AN ABSTRACT OF THE THESIS OF

Ryan Andrew Hass for the degree of Doctor of Philosophy in Mathematics presented on November 25, 2009.

Title: π-Line Reconstruction Formulas in Computed Tomography

Abstract approved: _____

Adel Faridani

X-ray computed tomography is a noninvasive imaging modality capable of reconstructing exact density values of 3D objects. Computed tomography machines are deployed across the world to provide doctors with an image that reveals more detail than a standard x-ray image. We investigate algorithms based on exact computed tomography reconstruction formulas where the backprojection depends on the π-line of the point to be reconstructed. This includes the helical inversion formulas of Katsevich and Pan et al. Our work provides numerical analysis, insight into the algorithms and practical applications of π-line reconstruction formulas.

A certain derivative with respect to source position appears in many π-line reconstruction formulas. Its accurate implementation is critical for the performance of the algorithms and there are several numerical methods available to calculate the derivative. New error estimates are derived for the numerical methods to calculate the derivative in the fan-beam setting with the curved detector geometry. Theoretical justification for previously proposed methods is provided. Numerical results from simulated data are presented to confirm the theoretical results.

The helical inversion formulas of Katsevich and Pan et al have been extended to a larger class of source trajectories. The generalized formulas depend on π-lines and a method is presented to calculate π-lines for a variable pitch and radius helix. We introduce the set called the region of backprojection that identifies all points that are reconstructed from the measured data of the current source position. The region of backprojection is described for the helix and the circle.

A characteristic artifact is found in reconstructions from formulas that depend on π-lines. The boundary of the region of backprojection is hypothesized as the cause of the artifact. A mathematical framework is presented to identify the location of the artifact in the reconstruction. This "comet tail artifact" is also present in reconstructions based on π-lines with improperly aligned measured data. An error term for the misaligned data is developed and an algorithm is presented to correctly align the data.

Implementation details for the reconstruction methods discussed are provided.

π-Line Reconstruction Formulas in Computed Tomography

by

Ryan Andrew Hass

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Presented November 25, 2009
Commencement June 2010

Doctor of Philosophy thesis of Ryan Andrew Hass presented on November 25, 2009.

APPROVED:

Major Professor, representing Mathematics

Chair of the Department of Mathematics

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Ryan Andrew Hass, Author

ACKNOWLEDGEMENTS

Academic

I am indebted to Adel Faridani for his patience and guidance over the last 6 years.

Personal

I wish to thank my parents, Terry and Roy Hass, for all the opportunities they have created for me.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

Page

TABLE OF CONTENTS (Continued)

Page

LIST OF FIGURES

Figure Page

LIST OF TABLES

1 Introduction

A patient who undergoes a Computer Tomography (CT) scan lies on a Table surrounded by a large donut shape containing an x-ray source and a detector array. The x-ray source and the detector rotate around the patient while emitting radiation and measuring the loss of energy. Figure 1.1 provides an illustration of such a machine. After the scan is complete an image, similar to one which appears in Figure 1.2, of the patient's interior is then provided to a radiologist. The process of creating the image of the patient's interior from x-ray measurements is modeled mathematically by recovering a function of density values from line integrals of the function.

One of the driving motivations of CT, or tomography in general, is to overcome the limits of natural human vision. Tomography provides a non destructive interior image of an opaque object from a series of x-ray images. Since 1973, CT imaging has become a fundamental tool of radiology departments world wide. The field moves quickly and new research provides better reconstructions by utilizing more advanced algorithms and hardware. Tomorrow's machines will be reconstructing objects, not just in a plane, but also across entire length of the object including the entire human body. Such volume reconstructions have applications in biomedical imaging and industrial engineering.

Fan-beam tomography is a popular scanning method to reconstruct 2D density functions. x-rays are emitted from a x-ray source and the intensities are measured on a single row of detectors. The x-ray source and detectors are rotated to a new viewing angle and the x-rays are emitted and measured again. The process repeats until the x-rays have been measured over a sufficient number of angles. The function that is reconstructed is supported in the plane of rotation. In cone-beam tomography the number of detector rows increases from the fan-beam case. The goal is to reconstruct a 3D density function on slices that are parallel to the plane of rotation. The reconstructions from cone-beam formulas are only exact in the plane of rotation.

The helix is a source position trajectory that meets the needs of exact 3D imaging. In 2002 Katsevich [12] proposed an exact formula to reconstruct 3D functions from a helical source trajectory. In 2004 Pan et al [32, 33] developed another exact reconstruction

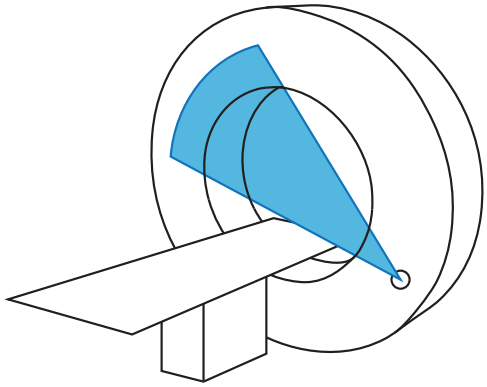

Figure 1.1: Typical CT scanner.

formula for helical tomography. Where the methods differ is in the direction of the filtering planes. Katsevich's formula filters the data along planes that intersect the source position trajectory at three positions. While Pan et al's formula filters the function along special lines that pass through the support of the function and intersect the source trajectory twice. Both methods apply a Hilbert transform to filter the data.

Definition 1.1. *Suppose we have a parametrized curve* $\mathbf{y}(s)$*. We say*

$$f(\mathbf{x}) = \int_{[s_b(\mathbf{x}),s_t(\mathbf{x})]} g(\mathbf{y}(s),\mathbf{x})\, ds$$

is a π*-line reconstruction formula if the domain of integration with respect to* s *is* $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$ *where* $\mathbf{x}$ *is on the line from* $\mathbf{y}(s_b(\mathbf{x}))$ *to* $\mathbf{y}(s_t(\mathbf{x}))$*. We call the line that passes through* $\mathbf{y}(s_b(\mathbf{x}))$ *and* $\mathbf{x}$ *the* π*-line of* $\mathbf{x}$*.*

There are π-line reconstruction formulas for fan-beam, helical, circle plus line, and saddle scanning trajectories. Our work encompasses the numerical analysis of π-line reconstruction formulas. We present results that provide insights in to the behavior of π-line reconstruction formulas and present practical applications for them.

The derivative in Katsevich's and Pan et al's formula is common to many π-line reconstruction formulas. Numerical methods for implementing the derivative have been developed by Noo et al [19, 18], Faridani et al [8], and by Yu and Wang [29]. Faridani et al proposed a method to calculate the derivative in the local detector coordinate system that

Figure 1.2: CT reconstruction of a human pelvis.

overcame performance issues found in older methods. Noo, Hoppe et al provided a very general method to implement the derivative for a variety of source position trajectories and defined it outside of the local detector coordinate system. However no mathematical theory has been provided to justify any advantages of the method of Noo, Hoppe et al. The goal is then to identify the strengths and weaknesses of the current discretization schemes. To do so we provide error estimates for the methods presented in [19, 8, 29]. Furthermore we provide numerical analysis of Noo, Hoppe et al's method in the fan-beam geometry and compare the method to Faridani et al's method.

The notion of a π-line for helical tomography was established in [3] and the existence and uniqueness of π-lines was developed in [5]. Izen [11] proposed a robust solution for calculating the π-lines of a point for the helix. Katsevich's and Pan et al's formula have been extended to more general scanning trajectories and the formulas depend on the existence and uniqueness π-lines. We extend Izen's method to calculate π-lines for helical scanning trajectories with a variable radius and variable pitch. The result is a simple algorithm to compute π-lines for general scanning trajectories.

We have observed a new type of artifact found in π-line reconstruction formulas. The comet tail artifact has occurred in high accuracy reconstructions of smooth density functions in both 2D and 3D π-line reconstructions. The artifact is also present in reconstructions from poorly aligned x-ray projection data. In this work we present

a novel mathematical framework to justify the location and shape of the comet tail artifact. We introduce a set called the region of backprojection and demonstrate the basic properties of the set. We hypothesize that the location of the artifact depends on the backprojection at the end of the interval $[s_b(\mathbf{x}), s_t(\mathbf{x})]$ and develop theory to support our numerical results. In particular the boundary of the region of backprojection contains points where $s \approx s_b(\mathbf{x})$ and $s \approx s_t(\mathbf{x})$ and the shape of the boundary dictates the location of the comet tail artifact. Naturally the choice of π-lines defines the boundary of the region of backprojection and therefore different families of π-lines can produce different shaped comet tail artifacts.

The remainder of Chapter 1 is devoted to the notation and geometry we will use for the remainder of this work. We introduce the π-line reconstruction formulas that we study and the standard filtered backprojection formula. We also describe in greater detail the filtering planes used in Katsevich's and Pan et al's reconstruction formulas.

In Chapter 2 we present the derivative in Katsevich's and Pan et al's formulas and discuss prior numerical implementations. We describe Noo, Hoppe et al's method in greater detail and present the numerical analysis of the method. Our estimate requires a decomposition of the method into three terms and to express the method in the local curved detector coordinate system. Numerical experiments, with exact projection data, are provided to support our theory

Chapter 3 presents Izen's construction for decomposing the helix cylinder into disjoint surfaces of π-lines called chips. We generalize the notion of chips to general scanning trajectories and in the process develop a method to calculate π-lines for the variable radius and variable pitch helix. We discuss the benefits of using the chips as a method to organize the computation necessary in Pan et al's 3D reconstruction formula. We also present results on the runtime of Katsevich's and Pan et al's reconstruction formulas.

We study comet tail artifacts in Chapter 4. Here we introduce the set that we call the region of backprojection and hypothesize that the boundary of the region of backprojection determines the shape and location of the comet tail artifact. We derive properties for the region of backprojection and determine the region of backprojection for helical CT. We define the region of backprojection for the helix in terms of the chips defined in Chapter 3. We then describe the region of backprojection for the circular scan for a families of 2D π-lines. In the 2D and 3D case we develop theory for the location of the comet tail artifact.

Chapter 5 describes the behavior of a π-line reconstruction formula with regards to misaligned data. After the projection data has been loaded into a computer the location of the detector bins must determined. The resolution of the reconstruction is affected by improperly aligned detector bins. We develop an error term for the reconstruction with misaligned data and observe that comet tail artifacts occur. We propose a heuristic method to properly align the data based on the norm of the reconstruction from a π-line reconstruction method. We extend our method to helical tomography and misaligned data.

In Chapter 6 we describe numerical implementation details for π-line reconstruction formulas described in the this work. In Chapter 7 we provide a convergence study of Katsevich's formula and discuss the performance of Katsevich's formula with regards to the selection of the pitch of the helix.

