A Survey of Algebraic Algorithms in Computerized Tomography

Martin A. Brooks

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Modelling and Computational Science (MSc)

in the

Faculty of Science

University of Ontario Institute of Technology

August 2010

Abstract

X-ray computed tomography (CT) is a medical imaging framework. It takes measured projections of X-rays through two-dimensional cross-sections of an object from multiple angles and incorporates algorithms in building a sequence of two-dimensional reconstructions of the interior structure. This thesis comprises a review of the different types of algebraic algorithms used in X-ray CT. Using simulated test data, I evaluate the viability of algorithmic alternatives that could potentially reduce over-exposure to radiation, as this is seen as a major health concern and the limiting factor in the advancement of CT [36, 34]. Most of the current evaluations in the literature [31, 39, 11] deal with low-resolution reconstructions and the results are impressive, however, modern CT applications demand very high-resolution imaging. Consequently, I selected five of the fundamental algebraic reconstruction algorithms (ART, SART, Cimmino's Method, CAV, DROP) for extensive testing and the results are reported in this thesis. The quantitative numerical results obtained in this study, confirm the qualitative suggestion that algebraic techniques are not yet adequate for practical use. However, as algebraic techniques can actually produce an image from corrupt and/or missing data, I conclude that further refinement of algebraic techniques may ultimately lead to a breakthrough in CT.

Contents

Acknowledgements

First and foremost, I would like to thank my supervisor Dr. Dhavide Aruliah for all his help and patience in the completion of this thesis.

Also, I would like to thank Dr. Gabor Herman for his assistance in understanding some of the aspects of the SNARK05 software package.

Finally, I would like to thank my father, my mother and Geneviève for their support throughout all the stages of this thesis.

List of Figures

List of Tables

Definitions and Notation

A	matrix of size $M \times N$
$A_{i,:}$	row i of A, where $i = 1 \dots M$
$A_{:,j}$	column j of$A$, where $j = 1 \dots N$
$A_{i,j}$	entry (i, j) of A
A^T	transpose of A
$A_{i,+}$	column vector of row sums of A, i.e., $A_{i,+} := \sum_{j=1}^{N} A_{i,j}$
$A_{+,j}$	row vector of column sums of A, i.e., $A_{+,j} := \sum_{i=1}^{M} A_{i,j}$
s_j	number of non-zero elements in column j of A
w_i	user chosen weight associated with row i of A
$x \cdot y$	standard dot product or inner product, $x \cdot y = \sum_{k=1}^{n} x_k y_k$
$\{m_i\}_{i=1}^{M}$	the set defined by $\{m_1, m_2, m_3, \dots, m_M\}$

Chapter 1

Introduction

Computed tomography (CT) is a medical imaging technique using two-dimensional projections through an object from various angles to generate a a model of the interior structure of the object through a sequence of two-dimensional cross-sections. CT has had a radical impact in the field of medicine, but it has also significantly helped in other areas, such as materials testing [19], microscopic imaging [13], and geology [46]. There are many different variations of CT and many ways to reconstruct an image. Johann Radon was the first to solve the image reconstruction problem analytically early in the 1900s [40]. Since then, there have been many algorithmic solutions developed. Image reconstruction algorithms fall mainly into two categories: direct methods based on filtered backprojection (FBP) and iterative algebraic methods. FBP, given noise-free data, is able to reconstruct and produce a result for any allowable error deviation. However, if we are missing data (i.e. on a specific angular interval), FBP is not usable and we turn to iterative methods.

Chapter 2 of this thesis deals with the evolution of computerized tomography. The chapter begins with a description of computerized tomography and the modeling assumptions used. It then follows with an explanation of the filtered backprojection algorithm (FBP), which presently is the method most used in CT machines [34, 36] (FBP is considered a direct method). This includes the definition of the Radon transform, the Fourier transform and the derivation of the FBP algorithm. Lastly, we discuss how the Radon transform is discretized to actually perform the necessary computations.

Chapter 3 is strictly concerned with algebraic reconstruction techniques, which all fall under the category of iterative methods. We first discuss the approach from an algebraic standpoint applied to X-ray CT and then break down and examine five different algorithms. These are Kaczmarz's method (ART), Cimmino's method (CIM), simultaneous algebraic reconstruction technique (SART), component averaging (CAV) and diagonally relaxed orthogonal projections (DROP). A timeline of their development is as follows

Year	**Algorithm**
1937	ART
1938	Cimmino
1984	SART
2001	CAV
2005	DROP

Table 1.1: Timeline of Algorithm Development

These algorithms were chosen out of a larger set of possible algorithms for the following reasons:

- ART and Cimmino's method were the first algebraic reconstruction algorithms developed on which most of the current research in the field is derived [29, 12, 28].

- SART was seen as a significant refinement of ART [3] and has continued to be researched recently [28].
- CAV and DROP were unique modern adaptations of ART, SART and Cimmino's method. To clarify, they have not been developed by simply adding a relaxation parameter or a weighting system. Both CAV and DROP had a novel approach to the reconstruction problem [28, 20, 11].

X-ray CT scanners used in modern hospitals typically generate images 512×512 pixels in size [38]. Yet, the current literature reporting theoretical test cases is dominated by reconstructions varying from 8×8 to 256×256 pixels in size [31, 39, 11, 15, 4]. Clearly algebraic reconstruction techniques involving theoretical test cases at the larger scale are merited if realistic comparisons are to be obtained.

Finally, in chapter 4, I present the results of three different studies and measure how each algorithm performs. The first experiment is a head phantom taken from SNARK05 [23]. Head phantoms are very common benchmarks in medical imaging, and we look at both the noiseless and noisy case. The second experiment is a two-dimensional cross section of a plant mitochondrion cell [18]. Many different cases are considered, including simulating missing data as well as variations in the resolution of the reconstruction for both noiseless and noisy cases. The last experiment that we consider is called a circle clock phantom (adapted from the sphere clock phantom of [47]).

The limiting factor in the advancement of CT is not the processing power of computers but the overexposure of radiation [34], which has been linked to cancer [14, 43].

Current CT machines expose patients to large doses of radiation to increase the signal-to-noise ratio and improve the accuracy of reconstructions [14, 43]. This means that efforts to decrease exposure to X-rays will also reduce the accuracy of a reconstruction. One approach to mitigate this degradation that can be considered is to limit exposure to simply the region of interest. This results in partial and incomplete data sets.

One of the main strengths of iterative algebraic reconstruction algorithms is their ability to compute compute physically reasonable reconstruction from incomplete or partial CT X-ray data. I wish to investigate whether this reconstruction can be performed adequately on large scale examples using limited viewing angles and noisy environments. This thesis identifies areas where the fundamental algebraic reconstruction algorithms (ART, CIM, SART) and modern adaptations (CAV, DROP) excel, where they fail, and if they can be viable for use in reconstructing X-ray images scaled to real-world proportions.

Chapter 2

Evolution of Computerized Tomography

2.1 What is Computed Tomography?

Computed tomography (CT) was introduced over 30 years ago, by Sir Godfrey Hounsfield of EMI Laboratories, England, and by Allan Cormack of Tufts University, Massachusetts. Tomography itself consists of the reconstruction of an image from its projections [25, 26]. The first mathematical solution to perform the reconstruction of these projections was published by Johann Radon in 1917 [40], but it was not until the 1970s that Hounsfield and Cormack made practical X-ray CT a reality. Hounsfield and Cormack both worked independently and came up with a similar process. Hounsfield's idea was announced in 1972, and both shared the 1979 Nobel Prize in Medicine [1].

Hounsfield originally used algebraic techniques similar to ART to compute CT reconstructions [25, 26]. At that time, algebraic techniques were much less feasible due to hardware restrictions in memory and processor speed. Ramachandran and Lakshminarayanan [41] were the first to apply backprojection algorithms to this problem. Later, Shepp and Logan [45] refined previous algorithms and applied their own backprojection algorithm to CT.

2.2 Modeling assumptions

An X-ray is comprised of a stream of photons, and when an X-ray passes through matter, there are three potential outcomes for those individual photons. The first option is transmission where a photon passes through the matter completely unaffected. The second option is absorption where all of a photon's energy is transferred to the matter. The last option is scattering where a photon is redirected and with a potential loss of energy. These processes follow principles rooted in quantum mechanics, which will not be discussed here [2]. Though it is impossible to predict whether an individual photon will be transmitted, absorbed or scattered , it is possible to be predict the percentage of overall photons in each category [2].

The model we adopt for the interaction of matter with X-rays is based on three assumptions [16]:

1. No refraction or diffraction is present, i.e., X-rays travel along straight lines and are not bent by objects they go through.
2. All X-rays are assumed to be monochromatic, i.e., all waves making up the X-ray are of the same wavelength.

3. All materials attenuate X-rays of a given energy linearly. That is, the X-ray beam intensity satisfies Beer's Law

$$\frac{dI}{ds} = -\mu[(x(s))]I. \tag{2.1}$$

   where s is the arclength along the straight-line trajectory $x(s)$ and μ is a material-dependent parameter referred to as the attenuation coefficient.

When some individual X-ray photons are absorbed by matter and some are transmitted, the intensity of the incident X-ray beam attenuates. Beer's law provides a quantitative description of the beam attenuation through matter. The numerical value of the attenuation coefficient depends on the material the X-ray is passing through.

Radiologists typically measure attenuation coefficients in Hounsfield units (HU). The value H_{tissue} of the attenuation coefficient of a particular tissue in HU is scaled relative to the value of the attenuation coefficient μ_{water} of water, i.e.,

$$H_{tissue} = \frac{\mu_{tissue} - \mu_{water}}{\mu_{water}} \times 1000. \tag{2.2}$$

Thus, with the scaling in (2.2), the attenuation coefficient of water is 0 HU and a substance with attenuation coefficient of 1000 HU attenuates X-rays twice as much as water. For X-ray CT to be beneficial, the reconstruction made must be accurate to around 10 HU due to the fact that the variation in the attenuation coefficients of soft tissue is about 2% (see Table 2.1).

Material	**Density** $(\mathrm{kg} \cdot \mathrm{m}^3)$ at 20°C	**Attenuation coefficient** (HU)
air	1.204	-1000
fat	918	-61
brain tissue	1050	-4
water	998.2071	0
muscle tissue	1060	41
blood	1060	53
bone	3880 (males)	1086

Table 2.1: Sample attenuation coefficients

2.3 Filtered Backprojection

Filtered backprojection (FBP) is the most popular and most widely used algorithm in applications of computerized tomography. The FBP algorithm is derived from the Fourier Slice Theorem [30]. The derivation of FBP involves polar coordinates in the inverse Fourier transform and manipulations of the limits of integration.

Filtered backprojection consists of two main steps:

- Data filtering (from the Fourier domain to the spatial domain)
- Backprojection

The presentation of FBP here is based on that in [30].

2.3.1 Projections and the Radon Transform

To define a projection, we first need to define what is meant by a line integral. Simply, a line integral represents the integral of a parameter of an object along a line. The line integral represents the total attenuation of a X-ray beam as it goes through the object. Figure 2.1 shows the coordinate system that is used. The object that we are trying to reconstruct is defined as a 2-D function $\mu(x, y)$ used in the Radon transform. The lines are represented by the parameters (t, θ); note that the line integrals are combined to form the projections of an object.

The line “Ray 1” from Figure 2.1 can be described algebraically by

$$x \cos\theta + y \sin\theta = t_1. \tag{2.3}$$

Therefore the line integral $\mathcal{R}[\mu](t, \theta)$ can be defined as

$$\mathcal{R}[\mu](t, \theta) = \int_{\ell_{(t,\theta)}} \mu(x, y) ds. \tag{2.4}$$

We can rewrite (2.4) using a delta function as

$$\mathcal{R}[\mu](t, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x, y) \delta(x \cos\theta + y \sin\theta - t) dx dy. \tag{2.5}$$

The function in (2.4) or (2.5) is known as the Radon transform of $\mu(x, y)$. In the context of X-ray computed tomography, the Radon transform is often referred to as a sinogram.

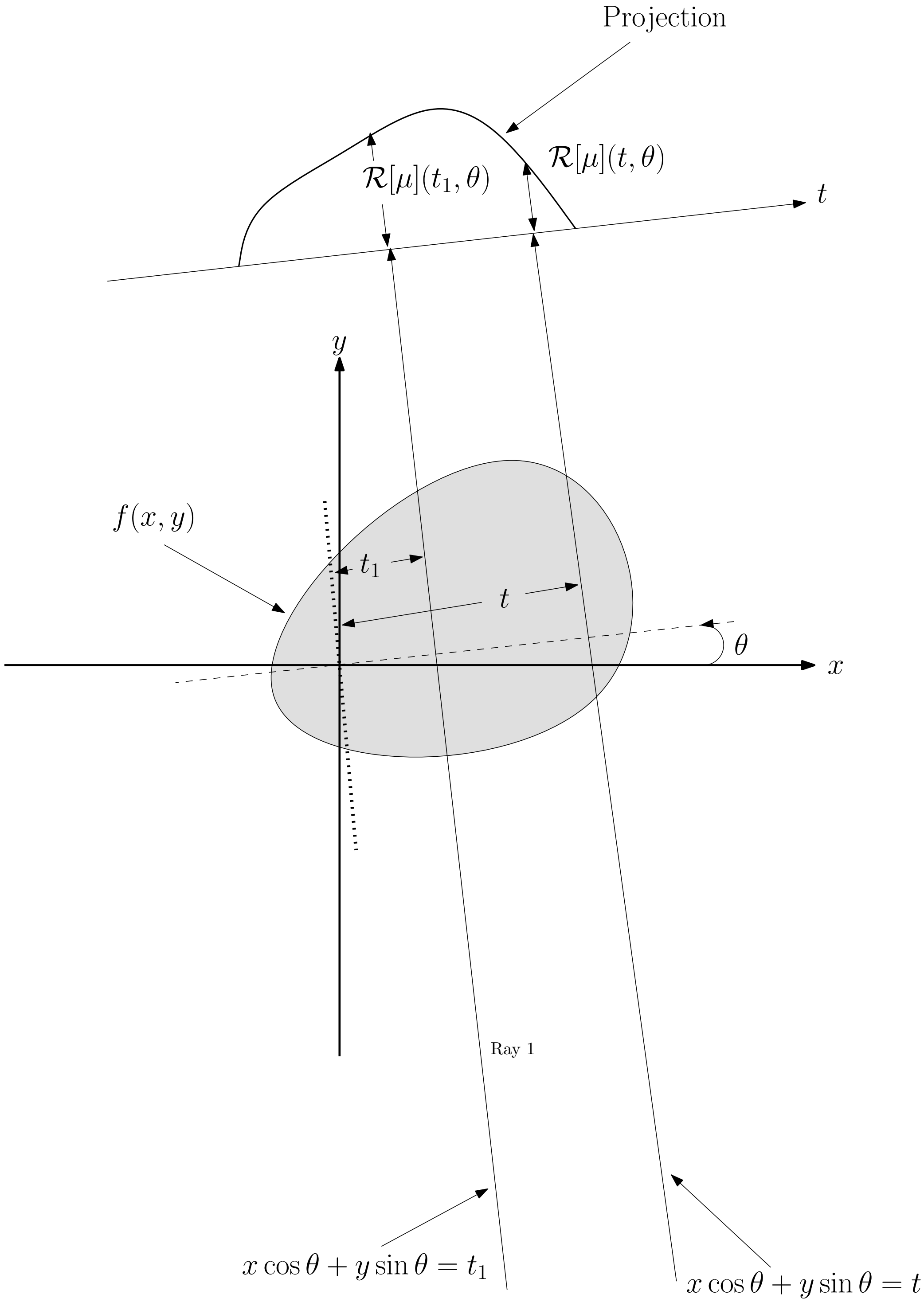


Figure 2.1: For angle θ, the object $\mu(x,y)$ and its projection $\mathcal{R}[\mu](t,\theta)$

2.3.2 The Fourier Transform

In describing the filtered backprojection algorithm, a few ideas from Fourier analysis are needed, namely the Fourier transform, the inverse Fourier transform and the Fourier Slice Theorem.

