey values. Figure 4.1 shows a flowchart of the DART algorithm. Traversing the purple boxes once constitutes one DART iteration.

Now the process of fixing pixels is investigated more thoroughly. Initially one is presented with the system:

$$\begin{pmatrix} | & & | \\ \mathbf{w}_{:,1} & \cdots & \mathbf{w}_{:,N} \\ | & & | \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_N \end{pmatrix} = \begin{pmatrix} p_1 \\ \vdots \\ p_M \end{pmatrix}. \tag{4.5}$$

By fixing pixel j the value of variable f_j is known. This means that $\mathbf{w}_{:,j} f_j \in \mathbb{R}^M$ can be computed beforehand and thus one may remove the variable f_j from $\mathbf{f}$ and column $\mathbf{w}_{:,j}$ from W and subtract $\mathbf{w}_{:,j} f_j$ from the right-hand-side. This results in the system

$$\begin{pmatrix} | & & | & | & & | \\ \mathbf{w}_{:,1} & \cdots & \mathbf{w}_{:,j-1} & \mathbf{w}_{:,j+1} & \cdots & \mathbf{w}_{:,N} \\ | & & | & | & & | \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_{j-1} \\ f_{j+1} \\ \vdots \\ f_N \end{pmatrix} = \begin{pmatrix} p_1 \\ \vdots \\ p_M \end{pmatrix} - \mathbf{w}_{:,j} f_j. \tag{4.6}$$

This system consists of the same number of equations as the initial system 4.5 but the number of variables is reduced by one. Eventually some number of ARM iterations are applied to the system

$$\tilde{W}\tilde{\mathbf{f}} = \mathbf{p} - \sum_{j \in F} \mathbf{w}_{:,f} f_j \tag{4.7}$$

which only has the free pixels as variables. The effects of noise, which is ever-present, acts therefore only on the free pixels. This causes the heavy fluctuations in the values of these pixels and thus the need of the smoothing operation is justified.

Below the DART algorithm is given in pseudo code. Let q be the number of initial ARM iterations that will be performed and r the number of ARM iterations during one DART iteration. If $\mathbf{f}$ is obtained by performing i ARM iterations with initial guess $\mathbf{x}$ on system W with right hand side $\mathbf{p}$ then this is denoted as $\mathbf{f} = ARM(\mathbf{x}, W, \mathbf{p}, i)$.

DART

$\mathbf{f}^0 \leftarrow ARM(\mathbf{0}, W, \mathbf{p}, q)$
$\mathbf{x}^0 \leftarrow \mathbf{f}^0$
$k \leftarrow 1$
while termination criterion is not met **do**
  Segment image: $\mathbf{s}^k \leftarrow r(\mathbf{x}^{k-1})$
  Identify boundary pixels U^k from $\mathbf{s}^k$
  Let $F^k = \{1, 2, \ldots, N\} \backslash U^k$ represent the set of fixed pixels
  Free pixels in F^k with probability $1 - p$
  Let $\hat{W}^k$ be the matrix with the columns corresponding to fixed variables removed
  $\hat{\mathbf{p}}^k \leftarrow \mathbf{p} - \sum_{i \in F^k} s_i^k \mathbf{w}_{:,i}$
  Let $\mathbf{y}^k$ be a vector containing the non-segmented values $\mathbf{x}^{k-1}$ of the free pixels U^k
  $\mathbf{f}^k \leftarrow ARM(\mathbf{y}^k, \hat{W}^k, \hat{\mathbf{p}}^k, r)$
  Smooth all the free pixels U^k to obtain $\mathbf{x}^k$
  $k \leftarrow k + 1$
end while

One should note that the initial guess $\mathbf{y}^k$ for the ARM iterations during a DART iteration consist of all the free variables that have their non-segmented values $\mathbf{x}^{k-1}$ rather than $\mathbf{s}^k$. Also observe that the smoothing is only applied to the free pixels U^k.