In Chapter 8 we end with some closing remarks.

1.1 Mathematical Model for CT

We will introduce the helical cone-beam scanning geometry and present the notation that will guide us through the rest of the paper. In particular we will describe the basic framework that allows us to reconstruct a function from its helical cone beam data and how there is a natural extension to the fan-beam model. This extension to the fan-beam model has proved to be a valuable resource in studying the behavior of the various exact 3D reconstruction formulas.

Let S^2 be the unit sphere and suppose Ω is the open unit ball. We assume that the x-ray is of high energy and that is does not scatter while passing through the object. Hence we take the line integral of $f(\mathbf{x}) \in L^1_0(\Omega)$, in the direction $\boldsymbol{\theta} \in S^2$ from position $\mathbf{y} \in \mathbb{R}^3$. The divergent beam x-ray transform of f is then given by

$$\mathcal{D}f(\mathbf{y}, \boldsymbol{\theta}) = \int_0^\infty f(\mathbf{y} + t\boldsymbol{\theta})\, dt$$

where we are assuming that $\mathbf{y}$ is outside the support of f.

1.2 Helical Reconstructions

Our setting is a helical source curve with cylindrical detector coordinate system. Here we assume a source curve $\mathbf{y}(s) = (R\cos(s), R\sin(s), \frac{p}{2\pi}s)$ with radius R and pitch p. We position a curved detector array at a distance D from the source position $\mathbf{y}(s)$. An orthonormal coordinate system that rotates with source position is given by

$$\begin{aligned}\mathbf{e}_u(s) &= [-\sin(s), \cos(s), 0]\\ \mathbf{e}_v(s) &= [-\cos(s), -\sin(s), 0]\\ \mathbf{e}_w(s) &= [0, 0, 1].\end{aligned}$$

Note that $\mathbf{y}(s) = -R\mathbf{e}_v(s) + \frac{ps}{2\pi}\mathbf{e}_w(s)$.

An x-ray emits from $\mathbf{y}(s)$ in the direction Θ, passes through the object f and is measured by the detector array at position

$$\begin{gathered}\mathbf{d}(s,\alpha,w) = \mathbf{y}(s) + D\sin(\alpha)\mathbf{e}_u(s) + D\cos(\alpha)\mathbf{e}_v(s) + w\mathbf{e}_w(s)\\ |\alpha| \le \alpha_{max}, \quad |w| \le w_{max}, \quad D \ge R.\end{gathered}$$

The vector between $\mathbf{y}(s)$ and detector position $\mathbf{d}(s,\alpha,w)$ is

$$\boldsymbol{\theta}(s,\alpha,w) = \frac{1}{\sqrt{D^2+w^2}}(D\sin(\alpha)\mathbf{e}_u(s) + D\cos(\alpha)\mathbf{e}_v(s) + w\mathbf{e}_w(s)).$$

Here α is the angle between the projections of $\mathbf{e}_v$ and $\boldsymbol{\theta}$ onto the plane $x_3 = 0$ and w is the difference in the x_3 coordinate between detector position $\mathbf{d}(s,\alpha,w)$ and $\mathbf{y}(s)$; cf. Figure 1.3. The measured data are given by

$$g(s,\alpha,w) = Df(\mathbf{y}(s), \boldsymbol{\theta}(s,\alpha,w)).$$

The line from $\mathbf{y}(s)$ to the point $\mathbf{x}$ intersects the curved detector at $(\alpha^*, w'(s,\alpha^*,\mathbf{x}))$ where

$$\alpha^* = \alpha^*(s,\mathbf{x}) = \arctan\left(\frac{\langle \mathbf{x}, \mathbf{e}_u\rangle}{R + \langle \mathbf{x}, \mathbf{e}_v\rangle}\right), \quad w'(s,\alpha^*,\mathbf{x}) = \frac{D\cos\alpha^*}{R + \langle \mathbf{x}, \mathbf{e}_v\rangle}\left(x_3 - \frac{ps}{2\pi}\right). \quad (1.1)$$

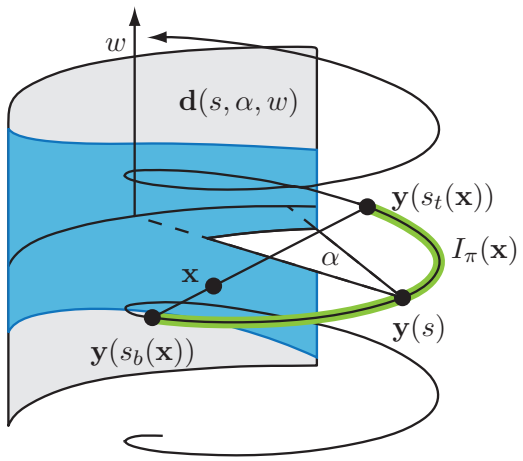

Figure 1.3: The Tam-Danielsson window is the shaded region of the detector surface $\mathbf{d}(s,\alpha,w)$. The π-line is shown as the line segment between $\mathbf{y}(s_b(\mathbf{x}))$ and $\mathbf{y}(s_t(\mathbf{x}))$.

We can express $\boldsymbol{\theta}$ in the local detector coordinates by

$$\alpha = \arctan\left(\frac{\boldsymbol{\theta}\cdot\mathbf{e}_u(s)}{\boldsymbol{\theta}\cdot\mathbf{e}_v(s)}\right), \quad w = \frac{D\boldsymbol{\theta}\cdot\mathbf{e}_w(s)}{\sqrt{1-(\boldsymbol{\theta}\cdot\mathbf{e}_w(s))^2}}. \tag{1.2}$$

The Tam-Danielsson window, shown in Figure 1.3 and Figure 1.4, is region of the detector array that lies between the stereographic projection from $\mathbf{y}(s)$ of the upper and lower turns of the helix onto the detector array. The angular width of the Tam-Danielsson window is $(-\alpha_{max},\alpha_{max})$, where $\alpha_{max} = \sin^{-1}(r/R)$, r is the radius of the support of the cylinder that contains the support of the function and R is the radius of the helix. Furthermore if $s \in I_\pi(\mathbf{x})$ then $\mathbf{x}$ is projected onto the Tam-Danielsson window corresponding to $\mathbf{y}(s)$. If the $\mathbf{x}$ is inside the helix cylinder and if $\mathbf{x}$ is projected onto the Tam-Danielsson window of $\mathbf{y}(s)$ then $s \in I_\pi(\mathbf{x})$ [16].

The choice of the filtering planes is one of Katsevich's important contributions to helical CT. Katsevich's inversion formula relies on filtering the measured data along a special family of planes called κ planes.

Definition 1.2. *A κ plane is any plane that intersects the helix at three points where one point of intersection is halfway between the other two points of intersection. We*

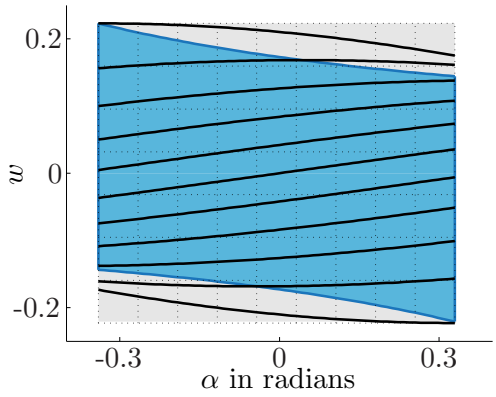

Figure 1.4: The κ curves covering a curved detector array. The shaded region represents the Tam-Danielsson Window.

call the intersection of the κ plane with the detector array a κ curve. We identify the κ curves by the points $\mathbf{y}(s)$, $\mathbf{y}(s+\phi)$, *and* $\mathbf{y}(s+2\phi)$.

The minimal amount of data to reconstruct $\mathbf{x}$ corresponds to measurements made within the Tam-Danielsson window, [22]. A point is on the Tam-Danielsson window if and only if the source position is in the point's π-interval [16]. As seen in Figure 1.4 the κ curves on the detector array may lie outside of the Tam-Danielsson window. Therefore when one filters along the κ curve data the does not lie within the Tam-Danielsson window is used in the reconstruction.

A point on the detector array may lie on more than one κ curve. This is because the κ curves tend to intersect near the top and bottom of the Tam-Danielsson window. If we increase α in Figure 1.4 we would have the κ curves intersecting. It is useful to classify κ planes by $\mathbf{y}(s)$ and ϕ the angle such that $\mathbf{y}(s)$, $\mathbf{y}(s+\phi)$ and $\mathbf{y}(s+2\phi)$ lie in the κ plane. We discuss how to handle a point that lies on multiple κ curves in Section 6.1.2.

We are now ready to present formulas that allow us to reconstruct a function from its x-ray data. The following formulas are by no means every option to invert x-ray data in two or three dimensions. We do however present formulas that follow a general theme. First the x-ray data is differentiated with respect to the detector coordinate system. Then the data is filtered along curves which represent the intersection of the detector surface and κ plane. Finally the data is backprojected along the line connecting the

detector coordinate's location and the x-ray's source position. Such a method is called a filtered backprojection (FBP) algorithm.

Another choice is to interchange the backprojection and the filtering steps. This involves filtering the data not in the detector coordinate but instead filtering the back-projected data along lines that pass through the reconstructed object. We call this type of method a backprojection filtration (BPF) method. We present 3D reconstruction formulas of the FBP and BPF type and describe advantages and drawbacks of each.

Katsevich's inversion formula is

$$f(\mathbf{x}) = \frac{1}{2\pi^2} \int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} \frac{\partial}{\partial q} \mathcal{D} f(\mathbf{y}(q), \cos\gamma\boldsymbol{\beta} + \sin\gamma\boldsymbol{\beta}^\perp)\Big|_{q=s} \frac{d\gamma}{\sin\gamma} ds \tag{1.3}$$

where

$$\boldsymbol{\beta} = \boldsymbol{\beta}(s, \mathbf{x}) = \frac{\mathbf{x} - \mathbf{y}(s)}{|\mathbf{x} - \mathbf{y}(s)|}. \tag{1.4}$$

The vector $\boldsymbol{\beta}^\perp$ is chosen orthogonal to $\boldsymbol{\beta}$ such that the two vectors span a κ plane.

This method is a FBP formula and is exact for sufficiently smooth functions. To reconstruct f at the point $\mathbf{x}$ the formula requires source positions between the endpoints of the π-interval $I_\pi(\mathbf{x})$ and hence does not require source positions far away from $\mathbf{x}$. For each point to be reconstructed we need to backproject, from source positions along its π-interval, the data that has been filtered along the κ plane that contains the point.

A Hilbert kernel is used to filter along κ curves. The same kernel is used at every source position and each κ curve. Furthermore the κ curves have the same detector coordinates for each source position. This observation will simplify our numerical implementation in Section 6.1.2.