Definition: The Fourier transform of an integrable function $\mu : \mathbb{R} \to \mathbb{C}$ can be defined as

$$\hat{\mu}(u) = \int_{-\infty}^{\infty} \mu(x) e^{-i2\pi(ux)} dx. \tag{2.6}$$

Definition: The Fourier transform of an integrable function $\mu : \mathbb{R}^2 \to \mathbb{C}^2$ can be defined as

$$\hat{\mu}(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x, y) e^{-i2\pi(ux+vy)} dxdy. \tag{2.7}$$

The Fourier transform is an operator mapping an integrable function to another integrable function. It can be thought of as the decomposition of a function into its harmonic components. For a complete overview of the Fourier transform, see [7, 17, 27, 2].

Definition: The one-dimensional inverse Fourier transform of an integrable function can be defined as

$$\mu(x) = \int_{-\infty}^{\infty} \hat{\mu}(u) e^{i2\pi(ux)} du. \tag{2.8}$$

Definition: The two-dimensional inverse Fourier transform of an integrable function can be defined as

$$\mu(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \hat{\mu}(u,v)e^{i2\pi(ux+vy)}dudv. \tag{2.9}$$

2.3.3 The Fourier Slice Theorem

Theorem 1. ***Fourier Slice Theorem***

Let the image $\mu(x,y)$ have a two-dimensional Fourier transform $\hat{\mu}(u,v)$ and Radon transform $\mathcal{R}[\mu](t,\theta)$. If $\widehat{\mathcal{R}[\mu]}(w,\theta)$ is the one-dimensional Fourier transform with respect to the affine distance t, then

$$\widehat{\mathcal{R}[\mu]}(w,\theta) = \hat{\mu}(w\cos\theta, w\sin\theta).$$

That is, the Fourier transform of all projections of $\mu(x,y)$ normal to the vector $\vec{n} = (\cos\theta, \sin\theta)$ is a slice through the origin of $\hat{\mu}(u,v)$ in the direction of $\vec{n}$.

The complete derivation of the Fourier Slice Theorem can be found in [30]. Its principle result states that each segment of projection data (at some angle θ) is identical to the Fourier transform of the multi-dimensional object at θ. The key significance of the Fourier Slice Theorem in medical imaging lies in the fact that the measured sinogram data in an X-ray CT scanner is in fact the Radon transform of the attenuation coefficient. This allows for the physical measurements from an X-ray CT scanner (the Radon transform data) to be analyzed by the tools of discrete Fourier analysis. Essentially, this lets the measured sinogram data be used to reconstruct the 2D Fourier transform of μ.

This can be thought of in three main steps:

1. Measure or obtain the Radon transform data, $\mathcal{R}[\mu](t,\theta)$.
2. Compute the Fourier transform of that data, $\widehat{\mathcal{R}[\mu]}(w,\theta)$.
3. From here, we can compute the values $\mu(x,y)$.

2.3.4 Derivation of the Filtered Backprojection Algorithm

Using the formula for the inverse Fourier transform (2.9), we change the coordinate system from rectangular in the frequency domain (u,v) to the polar coordinate system (w,θ) by performing the following substitution

$$\begin{aligned} u &= w\cos\theta \\ v &= w\sin\theta \\ du\,dv &= w\,dw\,d\theta \end{aligned}$$

obtaining

$$\mu(x,y) = \int_0^{2\pi}\int_0^{\infty} \hat{\mu}(w,\theta)e^{i2\pi w(x\cos\theta+y\sin\theta)}w\,dw\,d\theta. \tag{2.10}$$

We can break up the previous integral into two parts

$$\begin{aligned} \mu(x,y) = &\int_0^{\pi}\int_0^{\infty} \hat{\mu}(w,\theta)e^{i2\pi w(x\cos\theta+y\sin\theta)}w\,dw\,d\theta \\ &+ \int_0^{\pi}\int_0^{\infty} \hat{\mu}(w,\theta+\pi)e^{i2\pi w(x\cos(\theta+\pi)+y\sin(\theta+\pi))}w\,dw\,d\theta. \end{aligned}$$

Using the property $\mu(w,\theta+\pi) = \hat{\mu}(-w,\theta)$ [30] and the substitution $t = x\cos\theta + y\sin\theta$, $\mu(x,y)$ can now be written as

$$\mu(x, y) = \int_{0}^{\pi} \left[\int_{-\infty}^{\infty} \hat{\mu}(w, \theta)|w| \, e^{i2\pi wt} dw \right] d\theta. \tag{2.11}$$

The partial Fourier transform of $\mathcal{R}[\mu](t, \theta)$ with respect to t is defined as

$$\widehat{\mathcal{R}[\mu]}(w, \theta) = \int_{-\infty}^{\infty} \mathcal{R}[\mu](t, \theta) e^{-i2\pi wt} dt. \tag{2.12}$$

Using the Fourier Slice Theorem , substitute $\widehat{\mathcal{R}[\mu]}(w, \theta)$ for the 2-D Fourier transform $\hat{\mu}(w, \theta)$,

$$\mu(x, y) = \int_{0}^{\pi} \left[\int_{-\infty}^{\infty} \widehat{\mathcal{R}[\mu]}(w, \theta)|w| \, e^{i2\pi wt} dw \right] d\theta. \tag{2.13}$$

Defining the filtered operator (see section 2.3.5 for details on how filtering is performed) as

$$\widetilde{\mathcal{R}[\mu]}(t, \theta) = \int_{-\infty}^{\infty} \widehat{\mathcal{R}[\mu]}(w, \theta)|w| \, e^{i2\pi wt} dw, \tag{2.14}$$

and expressing (2.13) with $\widetilde{\mathcal{R}[\mu]}(t, \theta)$

$$\mu(x, y) = \int_{0}^{\infty} \widetilde{\mathcal{R}[\mu]}(x \cos\theta + y \sin\theta, \theta) \, d\theta \tag{2.15}$$

$\widetilde{\mathcal{R}[\mu]}(t, \theta)$ can be thought of as a filtered projection. For all angles θ, each resulting projection is added to form an estimate of $\mu(x, y)$. In (2.15) $\widetilde{\mathcal{R}[\mu]}(t, \theta)$ is backprojected for each angle θ.

This process can be thought of as follows. Each point (x, y) in the plane matches up with a value of $t = x\cos\theta + y\sin\theta$ for a value of θ. $\widetilde{\mathcal{R}[\mu]}(t,\theta)$ adds its value to the reconstruction at t also. So we can conclude that $\widetilde{\mathcal{R}[\mu]}(t,\theta)$ makes the absolute same contribution to the reconstruction at all of these points. Each filtered projection, $\widetilde{\mathcal{R}[\mu]}(t,\theta)$ is backprojected over the image plane.

2.3.5 Filtering

Filtering is used to, ideally, form a more precise reconstruction by removing the effects of noise. Let us consider an example where the noise present in a signal is random. Therefore, we can say the average amount of noise over time should be 0. Let μ represent a noisy signal and $\psi \in \mathbb{R}$ and $\psi > 0$. The average value of μ over $x - \psi \leq t \leq x + \psi$ for each value of x is

$$\mu_{ave}(x) = \frac{1}{2\psi}\int_{t=x-\psi}^{t=x+\psi} \mu(t)\, dt. \tag{2.16}$$

This function μ_{ave} is a filtered version of the original μ. For a suitable ψ value, μ_{ave} would be close to a noiseless signal since the noise should average out to 0. We can consider an alternate function Γ defined as follows

$$\Gamma_{\psi}(t) = \begin{cases} 1 & \text{if } -\psi \leq t \leq \psi, \\ 0 & \text{if } |t| > \psi. \end{cases} \tag{2.17}$$

Definition: A function φ which has a nonzero Fourier transform on some finite interval and has a value of zero outside that interval will be referred to as a band-limited function.

Looking at (2.14), we would like to replace $|w|$ by a filter that is the Fourier transform of a band limited function. The usual way to design a filter is to replace the absolute value function with a function β of the following form

$$\beta(\chi) = |\chi| \cdot F(\chi) \cdot \Gamma_L(\chi), \tag{2.18}$$

for $L > 0$. We can then see that

- $\beta(\chi)$ vanishes for $|\chi| > L$,
- $\beta(\chi)$ has the value $|\chi| \cdot F(\chi)$ when $|\chi| \leq L$.

F should be chosen as an even function for which $F(0) = 1$ since near the origin, the value of β is close to the absolute value.

There are many different filters used in medical imaging, some of which are

- the Ram-Lak filter,
- the cosine filter,
- the Shepp-Logan filter,
- the Hann filter,
- the Hamming filter.

For a description of individual filters, please see [16, 30, 41, 45, 17].

2.4 Discrete Formulations

2.4.1 Discrete Radon Transform

The central problem of X-ray computed tomography in two dimensions is to reconstruct the X-ray attenuation coefficient μ (a nonnegative, scalar-valued function of two spatial variables) over a certain spatial region from measurements of its Radon transform. Recall the definition of the Radon transform is

$$\mathcal{R}[\mu](t,\theta) = \iint_{\mathbb{R}^2} \mu(x,y)\delta(x\cos\theta + y\sin\theta - t)\,dxdy \tag{2.19a}$$

$$= \int_{\ell_{(t,\theta)}} \mu(x,y)\,ds. \tag{2.19b}$$

In (2.19), the line $\ell_{(t,\theta)}$ is described by the parameters (t,θ); $t \in \mathbb{R}$ is the (signed) displacement of the line from the origin and $\theta \in (-\pi,\pi]$ is the angle of orientation of the normal direction of the line. The line integral in (2.19b) is computed with respect to arclength along the line $\ell_{(t,\theta)}$.

Naturally, the number of lines along which the Radon transform can be sampled is finite. If a parallel-beam X-ray CT scanner has n_t detectors in a straight line at displacements $\{t_p\}_{p=1}^{n_t}$ from the centre and measurements are taken at n_θ projection angles $\{\theta_q\}_{q=1}^{n_\theta}$, then the total number of measurements taken is $M := n_t \times n_\theta$. The samples of the Radon transform can typically be represented as

$$\{r_{p,q} \mid r_{p,q} = \mathcal{R}[\mu](t_p,\theta_q),\ p = 1\colon n_t, q = 1\colon n_\theta\}. \tag{2.20}$$

2.4.2 Discrete filtering

Sampling is the process by which a discrete set of points are chosen from a function defined over all real numbers.

Theorem 2. ***Nyquist's Theorem***

If μ is an integrable band-limited function such that $\hat{\mu}(\chi) = 0$ when $|\chi| > L$, then, $\forall\, x \in \mathbb{R}$

$$\mu(x) = \sum_{n=-\infty}^{\infty} \mu\left(\frac{\pi n}{L}\right) \cdot \frac{\sin(Lx - n\pi)}{Lx - nx}. \tag{2.21}$$

Nyquist's Theorem states that any value of μ can be interpolated from $\left\{\mu\left(\frac{n\pi}{L}\right)\right\}$. We can also find out how many sampled values are needed to obtain an accurate representation of β (2.18). Specific examples of the use of Nyquist's Theorem with various filters can be found in [30, 17].

2.4.3 Discrete Fourier Transform

As seen previously (2.7), for a continuous and integrable function $\mu : \mathbb{R}^2 \to \mathbb{C}^2$, the Fourier transform is defined as

$$\hat{\mu}(u, v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mu(x, y) e^{-i2\pi(ux+vy)} dxdy. \tag{2.22}$$

Since most practical data is digitized, we need a discrete equivalent for (2.22). The discrete Fourier transform (DFT) takes regularly spaced samples and returns the Fourier transform for this set of data in it's respective frequency space. This is accomplished by replacing the integrals by summations [8, 6]. A step-size of 1 is assumed over an $N \times M$ grid in x and y. This yields the following definition for the discrete Fourier transform, denoted as $\widehat{\mu_{\mathcal{D}}}$

$$\widehat{\mu_{\mathcal{D}}}(u, v) = \frac{1}{NM} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \mu(x, y) e^{-2\pi i(\frac{ux}{N} + \frac{vy}{M})}. \tag{2.23}$$

The corresponding inverse becomes

$$\mu(x,y) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \widehat{\mu_{\mathcal{D}}}(u,v) e^{2\pi i (\frac{ux}{N} + \frac{vy}{M})}. \tag{2.24}$$

2.4.4 Discrete Filtered Backprojection

First, let us recall the definition of the continuous backprojection

$$\mu(x,y) = \int_0^{\infty} \widetilde{\mathcal{R}[\mu]}(x\cos\theta + y\sin\theta, \theta)\, d\theta. \tag{2.25}$$

In the discrete case of backprojection, the angle θ is replaced by a discrete set of angles $\{\frac{k\pi}{N} : 0 \leq k \leq N-1\}$. Therefore $d\theta$ will become $\frac{\pi}{N}$ and similarly to the discrete version of the Fourier transform, the integral is replaced by a summation. This yields the definition of the discrete backprojection $\mu_{\mathcal{D}}$

$$\mu_{\mathcal{D}}(x,y) = \left(\frac{1}{N}\right) \sum_{k=0}^{N-1} \widetilde{\mathcal{R}[\mu]} \left(x\cos\left(\frac{k\pi}{N}\right) + y\sin\left(\frac{k\pi}{N}\right), \left(\frac{k\pi}{N}\right)\right). \tag{2.26}$$

For more detailed information about any of these discrete formulations, please consult [30, 17].

Chapter 3

Algebraic Reconstruction Techniques

3.1 The algebraic approach to X-ray CT

There is a quite different conceptual approach to CT reconstruction that is simpler than the Fourier-transform techniques of the last chapter. Assuming that the image to be reconstructed consists of an array of unknowns, one uses the projection data, i.e. the Radon transform, to set up a system of linear algebraic equations which are subsequently solved using iterative algorithms. Algebraic techniques are well suited to solve problems when it is not possible to measure many projections or the projections are not distributed evenly. The central idea is to convert the problem of reconstructing the attenuation coefficient μ associated with a two dimensional object with a linear system of equations $Ax = b$ represented by a suitable matrix $A \in \mathbb{R}^{M\times N}$ and right-hand side vector $b \in \mathbb{R}^{M\times 1}$.

To compute a discrete reconstruction, we need to choose a suitable approximation $\hat{\mu}$ of μ over the domain of interest. Typically, $\hat{\mu}$ is chosen as a finite linear combination of functions, i.e.,

$$\mu(x, y) \simeq \hat{\mu}(x, y) := \sum_{J=1}^{N} C_J \Phi_J(x, y), \tag{3.1}$$

where $\{\Phi_J\}_{J=1}^{N}$ is a set of basis functions, C_J is the density of the Jth pixel and $N := n_x \times n_y$. Common bases include the pixel basis of piecewise constant functions, the set of piecewise bilinear basis functions [30], and the set of Kaiser-Bessel blobs [32] which are radially symmetric and smooth but not of bounded support.

Using pixel basis functions is the simplest and most straightforward option for understanding how the discretization is performed. The pixel basis functions are defined as follows,

$$\Phi_J(x, y) = \begin{cases} 1 & \text{if } (x, y) \text{ lies inside pixel } J, \\ 0 & \text{if } (x, y) \text{ lies outside pixel } J \end{cases} \tag{3.2}$$

If we apply the Radon transform to both sides of (3.1), use the pixel basis functions as defined above (3.2) and using the linearity property of the Radon transform [30], we obtain

$$\mathcal{R}[\hat{\mu}](t, \theta) = \sum_{J=1}^{N} C_J \mathcal{R}[\Phi_J](t, \theta). \tag{3.3}$$

If we had a CT machine, we would be given all the values $\mathcal{R}[\hat{\mu}](t, \theta)$ for a finite set of lines $\ell_{(t,\theta)}$. We let $b_k = \mathcal{R}[\hat{\mu}](t_k, \theta_k)$ for $k = 1\colon K$ where K is a positive integer. Therefore, we can write

$$b_k = \sum_{J=1}^{N} C_J \mathcal{R}[\Phi_J](t_k, \theta_k) \qquad \text{for } k = 1\colon K. \tag{3.4}$$

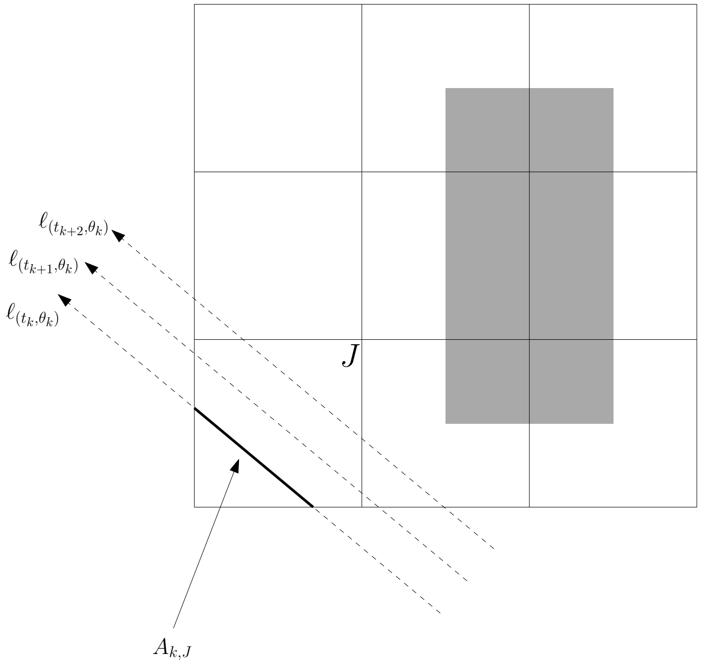


Figure 3.1: Matrix A is populated by all intersection values $A_{k,J}$.