4.3.2 Experiments

In this section the performance of DART will be investigated. The algorithm has several parameters: the ARM that will be used, the fix probability p, the smoothing filter, the threshold function, the number of grey values, the number of initial ARM iterations and the number of ARM iterations during one DART iteration. Since it will be impractical to consider the effect of each parameter some of them will be fixed:

- Smoothing filter: Gaussian filter with radius 1;
- Threshold function: Function $r(v)$ (4.3);

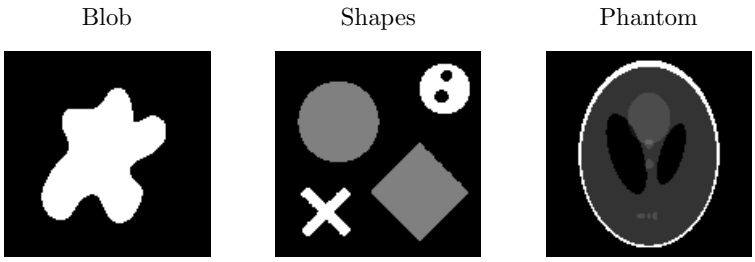


Figure 4.2: *The three test images for the DART experiments.*

- Initial ARM iterations: 50;
- ARM iterations: 20.

Hence only the used ARM, the number of grey values and p will be varied. Three ARMs will be considered: ART, SIRT and SART. The number of grey values are varied by the use of three different test images of 128 by 128 pixels, see Figure 4.2. *Blob* is a simple binary image with no holes. *Shapes* consists of three different grey values, the cross and the disk with holes have grey value 1, the big disk and diamond have grey value 0.5. Finally, the *phantom* is an image with six different grey values. Recall that the number of grey values for DART to perform efficiently should not be too large. For the following results the fix probability was 0.85 and reconstructions were made using 5, 10 and 15 equally spaced projection angles. Figures 4.3 – 4.5 show the reconstruction of the DART algorithm after 100 iterations using various ARMs and projection angles. The quality of the reconstruction is measured using the *pixel error* K, defined as the number of misclassified pixels by the DART algorithm. The pixel error is considered rather than the relative error, which was used in Section 3.7, since it is a more intuitive notion. Note that the the total number of pixels for each image is $128 \times 128 = 16384$. The reconstruction of the blob is practically perfect for all methods and angles, the shapes are only reconstructed accurately when data is available from 10 or more projection angles. The reconstruction of the head phantom shows the restriction of DART which possibly due to the high number of grey levels.

The various ARMs seem to produce comparable results. However, sometimes ART seems to produce divergent results as can be seen in Figure 4.6. Although this phenomenon was only observed for p close to 1 it opts for the conclusion that ART is not a good choice as ARM since SIRT and SART seem to produce similar results. SIRT iterations can be efficiently carried out in parallel, whereas SART is sequential by nature since the updates are done per angle. This suggests that SIRT is the algebraic reconstruction method of choice for DART.

To illustrate the need of the fix probability DART is applied to the blob with a small hole. If $p = 1$ this means that the set of free pixels U is not supplemented with a random subset of pixels, and hence, if the hole was not found in the initial reconstruction it will most likely never be found. Figure 4.7 shows the reconstructions of this image with a hole for the three ARMs, various projection angles and $p = 1$. Again ART exhibits divergent behaviour. Neither SIRT nor SART are able to reconstruct the image with the hole. Note that the ART reconstruction with 15 projection angles was able to find the hole and after 100 DART iterations the reconstruction is exact. Figure 4.8 shows the results with the same experiment, only now with $p = 0.99$. This small difference makes that all the reconstruction find the hole and are virtually perfect.

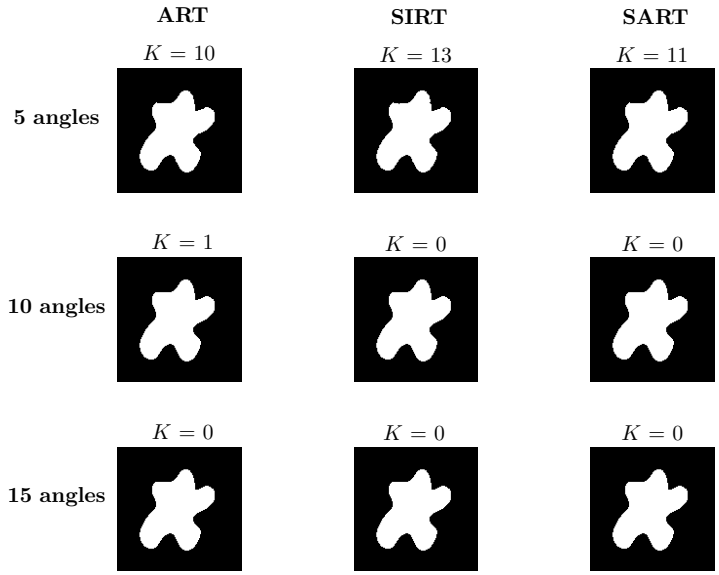


Figure 4.3: *DART reconstructions of the blob after 100 iterations using various ARM's and projection angels*