Katsevich's formula, as expressed in our local detector coordinates, is

$$\begin{aligned} f(\mathbf{x}) = \frac{1}{2\pi^2} \int_{I_\pi(\mathbf{x})} \frac{\cos\alpha^*}{R + \langle \mathbf{x}, \mathbf{e}_v \rangle} \int_0^{2\pi} \frac{D}{\sqrt{D^2 + w'^2}} \left(\frac{\partial g}{\partial s}(\alpha^*, w') + \frac{\partial g}{\partial \alpha}(\alpha^*, w') \right) \\ \times \frac{1}{\sin(\alpha^* - \alpha)} \, d\alpha \, ds. \end{aligned} \tag{1.5}$$

This representation of Katsevich's formula first appeared in [19]. We provide a complete set of implementation details for (1.5) in Section 6.1.

We now present an alternative formula for helical tomography. The formula relies on a different method of reconstruction developed by Pan et al [32] in the 3D setting which is based on Noo et al's work in the 2D case [4]. To reconstruct a point utilizing a π-line filtration method the data is first differentiated with respect to the local detector coordinates with the same derivative found in (1.3). Then one backprojects the data for $\mathbf{x}$ and all points along the π-line of $\mathbf{x}$. Once this has been done for all source positions in the π-interval, the data that has been backprojected onto the π-line of $\mathbf{x}$ is filtered with a 1D inverse Hilbert transform along the π-line of $\mathbf{x}$. We now have the function $f(\mathbf{x})$ on its π-line segments.

Let $\pi(\mathbf{x},t) = (1-t)\mathbf{y}(s_b(\mathbf{x})) + t\mathbf{y}(s_t(\mathbf{x}))$ be a parameterization of the π-line through $\mathbf{x}$. We suppose that $f(\mathbf{x})$ is supported inside the unit cylinder and so there exists a and b such that $f(\pi(\mathbf{x},t)) = 0$ for $0 \leq t < a$ and $b < t \leq 1$. Define

$$G(\pi(\mathbf{x},t)) = \int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x}-\mathbf{y}(s)|} \frac{\partial}{\partial q} Df(\mathbf{y}(q), \boldsymbol{\theta}(s,\mathbf{x}))\Big|_{q=s} \, ds. \tag{1.6}$$

Then we have

$$G(\pi(\mathbf{x},t)) = 2\int_a^b \frac{f(\pi(\mathbf{x},t'))}{t'-t} \, dt' \tag{1.7}$$

as shown in [32, 33]. Thus G is the Hilbert transform of f along the π-line given by $I_\pi(\mathbf{x})$. This inversion formula is a BPF algorithm. In local detector coordinates the formula reads as

$$G(\pi(\mathbf{x},t)) = \int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x}-\mathbf{y}(s)|} \left(\frac{\partial g}{\partial s}(\alpha^*, w') + \frac{\partial g}{\partial \alpha}(\alpha^*, w') \right) \, ds. \tag{1.8}$$

We have our π-line filtration formula

$$f(\pi(\mathbf{x},t)) = \frac{1}{2\pi^2\sqrt{(b-t)(t-a)}} \left(\int_a^b \frac{\sqrt{(b-t')(t'-a)}}{t-t'} G(\pi(\mathbf{x},t')) \, dt' + \pi c \right) \tag{1.9}$$

with

$$c = 2\frac{Df(\mathbf{y}(s_b(\mathbf{x})), \Theta(s_b(\mathbf{x}),\mathbf{x}))}{|\mathbf{y}(s_b(\mathbf{x})) - \mathbf{y}(s_t(\mathbf{x}))|}. \tag{1.10}$$

Equation (1.9) is different because it does not require filtering over any of the κ curves. Instead, the method is based on the idea that the backprojection of the derivative of

the x-ray data gives the Hilbert transform of the original image along its π-lines. One need only to invert the Hilbert transform assuming the function and its Hilbert data are compactly supported. The inversion of the Hilbert transform in this setting is ideal for CT. The solution of the integral equation (1.7) requires a constant c, the integral of $f(\mathbf{x})$ along the π-line, which is exactly what our measured x-ray data gives us at positions $\mathbf{y}(s_b(\mathbf{x}))$ and $\mathbf{y}(s_t(\mathbf{x}))$.

We shall now discuss the key differences between Katsevich's formula and the π-line filtration formula. The two differ in where the formulas choose to filter. The Katsevich formula filters the data in the local detector coordinates at each source position. The π-line filtration formula filters in the coordinates of $f(\mathbf{x})$ after all the points have been backprojected. Thus the amount of filtering required to reconstruct from Katsevich's formula grows with the number of source positions. The trade off is the π-line filtration may require more data to be backprojected. A segment of the π-line must be back-projected for each point we wish to reconstruct and for every source position. Thus if we only wish to reconstruct a function on a single π-line the π-line filtration method will be faster than the Katsevich formula. Furthermore if we restrict our reconstruction to only a plane then Katsevich's formula will be faster. The crux of the problem is the BPF formula cannot recover functions in a plane without reconstructing the volume that contains the π-lines of the points within the plane.

This downfall in the number of backprojections required in the BPF method can be overcome by utilizing a FBP of the π-line filtration method developed in [20]. Now instead of filtering along the π-line after every point has been backprojected, it is sufficient to filter the data on the detector surface along the projection of π-line segment. Hence a rebinning of measured data to the projection of the π-line onto the detector plane is necessary at each position. Unlike the Katsevich FBP formula, which filters along the same κ curves for all source positions, the filtering in the FBP π-line filtration algorithm has a different interpolation Table for each source position. This is caused by the fact that the projection of a π-line onto the detector surface is different for all source positions and must be calculated for each position.

Another issue to consider is which positions on the detector surface are used in each method. The π-line filtration method only uses data from points within the Tam-Danielsson window. Since every point along the π-line has the same π-interval and only source positions within the π-interval are used, every point on the π-line is within the

Tam-Danielsson window. However the Katsevich formula filters along the κ curves which do not necessarily intersect the Tam-Danielsson window (See Figure 1.4). This implies that the π-line filtration method uses less data on the detector surface to recover $f(\mathbf{x})$. For this reason the π-line filtration method, both the BPF and FBP varieties, are called minimum data reconstruction formulas. The Katsevich formula is not a minimum data reconstruction formula. Satisfying this property is only practical from a mathematical point of view.

The helical tomography reconstruction formulas discussed here solve the "long object" problem. That is the projection data can be truncated in the direction of the axis of the helix and still yield an exact reconstruction. The practical advantage of such a method is the detector that measures the x-rays does not need to be taller than the object to be reconstructed.

1.3 Fan-Beam Reconstructions

Now consider the source curve $\mathbf{y}(s) = (R\cos(s), R\sin(s))$ with radius R. We present old and new formulas for inverting the fan-beam x-ray data. If we let the pitch go to zero in equation (1.3) we have [12]

$$f(\mathbf{x}) = \frac{1}{2\pi^2}\int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x}-\mathbf{y}(s)|}\int_0^{2\pi}\frac{\partial}{\partial q}Df(\mathbf{y}(q),\Theta(s,x,\gamma))\Big|_{q=s}\frac{d\gamma}{\sin\gamma}ds \tag{1.11}$$

where $\mathbf{x}$ and $\mathbf{y}(s)$ are in the plane $x_3 = 0$. In local detector coordinates (1.11) is

$$f(\mathbf{x}) = \frac{1}{2\pi^2}\int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x}-\mathbf{y}(s)|}\int_0^{2\pi}\left(\frac{\partial g}{\partial s}(s,\alpha^*)+\frac{\partial g}{\partial \alpha}(s,\alpha^*)\right)\frac{1}{\sin(\alpha^*-\alpha)}\,d\alpha\,ds. \tag{1.12}$$

These formulas have been shown to hold in a more general setting by the work of Faridani et al [8]. Equation (1.12) has a striking resemblance to (1.5). Again the derivative is the same as before but now the filtering occurs strictly in the α coordinate before backprojecting. As we shall see in Section 4.3, we have far greater flexibility in the choice of the π-intervals compared to the helical trajectory.

We can remove the dependency of integrating over π-lines in formula (1.11) if we consider $[0,2\pi] = I_{\pi(\mathbf{x})} \cup I^c_{\pi(\mathbf{x})}$. This yields another inversion formula credited to Herman

and Naparstek [10]

$$f(\mathbf{x}) = \frac{1}{4\pi^2} \int_0^{2\pi} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} \left(\frac{\partial g}{\partial s} + \frac{\partial g}{\partial \alpha} \right) \frac{1}{\sin(\alpha^* - \alpha)} \, d\alpha \, ds. \tag{1.13}$$

It has been shown that the partial derivative with respect to s drops out when we have a circular source curve [27, 31]

$$0 = \frac{1}{4\pi^2} \int_0^{2\pi} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} \frac{1}{\sin(\alpha^* - \alpha)} \frac{\partial g}{\partial s}(s, \alpha) \, d\alpha \, ds. \tag{1.14}$$

This yields the formula

$$\begin{aligned} f(\mathbf{x}) &= \frac{1}{4\pi^2} \int_0^{2\pi} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} k(\alpha^* - \alpha) \frac{\partial g}{\partial \alpha}(s, \alpha) \, d\alpha \, ds \\ &= \frac{1}{4\pi^2} \int_0^{2\pi} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} k'(\alpha^* - \alpha) g(s, \alpha) \, d\alpha \, ds \end{aligned} \tag{1.15}$$

where

$$k(\alpha) = \frac{1}{\sin(\alpha)}. \tag{1.16}$$

The above formula contains the same filtering as the standard parallel beam formula in 2D.

2 View Dependent Derivatives

There have been two recent methods for implementing a data derivative found in fan-beam, cone-beam and helical tomography formulas. Each method's aim is to efficiently implement the derivative without decreasing resolution in the reconstruction. Calculated with respect to the source position while holding the line integral direction fixed, the derivative is given by

$$\frac{\partial}{\partial q}\mathcal{D}f(\mathbf{y}(q),\boldsymbol{\theta}(s,\alpha))\Big|_{q=s}. \tag{2.1}$$

The term appears in the reconstruction formulas given by (1.3), (1.6), and (1.11) and the subsequent formulas (1.5), (1.13), (1.12) and (1.8) when it is expressed in local detector coordinates. Geometrically, the derivative can be understood as the difference quotient of x-ray data from parallel lines that pass through the object. The direct implementation of the derivative provides poor results and efforts have been made to increase accuracy. The derivative is considered unfavorable because it can reduce tangential resolution in the reconstruction and the early implementation schemes reduced the competitiveness of the reconstruction algorithm compared to other reconstruction methods [31]. The derivative can be removed by an integration by parts [10, 12]. However this choice increases the computational complexity of the numerical inversion in the 3D case [18].