Since we know that the pixel basis function Φ_J only has the value 1 on its pixel and 0 everywhere else, we can make the observation that the value of the integral $\mathcal{R}[\Phi_J](t_k, \theta_k)$ is equal to the length of the intersection of the line $\ell_{(t_k,\theta_k)}$ with pixel J. These intersection values (see Fig 3.1) are typically easy to compute.Assigning $A_{k,J}$ as the length of the intersection of $\ell_{(t_k,\theta_k)}$ with pixel J

$$A_{k,J} = \mathcal{R}[\Phi_J](t_k, \theta_k) \qquad \text{for } k = 1\colon K \text{ and } J = 1\colon N. \tag{3.5}$$

Finally, we can write (3.3) as

$$b_k = \sum_{J=1}^{N} C_J A_{k,J} \qquad \text{for } k = 1\colon K. \tag{3.6}$$

As not to confuse the reader, the notation will be changed as follows for the rest of the chapter:

- Matrix A will use the indices i and j.
- C will be replaced by the variable x.

Using this notation, (3.3) becomes

$$\sum_{j=1}^{N} A_{i,j} x_j = b_i, i = 1\colon M, \tag{3.7}$$

or the commonly seen standard system of equations,

$$A \cdot x = b. \tag{3.8}$$

Assuming that the system of equations expressed in (3.3) is square and invertible, matrix inversion is a possibility, however, not a practical one. Let A be $M \times N$ and if M and N are large, an image with dimensions 512×512 yields more than $256,000$ for N, and makes the matrix of $A_{i,j}$ values larger than $256,000 \times 256,000$. This eliminates most possibilities of using direct methods (e.g., Gaussian elimination, QR decomposition, etc.). Also, most systems of equations that arise in practice are often underdetermined (i.e. from missing data). This leads to the iterative method derived by Kaczmarz [29].

3.2 Kaczmarz's Method

Expanding the summation in equation (3.7),

$$\begin{aligned}A_{1,1}x_1 + A_{1,2}x_2 + \ldots + A_{1,N}x_N &= b_1\\ A_{2,1}x_1 + A_{2,2}x_2 + \ldots + A_{2,N}x_N &= b_2\\ &\vdots\\ A_{M,1}x_1 + A_{M,2}x_2 + \ldots + A_{M,N}x_N &= b_M\end{aligned} \tag{3.9}$$

Each equation in the system of equations above represents a hyperplane, $\mathbb{H} \in \mathbb{R}^N$. If there are N degrees of freedom when the grid has N cells, consider $(x_1\colon x_N)$ to be a single point in N dimensional space. The solution of the system of equations (3.9) constitutes the intersection of all the hyperplanes

$$\mathbb{H}_k := \{x \in \mathbb{R}^N | A_{k,:}x = b_k\} \qquad\qquad (k = 1 : M) \tag{3.10}$$

defined by each scalar equation. The set of solutions may have zero, one or infinitely many elements depending on the rank of the coefficient matrix and whether the right-hand side vector b lies in the range of A.

This is an example with two variables, to make the idea easier to understand (see Fig 3.2). The system of equations would be, where $M = 2$ and $N = 2$

$$\begin{aligned}A_{1,1}x_1 + A_{1,2}x_2 &= b_1\\ A_{2,1}x_1 + A_{2,2}x_2 &= b_2\end{aligned} \tag{3.11}$$

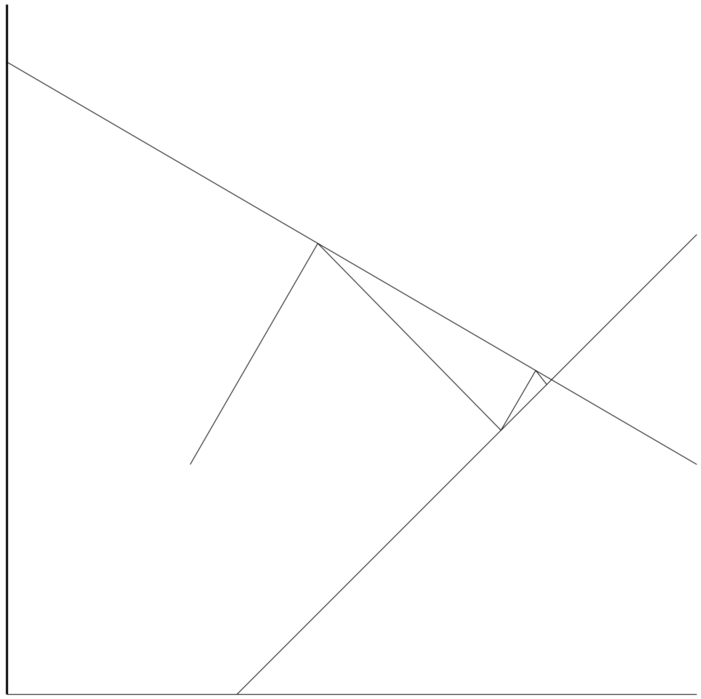


Figure 3.2: Kaczmarz's method converges to the point of intersection of two lines.

In other words, for $k = 1, 2$, a line $\mathbb{H}_R$ on the plane is defined by

$$A_{k,1}x_1 + A_{k,2}x_2 = p_k \tag{3.12}$$

To solve this system, the point of intersection of $\mathbb{H}_1$ and $\mathbb{H}_2$ must be found.

To form the basic Kaczmarz iteration, consider rewriting the system of equations (3.9) as,

$$A_{k,:} \cdot x = b_k, \qquad\qquad (k = 1 : M) \tag{3.13}$$

$A_{k,:}$ is the kth row of the matrix A. Thus, each pair $(A_{k,:}, b_k)$ defines a hyperplane in $\mathbb{R}^N$.

The Algebraic Reconstruction Technique (ART) or Kaczmarz's method is summarized in the following algorithm [30].

```
Data: $x^{(\nu)} \in \mathbb{R}^N, A \in \mathbb{R}^{M \times N}, b \in \mathbb{R}^M$
$\nu \leftarrow 0$
while not converged do
    $z \leftarrow x^{(\nu)}$
    for $i = 1 : M$ do
        $z \leftarrow$ Projection of $z$ onto hyperplane $\mathbb{H}_i$
    end
    $\nu \leftarrow \nu + 1$
    $x^{(\nu)} \leftarrow z$
end
```

Algorithm 1: Kaczmarz's Algorithm

It is important to note that the number of iterations needed to achieve a solution is dependent on the angle between successive hyperplanes. For example, consider the case where the hyperplanes are orthogonal and any initial guess will allow the solution to be reached in one step. As the angle between the hyperplanes becomes smaller, more and more iterations are needed to reach a solution. There have been a few schemes to improve convergence speed. The first proposed by Ramakrishnan et al. [42], consists of a pairwise orthogonalization scheme. However, another technique proposed by Hounsfield [25] simply consists of choosing the order of hyperplanes in a manner to optimize convergence, ideally choosing hyperplanes that are not adjacent to each other, and more likely to be parallel.

3.2.1 Derivation of the Kaczmarz algorithm

Computationally, if one wishes to use this algorithm, an actual iterative formula is needed. Let g be a vector and let $g_i = g - \alpha A_{i,:}$. The following facts are known,

1. $A_{i,:}$ is orthogonal to the hyperplane $A_{i,:}\, x = b_i$

2. The vector g_i is the orthogonal projection of the vector g onto the hyperplane$A_{i,:}\, x = b_i$.

Substituting g_i into (3.13),

$$g_i \cdot A_{i,:} = b_i = g \cdot A_{i,:} - \alpha A_{i,:} \cdot A_{i,:} \tag{3.14}$$

Isolating for α,

$$\alpha A_{i,:} \cdot A_{i,:} = b_i - g \cdot A_{i,:}$$

$$\alpha = \frac{g \cdot A_{i,:} - b_i}{A_{i,:} \cdot A_{i,:}}$$

Deriving an explicit form of the algorithm will produce the following equation [30],

$$x^{(\nu+1)} = x^{(\nu)} - \lambda \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{||A_{i,:}||^2} A_{i,:}^T \tag{3.15}$$

where λ is a relaxation parameter, ν is the current iteration and $i = 1\colon M$. One iteration is considered to be complete when we have gone through $i = 1\colon M$.

3.3 Simultaneous Algebraic Reconstruction Technique - SART

ART was, historically, the first iterative technique used in the field of CT [21]. The simultaneous Algebraic Reconstruction Technique was proposed as a refinement of

ART [30], and we will be testing the algorithm to ascertain if it is indeed superior to ART in terms of accuracy and convergence speed in chapter 4.

The idea behind SART was to try to achieve a reduction in background noise by updating the contributions of all rays for a specific projection simultaneously. This method considers a subset of our ray sums relating to a particular angle [3]. As we progress through the algorithm, the estimation of our reconstruction is updated by the back-distribution of the forward projection error along a series of rays for a single angle [3].

To relate this to the standard system of equations that has been used throughout this paper,

$$A \cdot x = b,$$

where the matrix A has size $M \times N$, $b = (b_1 \colon b_M)^T \in \mathbb{R}^M$ and represents the measured data, and the reconstruction $x = (x_1 \colon x_N)^T \in \mathbb{R}^N$. To help us express the algorithm for SART, we define

$$A_{i,+} = \sum_{j=1}^{N} A_{i,j} \text{ for } i = 1 \colon M, \quad \text{and} \tag{3.16}$$

$$A_{+,j} = \sum_{i=1}^{M} A_{i,j} \text{ for } j = 1 \colon N. \tag{3.17}$$

The SART algorithm [3] can be expressed as follows,

$$x_j{}^{(\nu+1)} = x_j{}^{(\nu)} + \frac{\lambda}{A_{+,j}} \sum_{i=1}^{M} \frac{A_{i,j}}{A_{i,+}} (b_i - A_{i,:} \cdot (x^{(\nu)})) \tag{3.18}$$

for $j = 1\colon N$ and $\nu = 0, 1, 2, \ldots$. The following assumptions are also made,

- $A_{i,j} \geq 0,$ for $i = 1\colon M$ and $j = 1\colon N$.
- $A_{+,j} \neq 0$ and $A_{i,+} \neq 0$ for $i = 1\colon M$ and $j = 1\colon N$.

In contrast to Kaczmarz's method (ART) that updates the solution with each projection, SART applies a single correction to the solution only after computing all the projections of the current solution onto the hyperplanes determined by the individual rays.

3.4 Cimmino's Method

Consider the system of linear algebraic equations, defined by

$$A \cdot x = b \tag{3.19}$$

where A is an $M \times N$ matrix and $b \in \mathbb{R}^M$.

Using the initial approximation $x^{(0)} \in \mathbb{R}^N$, Cimmino's method takes the mirror image or reflection $x^{i,(0)}$ of $x^{(0)}$ for $i = 1\colon N$ with respect to the hyperplanes (3.19) in the following manner [5, 12] :

$$x^{i,(0)} = x^{(0)} + 2\frac{b_i - A_{i,:} \cdot x^{(0)}}{\|A_{i,:}\|^2} A_{i,:} \tag{3.20}$$

In Cimmino's algorithm, we form the next iterate $x^{(1)}$ by computing the reflection points $x^{i,(0)}$ and computing the "centre of gravity" of those points assuming that point $x^{i,(0)}$ has "mass" $m_i \geq 0$ ($i = 1\colon M$).

Since the initial point $x^{(0)}$ and all the reflections, $x^{i,(0)}$, with respect to the M hyperplanes (3.19) reside on a hypersphere, whose center is solution of the linear system. The centre of gravity of the system of masses $\{m_i\}_{i=1}^{M}$ must fall within this hypersphere, therefore the iterate $x^{(1)}$ is a better approximation to the solution. Therefore, the next iterate $x^{(1)}$ approximates the solution better than $x^{(0)}$ since the center of gravity of $\{m_i\}_{i=1}^{M}$ must fall inside the hypersphere.

This procedure is now repeated with the new approximation $x^{(1)}$.

Cimmino's method can now be written, in matrix form, as :

$$x^{(\nu+1)} = x^{(\nu)} + \lambda A^T \widetilde{D}(b - A \cdot x^{(\nu)}) \tag{3.21}$$

where $\widetilde{D} \in \mathbb{R}^{M \times M}$ is set as:

$$\widetilde{D} = D^T D = \textsc{diag}(\frac{\sqrt{m_1}}{\|A_{1,:}\|}, \frac{\sqrt{m_2}}{\|A_{2,:}\|}, \ldots, \frac{\sqrt{m_M}}{\|A_{M,:}\|}) \tag{3.22}$$

and λ is a user chosen relaxation parameter. The quantities $m_1, m_2, \ldots, m_M$ are set to a system of user chosen weights $\{\omega_i\}_{i=1}^{M}$ for uniformity with other algorithms, where $\omega_i = \sqrt{m_i}$. Algebraically manipulating (3.21), the following is obtained:

$$x^{(\nu+1)} = x^{(\nu)} + \lambda \sum_{i=1}^{M} \omega_i \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{||A_{i,:}||^2} A_{i,:} \tag{3.23}$$

Setting all the ω_i's to $\frac{1}{M}$ the final representation of Cimmino's method becomes

$$x^{(\nu+1)} = x^{(\nu)} + \frac{\lambda}{M} \sum_{i=1}^{M} \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{||A_{i,:}||^2} A_{i,:}. \tag{3.24}$$

3.5 CAV

Component averaging is a method that projects the current iterate onto all the hyperplanes of the system. In comparison to Cimmino's method, which uses orthogonal projections and scalar weights, CAV uses oblique projections and diagonal weighting matrices.

Consider the following set of $N \times N$ diagonal matrices $\{G_i\}_{i=1}^{M}$ where $G_i = \textsc{diag}(g_{i1} \colon g_{iN})$ and $g_{ij} \geq 0 \ \forall \ i = 1\colon M$ and $j = 1\colon N$ such that $\sum_{i=1}^{M} G_i = I$.

The linear system

$$A \cdot x = b$$

can be represented by the M hyperplanes defined below

$$A_{i,:} \cdot x = b_i, \qquad i = 1\colon M. \tag{3.25}$$

The component averaging algorithm has three important features:

- Every single orthogonal projection onto the ith hyperplane defined at (3.25) is substituted by an oblique projection with respect to G_i.
- The set $\{w_i\}_{i=1}^{M}$ is replaced by the set $\{G_i\}_{i=1}^{M}$.
- The weights of $\{G_i\}_{i=1}^{M}$ are $G_i = \text{DIAG}(g_{i1}\colon g_{iN})$ and are inversely proportional to the number of nonzero elements in each column of matrix A.