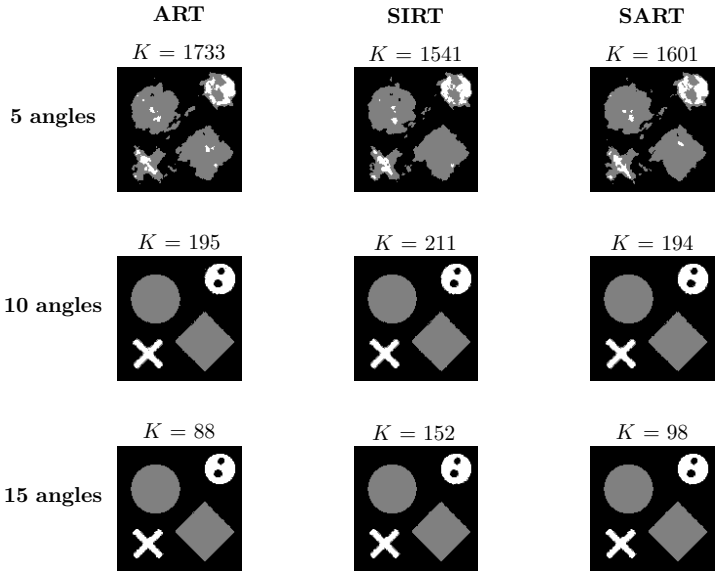


Figure 4.4: *DART reconstructions of the shapes after 100 iterations using various ARM's and projection angels*

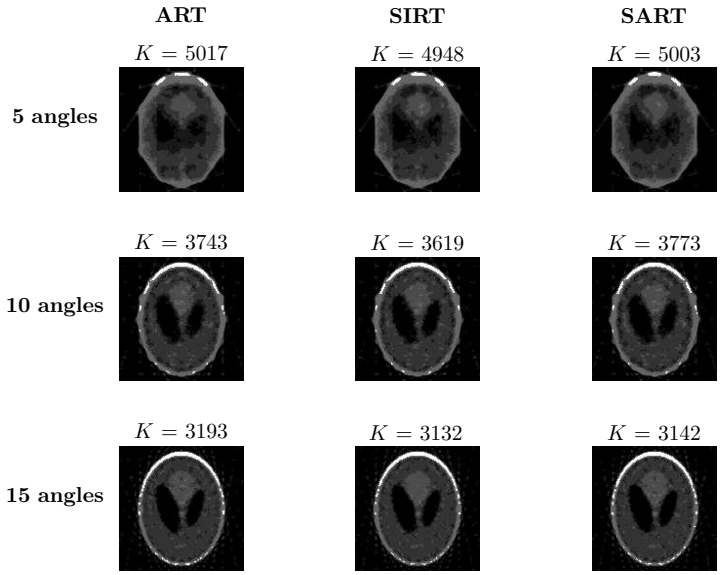


Figure 4.5: *DART reconstructions of the phantom after 100 iterations using various ARM's and projection angels*

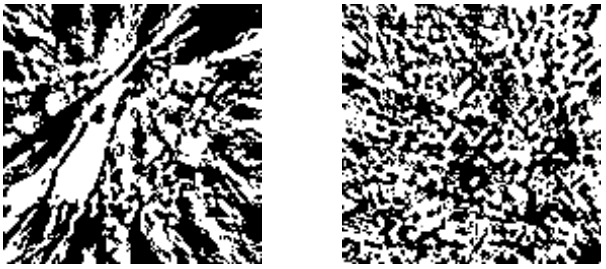


Figure 4.6: *The divergence of ART. Left: Reconstruction of the blob with* $p = 1$ *and 10 angles. Right: Reconstruction of shapes with* $p = 0.99$ *and 10 angles.*