The method proposed by Noo et al [18] is robust with regards to adapting to many different scanning geometries. The Noo et al method requires linear interpolation in the data specified by a free parameter ε and then a differentiation step. A straightforward method has been proposed by Faridani et al [8]. Faridani et al's method relies on standard central difference methods over the native detector geometry and requires no ε parameter. We will show that the Noo et al method is a second order method. Our analysis depends on a decomposition of Noo et al's method into 3 difference quotients and expressing them in the fan-beam curved detector geometry. We then perform a Taylor expansion to describe the error terms from interpolation and the difference quotients found in Noo et al's method. Our new analysis provides a uniform framework to compare the error

terms of the current methods proposed in the literature for the implementation of the derivative.

We begin with an overview of our notation and introduce the difference schemes in better detail in Section 2.1. An analysis of Faridani et al's and Noo et al's derivatives in the local detector coordinates is presented in Section 2.2 and 2.3. We conclude in Section 2.4 with a comparison of accuracy of the two methods. We also design numerical experiments to illustrate our results from Section 2.2 and 2.3 and investigate claims made about the undesirable behavior of the derivative term in the reconstruction formula.

2.1 View Dependent Derivatives

The goal of this Section is to describe methods for effective discretization of (2.1). Let $\mathbf{y}(s) = (R\cos(s), R\sin(s))$. We assume that we have angular discretization $\alpha_i = i\Delta\alpha$, for $i = -q \dots q-1$, and $\Delta\alpha = \sin^{-1}(r/R)/q$. Our source position curve is discretized by $s_k = k\Delta s$. We denote $I\mathcal{D}(\mathbf{y}(s), \boldsymbol{\theta})$ and $Ig(s, \alpha)$ as terms that may require linear interpolation in the second variable.

The first method we consider is [19, equation (19)], the direct scheme,

$$\begin{aligned} &\frac{\partial}{\partial q}\mathcal{D}f(\mathbf{y}(q), \boldsymbol{\theta}(s_{k+1/2}, \alpha_i))\Big|_{q=s_{k+1/2}} \\ &\quad\approx \frac{I\mathcal{D}f(\mathbf{y}(s_{k+1}), \boldsymbol{\theta}(s_{k+1/2}, \alpha_i) - I\mathcal{D}f(\mathbf{y}(s_k), \boldsymbol{\theta}(s_{k+1/2}, \alpha_i))}{\Delta s} \end{aligned} \tag{2.2}$$

and we denote this method as M0. Method M0 is a natural choice because it is faithful to the derivative with respect to source position but with a fixed $\boldsymbol{\theta}$; cf. Figure 2.1.

As stated in [19] if we use (1.2) then we have

$$\begin{aligned} &\arctan\left(\frac{\boldsymbol{\theta}(s_{k+1/2}, \alpha_i)\cdot\mathbf{e}_u(s)}{\boldsymbol{\theta}(s_{k+1/2}, \alpha_i)\cdot\mathbf{e}_v(s)}\right) \\ &\quad= \arctan\left(\frac{(\sin(\alpha_i)\mathbf{e}_u(s+\Delta s/2) + \cos(\alpha_i)\mathbf{e}_v(s+\Delta s/2))\cdot\mathbf{e}_u(s)}{(\sin(\alpha_i)\mathbf{e}_u(s+\Delta s/2) + \cos(\alpha_i)\mathbf{e}_v(s+\Delta s/2))\cdot\mathbf{e}_v(s)}\right) \\ &\quad= \alpha_i - \Delta s/2 \\ &\arctan\left(\frac{\boldsymbol{\theta}(s_{k+1/2}, \alpha_i)\cdot\mathbf{e}_u(s+\Delta s)}{\boldsymbol{\theta}(s_{k+1/2}, \alpha_i)\cdot\mathbf{e}_v(s+\Delta s)}\right) = \alpha_i + \Delta s/2. \end{aligned}$$

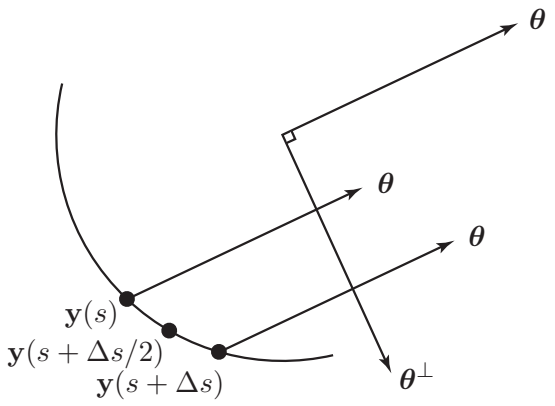

Figure 2.1: The directions $\boldsymbol{\theta}$ used in the direct scheme M0.

Hence expressed in detector coordinates, the direct scheme becomes

$$\frac{Ig(s_{k+1}, \alpha_i + \Delta s/2) - Ig(s_k, \alpha_i - \Delta s/2)}{\Delta s}. \tag{2.3}$$

It may be necessary to apply linear interpolation in α to determine the values of g at $\alpha_i \pm \Delta s/2$. Numerical results have shown M0 to produce low resolution reconstructions [19, 28].

If we write (2.1) in the local detector coordinate system and apply the chain rule we have the following expression

$$\frac{\partial}{\partial q}\mathcal{D}f(\mathbf{y}(q), \boldsymbol{\theta}(s,\alpha))\bigg|_{q=s} = \frac{\partial g}{\partial s}(s,\alpha) + \frac{\partial g}{\partial \alpha}(s,\alpha). \tag{2.4}$$

as shown in [19]. One approach, introduced in [29] and denoted M1 in [8], to discretizing (2.4) is to use central differences in each term

$$\begin{aligned} \frac{\partial g}{\partial s}(s_k, \alpha_i) &\approx \frac{g(s_{k+1}, \alpha_i) - g(s_{k-1}, \alpha_i)}{2\Delta s} \\ \frac{\partial g}{\partial \alpha}(s_k, \alpha_i) &\approx \frac{g(s_k, \alpha_{i+1}) - g(s_k, \alpha_{i-1})}{2\Delta \alpha}. \end{aligned} \tag{2.5}$$

Another approach is to calculate the derivative on the grid

$$(s_{k+1/2}, \alpha_{i+1/2}) = (s_k + \Delta s/2, \alpha_i + \Delta\alpha/2)$$

by

$$\begin{aligned}
\frac{\partial g}{\partial s}(s_{k+1/2}, \alpha_{i+1/2}) &\approx \frac{1}{2\Delta s}[g(s_{k+1}, \alpha_i) - g(s_k, \alpha_i) \\
&\quad + g(s_{k+1}, \alpha_{i+1}) - g(s_k, \alpha_{i+1})] \\
\frac{\partial g}{\partial \alpha}(s_{k+1/2}, \alpha_{i+1/2}) &\approx \frac{1}{2\Delta\alpha}[g(s_k, \alpha_{i+1}) - g(s_k, \alpha_i) \\
&\quad + g(s_{k+1}, \alpha_{i+1}) - g(s_{k+1}, \alpha_i)].
\end{aligned} \tag{2.6}$$

This method appeared in [19] and following [8] we denote it as M3. Note that the stepsize in M3 is half that of M1.

Faridani et al [8] proposed the following method for the discretization of (2.4)

$$\begin{aligned}
\frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}) &\approx \frac{1}{4\Delta s}[g(s_{k+1}, \alpha_i) - g(s_{k-1}, \alpha_i) \\
&\quad + g(s_{k+1}, \alpha_{i+1}) - g(s_{k-1}, \alpha_{i+1})] \\
\frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) &\approx \frac{1}{\Delta\alpha}[g(s_k, \alpha_{i+1}) - g(s_k, \alpha_i)].
\end{aligned} \tag{2.7}$$

and as in [8] we designate this scheme as M4. Here a central difference scheme is applied to both the α and s derivative on the grid $(s_k, \alpha_{i+1/2})$.

The Noo et al method for the derivative [18] is based on the following direct approximation

$$\left.\frac{\partial}{\partial q}\mathcal{D}f(\mathbf{y}(q), \boldsymbol{\theta}(s, \alpha))\right|_{q=s} \approx \frac{\mathcal{D}f(\mathbf{y}(s + \varepsilon\Delta s), \boldsymbol{\theta}(s, \alpha)) - \mathcal{D}f(\mathbf{y}(s - \varepsilon\Delta s), \boldsymbol{\theta}(s, \alpha))}{2\varepsilon\Delta s} \tag{2.8}$$

where ε is a free parameter in $(0, 1]$. We designate this scheme M5. Noo et al approximate the projection data from source position $\mathbf{y}(s + \varepsilon\Delta s)$ by linear interpolation in both s

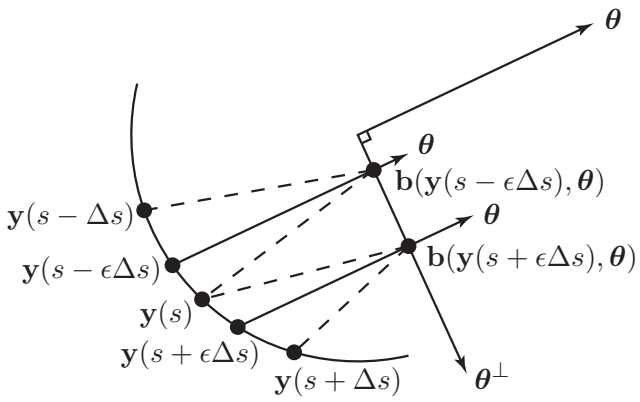

Figure 2.2: The point of interest $\mathbf{b}$ for the source position $\mathbf{y}(s \pm \varepsilon\Delta s)$. The dashed lines represent $\boldsymbol{\eta}$.

and α

$$\begin{aligned}\mathcal{D}f(\mathbf{y}(s+\varepsilon\Delta s),\boldsymbol{\theta}(s,\alpha))) \approx (1-\varepsilon)I\mathcal{D}f(\mathbf{y}(s),\boldsymbol{\eta}(\mathbf{y}(s),s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))\\ +\varepsilon I\mathcal{D}f(\mathbf{y}(s+\Delta s),\boldsymbol{\eta}(\mathbf{y}(s+\Delta s),s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))\\ \boldsymbol{\eta}(\mathbf{y},s,\boldsymbol{\theta}) = \frac{\mathbf{b}(s,\boldsymbol{\theta})-\mathbf{y}}{|\mathbf{b}(s,\boldsymbol{\theta})-\mathbf{y}|}.\end{aligned} \tag{2.9}$$

The authors of [18] designate

$$\mathbf{b}(s,\boldsymbol{\theta}) = \mathbf{y}(s) - \langle \mathbf{y}(s),\boldsymbol{\theta}\rangle\boldsymbol{\theta} \tag{2.10}$$

as the point of interest; cf. Figure 2.2. This expression for $\mathbf{b}$ is the fan-beam geometry point of interest where $\mathbf{n} = (0,0,1)^T$ and $\mathbf{x}_0 = (0,0,0)^T$ for [18, equation (26)]. We define the approximation for $\mathcal{D}f(\mathbf{y}(s-\varepsilon\Delta s)$ by replacing Δs with $-\Delta s$ in (2.9). Here the expression for the derivative is not defined in the local detector coordinates. The difference scheme given by M5 is designed to overcome the fact that sampling in s is often coarser than that of α and to improve over the results of M3.