Definition: The generalized oblique projection of a point $z \in \mathbb{R}^N$ onto the ith hyperplane as defined by (3.25) with respect to G_i is,

$$(P_i^G(z))_j = \begin{cases} z_j + \dfrac{b - A_{i,:} \cdot z}{\sum_{\substack{l=1 \\ g_{il} \neq 0}}^{N} A_{i,l}/g_{il}} \dfrac{A_{i,j}}{g_{ij}} & \text{if } g_{ij} \neq 0, j = 1, 2, 3, \ldots, N \\ z_j & \text{if } g_{ij} = 0, j = 1, 2, 3, \ldots, N. \end{cases} \tag{3.26}$$

Beginning from the matrix form of Cimmino's method (3.21) and using the relaxation parameter λ, we find

$$x^{(\nu+1)} = x^{(\nu)} + \lambda A^T D^T D(b - A \cdot x^{(\nu)}).$$

From the first two points above, the consequent substitution is made and one obtains

$$x^{(\nu+1)} = x^{(\nu)} + \lambda A^T G(b - A \cdot x^{(\nu)}).$$

With some algebraic manipulation and making the substitution using the definition (3.26), we obtain

$$x_j^{(\nu+1)} = x_j^{(\nu)} + \lambda \sum_{\substack{i=1 \\ g_{ij} \neq 0}}^{M} \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{\sum_{\substack{l=1 \\ g_{il} \neq 0}}^{N} A_{i,l}^2 / g_{il}} \cdot A_{i,j}. \tag{3.27}$$

Lastly, defining

$$g_{ij} = \begin{cases} \dfrac{1}{s_j} & \text{if } A_{i,j} \neq 0, \\ 0 & \text{if } A_{i,j} = 0, \end{cases} \tag{3.28}$$

and substituting this definition into (3.27), the CAV algorithm presents itself:

$$x_j^{(\nu+1)} = x_j^{(\nu)} + \lambda \sum_{\substack{i=1 \\ g_{ij} \neq 0}}^{M} \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{\sum_{l=1}^{N} s_l A_{i,l}^2} \cdot A_{i,j}. \tag{3.29}$$

3.6 DROP

The method of Diagonally Relaxed Orthogonal Projections (DROP) begins from Cimmino's method (3.24),

$$x^{(\nu+1)} = x^{(\nu)} + \frac{\lambda}{M} \sum_{i=1}^{M} \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{\|A_{i,:}\|^2} A_{i,:}$$

How DROP differs from Cimmino's method is in using diagonal componentwise relaxation. Instead of using equal weights, in the case of Cimmino's method as $\omega_i = \frac{1}{M}$ for $i = 1\colon M$, allow the weights to depend on the index j of the approximate solution vector $x = \{x_1, x_2, \ldots, x_j, \ldots, x_N\}$. Described mathematically as

$$x_j^{(k+1)} = x_j^{(k)} + \lambda \sum_{i=1}^{M} \omega_{ij} \frac{b_i - A_{i,:} \cdot x^{(k)}}{\|A_{i,:}\|^2} A_{i,j}, \quad (j = 1 : N), \tag{3.30}$$

where $\{\omega_{ij}\}_{i=1}^{M}$ is a nonnegative systems of weights and $j = 1 : N$. This relaxation can be used to exploit the sparsity of the problem [11].

When A is sparse, only a small number of elements in the jth column of A are non-zero, $\{A_{1,j}\colon A_{M,j}\}$. Observe that, in (3.30), the sum of the contributions of $\{A_{1,j}\colon A_{M,j}\}$ are divided by the number M. This division impedes the efficient progress of the method. This observation led to the idea [11] of considering the replacement of the factor $\frac{1}{M}$ by a factor that is solely dependent on the number of nonzero items in the set $\{A_{1,j}\colon A_{M,j}\}$. This new factor is denoted as s_j for each $j = 1\colon N$ and represents the number of nonzero elements in the column j of A.

Substituting the factor s_j for M in (3.24):

$$x_j^{(\nu+1)} = x_j^{(\nu)} + \frac{\lambda}{s_j} \sum_{i=1}^{M} \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{\|A_{i,:}\|^2} A_{i,j} \qquad \text{for } j = 1\colon N \tag{3.31}$$

The assumption is made that all columns of A are nonzero, so for all j, $s_j \neq 0$. Generalizing (3.31) to a weighted case one obtains

$$x_j^{(\nu+1)} = x_j^{(\nu)} + \frac{\lambda}{s_j} \sum_{i=1}^{M} \omega_i \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{\|A_{i,:}\|^2} A_{i,j} \qquad \text{for } j = 1\colon N \tag{3.32}$$

Chapter 4

Numerical Results

4.1 Description of Computational Experiments

We examine several different examples to measure the performance of the algorithms studied in this thesis. All numerical experiments are carried out using the SNARK05 software package [23]. The performance of individual algorithms can depend on how the equations are processed. If the ordering is random, and sequential data vectors are close to parallel, this can reduce the speed at which we converge to a solution. The idea is to choose sequential data vectors to be as close to orthogonal as possible. The ordering proposed by Herman and Meyer [24] accomplishes this and is used in all of our experiments. I used the package SNARK05 to implement all reconstruction algorithms (SNARK05 provides an interface for experimenting with new user-defined algorithms).

For each example, we construct a mathematical phantom (i.e., a known function $\mu(x, y)$), we choose a scanning geometry for the rays, and we compute the associated line integrals (i.e. Radon transform $\mathcal{R}[\mu](t, \theta)$). We then use this sinogram data as the right-hand side vector b and we construct the reconstruction matrix A associated with the scanning geometry. This matrix A and vector b are the input to various algebraic reconstruction algorithms. After n iterations, the computed result $x^{(n)}$ is compared to the true image $\tilde{x}$ with the relative error. The relative error is computed by the formula

$$\frac{\|x^{(n)} - \breve{x}\|_1}{\|\breve{x}\|_1} \tag{4.1}$$

where $\| \cdot \|_1$ denotes the ℓ_1-norm, $x^{(n)}$ is the computed reconstruction after n iterations, and $\breve{x}$ is the original phantom picture. In practice, the relative error is not computable since the "original" or phantom picture is not known. The relative residual error would be a better metric, however, in SNARK05, due to the implementation of how data is accessed, it was prohibitively expensive and time consuming to compute.

In certain cases, noise is introduced. This is done by feeding each raysum to a Poisson random number generator whose mean is given by the value of the raysum. Poisson noise occurs when the discrete number of photons is small enough to allow statistical fluctuations in a measurement. The magnitude of the noise increases with the mean magnitude of the light. However, since the average magnitude of the signal increases more quickly than that of the noise, it is usually only a problem for low light intensities. For a more detailed description, please consult [23].

In all cases, the initial iterate $x^{(0)}$ is the zero vector. For algorithms using a weighting factor, $\omega_i = 1$. This choice was made to remain consistent with other examples

[11, 39, 31] in the literature. This choice also lets the convergence speed and accuracy of a specific algorithm depend solely on the relaxation parameter, λ.

ART, as well as all the other algorithms examined in this thesis (Cimmino, SART, CAV, DROP) are tested extensively. In trying to find an ideal relaxation parameter λ, each algorithm is run multiple times with distinct values of λ. For each algorithm, the relaxation parameter resulting in the minimal relative error between iterations 1 to 50 is chosen as the "optimal" value.

For all experiments, I limited the number of iterations to 50. In hospitals there is a fine balance between image resolution and the time is takes to obtain a reconstruction. The typically agreed upon middle ground is 512×512 pixels and between 5 to 8 minutes for a reconstruction [38]. Through experimentation, 50 iterations satisfied that criteria. Larger iteration counts were also tested (75,100,125,150), but the relative error was reduced by very minimal amounts varying between 10^{-4} to 10^{-8}. Please note that 50 iterations is already a very large number compared to the limited amount of experiments done in the literature on similarly scaled examples [11, 15]. For many optimal λ values, 50 iterations are not needed to obtain the minimal relative error.

As a point of reference, I present some comparable results from the literature (see Table 4.1). There are three papers that are included:

- Censor et al, 2008, abbreviated as CEN.
- Popa and Zdunek, 2004, abbreviated as POP.
- Kostler et al, 2006, abbreviated as KOS.

Two other papers were examined in depth (Bautu et al, 2006 and Duluman and Popa, 2006), however, the measure used was the residual norm and I will not present the results here. In Bautu et al, 2006, experiments were of sizes 8×8 to 20×20 pixels. In Duluman and Popa, 2006, the experiment was of size 256×256 pixels.

Source	**ALG**	Exp. Name	Size	Stop. Cycle	Rel. Error
CEN	ART	Head Phant. (NL)	63×63	10	0.2
CEN	CIM	Head Phant. (NL)	63×63	10	0.27
CEN	DROP	Head Phant. (NL)	63×63	10	0.28
CEN	CAV	Head Phant. (NL)	63×63	10	0.269
CEN	ART	Head Phant. (N)	63×63	10	0.3
CEN	CIM	Head Phant. (N)	63×63	7	0.3
CEN	DROP	Head Phant. (N)	63×63	6	0.3
CEN	CAV	Head Phant. (N)	63×63	6	0.3
CEN	ART	Mitochondrion (NL)	341×341	10	0.2
CEN	DROP	Mitochondrion (NL)	341×341	10	0.2
CEN	ART	Mitochondrion (N)	341×341	3	0.41
CEN	DROP	Mitochondrion (N)	341×341	3	0.41
KOS	ART	Head Phant. (NL)	24×24	10	0.6
KOS	AFMG	Head Phant. (NL)	24×24	10	0.3
POP	ART	Rock Struc. (N)	30×30	50	0.161
POP	ART	Rock Struc. (N)	30×30	1000	0.154
POP	CEG	Rock Struc. (N)	30×30	50	0.164
POP	CEG	Rock Struc. (N)	30×30	1000	0.154
POP	KERP	Rock Struc. (N)	30×30	50	0.16
POP	KERP	Rock Struc. (N)	30×30	1000	0.1535
POP	ART	Drawing (N)	12×12	1000	not given
POP	CEG	Drawing (N)	12×12	1000	not given
POP	KERP	Drawing (N)	12×12	1000	not given

Table 4.1: Summarized results from the literature.

Under the heading “Exp. Name”, (NL) refers to a noiseless experiment and (N) refers to a noisy experiment. The size of each experiment is given in pixels and the unfamiliar algorithm abbreviations are as follows:

- **AFMG**: Algebraic Full Multi-Grid [31].
- **CEG**: Censor, Eggermont, Gordon [39].
- **KERP**: Kaczmarz Extended with Relaxation Parameters [39].

4.1.1 SNARK05 Software Package

SNARK05 is a programming interface used for the creation and evaluation of reconstruction algorithms. There have been many previous releases dating back to 1970. SNARK05 was designed to be capable of the following [23]

- Using various modes of data collection such as different geometrical arrangements for X-ray sources and detectors.
- Easily creating mathematically designed phantoms that can realistically represent 2-D cross sections of real world phenomena.
- Customizable display modes.
- Using several routines for the statistical evaluation of reconstruction algorithms.

SNARK05 provides many advantages over its previous incarnations. They are

- It is implemented in C++ as opposed to Fortran.

- XML headers are used in the file structures of the projection data, phantoms and algorithms.

- Iterative algorithms are capable of performing reconstructions on using the blob [32] basis as well as the pixel basis.

- No restrictions on the size of data structures or phantoms. The only limitations are imposed by the hardware, compiler and operating system.

- It has graphical capabilities for inputting data as well as viewing results.

4.2 Algorithm Pseudocode

For convenience, we summarize below the key formulas defining each iterative algorithm used in our experiments.

ART

$$x^{(\nu+1)} = x^{(\nu)} - \lambda \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{||A_{i,:}||^2} A_{i,:}^T \tag{4.2}$$

SART

$$x_j{}^{(\nu+1)} = x_j{}^{(\nu)} + \frac{\lambda}{A_{+,j}} \sum_{i=1}^{M} \frac{A_{i,j}}{A_{i,+}} (b_i - A_{i,:} \cdot (x^{(\nu)})) \tag{4.3}$$

where $j = 1\colon N$.

Cimmino

$$x_j{}^{(\nu+1)} = x_j{}^{(\nu)} + \frac{\lambda}{M} \sum_{i=1}^{M} \frac{A_{i,j}}{||A_{i,:}||^2} (b_i - A_{i.:} \cdot x^{(\nu)}) \tag{4.4}$$

where $j = 1\colon N$.

CAV

$$x_j^{(\nu+1)} = x_j^{(\nu)} + \lambda \sum_{\substack{i=1 \\ g_{ij} \neq 0}}^{M} \frac{b_i - A_{i,:} \cdot x^{(\nu)}}{\sum_{l=1}^{N} s_l A_{i,l}^2} \cdot A_{i,j}. \tag{4.5}$$

where $j = 1\colon N$.

Fully Simultaneous DROP method

$$x_j{}^{(\nu+1)} = x_j{}^{(\nu)} + \frac{\lambda}{s_j} \sum_{i=1}^{M} w_i \frac{A_{i,j}}{||A_{i,:}||^2} (b_i - A_{i,:} \cdot x^{(\nu)}) \tag{4.6}$$

where $j = 1\colon N$.

4.3 Snark Head Phantom

4.3.1 Description and analysis of the phantom

The first example that we examine is the head phantom from SNARK05 [23]. Head phantoms are very common in the medical imaging literature and often serve as a benchmark [45]. This example is discretized to 511 × 511 pixels. The scanning geometry consists of 300 projections with each projection having 725 uniformly spaced rays, each projection taken at angles distributed uniformly between 0° and 358.8°. The weighting matrix therefore has dimensions $217,500 \times 261,121$. The experiments consists of sinogram data computed with and without noise. This results in an underdetermined system of equations for both options.

Table 4.2 lists the optimal λ and stopping cycles for the SNARK05 head phantom.

Algorithm	Noiseless	Relative Error	Noisy	Relative Error
ART	(0.01,32)	0.7066	(0.01,8)	0.7189
CIM	(29,50)	0.8218	(29,50)	0.8218
SART	(1.7,48)	0.7067	(1.5,16)	0.7185
CAV	(2.1,49)	0.7067	(1.9, 15)	0.7185
DROP	(1.9,50)	0.7069	(1.9,15)	0.7186

Table 4.2: Pairs of optimal relaxation parameter λ and iteration number with the relative error for the noiseless and noisy cases of the SNARK05 head phantom.

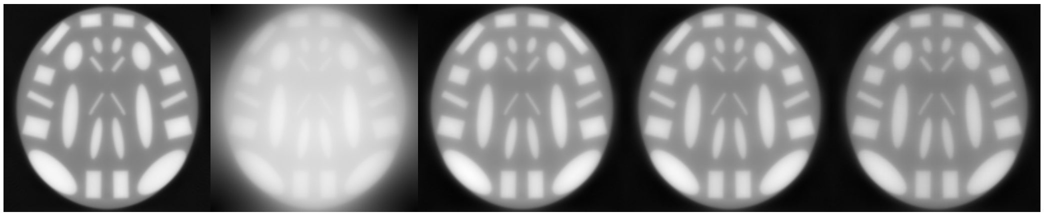
In order to to ascertain which algorithm is superior to which, I take into account two factors, the minimal relative error obtained and qualitative observation. There is a superiority with the ART algorithm compared to all others in the noiseless experiment (see Fig 4.1). Using the optimal relaxation parameter, we can see a faster convergence to the lowest relative error in figure 4.3 and in more detail in figure 4.4. None of the other algorithms compare. Also, observing figure 4.1, ART provides the least blurry reconstruction.

However, choosing the ideal algorithm in the noisy case isn't as obvious. We can make an argument for using ART, SART, CAV or DROP, as they all have their own positive aspects. If we look at the figure 4.2, we can observe that although ART gives us the sharpest reconstruction after 50 iterations, the densities of the shapes are not retained as accurately. SART, CAV and DROP give us a better reconstruction in terms of the density of the shapes. Figure 4.5 shows the convergence behaviour for ART in the noisy case for varying λ values.

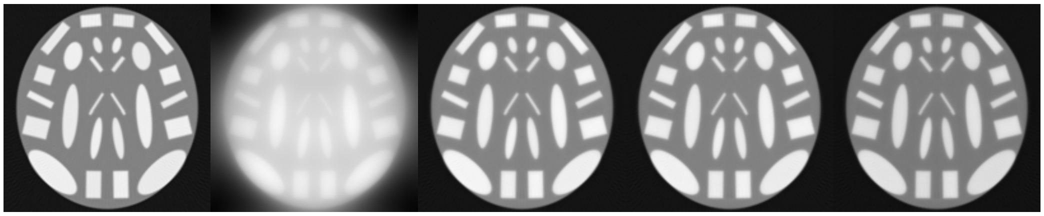
The oscillatory behaviour that can be seen in figure 4.3 and figure 4.4 I believe is due to the algorithm converging too quickly or too "far". This error then gets corrected in the following few iterations, but reappears cyclically as iterations continue. This phenomenon is due to λ being too large and $x^{\nu+1}$ is over-adjusted.