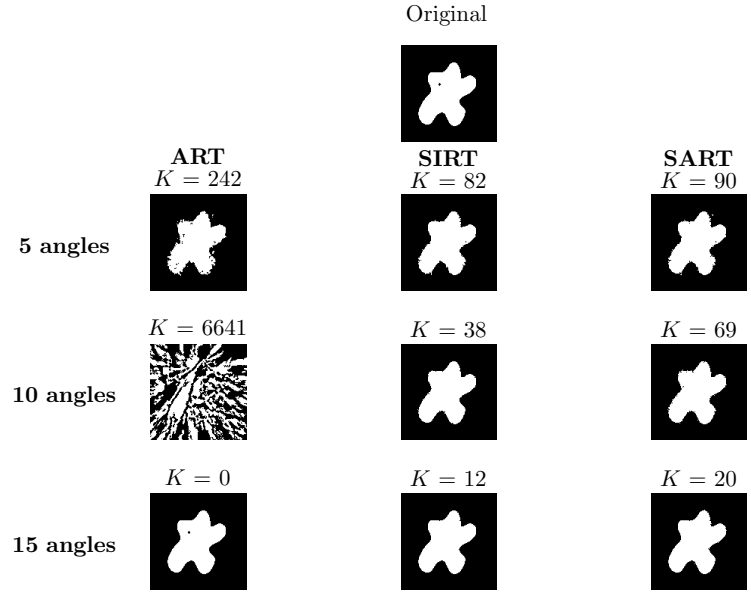


Figure 4.7: *DART reconstructions of the blob with an hole after 100 iterations using various ARM's and projection angels*

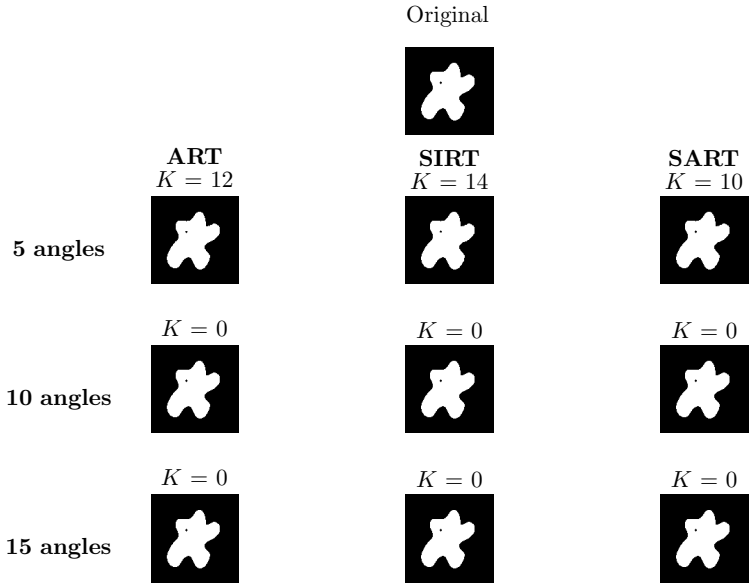


Figure 4.8: *DART reconstructions of the blob with a hole after 100 iterations using various ARM's and projection angels, fix probability* $p = 0.99$