We must first express M5 in terms of the local detector geometry. To do this we rewrite M5 as the sum of three terms d_s, d_α and d_ε where

$$\begin{aligned}
d_s(s,\alpha,\varepsilon) &= \frac{1}{2\Delta s}[I\mathcal{D}f(\mathbf{y}(s+\Delta s),\boldsymbol{\eta}(\mathbf{y}(s+\Delta s),s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha))) \\
&\quad - I\mathcal{D}f(\mathbf{y}(s-\Delta s),\boldsymbol{\eta}(\mathbf{y}(s-\Delta s),s-\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))] \\
d_\alpha(s,\alpha,\varepsilon) &= \frac{1}{2\varepsilon\Delta s}[I\mathcal{D}f(\mathbf{y}(s),\boldsymbol{\eta}(\mathbf{y}(s),s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha))) \\
&\quad - I\mathcal{D}f(\mathbf{y}(s),\boldsymbol{\eta}(\mathbf{y}(s),s-\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))] \\
d_\varepsilon(s,\alpha,\varepsilon) &= -\varepsilon d_\alpha(s,\alpha,\varepsilon).
\end{aligned}$$

We note that to calculate the terms d_s, d_α, and d_ε requires linear interpolation in α.

We will now express d_s in the local detector coordinates. Suppose $R = 1$

$$\begin{aligned}
\mathbf{y}(s+\varepsilon\Delta s) &= \sin(\varepsilon\Delta s-\Delta s)\mathbf{e}_u(s+\Delta s)-\cos(\varepsilon\Delta s-\Delta s)\mathbf{e}_v(s+\Delta s) \\
\boldsymbol{\theta}(s,\alpha) &= \sin(\alpha+\Delta s)\mathbf{e}_u(s+\Delta s)+\cos(\alpha+\Delta s)\mathbf{e}_v(s+\Delta s) \\
\mathbf{y}(s+\Delta s) &= -\mathbf{e}_v(s+\Delta s) \\
\mathbf{b}(s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)) &= \mathbf{y}(s+\varepsilon\Delta s)-\langle\mathbf{y}(s+\varepsilon\Delta s),\boldsymbol{\theta}(s,\alpha)\rangle\boldsymbol{\theta}(s,\alpha) \\
&= (-\cos(\alpha-2s-\Delta s)\sin(\alpha+\varepsilon\Delta s))\mathbf{e}_u(s+\Delta s) \\
&\quad + (\sin(\alpha-2s-\Delta s)\sin(\alpha+\varepsilon\Delta s))\mathbf{e}_v(s+\Delta s).
\end{aligned}$$

Define

$$\begin{aligned}
h(u,\alpha,\varepsilon) &= \arctan\left(\frac{\langle\boldsymbol{\eta}(\mathbf{y}(s+u),s+\varepsilon u,\boldsymbol{\theta}(s,\alpha)),\mathbf{e}_u(s+u)\rangle}{\langle\boldsymbol{\eta}(\mathbf{y}(s+u),s+\varepsilon u,\boldsymbol{\theta}(s,\alpha)),\mathbf{e}_v(s+u)\rangle}\right) \\
&= \arctan\left(\frac{2\cos(\alpha+u)\sin(\alpha+\varepsilon u)}{\cos(2\alpha+u+\varepsilon u)-\cos(u-\varepsilon u)+2}\right)
\end{aligned} \tag{2.11}$$

which is independent of s. Note we get the same expression for h for an arbitrary $R > 0$. Now the terms in d_s can be expressed in local detector coordinates as

$$I\mathcal{D}f(\mathbf{y}(s+\Delta s),\boldsymbol{\eta}(\mathbf{y}(s+\Delta s),s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha))) = Ig(s+\Delta s,h(\Delta s,\alpha,\varepsilon)) \tag{2.12}$$

and

$$I\mathcal{D}f(\mathbf{y}(s-\Delta s),\boldsymbol{\eta}(\mathbf{y}(s+\Delta s),s-\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))=Ig(s-\Delta s,h(-\Delta s,\alpha,\varepsilon)). \tag{2.13}$$

For the d_α term we have

$$I\mathcal{D}f(\mathbf{y}(s),\boldsymbol{\eta}(\mathbf{y}(s),s+\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))=Ig(s,\nu(\varepsilon\Delta s,\alpha)) \tag{2.14}$$

$$I\mathcal{D}f(\mathbf{y}(s),\boldsymbol{\eta}(\mathbf{y}(s),s-\varepsilon\Delta s,\boldsymbol{\theta}(s,\alpha)))=Ig(s,\nu(-\varepsilon\Delta s,\alpha)) \tag{2.15}$$

with

$$\begin{aligned}\nu(u,\alpha)&=\arctan\left(\frac{\langle\boldsymbol{\eta}(\mathbf{y}(s),s+u,\boldsymbol{\theta}(s,\alpha)),\mathbf{e}_u(s)\rangle}{\langle\boldsymbol{\eta}(\mathbf{y}(s),s+u,\boldsymbol{\theta}(s,\alpha)),\mathbf{e}_v(s)\rangle}\right)\\&=\arctan\left(\frac{2\cos(\alpha)\sin(\alpha+u)}{2-\cos(u)+\cos(2\alpha+u)}\right).\end{aligned} \tag{2.16}$$

Again we get the same expression for ν for an arbitrary $R>0$. Thus M5, in local detector coordinates, is the sum of the following terms

$$d_s(s,\alpha,\varepsilon)=\frac{Ig(s+\Delta s,h(\Delta s,\alpha,\varepsilon))-Ig(s-\Delta s,h(-\Delta s,\alpha,\varepsilon))}{2\Delta s} \tag{2.17}$$

$$d_\alpha(s,\alpha,\varepsilon)=\frac{Ig(s,\nu(\varepsilon\Delta s,\alpha))-Ig(s,\nu(-\varepsilon\Delta s,\alpha))}{2\varepsilon\Delta s} \tag{2.18}$$

$$d_\varepsilon(s,\alpha,\varepsilon)=-\frac{Ig(s,\nu(\varepsilon\Delta s,\alpha))-Ig(s,\nu(-\varepsilon\Delta s,\alpha))}{2\Delta s}. \tag{2.19}$$

Let us remark on the features of the methods presented. In helical and fan-beam tomography Δs is typically larger than $\Delta\alpha$. Method M3 attempts to use a smaller step size in Δs to improve accuracy. Suppose $\Delta s=4\Delta\alpha$. Then method M0 depends on values of α at $\alpha_{i\pm2}$. The step size in α is then too coarse for an accurate estimate of the derivative at α_i.

To highlight the drawbacks of M0 observe that

$$\begin{aligned}\frac{g(s_{k+1},\alpha_i+\Delta s/2)-g(s_k,\alpha_i-\Delta s/2)}{\Delta s} &= \frac{1}{2\Delta s}[g(s_{k+1},\alpha_i+\Delta s/2)-g(s_{k+1},\alpha_i-\Delta s/2)\\ &\quad+g(s_{k+1},\alpha_i-\Delta s/2)-g(s_k,\alpha_i-\Delta s/2)\\ &\quad+g(s_{k+1},\alpha_i+\Delta s/2)-g(s_k,\alpha_i+\Delta s/2)\\ &\quad+g(s_k,\alpha_i+\Delta s/2)-g(s_k,\alpha_i-\Delta s/2)]\end{aligned}$$

and therefore it can viewed as an approximation for (2.4) by

$$\begin{aligned}\frac{\partial g}{\partial s}(s_{k+1/2},\alpha_i) &\approx \frac{1}{2\Delta s}[g(s_{k+1},\alpha_i+\Delta s/2)-g(s_k,\alpha_i+\Delta s/2)\\ &\quad+g(s_{k+1},\alpha_i-\Delta s/2)-g(s_k,\alpha_i-\Delta s/2)]\\ \frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_i) &\approx \frac{1}{2\Delta s}[g(s_{k+1},\alpha_i+\Delta s/2)-g(s_{k+1},\alpha_i-\Delta s/2)\\ &\quad+g(s_k,\alpha_i+\Delta s/2)-g(s_k,\alpha_i-\Delta s/2)].\end{aligned} \tag{2.20}$$

We see that the partial with respect to α is discretized with step size $\Delta s \gg \Delta\alpha$ instead of $\Delta\alpha$. Thus this natural choice for the discretization is the least effective method discussed thus far because of the step size for s used for approximating the partial with respect to α.

We note that Noo et al distinguish between direct schemes, schemes defined in terms of $\boldsymbol{\theta}$, and chain-rule based schemes, those based on (2.4). The above decomposition of M0 allow us to interpret M0 within the framework of chain-rule based schemes and thereby allow for a direct comparison. We are motivated to provide a similar analysis of M5 by studying our decomposition of M5 in the local detector coordinate system.

Following the appearance of [19], M3 was considered the standard method for the discretization of (2.1). Method M3 is more accurate than M0 and M1 but it shares an undesirable feature with M0 in the partial derivative with respect to α. The derivative is the average of derivatives computed at $s = s_k$ and $s = s_{k+1}$. This suggests the reconstruction by M3 is the average of two slightly rotated images and we shall see such a result in Figure 2.6 in Section 2.4. It was observed in [31] that M3 lead to distortions in the reconstruction that reduced the resolution of circular objects. In [8] it was suggested to use M4 because the derivative with respect to α is calculated directly at $s = s_k$ and

retains a short step size in α compared to M1. The compromise made in M4 is a larger step size in s. The results presented by Faridani et al showed empirically that M4 is the preferred method for discretizing (2.1). We will develop rationale to justify the increased performance of M4 over M3 in the following section.

2.2 Analysis of M0, M3 and M4

We begin by presenting estimates for the discretization of methods M0, M3 and M4. The estimates for M1 follow directly from the standard estimates on central difference quotients [1].