Snark Head Phantom

ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1



ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10

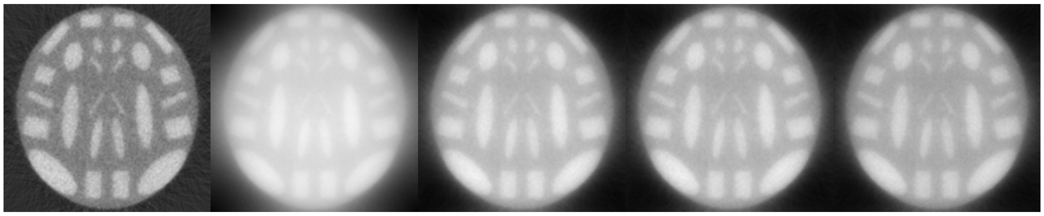


ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

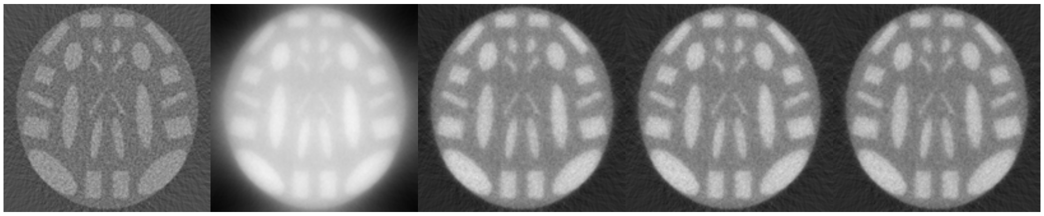
Figure 4.1: Reconstruction of the Noiseless 511 × 511 pixel Snark Head phantom using ART, CIM, SART, CAV and DROP algorithms

Snark Head Phantom

ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1



ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10



ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

Figure 4.2: Reconstruction of the Noisy 511 × 511 pixel Snark Head phantom using ART, CIM, SART, CAV and DROP algorithms

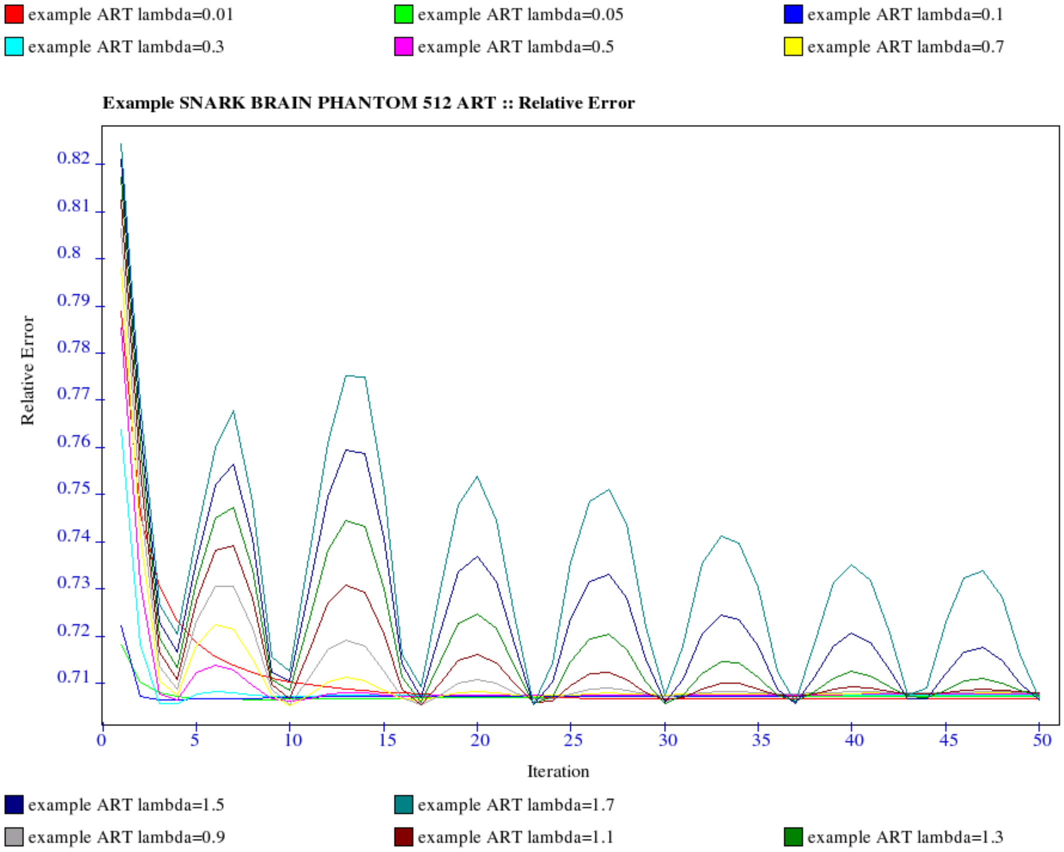


Figure 4.3: Plot of relative error against iteration count for ART with 511 pixels on the snark head phantom noiseless data for varying values of λ.

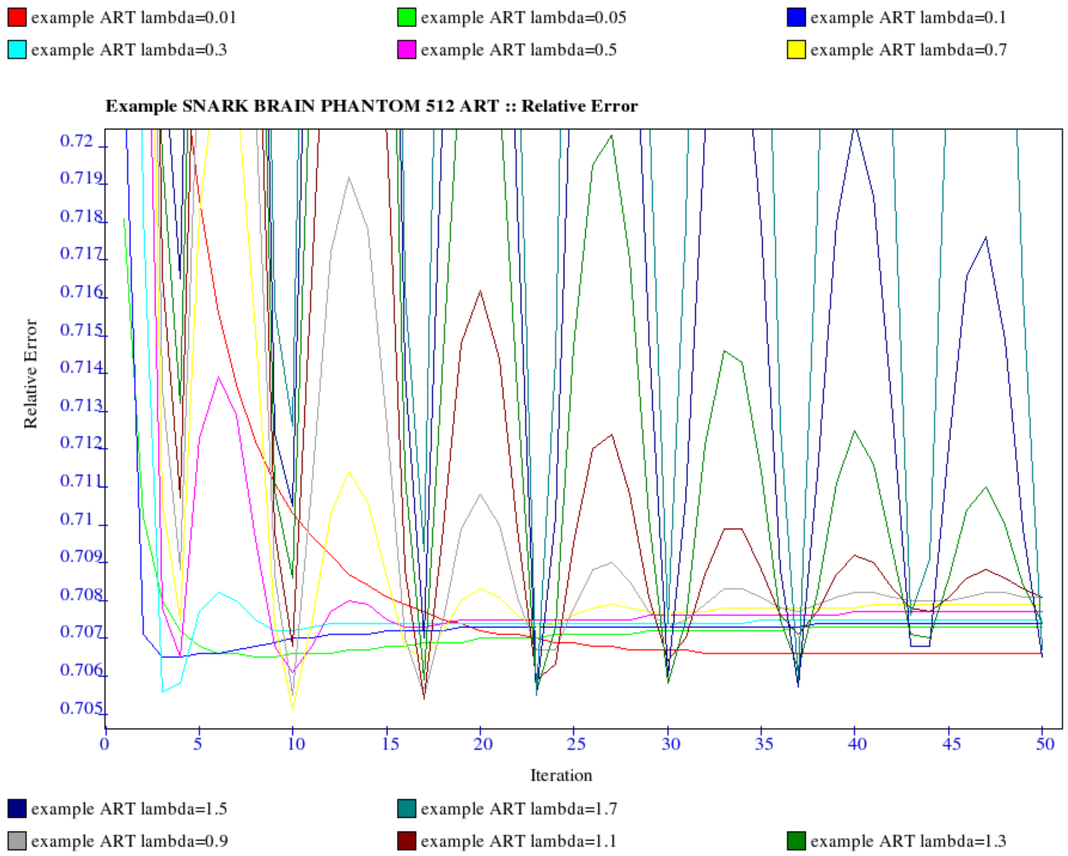


Figure 4.4: A closer view of the plot of relative error against iteration count for ART with 511 pixels on the snark head phantom noiseless data for varying values of λ.

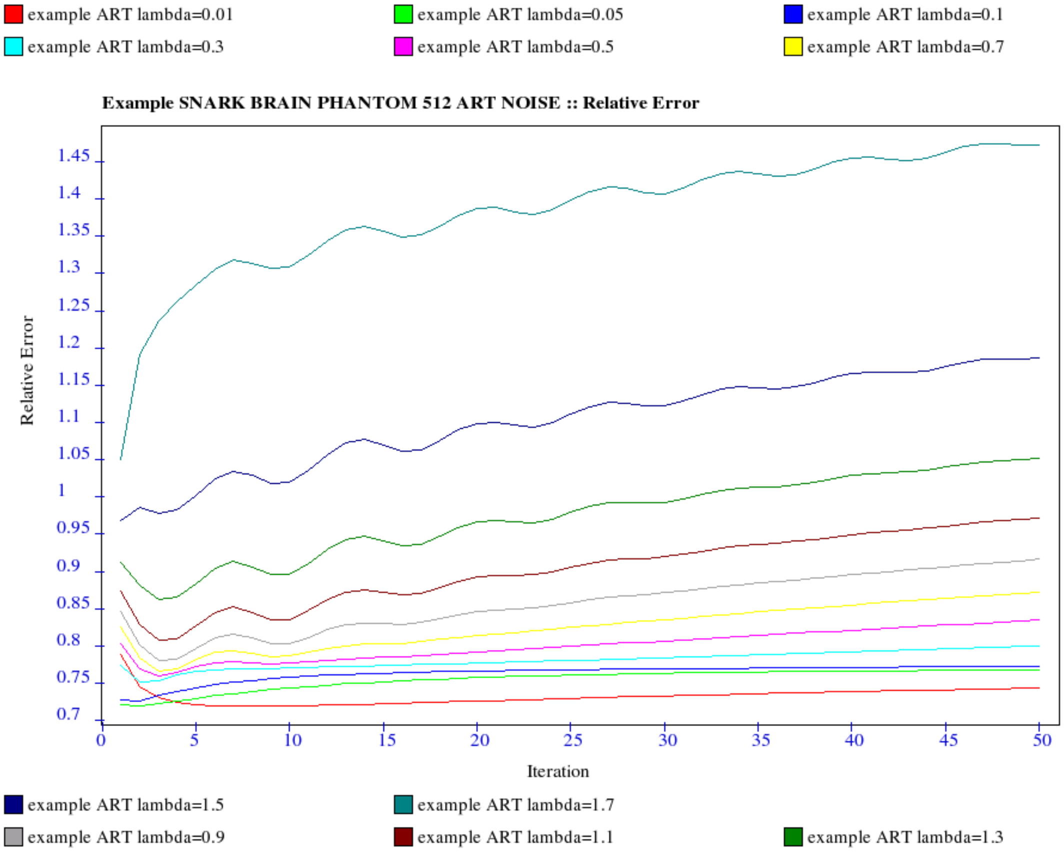


Figure 4.5: Plot of relative error against iteration count for ART with 511 pixels on the snark head phantom noisy data for varying values of λ.

4.4 Mitochondrion Phantom

4.4.1 Description and analysis of the phantom

The second example that we consider is the mitochondrion phantom [18, 11]. It simulates a mitochondrion with hollow cylinders for the membranes and solid cylinders for the cristae. In practice, to obtain the necessary data to perform a reconstruction on mitochondria, a high-voltage electron microscope (HVEM) would be used. Electron tomography then allows us to determine the internal structure from a set of projection HVEM images. These images are taken from different directions by adjusting how the sample is tilted. This usually follows one of two geometries: single-tilt axis [37] or double-tilt axis [33]. Structural analysis of specimens that require HVEMs are very complex and require high-resolution reconstructions (from 256×256 to 1024×1024) [18]. This was motivation to scale up this example to a size that researchers would find adequate.

This example is discretized to 511×511 pixels and the scanning geometry consists of projections of 725 uniformly spaced rays sampled at 72 equispaced angles between 0° and 140°. We also consider a second mitochondrion phantom example using 300 projections spaced evenly between 0° and 358.8°. These two examples contrast trying to reconstruct an image from complete and incomplete or corrupted data, where the second phantom with projections spaced between 0° and 358.8° is the complete data case.

First, for 72 projections, the weighting matrix A has dimensions $52,200 \times 261,121$. The data collected is split into two categories, with and without noise. Table 4.3 lists the optimal λ and stopping cycle pairs for the 72 projection case.

Algorithm (72 proj)	Noiseless	Relative Error	Noisy	Relative Error
ART	(0.1,9)	0.4587	(0.01,13)	0.5691
CIM	(29,50)	0.7757	(29,50)	0.775
SART	(0.7,50)	0.4547	(1.7,5)	0.5467
CAV	(0.9,50)	0.4594	(2.1, 5)	0.5501
DROP	(0.9,50)	0.4605	(1.9,5)	0.5581

Table 4.3: Pairs of optimal relaxation parameter λ and iteration number with the relative error for the noiseless and noisy cases of the mitochondrion head phantom.

Secondly, for 300 projections, the weighting matrix A has dimensions $217,500 \times 261,121$. The data collected is split into two categories, with and without noise.

Optimal pairs of relaxation parameters and stopping cycles are shown in Table 4.4.

Algorithm (300 proj)	Noiseless	Relative Error	Noisy	Relative Error
ART	(0.1,3)	0.364	(0.01,5)	0.4777
CIM	(29,50)	0.7563	(29,50)	0.7763
SART	(1.7,50)	0.3641	(1.7,7)	0.4592
CAV	(2.1,50)	0.3648	(2.1, 7)	0.4584
DROP	(1.9,50)	0.3656	(1.9,7)	0.471

Table 4.4: Pairs of optimal relaxation parameter λ and iteration number with the relative error for the noiseless and noisy cases of the mitochondrion head phantom.

Observing the noiseless case with 72 projections (figure 4.8), the appearance of the reconstruction is impacted by the fact that the projection angles range only from 0° to 140°. The ability to collect data from only this range is a limitation of single-tilt axis geometry [18, 37, 33].

Increasing the number of projections to 300 and spreading them around more completely ($0° - 358.8°$) gives a much better reconstruction (see fig 4.9). Please note that this particular experiment is theoretical and I do not have the familiarity with electron microscopy to validate how feasible it is to actually use a range of projections from ($0° - 358.8°$). The slight tilt in the noiseless case for 72 projections has disappeared and the densities are better represented. The noisy case is also much improved and SART, CAV and DROP outperform ART and Cimmino's algorithm in terms of minimal relative error and quality of the reconstruction.

Figure 4.6 shows us the convergence behaviour of the noiseless case for 300 projections. However, for the noisy case, if we look at figure 4.7, we can notice that after a certain number of iterations, the relative error begins to increase. A possible reason for this is the error in the right hand side of $A \cdot x = b$ compounded by the ill-conditioning of A. For a fixed error in b, the more ill-conditioned matrix A is, the faster the relative error will diverge (and vice versa). In this context, ill-conditioning refers to how close to parallel the hyperplanes are that make up the matrix. The more ill-conditioned a matrix is, the closer to parallel they are. This phenomenon is sometimes referred to as semi-convergence [11].