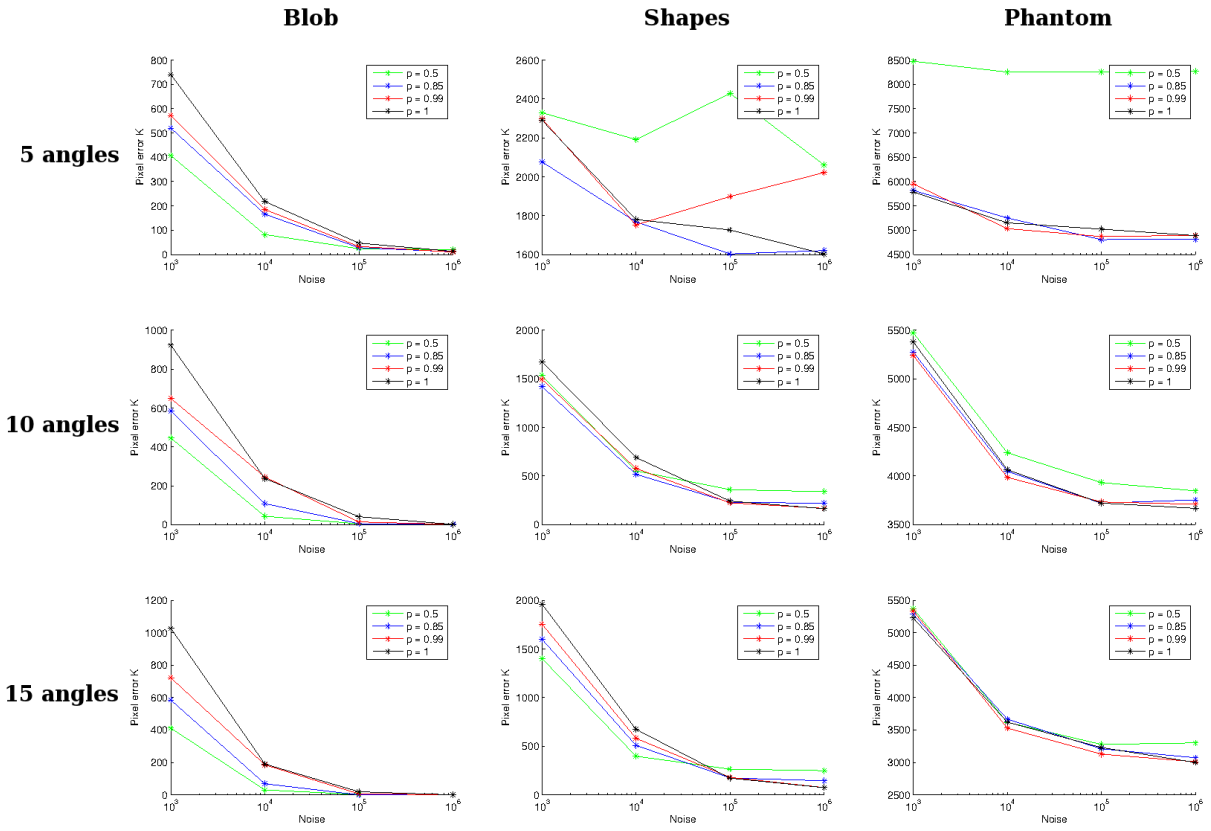
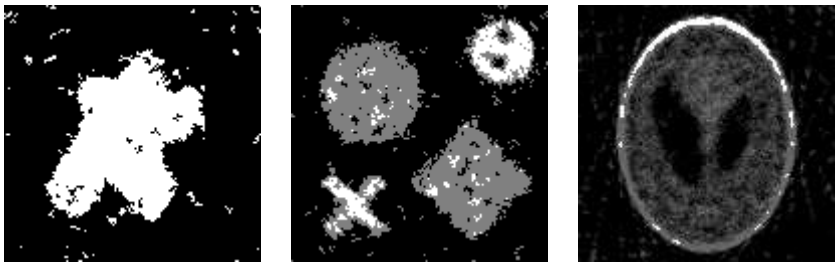


Figure 4.9: *The pixel error corresponding to four different noise levels for various p.*

For the experiments with noise only one ARM will be considered for the sake of clarity. SIRT is chosen as the ARM since the foregoing experiments and reasoning suggested that this is the best choice. The sinograms of the images are polluted with Poisson distributed noise. The *number of counts per detector element* give a measure of the magnitude of the noise. A higher number of counts corresponds to less noise. Four noise levels are considered: $10^3, 10^4, 10^5$ and 10^6. Figure 4.9 shows the pixel error K after 100 DART iterations corresponding to the four different noise levels for various fix probabilities p.

From Figure 4.3 it was deduced that the blob was reconstructed virtually exact for all considered angles, this is still true for the high number of counts 10^5 and 10^6. For 10^3 the blob can still be seen, but the reconstruction suffers greatly, see Figure 4.10(a). Note that lower fix probabilities give better results for low number of counts.