Lemma 2.1. *Let $g \in C^4(\mathbb{R})$, $\alpha_i = i\Delta\alpha$ and $\alpha_i < t < \alpha_{i+1}$ then*

$$g(t) - Ig(t) = -\frac{(1-c)c\Delta\alpha^2}{2}g''(t) - \frac{(2c^3 - 3c^2 + c)\Delta\alpha^3}{6}g'''(t) + O(\Delta\alpha^4)$$

where $c = (t - \alpha_i)/\Delta\alpha$ and $Ig(t) = (1-c)g(\alpha_i) + cg(\alpha_{i+1})$.

Lemma 2.2. *(Error for method M0) Let $g(s,\alpha)$ be C^4 on $\mathbb{R} \times [-\pi, \pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. Then*

$$\begin{aligned}
&\frac{\partial g}{\partial \alpha}(s_{k+1/2}, \alpha_i) + \frac{\partial g}{\partial s}(s_{k+1/2}, \alpha_i) - \frac{Ig(s_{k+1}, \alpha_i + \Delta s/2) - Ig(s_k, \alpha_i - \Delta s/2)}{\Delta s} \\
&\quad = -c(1-c)\Delta\alpha^2 \frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2}, \alpha_i) - c(1-c)\Delta\alpha^2 \frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2}, \alpha_i) \\
&\qquad - \frac{\Delta s^2}{24}\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2}, \alpha_i) - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2}, \alpha_i) \\
&\qquad - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial \alpha \partial s^2}(s_{k+1/2}, \alpha_i) - \frac{\Delta s^2}{24}\frac{\partial^3 g}{\partial s^3}(s_{k+1/2}, \alpha_i) + O(\Delta s^3 + \Delta\alpha^3)
\end{aligned}$$

with

$$\begin{aligned}
Ig(s_{k+1}, \alpha_i + \Delta s/2) &= (1-c)g(s_{k+1}, \alpha_{i+J}) + cg(s_{k+1}, \alpha_{i+J+1}) \\
Ig(s_k, \alpha_i + \Delta s/2) &= cg(s_k, \alpha_{i-J-1}) + (1-c)g(s_k, \alpha_{i-J})
\end{aligned}$$

where $c = (\alpha_i + \Delta s/2 - \alpha_{i+J})/\Delta\alpha$ and J is such that $\alpha_{i+J} \le \alpha_i + \Delta s/2 < \alpha_{i+J+1}$.

Proof. Let $\alpha_i = i\Delta\alpha$ and $\alpha_{i+J} \le \alpha_i + \Delta s/2 < \alpha_{i+J+1}$. Then $\alpha_i + \Delta s/2 = (1-c)\alpha_{i+J} + c\alpha_{i+J+1}$ and $c = (\alpha_i + \Delta s/2 - \alpha_{i+J})/\Delta\alpha$. Likewise $\alpha_i - \Delta s/2 = c\alpha_{i-J-1} + (1-c)\alpha_{i-J}$.

The error for the linear interpolation in α is

$$\begin{aligned}
&g(s_{k+1},\alpha_i+\Delta s/2)-(1-c)g(s_{k+1},\alpha_{i+J})-cg(s_{k+1},\alpha_{i+J+1})\\
&\quad=-c(1-c)\Delta\alpha^2\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},\alpha_i+\Delta s/2)-\frac{c-3c^2+2c^2}{6}\Delta\alpha^3\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1},\alpha_i+\Delta s/2)\\
&\qquad+O(\Delta\alpha^4)
\end{aligned}$$

and

$$\begin{aligned}
&g(s_k,\alpha_i-\Delta s/2)-cg(s_k,\alpha_{i-J-1})-(1-c)g(s_k,\alpha_{i-J})\\
&\quad=-c(1-c)\Delta\alpha^2\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_i-\Delta s/2)-\frac{c-3c^2+2c^2}{6}\Delta\alpha^3\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_i-\Delta s/2)\\
&\qquad+O(\Delta\alpha^4).
\end{aligned}$$

Since $c\Delta\alpha=\alpha_i+\Delta s/2-\alpha_{i+J}$ and $\alpha_{i+J}\le\alpha_i+\Delta s/2<\alpha_{i+J+1}$ we have $c(1-c)\Delta\alpha^2=O(\Delta\alpha^2)$. We shall now show the difference of the errors from the interpolation in α is $O(\Delta\alpha^2\Delta s)$. The 2D Taylor series of the error terms are

$$\begin{aligned}
\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},\alpha_i+\Delta s/2)&=\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1/2},\alpha_i)+\frac{\Delta s}{2}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1/2},\alpha_i)+\frac{\Delta s}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_{k+1/2},\alpha_i)\\
&\quad+\frac{\Delta s^2}{8}\frac{\partial^4 g}{\partial\alpha^4}(s_{k+1/2},\alpha_i)+\frac{\Delta s^2}{4}\frac{\partial^4 g}{\partial\alpha^3 s}(s_{k+1/2},\alpha_i)\\
&\quad+\frac{\Delta s^2}{8}\frac{\partial^4 g}{\partial\alpha^2\partial s^2}(s_{k+1/2},\alpha_i)+O(\Delta s^3)\\
\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_i-\Delta s/2)&=\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1/2},\alpha_i)-\frac{\Delta s}{2}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1/2},\alpha_i)-\frac{\Delta s}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_{k+1/2},\alpha_i)\\
&\quad+\frac{\Delta s^2}{8}\frac{\partial^4 g}{\partial\alpha^4}(s_{k+1/2},\alpha_i)+\frac{\Delta s^2}{4}\frac{\partial^4 g}{\partial\alpha^3 s}(s_{k+1/2},\alpha_i)\\
&\quad+\frac{\Delta s^2}{8}\frac{\partial^4 g}{\partial\alpha^2\partial s^2}(s_{k+1/2},\alpha_i)+O(\Delta s^3).
\end{aligned}$$

If we take the difference of the two second order error terms we have

$$\begin{aligned}
&\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},\alpha_i+\Delta s/2)-\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_i-\Delta s/2)\\
&\quad=\Delta s\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1/2},\alpha_i)+\Delta s\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_{k+1/2},\alpha_i)+O(\Delta s^3)
\end{aligned}$$

and for the third order error terms we have

$$\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1}, \alpha_i + \Delta s/2) - \frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_i - \Delta s/2)$$
$$= \Delta s \frac{\partial^4 g}{\partial \alpha^4}(s_{k+1/2}, \alpha_i) + \Delta s \frac{\partial^4 g}{\partial \alpha^3 \partial s}(s_{k+1/2}, \alpha_i) + O(\Delta s^3).$$

This also implies

$$-\frac{c - 3c^2 + 2c^2}{6}\Delta\alpha^3 \left(\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1}, \alpha_i + \Delta s/2) - \frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_i - \Delta s/2) \right) = O(\Delta\alpha^3 \Delta s).$$

Therefore

$$\begin{aligned}
&\frac{1}{\Delta s}[g(s_{k+1}, \alpha_i + \Delta s/2) - (1-c)g(s_{k+1}, \alpha_{i+J}) - cg(s_{k+1}, \alpha_{i+J+1}) \\
&\quad - g(s_k, \alpha_i - \Delta s/2) + cg(s_k, \alpha_{i-J-1}) + (1-c)g(s_k, \alpha_{i-J})] \\
&= -c(1-c)\Delta\alpha^2 \frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2}, \alpha_i) - c(1-c)\Delta\alpha^2 \frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2}, \alpha_i) + O(\Delta s^3 + \Delta\alpha^3).
\end{aligned} \tag{2.21}$$

Next we estimate the following error

$$\frac{\partial g}{\partial \alpha}(s_{k+1/2}, \alpha_i) + \frac{\partial g}{\partial s}(s_{k+1/2}, \alpha_i) - \frac{g(s_{k+1}, \alpha_i + \Delta s/2) - g(s_k, \alpha_i - \Delta s/2)}{\Delta s}$$

with the Taylor series

$$\begin{aligned}
g(s_{k+1}, \alpha_i + \Delta s/2) = g(s_{k+1/2}, \alpha_i) &+ \frac{\Delta s}{2}\frac{\partial g}{\partial \alpha}(s_{k+1/2}, \alpha_i) + \frac{\Delta s}{2}\frac{\partial g}{\partial \alpha}(s_{k+1/2}, \alpha_i) \\
&+ \frac{\Delta s^2}{8}\frac{\partial^2 g}{\partial \alpha^2}(s_{k+1/2}, \alpha_i) + \frac{\Delta s^2}{4}\frac{\partial^2 g}{\partial \alpha \partial s}(s_{k+1/2}, \alpha_i) \\
&+ \frac{\Delta s^2}{8}\frac{\partial^2 g}{\partial s^2}(s_{k+1/2}, \alpha_i) + \frac{\Delta s^3}{48}\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2}, \alpha_i) \\
&+ \frac{\Delta s^3}{16}\frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2}, \alpha_i) + \frac{\Delta s^3}{16}\frac{\partial^3 g}{\partial \alpha \partial s^2}(s_{k+1/2}, \alpha_i) \\
&+ \frac{\Delta s^3}{48}\frac{\partial^3 g}{\partial s^3}(s_{k+1/2}, \alpha_i) + O(\Delta s^4)
\end{aligned}$$

and

$$\begin{aligned} g(s_{k-1},\alpha_i-\Delta s/2) = g(s_{k+1/2},\alpha_i) &- \frac{\Delta s}{2}\frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_i) - \frac{\Delta s}{2}\frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_i) \\ &+ \frac{\Delta s^2}{8}\frac{\partial^2 g}{\partial \alpha^2}(s_{k+1/2},\alpha_i) + \frac{\Delta s^2}{4}\frac{\partial^2 g}{\partial \alpha \partial s}(s_{k+1/2},\alpha_i) \\ &+ \frac{\Delta s^2}{8}\frac{\partial^2 g}{\partial s^2}(s_{k+1/2},\alpha_i) - \frac{\Delta s^3}{48}\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2},\alpha_i) \\ &- \frac{\Delta s^3}{16}\frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2},\alpha_i) - \frac{\Delta s^3}{16}\frac{\partial^3 g}{\partial \alpha \partial s^2}(s_{k+1/2},\alpha_i) \\ &- \frac{\Delta s^3}{48}\frac{\partial^3 g}{\partial s^3}(s_{k+1/2},\alpha_i) + O(\Delta s^4). \end{aligned}$$

Thus

$$\begin{aligned} &\frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_i) + \frac{\partial g}{\partial s}(s_{k+1/2},\alpha_i) - \frac{g(s_{k+1},\alpha_i+\Delta s/2) - g(s_k,\alpha_i-\Delta s/2)}{\Delta s} \\ &\quad = -\frac{\Delta s^2}{24}\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2},\alpha_i) - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2},\alpha_i) \\ &\qquad - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial \alpha \partial s^2}(s_{k+1/2},\alpha_i) - \frac{\Delta s^2}{24}\frac{\partial^3 g}{\partial s^3}(s_{k+1/2},\alpha_i) + O(\Delta s^3). \end{aligned} \tag{2.22}$$

By our bounds (2.21) and (2.22) we conclude

$$\begin{aligned} &\frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_i) + \frac{\partial g}{\partial s}(s_{k+1/2},\alpha_i) \\ &\quad - \frac{1}{\Delta s}[(1-c)g(s_{k+1},\alpha_{i+J}) - cg(s_{k+1},\alpha_{i+J+1}) - cg(s_k,\alpha_{i-J-1}) + (1-c)g(s_k,\alpha_{i-J})] \\ &= -c(1-c)\Delta\alpha^2\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2},\alpha_i) - c(1-c)\Delta\alpha^2\frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2},\alpha_i) \\ &\quad - \frac{\Delta s^2}{24}\frac{\partial^3 g}{\partial \alpha^3}(s_{k+1/2},\alpha_i) - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_{k+1/2},\alpha_i) \\ &\quad - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial \alpha \partial s^2}(s_{k+1/2},\alpha_i) - \frac{\Delta s^2}{24}\frac{\partial^3 g}{\partial s^3}(s_{k+1/2},\alpha_i) + O(\Delta s^3 + \Delta\alpha^3). \end{aligned}$$

□

The techniques we use in the proof of Lemma 2.2 will be used throughout this Chapter. We will use similar estimates on the errors from linear interpolation in α for M5

because this method also uses linear interpolation in α. Let us now state and prove the error estimate for M3.