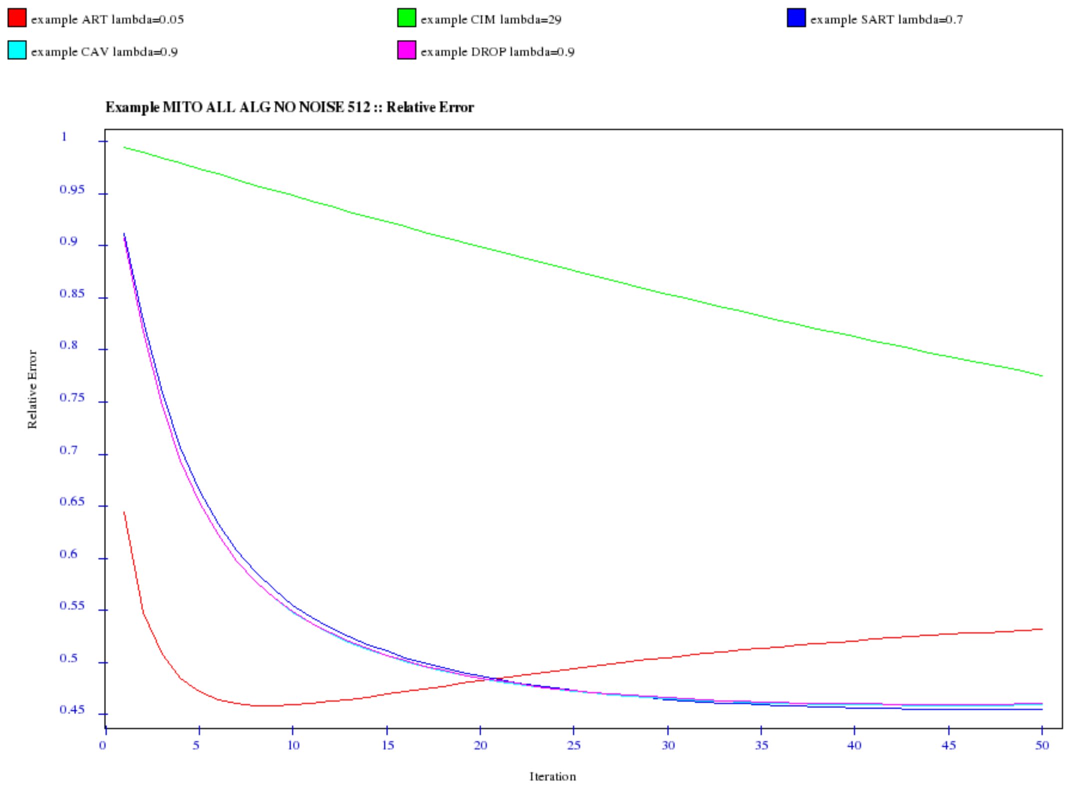


Figure 4.6: Plot of relative error against iteration count for ART, CIM, SART, CAV, DROP with 511 pixels on the mitochondrion phantom noiseless data for optimal values of λ.

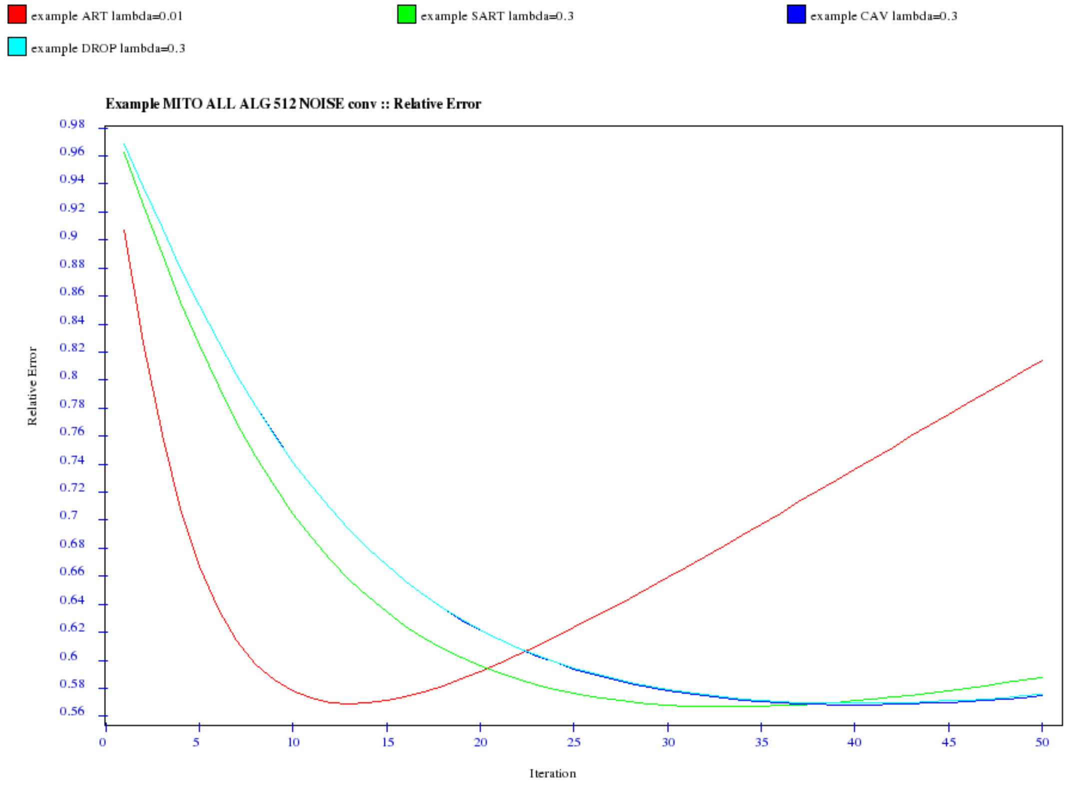
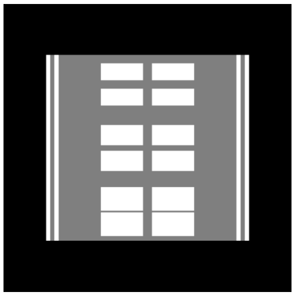
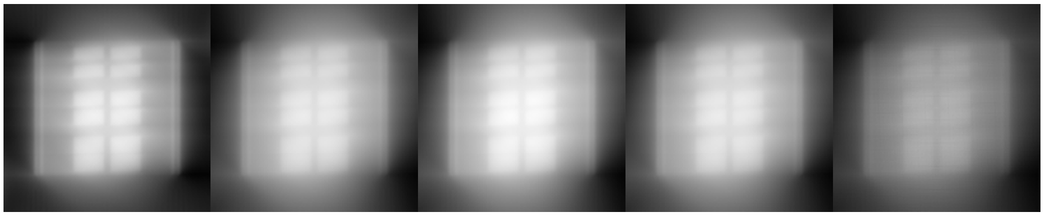


Figure 4.7: Plot of relative error against iteration count for ART, SART, CAV, DROP with 511 pixels on the mitochondrion phantom noisy data for optimal values of λ.

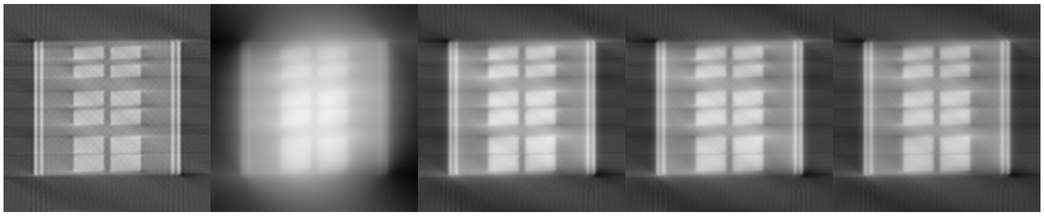


Mitochondrion phantom



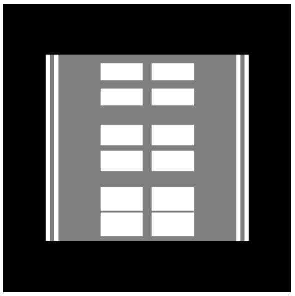
ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10

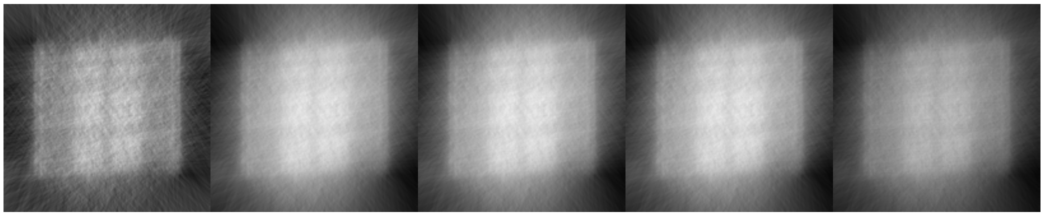


ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

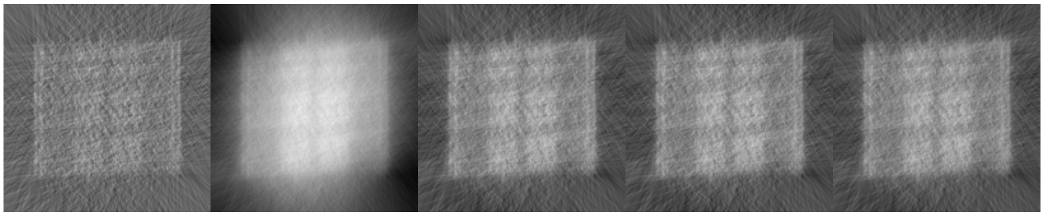
Figure 4.8: Reconstruction of the noiseless 511 × 511 pixel Mitochondrion phantom using ART, CIM, SART, CAV and DROP algorithms with 72 projections



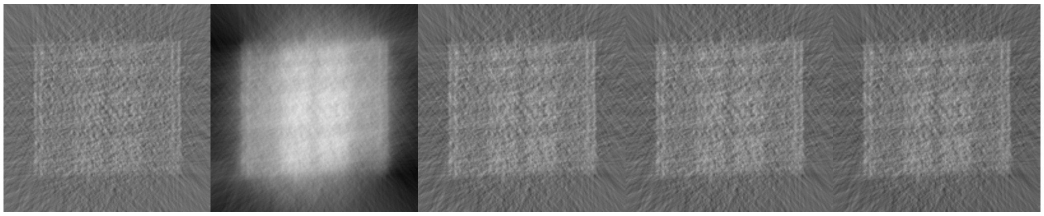
Mitochondrion phantom



ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

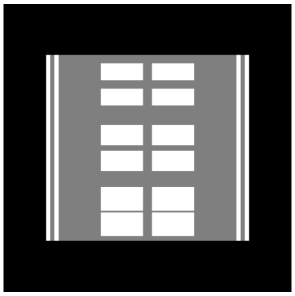


ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10



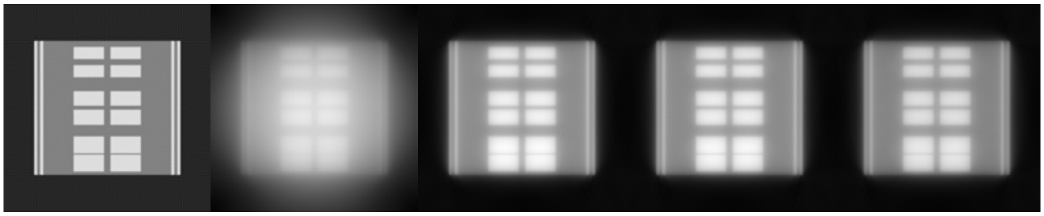
ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

Figure 4.9: Reconstruction of the noisy 511 × 511 pixel Mitochondrion phantom using ART, CIM, SART, CAV and DROP algorithms with 72 projections

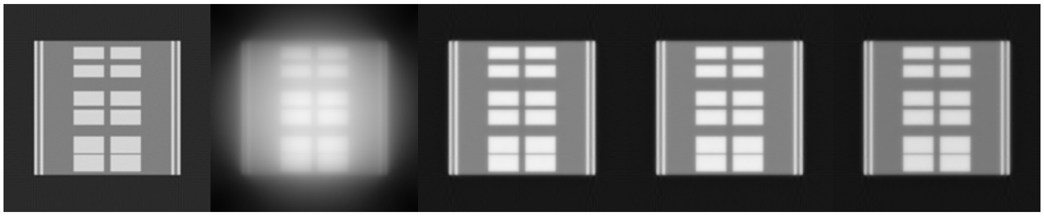


Mitochondrion phantom

ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

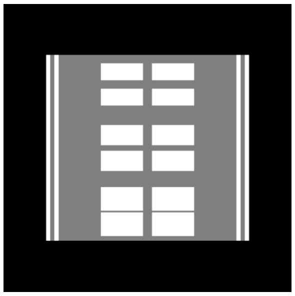


ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10

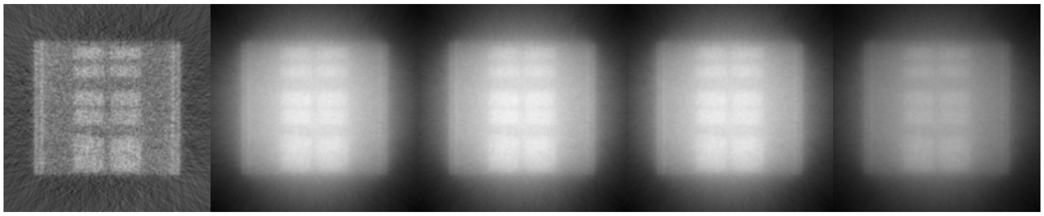


ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

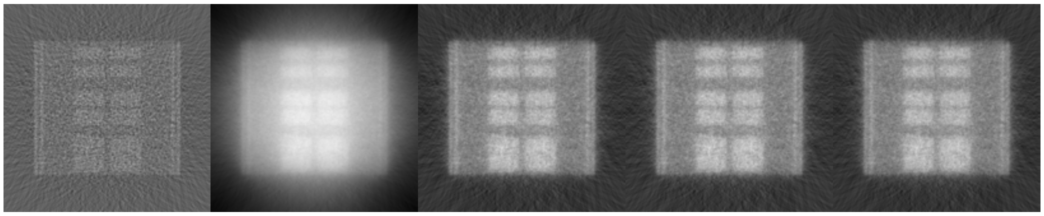
Figure 4.10: Reconstruction of the noiseless 511 × 511 pixel Mitochondrion phantom using ART, CIM, SART, CAV and DROP algorithms with 300 projections



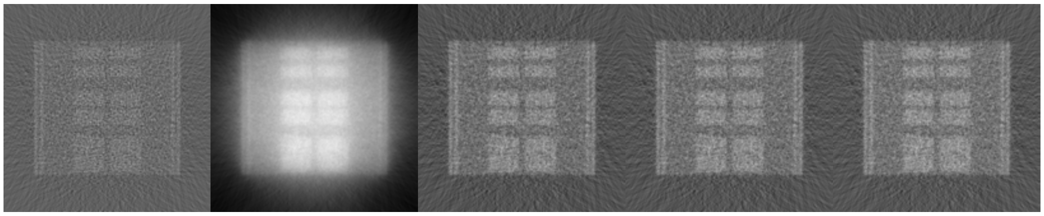
Mitochondrion phantom



ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1



ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10



ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

Figure 4.11: Reconstruction of the noisy 511 × 511 pixel Mitochondrion phantom using ART, CIM, SART, CAV and DROP algorithms with 300 projections

4.5 Circle Clock Phantom

4.5.1 Description and analysis of the phantom

The third example that we consider is the circle clock phantom, adapted from the sphere clock phantom in Dr. Henrik Turbell's Ph.D thesis [47]. The sphere clock phantom consists of two rings of spheres. One larger one placed outside at a certain pitch and a smaller one placed inside in the opposite direction. All spheres have the same density. The circle clock phantom can be thought of as a horizontal cross section of the sphere clock phantom. It comprises 12 individual circles placed on a circle, with 12 smaller circles placed inside, all of which have the same density. The individual circles in the inner ring have a radius half the size of the outer circles.

This example was created because I was interested if all the algebraic algorithms would be able to reconstruct this image due to using a parallel trajectory and a varying viewing angle. The sphere clock phantom itself was designed for use in testing helical cone-beam reconstruction algorithms [47].

This example is discretized on a grid of 511×511 pixels and uses 72 projections spaced evenly between 0° and 140° formed of 725 rays. We also consider variations using 32 projections and 300 projections.

First, for 72 projections, the weighting matrix A has dimensions $52,200 \times 261,121$. The data collected is split into two categories, with and without noise.