Earlier it was seen that the shapes could not be properly reconstructed from only 5 projection angles. It is therefore not to be expected that the reconstruction with noise would be any better. Indeed, Figure 4.9 shows that the pixel errors for the concerning reconstructions are very high. For 10 and 15 projection angles the reconstructions are much better if the number of counts are high. Again, for the low number of counts the pixel error is significant resulting in poor, but somewhat usable, reconstructions (Figure 4.10(b)). Also for this image low fix probabilities seem to produce better images in the case of low number of counts.



(a) (b) (c)

Figure 4.10: *DART reconstruction with noisy data (10^3) of (a) the blob with 5 angles and $p = 0.99$, (b) shapes with 10 angles and $p = 0.5$, (c) head phantom with 15 angles and $p = 0.85$.*

The reconstructions of the head phantom have very large pixel errors, this was to be expected since even without noise the reconstructions were particularly poor. Figure 4.10(c) shows the reconstruction of the head phantom using very noisy data originating from 15 angles with fix probability 0.85.

From Figure 4.9 one could deduce that low fix probabilities, such as $p = 0.5$, produce the best results if the projection data is very noisy. If the level of noise in the sinograms is relatively low then one would be wise to choose a higher fix probability, such as $p = 0.85$, since this might yield better results (see the pixel error of the shapes) and fewer computations have to be carried out. Choosing the fix probability too high might lead to reconstructions that overlooked holes in the object, as seen before.

Chapter 5

Research Goals

This chapter will formulate the research questions and goals that will form the basis for the upcoming research. Some test cases will be described that will be involved in answering the posed questions.

5.1 Research Questions

Chapter 4 described the DART algorithm in detail. Although the performance of DART is reasonable, its approach is very heuristic. The random subset construct, for example, is needed to find any holes in the object. The reduced system (4.7) contains only the free pixels as variables, which are the boundary pixels supplemented with the random subset. The effects of the noise, which is present in the projection data $\mathbf{p}$, is distributed over these free variables only. This causes heavy fluctuations in their values. Smoothing with the Gaussian filter counters this effect, but there is no theoretical motivation which justifies the use of this blurring operator. In Chapter 3 the concept of regularization was introduced as a tool to reduce the effects of noise on the reconstructions. DART might benefit from the use of regularization on the set of free pixels U. Although the experiments carried out in Section 4.3.2 suggested that SIRT was the ARM of choice for DART, it was not really motivated by sound arguments.

The main goal of the upcoming research will be to investigate if the DART algorithm can be improved. Hence the general research question will be:

Can the DART algorithm be improved?

More specific questions can be asked in the context of this general research question:

- Which algorithm should be used as ARM in DART and does it matter?
- Can better results be obtained by introducing regularization directly onto the set of free pixels U?
- Are there alternatives for the random subset construct?

5.2 Methodology

This section will describe what approach will be used to answer each of the research questions.

Which algorithm should be used as ARM in DART and does it matter?

To answer this question further experiments have to be carried out. Analysis of the results of these experiments will most likely yield what the best ARM is.

Can better results be obtained by introducing regularization directly onto the set of free pixels U?

Regularization can be interpreted in many different ways. One approach would be to use Tikhonov regularization, see (3.39), on the free pixels. One would in that case solve to following problem:

$$\begin{pmatrix} W \\ \lambda I \end{pmatrix} \mathbf{f}^U = \begin{pmatrix} \mathbf{p} - \sum_{j \in F} \mathbf{w}_{:,j} f_j \\ 0 \end{pmatrix}, \tag{5.1}$$

with $\mathbf{f}^U$ the vector containing the free pixels.

Are there alternatives for the random subset construct?

DART typically attempts to solve the discrete minimization problem

$$\min_{\mathbf{f} \in \{\rho_1, \ldots, \rho_l\}^N} \|W\mathbf{f} - \mathbf{p}\|. \tag{5.2}$$

The problem with DART is that the fixing of pixels might cause holes in the object to be overlooked. One can alternatively consider the following problem:

$$\begin{pmatrix} W \\ D \end{pmatrix} \mathbf{f} = \begin{pmatrix} \mathbf{p} \\ D\mathbf{v} \end{pmatrix}, \tag{5.3}$$