Lemma 2.3. *(Error for method M3) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. Then*

$$\begin{aligned}
&\frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_{i+1/2}) - \frac{g(s_k,\alpha_{i+1}) - g(s_k,\alpha_i) + g(s_{k+1},\alpha_{i+1}) - g(s_{k+1},\alpha_i)}{2\Delta\alpha} \\
&\quad = -\frac{\Delta\alpha^2}{24}\frac{\partial^3}{\partial\alpha^3}g(s_{k+1/2},\alpha_{i+1/2}) - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_{k+1/2},\alpha_{i+1/2}) \\
&\qquad + O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3)
\end{aligned}$$

and

$$\begin{aligned}
&\frac{\partial g}{\partial s}(s_{k+1/2},\alpha_{i+1/2}) - \frac{g(s_{k+1},\alpha_{i+1}) - g(s_k,\alpha_{i+1}) + g(s_{k+1},\alpha_i) - g(s_k,\alpha_i)}{2\Delta s} \\
&\quad = -\frac{\Delta s^2}{24}\frac{\partial^3}{\partial s^3}g(s_{k+1/2},\alpha_{i+1/2}) - \frac{\Delta\alpha^2}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_{k+1/2},\alpha_{i+1/2}) \\
&\qquad + O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3).
\end{aligned}$$

Proof. We shall show the result for the partial derivative with respect to α. The other error estimate follows by a similar calculation. We use the following approximation

$$\begin{aligned}
g(s_{k+1},\alpha_{i+1}) = g(s_{k+1/2},\alpha_{i+1/2}) &+ \frac{\Delta\alpha}{2}\frac{\partial g}{\partial\alpha}(s_{k+1/2},\alpha_{i+1/2}) + \frac{\Delta s}{2}\frac{\partial g}{\partial s}(s_{k+1/2},\alpha_{i+1/2}) \\
&+ \frac{\Delta\alpha^2}{8}\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1/2},\alpha_{i+1/2}) + \frac{\Delta\alpha\Delta s}{4}\frac{\partial^2 g}{\partial\alpha\partial s}(s_{k+1/2},\alpha_{i+1/2}) \\
&+ \frac{\Delta s^2}{8}\frac{\partial^2 g}{\partial s^2}(s_{k+1/2},\alpha_{i+1/2}) + \frac{\Delta\alpha^3}{48}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1/2},\alpha_{i+1/2}) \\
&+ \frac{\Delta\alpha^2\Delta s}{16}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_{k+1/2},\alpha_{i+1/2}) + \frac{\Delta\alpha\Delta s^2}{16}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_{k+1/2},\alpha_{i+1/2}) \\
&+ \frac{\Delta s^3}{48}\frac{\partial^3 g}{\partial s^3}(s_{k+1/2},\alpha_{i+1/2}) \\
&+ O(\Delta\alpha^4 + \Delta\alpha^3\Delta s + \Delta\alpha^2\Delta s^2 + \Delta\alpha\Delta s^3 + \Delta s^4)
\end{aligned}$$

and therefore

$$\begin{aligned} g(s_{k+1},\alpha_{i+1}) - g(s_{k+1},\alpha_i) = \Delta\alpha \frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_{i+1/2}) + \frac{\Delta\alpha\Delta s}{2}\frac{\partial^2 g}{\partial\alpha\partial s}(s_{k+1/2},\alpha_{i+1/2}) \\ + \frac{\Delta\alpha^3}{24}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1/2},\alpha_{i+1/2}) \\ + \frac{\Delta\alpha\Delta s^2}{8}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_{k+1/2},\alpha_{i+1/2}) \\ + O(\Delta\alpha^4 + \Delta\alpha^3\Delta s + \Delta\alpha^2\Delta s^2 + \Delta\alpha\Delta s^3 + \Delta s^4). \end{aligned}$$

By a similar argument we have

$$\begin{aligned} g(s_k,\alpha_{i+1}) - g(s_k,\alpha_i) = \Delta\alpha \frac{\partial g}{\partial \alpha}(s_{k+1/2},\alpha_{i+1/2}) - \frac{\Delta\alpha\Delta s}{2}\frac{\partial^2 g}{\partial\alpha\partial s}(s_{k+1/2},\alpha_{i+1/2}) \\ + \frac{\Delta\alpha^3}{24}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1/2},\alpha_{i+1/2}) + \frac{\Delta\alpha\Delta s^2}{8}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_{k+1/2},\alpha_{i+1/2}) \\ + O(\Delta\alpha^4 + \Delta\alpha^3\Delta s + \Delta\alpha^2\Delta s^2 + \Delta\alpha\Delta s^3 + \Delta s^4). \end{aligned}$$

We have shown

$$\begin{aligned} \frac{\partial g}{\partial\alpha}(s_{k+1/2},\alpha_{i+1/2}) - \frac{g(s_k,\alpha_{i+1}) - g(s_k,\alpha_i) + g(s_{k+1},\alpha_{i+1}) - g(s_{k+1},\alpha_i)}{2\Delta\alpha} \\ = -\frac{\Delta\alpha^2}{24}\frac{\partial^3}{\partial\alpha^3} g(s_{k+1/2},\alpha_{i+1/2}) - \frac{\Delta s^2}{8}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_{k+1/2},\alpha_{i+1/2}) \\ + O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3). \end{aligned}$$

□

We now present our estimate for M4.

Lemma 2.4. *(Error for method M4) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. Then*

$$\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) - \frac{1}{\Delta\alpha}[g(s_k,\alpha_{i+1}) - g(s_k,\alpha_i)] = -\frac{\Delta\alpha^2}{24}\frac{\partial^3}{\partial\alpha^3} g(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^3)$$

and

$$\begin{aligned}
&\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) - \frac{1}{4\Delta s}[g(s_{k+1},\alpha_i) - g(s_{k-1},\alpha_i) + g(s_{k+1},\alpha_{i+1}) - g(s_{k-1},\alpha_{i+1})] \\
&\quad = -\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) - \frac{\Delta\alpha^2}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2}) \\
&\qquad + O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3).
\end{aligned}$$

Proof. The error term for the derivative with respect to α follows directly from the central difference scheme error estimate [1]

$$\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) - \frac{1}{\Delta\alpha}[g(s_k,\alpha_{i+1}) - g(s_k,\alpha_i)] = -\frac{\Delta\alpha^2}{24}\frac{\partial^3}{\partial\alpha^3}g(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^3).$$

For the derivative with respect to s we again use a Taylor series estimate

$$\begin{aligned}
g(s_{k+1},\alpha_{i+1}) = g(s_k,\alpha_{i+1/2}) &+ \frac{\Delta\alpha}{2}\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) + \Delta s\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) \\
&+ \frac{\Delta\alpha^2}{8}\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2}) + \frac{\Delta\alpha\Delta s}{2}\frac{\partial^2 g}{\partial\alpha\partial s}(s_k,\alpha_{i+1/2}) \\
&+ \frac{\Delta s^2}{2}\frac{\partial^2 g}{\partial s^2}(s_k,\alpha_{i+1/2}) + \frac{\Delta\alpha^3}{48}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \\
&+ \frac{\Delta\alpha^2\Delta s}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2}) + \frac{\Delta\alpha\Delta s^2}{4}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2}) \\
&+ \frac{\Delta s^3}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^4 + \Delta\alpha^3\Delta s + \Delta\alpha^2\Delta s^2 + \Delta\alpha\Delta s^3 + \Delta s^4)
\end{aligned}$$

and likewise for $g(s_{k+1},\alpha_i), g(s_{k-1},\alpha_i)$ and $g(s_{k-1},\alpha_{i+1})$. We then have

$$\begin{aligned}
g(s_{k+1},\alpha_i) - g(s_{k-1},\alpha_i) = 2\Delta s\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) &- \Delta\alpha\Delta s\frac{\partial^2 g}{\partial\alpha\partial s}(s_k,\alpha_{i+1/2}) \\
&+ \frac{\Delta\alpha^2\Delta s}{4}\frac{\partial^3 g}{\partial s}(s_k,\alpha_{i+1/2}) + \frac{\Delta s^3}{3}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) \\
&+ O(\Delta\alpha^4 + \Delta\alpha^3\Delta s + \Delta\alpha^2\Delta s^2 + \Delta\alpha\Delta s^3 + \Delta s^4)
\end{aligned}$$

and

$$g(s_{k+1},\alpha_{i+1}) - g(s_{k-1},\alpha_{i+1}) = 2\Delta s\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) + \Delta\alpha\Delta s\frac{\partial^2 g}{\partial\alpha\partial s}(s_k,\alpha_{i+1/2}) + \frac{\Delta\alpha^2\Delta s}{4}\frac{\partial^3 g}{\partial s}(s_k,\alpha_{i+1/2}) + \frac{\Delta s^3}{3}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^4 + \Delta\alpha^3\Delta s + \Delta\alpha^2\Delta s^2 + \Delta\alpha\Delta s^3 + \Delta s^4).$$

Hence

$$\begin{aligned}&\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) - \frac{1}{4\Delta s}[g(s_{k+1},\alpha_i) - g(s_{k-1},\alpha_i) + g(s_{k+1},\alpha_{i+1}) - g(s_{k-1},\alpha_{i+1})]\\ &\quad= -\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) - \frac{\Delta\alpha^2}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})\\ &\qquad+ O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3).\end{aligned} \tag{2.23}$$

□

Let us decompose f into the sum of f_s and f_α where

$$f_\alpha(\mathbf{x}) = \frac{1}{2\pi^2}\int_{I_\pi(\mathbf{x})}\frac{1}{|\mathbf{x}-\mathbf{y}(s)|}\int_0^{2\pi}\frac{\partial g}{\partial\alpha}(s,\alpha)\frac{1}{\sin(\alpha^*-\alpha)}\,d\alpha\,ds \tag{2.24}$$

and

$$f_s(\mathbf{x}) = \frac{1}{2\pi^2}\int_{I_\pi(\mathbf{x})}\frac{1}{|\mathbf{x}-\mathbf{y}(s)|}\int_0^{2\pi}\frac{\partial g}{\partial s}(s,\alpha)\frac{1}{\sin(\alpha^*-\alpha)}\,d\alpha\,ds. \tag{2.25}$$

A comparison of error terms of M3 and M4 is now possible. Empirical evidence from the phantoms we have studied in Section 2.4 suggest the larger of the two terms (2.24) and (2.25) is f_α. Thus we hypothesize the largest contribution of the discretization error of (2.4) is the partial with respect to α. Under this assumption we see that the error term from interpolation in s for M3 is not found in M4. The remaining error term for the partial with respect to α are identical for M3 and M4. Therefore we can use this reasoning to justify the greater resolution in numerical reconstructions from M4 compared to M3.