Algorithm (72 proj)	Noiseless	Relative Error	Noisy	Relative Error
ART	(1.3,44)	0.9989	(0.01,30)	1.0016
CIM	(0.4,1)	1.0	(0.4,1)	1
SART	(1.9,36)	0.9994	(0.1,1)	1.0002
CAV	(2.1,41)	0.9994	(0.1, 1)	1.0001
DROP	(1.9,45)	0.9994	(0.1,1)	1.0001

Table 4.5: Pairs of optimal relaxation parameter λ and iteration number with the relative error for the noiseless and noisy cases of the circle clock phantom.

Algorithm (32 proj)	Noiseless	Relative Error
ART	(1.3,25)	0.9995
CIM	(0.4,1)	1.0
SART	(1.9,32)	0.9998
CAV	(2.1,37)	0.9998
DROP	(1.9,42)	0.9998

Table 4.6: Pairs of optimal relaxation parameter λ and iteration number with the relative error for the noiseless case of the circle clock phantom.

Optimal pairs of relaxation parameters and stopping cycles are shown in Table 4.5.

Secondly, for 32 projections spaced evenly between 0 and 140°, the weighting matrix A has dimensions $23,200 \times 261,121$. The data collected is for the noiseless and noisy cases, but only the noiseless results will be shown here (Table 4.6). The results for the noisy case exhibit semi-convergence [11] as do the experiments with 72 and 300 projections.

Lastly, for 300 projections spaced evenly between 0 and 358.8°, the weighting matrix A has dimensions $217,500 \times 261,121$. The data collected was for the noiseless and noisy cases seen in Table 4.7.

Let us analyze first the noiseless cases. Comparing Figure 4.12 and 4.13, we see a clearer image with 72 projections, as would be expected, but the “tilting” is still visible in both due to the limited range of the projection angles. The background noise in both cases is also very prevalent. All algorithms apart from Cimmino’s seem to give a similar result for both cases respectively.

Results improve significantly upon observing the case with 300 projections. The tilting is gone as well as the background noise. Looking at the 10th iteration, the reconstruction is fairly clear, and only getting sharper as we move toward the 50th. SART, CAV and DROP all perform similarly, reducing the background noise and producing a sharp reconstruction, whereas ART does produce a sharp reconstruction, we get a gray layer throughout the background.

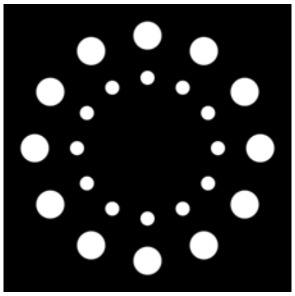
In the noisy case, all of the algorithms fail in providing a reconstruction. At first this was thought to be a bug, but extensive retesting was done to no avail, the results were the same. At this point in the research, the reason is not known, so further analysis is required. However, we could infer that the background noise is interfering and due to the uniform density of the circles, as the reconstruction progresses, the circles and noise become indistinguishable, giving us the result in Figure 4.15.

For all cases, the relative error is very high, not budging very much from 100%. Visually we can see the improvement of the reconstruction in the noiseless case, so using the relative error as defined in section 4.1, may not be the best measure overall. However, the relative error of the reconstructed image is used widely in the literature on iterative algebraic techniques in image reconstruction (for example, see [11, 31, 39]), and being able to compare the results obtained in this thesis to previously published results was the reasoning for using this measure of error.

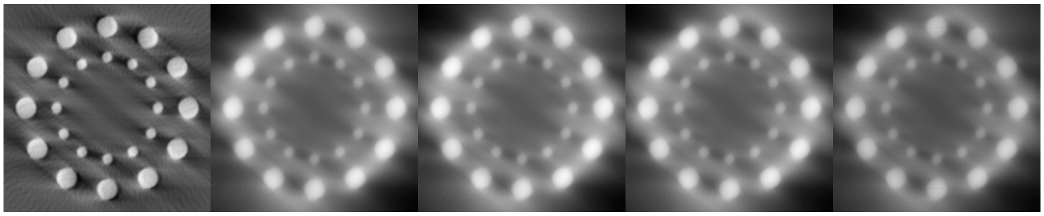
Algorithm (300 proj)	Noiseless	Relative Error	Noisy	Relative Error
ART	(0.01,30)	0.9971	(0.01,1)	1.00178
CIM	(0.4,1)	1.0	(0.4,1)	1
SART	(1.9,47)	0.9971	(0.1,1)	1.0003
CAV	(2.1,42)	0.9972	(0.1, 1)	1.0002
DROP	(1.9,46)	0.9972	(0.1,1)	1.002

Table 4.7: Pairs of optimal relaxation parameter λ and iteration number with the relative error for the noiseless and noisy cases of the circle clock phantom.

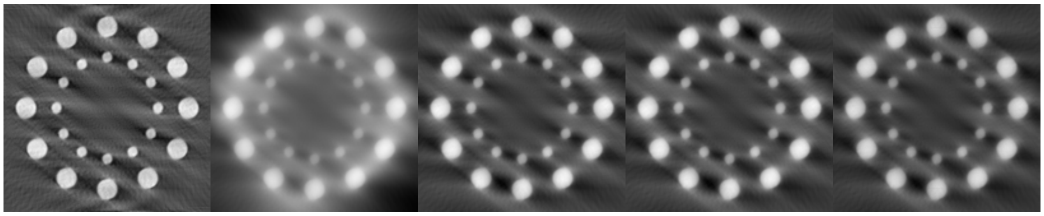
It is interesting that despite having a relative error of nearly 100%, there is a distinct improvement in the results. Since the relative error is a measure that is proportional, this could mean that the artifacts (white lines in the background) present in the cases with 32 and 72 projections may account for this behaviour. However, for the case with 300 projections, there no longer are any visible artifacts, and more extensive testing would need to be done to find out why the relative error is higher than expected.



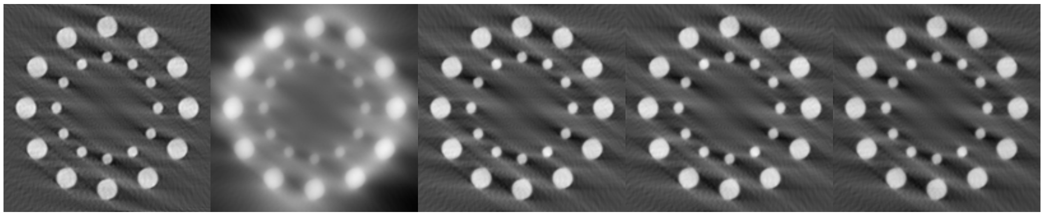
Circle clock phantom



ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

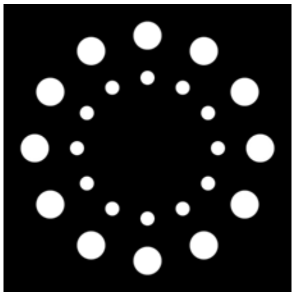


ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10

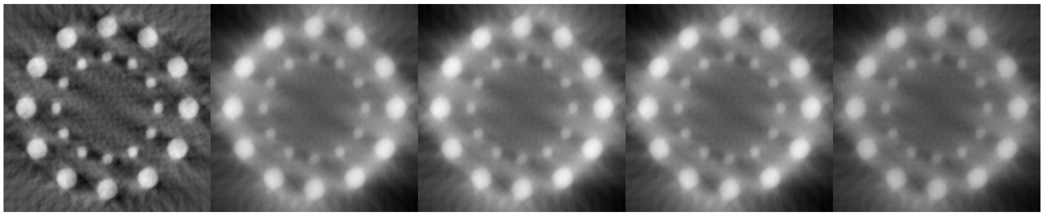


ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

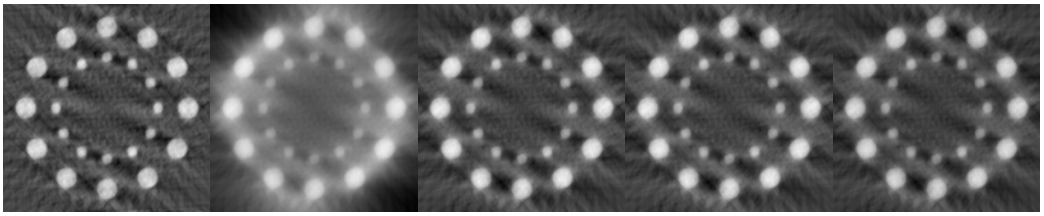
Figure 4.12: Reconstruction of the noiseless 511 × 511 pixel circle clock phantom using ART, CIM, SART, CAV and DROP algorithms with 72 projections



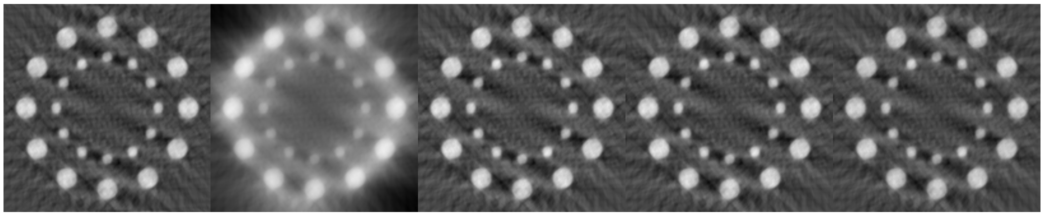
Circle clock phantom



ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

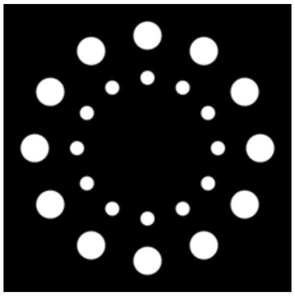


ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10

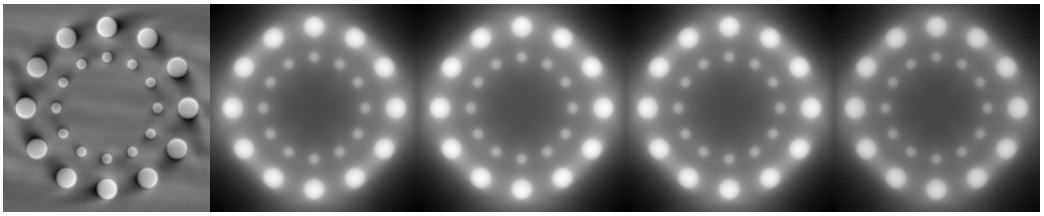


ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

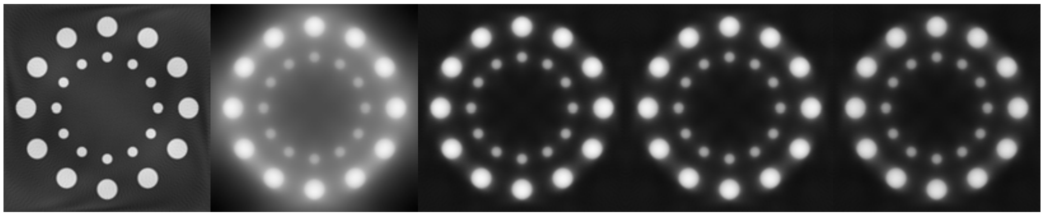
Figure 4.13: Reconstruction of the noiseless 511×511 pixel circle clock phantom using ART, CIM, SART, CAV and DROP algorithms with 32 projections



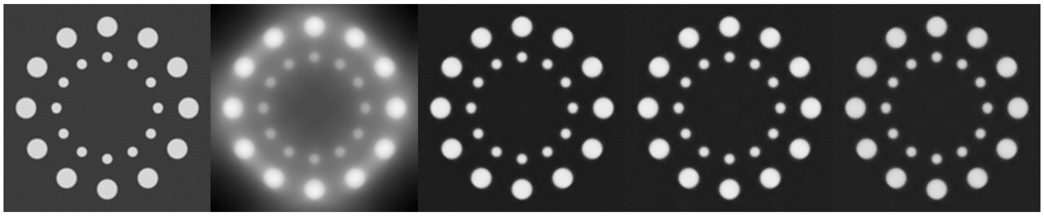
Circle clock phantom



ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

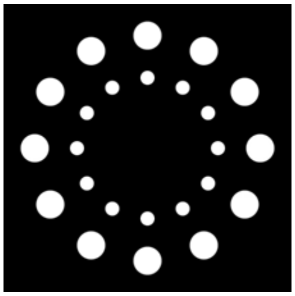


ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10



ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

Figure 4.14: Reconstruction of the noiseless 511×511 pixel circle clock phantom using ART, CIM, SART, CAV and DROP algorithms with 300 projections



Circle clock phantom

ART iter 1 CIM iter 1 SART iter 1 CAV iter 1 DROP iter 1

ART iter 10 CIM iter 10 SART iter 10 CAV iter 10 DROP iter 10

ART iter 50 CIM iter 50 SART iter 50 CAV iter 50 DROP iter 50

Figure 4.15: Reconstruction of the noisy 511 × 511 pixel circle clock phantom using ART, CIM, SART, CAV and DROP algorithms with 300 projections

4.5.2 Comparison with Filtered Backprojection

The following is a comparison between FBP and the iterative algorithms seen in this thesis. All experiments with filtered backprojection were done using a bandlimiting filter. The bandlimiting filter $F(R)$ is defined as follows

$$\Pi(x) = \begin{cases} 1, & \text{if } x \leq \frac{1}{2}, \\ 0, & \text{if } x > \frac{1}{2}, \end{cases} \tag{4.7}$$

and

$$F(R) = \Pi\left(\frac{R}{2C}\right) \times |R|, \tag{4.8}$$

where C is a cutoff value. Tables 4.8 and 4.9 show the relative error in the noiseless and noisy case for filtered backprojection and all iterative reconstruction algorithms.

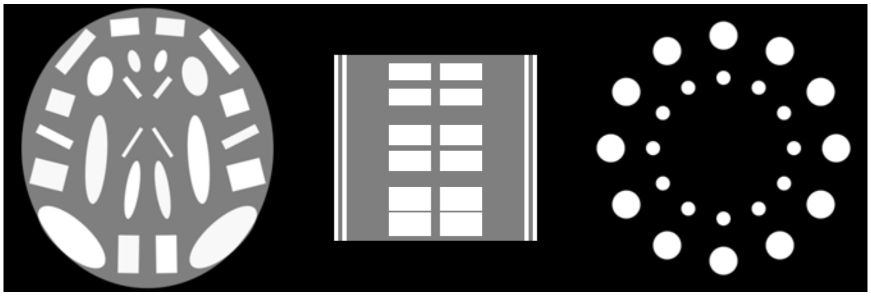
Experiment (noiseless)	FBP	ART	CIM	SART	CAV	DROP
Snark head	0.0293	0.7066	0.8218	0.7067	0.7067	0.7069
Mitochondrion	0.0657	0.364	0.7563	0.3641	0.3648	3656
Circle clock	0.1902	0.9971	1.0	0.9971	0.9972	0.9972

Table 4.8: Relative error measurements comparing FBP and iterative algebraic techniques in the noiseless case

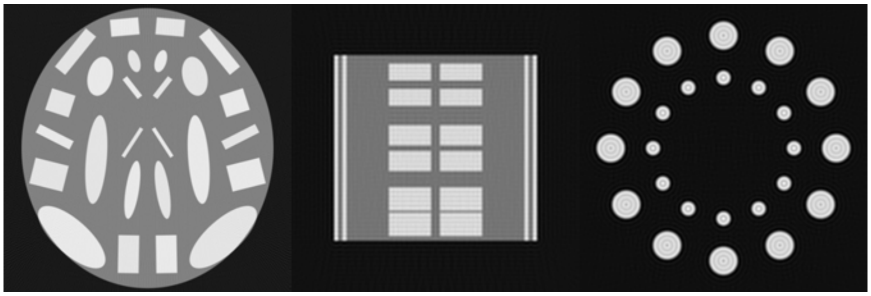
Experiment (noisy)	FBP	ART	CIM	SART	CAV	DROP
Snark head	0.1418	0.7189	0.8218	0.7185	0.7185	0.7186
Mitochondrion	0.2522	0.4777	0.7763	0.4592	0.4584	0.471
Circle clock	1.0	1.00178	1.0	1.0003	1.0002	1.002

Table 4.9: Relative error measurements comparing FBP and iterative algebraic techniques in the noisy case

Figure 4.16 shows the reconstruction on all three phantoms using filtered backprojection in both the noiseless and noisy cases. In the noiseless cases, FBP reconstructs the images almost perfectly. However, in the noisy case, the results aren't as good.