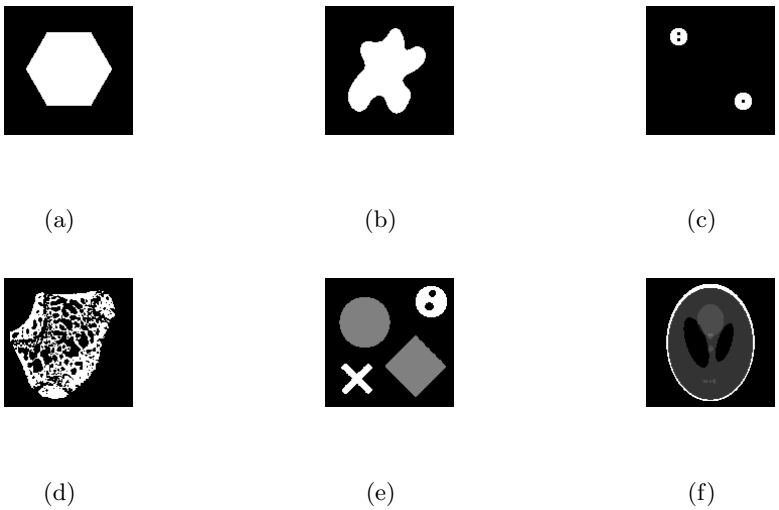
with D a diagonal matrix and $\mathbf{v}$ a vector containing the presumed values of the corresponding pixels. This system will steer the solution towards the presumed values in $\mathbf{v}$ but the pixels are allowed to deviate from these values. The entries d_i, $i = 1, 2, \ldots, N$, of the matrix D represent the degree to which the pixels of the solution should be steered towards the values in $\mathbf{v}$. Solving 5.3 is equivalent to the minimization problem

$$\min_{\mathbf{f} \in \mathbb{R}^{\mathbf{N}}} \|W\mathbf{f} - \mathbf{p}\| + \|D\mathbf{f} - D\mathbf{v}\|. \tag{5.4}$$

Note that deviation of the proposed values v_i is penalised proportional to d_i. Very high values of d_i will mostly likely result into a system where pixel i has grey values v_i. A smaller d_i indicate some uncertainty associated with the value v_i, the grey value of these pixels are more likely to deviate from the presumed values.

This approach gives rise to the following sub questions:

- Can this approach lead to similar behaviour as the original DART?
- How should the d_i be chosen?



(a) (b) (c)

(d) (e) (f)

Figure 5.1: *Test problems (a) Hexagon (b) The blob (c) Two circles with holes (d) Bone (e) Shapes (f) Phantom*

5.3 Test Problems

The research questions posed in the previous section call for modifications of the DART implementation. Naturally these new implementations have to be validated, the results of experiments have to be verified and compared to the results of the original algorithm. To do this validation some *test problems* will be presented. The test problems have some hierarchical structure ranging from simple objects to more intricate shapes. At first only binary images will be investigated. When the results of the binary images are satisfying DART is used to tackle images with more than two grey values. The problems are motivated by a range of features:

- The shape of the object;
- The ratio of edges to the total number of pixels;
- The number of grey values.

The shape of the object is a very broad notion. Objects can be very simple, e.g. symmetric, homogeneous and convex, or more complex, the image might even consist of multiple objects. Edges in the image are very important in DART since these are the only pixels that are subject to change. Images with a low ratio of edges require less computation time since a great number of pixels are fixed and thus a smaller system is considered. It is obvious why the number of different grey values is an important feature. Figure 5.1 shows the six test problems that will be used to answer the research questions.

The main interest of this research is the performance of the improved implementation when one deals with noisy projection data. Therefore, the sinograms of the images will be polluted with Poisson distributed noise. Four different number of counts will be used: $10^3, 10^4, 10^5$ and 10^6. Recall that low number of counts correspond to more noise.

The experiments will reconstruct the objects from data acquired from a varying number of angles. Naturally, simple objects such as 5.1(a) will require fewer angles to obtain a perfect reconstruction than the more complex problems such as 5.1(d). Also the range from which

the angles are acquired will be investigated, smaller angular ranges will most likely result in poorer reconstructions. Also the size, mainly the ratio of edges to total number of pixels, is an important feature which needs to be taken into account.

In all cases the performance of the improved algorithm will be measured by looking at the pixel error of the reconstructions, similarly as was done in Section 4.3.2.