Snark head phantom Mitochondrion phantom Circle clock phantom



FBP Noiseless FBP Noiseless FBP Noiseless

FBP Noisy FBP Noisy FBP Noisy

Figure 4.16: Reconstruction using filtered backprojection of the snark head phantom, mitochondrion phantom and the circle clock phantom with a resolution of 511 × 511 pixels.

The noisy snark head phantom reconstruction is the most accurate of all the algorithms seen. Visually the noisy mitochondrion phantom reconstruction isn't perfect, however, the relative error is quite low, and the density of the shapes is retained fairly well. As with the other experiments, the circle clock phantom still cannot be reconstructed. It is important to note that it takes only a single iteration to obtain each result with filtered backprojection.

4.6 Timings

The tables 4.10, 4.11, 4.12 show the mean run times for a single iteration. It was evaluated over 5 different runs of 50 iterations. The higher resolution experiments, especially those with 300 projections are very expensive computationally. All experiments were performed on a single cpu. The details of the machine used are as follows:

- Processor: AMD Opteron 254 with 2.8 GHz.
- Memory: 8 Gb RAM.
- OS: Red Hat Enterprise Linux ES release 4 (64 bit)

As mentioned before, I used the SNARK05 programming interface.

	Time (s) 1 it
ART	5.55
CIM	6.63
SART	7.05
CAV	7.13
DROP	6.99
FBP	4.01

Table 4.10: Timings for the Snark head phantom

	Projections	Resolution	Time (s) 1 it
ART	72	512 × 512	1.34
	300	64 × 64	0.09
		256 × 256	1.29
		512 × 512	5.48
CIM	72	512 × 512	1.63
	300	64 × 64	0.09
		256 × 256	1.41
		512 × 512	6.61
SART	72	512 × 512	1.71
	300	64 × 64	0.1
		256 × 256	1.5
		512 × 512	6.93
CAV	72	512 × 512	1.74
	300	64 × 64	0.1
		256 × 256	1.46
		512 × 512	7.03
DROP	72	512 × 512	1.7
	300	64 × 64	0.09
		256 × 256	1.39
		512 × 512	6.88
FBP	300	512 × 512	3.99

Table 4.11: Timings for the mitochondrion phantom

	Projections	Time (s) 1 it
ART	32	0.6
	72	1.29
	300	5.49
CIM	32	0.71
	72	1.57
	300	6.6
SART	32	0.75
	72	1.65
	300	6.93
CAV	32	0.77
	72	1.68
	300	7.02
DROP	32	0.76
	72	1.64
	300	6.88
FBP	300	3.97

Table 4.12: Timings for the circle clock phantom

4.6.1 Parallelization

All iterative algebraic reconstruction techniques seen in this thesis are candidates for parallelization. In fact, they all fall under the “perfectly parallelizable” [35] category. This means that the algorithms can obtain linear speed-up when they are parallelized and run with p processors [35].

Consider the DROP algorithm

$$x_j^{(\nu+1)} = x_j^{(\nu)} + \frac{\lambda}{s_j} \sum_{i=1}^{M} w_i \frac{A_{i,j}}{||A_{i,:}||^2}(b_i - A_{i,:} \cdot x^{(\nu)}) \tag{4.9}$$

where $j = 1 \colon N$. One way to parallelize this algorithm would be to send each x_j of $x_j^{(\nu+1)}$ to be calculated on a separate processor. Note that there is overhead for sending data to and from processors, so experimentation would be required to find an optimal parallelization scheme.

Chapter 5

Summary

In this thesis, we have examined the evolution of computerized tomography and studied five different algebraic reconstruction techniques. In addition, we have tried to extensively test all algorithms through many different cases in three main experiments.

In chapter 3, we looked at the algebraic approach to X-ray CT and reviewed the various reconstruction techniques and how they were derived. These methods were

1. Kaczmarz's Method, otherwise known as the algebraic reconstruction technique (ART)
2. Cimmino's Method (CIM)
3. Simultaneous algebraic reconstruction technique (SART)
4. Component averaging (CAV)
5. Diagonally relaxed orthogonal projections (DROP).

In chapter 4, we described our three main experiments, the SNARK05 Head phantom, the mitochondrion phantom and the circle clock phantom. Many different test cases were used, including variations in the number of projections and the angular intervals. The main criterion for the evaluation of the reconstructions was the relative error as defined in section 4.1. We notice, especially in the noisy cases, that the relative error increases after a certain number of iterations; a phenomenon known as semi-convergence [11]. This emphasizes the need to be able to identify ideal stopping conditions for algebraic methods.

Our best results, in terms of relative error, were with the mitochondrion phantom, as the relative error was quite high for both the SNARK05 head phantom and the circle clock phantom. Qualitatively, we can definitely see an improvement in the latter two experiments, however, the actual densities as the reconstruction progresses may be diverging from their intended values. The most suprising result was the failure of all algorithms to be able to reconstruct the noisy circle clock phantom. The exact cause of this failure is unknown at this point and further investigation would be required.

Let us look at some of the key results for each algorithm.

ART

- Generally sharper reconstruction than other algorithms.
- Loss of accuracy of the density of the reconstructed shapes.
- Least computationally demanding algorithm.
- Not accurate enough for practical use.

CIM

- Fundamental building block for later algorithms.
- Fails in reconstructing images accurately.
- Not viable for use.

SART, CAV, DROP

- Adequate accuracy of reconstruction, on par with each other.
- Image reconstruction slightly more blurry than ART.
- More computationally intensive than ART.
- Not accurate enough for practical use.

ART, SART, CAV and DROP have promising results, especially with the mitochondrion phantom. They are however, not accurate enough to be put into practical use yet. ART is the least computationally demanding of the four, as seen by observing Tables 4.10, 4.11 and 4.12. If Moore's Law is sustained, the issue of computing speed between ART and SART/CAV/DROP would be negligible in my opinion. Already, all algorithms are candidates for parallel computing, which would greatly reduce runtimes and with the advent of cloud computing, efficiency would only be increased.

In comparison to the literature (see Table 4.1), all experiments performed had a higher relative error. The most significant of which is the mitochondrion phantom tested in [11]. The reconstruction performed in that paper was of size 341 × 341 pixels and my findings were that as we increased to 512 × 512 pixels, the relative error increased significantly. Larger test cases would be needed to see if increasing the resolution of the reconstruction increases the relative error proportionally. In terms of the other cases from the literature, it is difficult to make a comparison, as some are very small and it becomes difficult to compare results of a reconstruction from a matrix with 900 (30×30 pixels) rows and one with over $200,000$ (512×512 pixels) rows.

Some of the questions and issues that I would have liked to follow up on are

1. Including and testing block-iterative versions of all applicable algorithms for varying block sizes. Block-iterative algorithms use a “block” of equations at each step of an iteration rather than all equations or simply a single equation. Many algorithms have block-iterative versions, some of which are ART, CAV, and DROP [11, 9, 10].
2. Scale all experiments to larger sizes (e.g. 1024×1024 pixels).
3. Why did all algorithms fail on the reconstruction of the noisy circle clock phantom? How can we identify phantom examples that may fail to be reconstructed?
4. Is there a way to reduce and/or eliminate the semi-convergence phenomenon?
5. Test other measures of accuracy such as distance or residual norm.

Algebraic reconstruction techniques are not currently used in commercial X-ray CT systems in favor of algorithms based on filtered backprojection [36, 34]. This is due to varying factors, mainly that manufacturers have little motivation to implement new algorithms when there are other practical issues that can be dealt with [36, 34], such as detector arrays, display and analysis of data sets and mechanical detector-source rotation speed.

Radiation exposure should be limited to a minimum to avoid damaging biological effects [14, 43, 48, 44]. The problem lies in the fact that, for a fixed signal to noise ratio in an X-ray CT image voxel (3-D pixel), the X-ray exposure must increase by a factor that is at least the cube of the resolution. For example, to increase resolution from 1mm^3 to 0.5mm^3, the exposure would increase at least 8 fold [22].

The strengths of algebraic techniques lie in being able to perform reconstruction with missing data, or for a particular angular interval. This is in contrast to FBP, which can provide a theoretically perfect reconstruction, however, the data gathered will always have noise, which leads to errors in the reconstruction. FBP also needs the complete set of data to perform its reconstruction correctly. Iterative algebraic techniques refine their solution at each step, and can proceed with a reconstruction regardless of how much data is lost or missing. Harnessing algebraic techniques properly could translate to lower radiation doses for patients and less costs to hospitals and manufacturers.

Bibliography

[1] Nobelprize.org, "The Nobel Prize in Physiology or Medicine 1979". `http://nobelprize.org/nobel_prizes/medicine/laureates/1979/`.

[2] P. J. Allisy-Roberts and J. Williams. *Farr's Physics for Medical Imaging.* Saunders, 2007.

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

[4] A. Bautu, E. Bautu, and C. Popa. Hybrid algorithms in image reconstruction. *Proc. Appl. Math. Mech.*, 6:707–708, 2006.

[5] M. Benzi. Gianfranco Cimmino's contributions to numerical mathematics. *Ciclo di Conferenze in Ricordo di Gianfranco Cimmino*, pages 87–109, 2005.

[6] G. Beylkin. Discrete Radon transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35 (2), 1987.

[7] R. N. Bracewell. *The Fourier Transform and Its Applications.* McGraw-Hill, 1986.

[8] W. L. Briggs and V. E. Henson. *The DFT: An Owner's Manual for the Discrete Fourier Transform.* SIAM, 1995.

[9] C. Byrne. Convergent block-iterative algorithms for image reconstruction from inconsistent data. *IEEE Transactions on Image Processing*, 6:1296–1304, 1997.

[10] C. Byrne. Block-iterative algorithms. *International Transactions in Operational Research*, 16:427–463, 2008.

[11] Y. Censor, T. Elfving, G. T. Herman, and T. Nikazad. On diagonally relaxed orthogonal projection methods. *SIAM J. Sci. Comput.*, 30:473–504, 2008.

[12] G. Cimmino. Calcolo approssimato per le soluzioni dei sistemi di equazioni lineari. *La Ricerca Scientifica*, II,9:326–333, 1938.

[13] R.A. Crowther, D.J. DeRosier, and A. Klug. The reconstruction of a three-dimensional structure from its projections and its application to electron microscopy. *Proc R. Soc London*, A317:319–340, 1970.

[14] A. Berrington de Gonzalez, M. Mahesh, K. Kim, M. Bhargavan, R. Lewis, F. Mettler, and C.Land. Projected cancer risks from computed tomographic scans performed in the United States in 2007. *Arch Intern Med*, 169 (22), 2009.

[15] T. Duluman and C. Popa. Algebraic reconstruction technique versus conjugate gradient in image reconstruction from projections. *Mathematical Modelling of Environmental and Life Sciences Problems*, pages 67–78, 2006.

[16] C. L. Epstein. *Introduction to the Mathematics of Medical Imaging.* SIAM, second edition, 2008.

[17] T.G. Feeman. *The Mathematics of Medical Imaging.* Sringer, first edition, 2010.

[18] J.-J Fernndez, A.F. Lawrence, J. Roca, I. Garcia, M.H. Ellisman, and J.-M Carazo. High-performance electron tomography of complex biological specimens. *J. Structural Biol.*, 138:6–20, 2002.

[19] B.P. Flannery, H.W. Deckman, W.G. Roberge, and K.L. D'Amico. Three dimensional X-ray microtomography. *Science*, 237:1439–1444, 1987.

[20] D. Gordon and R. Gordon. Component-averaged row projections: A robust block-parallel scheme for sparse linear systems. *SIAM J. Sci. Comput.*, 27:1092–1117, 2005.

[21] R. Gordon, R. Bender, and G.T. Herman. Algebraic reconstruction technique (ART) for three dimensional electron microscopy and x-ray photography. *Journal of Theoretical Biology*, 29:471–482, 1970.

[22] L. Grodzins. Optimal energies for X-ray transverse tomography of small samples. *Nucl. Instrum. Methods*, 206:541–545, 1983.

[23] G.T. Herman. A Programming System for Image Reconstruction. *Technical Report*, 2005.

[24] G.T. Herman and L.B. Meyer. Algebraic reconstruction technique can be made computationally efficient. *IEEE Trans. Medical Imaging*, 12:600–609, 1993.

[25] G. N. Hounsfield. Computerized transverse axial scanning (tomography): Part 1. description of system. *Br. J. Radiol.*, 46:1016–1022, 1973.

[26] G. N. Hounsfield and J. Ambrose. Computerized transverse axial scanning (tomography): Part 2. clinical application. *Br. J. Radiol.*, 46:1023–1047, 1973.

[27] J.F. James. *A student's guide to Fourier transforms: with applications in physics and engineering*. Cambridge University Press, 2002.

[28] M. Jiang and G. Wang. Convergence studies on iterative algorithms for image reconstruction. *IEEE Transactions on Medical Imaging*, 22, 2003.

[29] S. Kaczmarz. Angenäherte auflösung von sysmen linearer gleichungen. *Bull. Acad. Poli. Sci. Lett. A*, 35:355–357, 1937.

[30] A. C. Kak and Malcolm Slaney. *Principles of Computerized Tomographic Imaging*. IEEE Press, 1988.

[31] H. Kostler, C. Popa, M. Prummer, and U. Rude. Algebraic full multigrid in image reconstruction. *Mathematical Modelling of Environmental and Life Sciences Problems*, pages 123–130, 2006.

[32] R.M. Lewitt. Multidimensional digital image representations using generalized Kaiser-Bessel window functions. *J. Opt. Soc. Amer.*, 7:1834–1846, 1990.

[33] D.N. Mastronarde. Dual-axis tomography: An approach with alignment methods that preserve resolution. *J. Struct. Biol.*, 120:343–352, 1997.

[34] F. Natterer and E. L. Ritman. Past and future directions in X-ray computed tomography (CT). *International Journal of Imaging Systems and Technology*, 12:175 – 187, 2002.

[35] P. S. Pacheco. *Parallel Programming with MPI.* Morgan Kaufmann, 1997.

[36] X. Pan, E. Y Sidky, and M. Vannier. Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction? *Inverse Problems*, 25, 2009.

[37] G.A Perkins, C.W. Renken, J.Y. Song, T.G. Frey, S.J. Young, S. Lamont, M.E. Martone, S. Lindsey, and M.H. Ellisman. Electron tomography of large multicomponent biological structures. *J. Struc. Biol.*, 120:219–227, 1997.

[38] D. Pointer. Computed tomography (CT) scan image reconstruction on the SRC-7. *RSSI*, 2008.

[39] C. Popa and R. Zdunek. Kaczmarz extended algorithm for tomographic image reconstruction from limited data. *Mathematics and Computers in Simulation*, 65:579–598, 2004.

[40] J. Radon. Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten. *Ber. Sachs. Akad. Wiss.*, 69:262–267, 1917.

[41] G. N. Ramachandran and A. V. Lakshminarayanan. Three dimensional reconstructions from radiographs and electron micrographs: Application of convolution instead of Fourier transforms. *Proc. Nat. Acad. Sci.*, 68:2236–2240, 1971.

[42] R.S. Ramakrishnan, S.K. Mullick, R.K.S Rathore, and R. Subramanian. Orthogonalization, Bernstein polynomials, and image restoration,. *Appl. Opt.*, 18:464–468, 1979.

[43] R.F. Redberg. Cancer risks and radiation exposure from computed tomographic scans. *Arch Intern Med*, 169 (22), 2009.

[44] E.L. Ritman. Medical X-ray imaging, current status and some future challenges. *JCPDS*, pages 1–12, 2006.

[45] L. A. Shepp and B. F. Logan. The Fourier reconstruction of a head section. *IEEE Trans. Nucl. Sci.*, NS-21:21–43, 1974.

[46] R. Snieder and J. Tramper. Linear and nonlinear inverse problems. *Geomatic methods for the analysis of data in the earth sciences*, pages 93–164, 2000.

[47] H. Turbell. Cone-beam reconstruction using filtered backprojection. *Linkopings Universitet*, 2001.

[48] L.K. Wagner, P.J. Eifel, and R.A. Geise. Potential biological effects following high X-ray dose interventional procedures. *J. Vasc. and Intervent. Radiol.*, 5:71–84, 1994.