Bibliography

[1] A. Andersen and A. C. Kak. Simultaneous algebraic reconstruction technique (SART): A superior implementation of the ART algorithm. *Ultrasonic Imaging*, 6(1):81–94, 1984.

[2] K. J. Batenburg. A network flow algorithm for reconstructing binary images from continuous x-rays. *J. Math. Imaging Vis.*, 30(3):231–248, 2008.

[3] K. .J Batenburg and J. Sijbers. *Dart: A Fast Heuristic Algebraic Reconstruction Algorithm for Discrete Tomography*, pages IV – 133–IV – 136. IEEE, 2007.

[4] K. J. Batenburg and J. Sijbers. DART: A practical reconstruction algorithm for discrete tomography. *IEEE Transactions on Image Processing*, 20(9):2542–2553, 2011.

[5] T. M. Buzug. *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Springer, Berlin, 2008.

[6] J. W. Demmel. *Applied numerical linear algebra*. SIAM, Philadelphia, 1997.

[7] L. A. Feldkamp, L. C. Davis, and J. W. Kress. Practical cone-beam algorithm. *Journal of the Optical Society of America A*, 1(6):612–619, June 1984.

[8] D. Gale. A theorem on flows in networks. *Pacific Journal of Mathematics*, 7(2):1073–1082, 1957.

[9] R.J. Gardner, P. Gritzmann, and D. Prangenberg. On the computational complexity of reconstructing lattice sets from their x-rays. *Discrete Mathematics*, 202(13):45–71, 1999.

[10] J. Gregor and T. Benson. Computational analysis and improvement of SIRT. *IEEE Transactions on Medical Imaging*, 27(7):918–924, 2008.

[11] P. Gritzmann, S. de Vries, and M. Wiegelmann. Approximating binary images from discrete x-rays. *SIAM Journal on Optimization*, 11(2):522–546, 2000.

[12] P. C. Hansen. *Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion*. SIAM, Philadelphia, 1998.

[13] P. C. Hansen and M. Saxild-Hansen. AIR tools a MATLAB package of algebraic iterative reconstruction methods. *Journal of Computational and Applied Mathematics*, 236(8):2167–2178, 2012.

[14] G. T. Herman. *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography*. Academic Press, New York, 1980.

[15] G. T. Herman. *Fundamentals of Computerized Tomography: Image Reconstruction from Projections*. Springer, London, 2009.

[16] M. Jiang and G. Wang. Convergence of the simultaneous algebraic reconstruction technique (SART). *IEEE Transactions on Image Processing*, 12(8):957–961, August 2003.

[17] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging*. IEEE Press, New York, 1987.

[18] A. Kuba and G. T. Herman. Discrete tomography: A historical overview. In A. Kuba and G. T. Herman, editors, *Discrete Tomography: Foundations, Algorithms and Applications*, pages 3–34. Birkhäuser, Boston, 1999.

[19] H. Y. Liao and G. T. Herman. A coordinate ascent approach to tomographic reconstruction of label images from a few projection. *Discrete Applied Mathematics*, 151(1-3):184–197, 2005.

[20] F. Riesz and B. S. *Functional Analysis*. Dover Publications, Mineola, New York, 1990.

[21] H. J. Ryser. Combinatorial properties of matrices of zeros and ones. *Canadian Journal of Mathematics*, 9:371–377, 1957.

[22] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, second edition, 2003.

[23] T. Schüle, C. Schnörr, S. Weber, and J. Hornegger. Discrete tomography by convex-concave regularization and d.c. programming. *Discrete Applied Mathematics*, 151(1-3):229–243, 2005.

[24] H. Stark, J. W. Woods, I. Paul, and R. Hingorani. Direct rourier reconstruction in computer tomography. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29(2):237–245, 1981.

[25] A. van der Sluis and H. A. van der Vorst. Numerical solution of large, sparse linear algebraic systems arising from tomographic problems. In G. Nolet, editor, *Seismic Tomography With Applications in Global Seismology and Exploration Geophysics*, pages 49–83. D. Reidel Publishing Company, Dordrecht, 1987.

[26] S. Webb. *From the Watching of Shadows: The Origins of Radiological Tomography*. Taylor & Francis, Bristol, 1990.