2.3 Analysis of M5

Now we shall focus on deriving our estimate for M5. We will show that $d_s + d_\varepsilon$ behaves as a central difference scheme with respect to s and d_α behaves as a central difference scheme with respect to α. We have the following result on the discretization errors in M5.

Theorem 2.5. *(Error for method M5) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. If*

$$0 < \varepsilon \le \frac{\Delta\alpha - \Delta s^2 M}{2\Delta s}$$

where

$$M = \max\left(\sup_{t'\in(0,\Delta s)}\left|\frac{\partial^2 h}{\partial u^2}(t',\alpha_{i+1/2},\varepsilon)\right|, \sup_{t'\in(0,\varepsilon\Delta s)}\left|\frac{\partial^2 \nu}{\partial u^2}(t',\alpha_{i+1/2})\right|\right)$$

then

$$\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) + \frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) - d_s(s_k,\alpha_{i+1/2}) - (1-\varepsilon)d_\alpha(s_k,\alpha_{i+1/2}) = \sum_{i=1}^{7} E_i$$

where

$$\begin{aligned}
E_1 &= \frac{\Delta s^2}{4}\big((\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2}) \\
&\quad + 2(1-\varepsilon)\varepsilon^2\big)\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) \\
E_2 &= \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha\partial s}(s_k,\alpha_{i+1/2})
\end{aligned}$$

and

$$\begin{aligned}
E_3 &= -\frac{\Delta\alpha^2}{24}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \\
E_4 &= -\frac{\Delta\alpha^2}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2}) \\
E_5 &= -\varepsilon\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2}) \\
E_6 &= -\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) \\
E_7 &= O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3 + \varepsilon^3\Delta s^3).
\end{aligned}$$

Lemma 2.6. *Suppose h and ν are defined by* (2.11) *and* (2.16). *Let $\alpha_i = i\Delta\alpha$, $\Delta s > 0$ and $\varepsilon > 0$ then*

$$\begin{aligned} h(\Delta s, \alpha_{i+1/2}, \varepsilon) = \alpha_{i+1/2} + \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) \\ - \frac{\Delta s^3}{4}(\varepsilon-1)^2\sec^2(\alpha_{i+1/2})((\varepsilon-1)\cos(2\alpha_{i+1/2}) + \varepsilon + 1) + O(\Delta s^4) \end{aligned}$$

and

$$\nu(\varepsilon\Delta s, \alpha_{i+1/2}) = \alpha_{i+1/2} + \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \frac{(\varepsilon\Delta s)^3}{2} + O(\varepsilon^4\Delta s^4).$$

Proof. A tedious but routine calculation shows

$$\begin{aligned} h(\Delta s, \alpha_{i+1/2}, \varepsilon) &= h(0, \alpha_{i+1/2}, \varepsilon) + \Delta s\frac{\partial h}{\partial u}(0, \alpha_{i+1/2}, \varepsilon) + \frac{\Delta s^2}{2}\frac{\partial^2 h}{\partial u^2}(0, \alpha_{i+1/2}, \varepsilon) \\ &\quad + \frac{\Delta s^3}{6}\frac{\partial^3 h}{\partial u^3}(0, \alpha_{i+1/2}, \varepsilon) + O(\Delta s^4) \\ &= \alpha_{i+1/2} + \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) \\ &\quad - \frac{\Delta s^3}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2}) + O(\Delta s^4) \\ h(-\Delta s, \alpha_{i+1/2}, \varepsilon) &= \alpha_{i+1/2} - \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) \\ &\quad + \frac{\Delta s^3}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2}) + O(\Delta s^4). \end{aligned}$$

For the term ν we have

$$\begin{aligned} \nu(\varepsilon\Delta s, \alpha_{i+1/2}) &= \nu(\varepsilon\Delta s, \alpha_{i+1/2}) + \varepsilon\Delta s\frac{\partial\nu}{\partial u}(0, \alpha_{i+1/2}) + \frac{(\varepsilon\Delta s)^2}{2}\frac{\partial^2\nu}{\partial u^2}(0, \alpha_{i+1/2}) \\ &\quad + \frac{(\varepsilon\Delta s)^3}{6}\frac{\partial^3\nu}{\partial u^3}(0, \alpha_{i+1/2}) + O(\varepsilon^4\Delta s^4) \\ &= \alpha_{i+1/2} + \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \frac{(\varepsilon\Delta s)^3}{2} + O(\varepsilon^4\Delta s^4) \\ \nu(-\varepsilon\Delta s, \alpha_{i+1/2}) &= \alpha_{i+1/2} - \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) + \frac{(\varepsilon\Delta s)^3}{2} + O(\varepsilon^4\Delta s^4). \end{aligned}$$

□

In M4 we calculate the derivative at the point $(s_k, \alpha_{i+1/2})$. Suppose we calculate M5 over the same grid as M4. In [18] the authors state a geometrical reason to justify the calculation of M5 with this half detector shift in α, *"We apply this shift so that the scheme closely resembles a difference between consecutive detector samples when ε tends to zero and $\mathbf{y}(s)$ is far from the object."* We can give another justification for this shift in α if we use the results of Lemma 2.6 and choose a specific ε.

Suppose $\varepsilon = \Delta\alpha/(2\Delta s)$, which is slightly larger than is required by Theorem 2.5. Then by the Taylor series in Lemma 2.6 we know

$$h(\Delta s, \alpha_{i+1/2}, \Delta\alpha/(2\Delta s)) = \alpha_{i+1/2} + \Delta s \frac{\Delta\alpha}{2\Delta s} + O(\Delta\alpha^2) = \alpha_{i+1} + O(\Delta\alpha^2)$$

and

$$h(-\Delta s, \alpha_{i+1/2}, \Delta\alpha/(2\Delta s)) = \alpha_i + O(\Delta\alpha^2).$$

Therefore the terms in d_s are approximately the difference of two adjacent detectors. Similar reasoning can be applied the terms in d_α using the Taylor series of ν.

Let us investigate the idea of the shift in α further and determine what would occur if we calculated $g(s, \nu(\pm\Delta s, \alpha_i))$ instead of $g(s, \nu(\pm\Delta s, \alpha_{i+1/2}))$. By Lemma 2.6 we know $\nu(\Delta\alpha/2, \alpha_i) \approx \alpha_{i+1/2}$ and $\nu(-\Delta\alpha/2, \alpha_i) \approx \alpha_{i-1/2}$. We would then need the following linear approximations

$$\begin{aligned} g(s, \nu(\Delta\alpha/2, \alpha_i)) &= \frac{1}{2}(g(s, \alpha_i) + g(s, \alpha_{i+1})) \\ g(s, \nu(-\Delta\alpha/2, \alpha_i)) &= \frac{1}{2}(g(s, \alpha_{i-1}) + g(s, \alpha_i)) \end{aligned}$$

to approximate

$$\begin{aligned} d_\alpha(s, \alpha_i, \varepsilon) &= \frac{Ig(s, \nu(\Delta\alpha/2, \alpha_i)) - Ig(s, \nu(-\Delta\alpha/2, \alpha_i))}{2\varepsilon\Delta s} \\ &\approx \frac{g(s, \alpha_i) + g(s, \alpha_{i+1}) - g(s, \alpha_{i-1}) - g(s, \alpha_i)}{4\varepsilon\Delta s} \\ &= \frac{g(s, \alpha_{i+1}) - g(s, \alpha_{i-1})}{2\Delta\alpha}. \end{aligned}$$

Thus we are approximating d_α with a central difference scheme with step size $2\Delta\alpha$. If we compare this to the approximation with the half detector shift in α

$$d_\alpha(s, \alpha_{i+1/2}, \varepsilon) \approx \frac{g(s, \alpha_{i+1}) - g(s, \alpha_i)}{\Delta\alpha}$$

we find that the half detector shift is equivalent to a central difference scheme with a smaller stepsize than with no shift in α. Therefore the choice of the half detector shift will increase the accuracy of the approximation for d_α. We shall see that our choice of ε must be smaller than $\Delta\alpha/(2\Delta s)$ to get the necessary behavior for the interpolation in α.

Lemma 2.7. *If*

$$0 < \varepsilon \leq \frac{\Delta\alpha - \Delta s^2 M}{2\Delta s}$$

where

$$M = \max\left(\sup_{t' \in (0, \Delta s)} \left| \frac{\partial^2 h}{\partial u^2}(t', \alpha_{i+1/2}, \varepsilon) \right|, \sup_{t' \in (0, \varepsilon\Delta s)} \left| \frac{\partial^2 \nu}{\partial u^2}(t', \alpha_{i+1/2}) \right| \right)$$

then

$$\alpha_i < h(\pm\Delta s, \alpha_{i+1/2}, \varepsilon) < \alpha_{i+1}$$
$$\alpha_i < \nu(\pm\varepsilon\Delta s, \alpha_{i+1/2}) < \alpha_{i+1}.$$

Proof. From Lemma 2.6 we have

$$h(\Delta s, \alpha_{i+1/2}, \varepsilon) = \alpha_{i+1/2} + \varepsilon\Delta s + \frac{\Delta s^2}{2} \frac{\partial^2 h}{\partial u^2}(t, \alpha_{i+1/2}, \varepsilon)$$

where $t \in (0, \Delta s)$. Now we want

$$\left| \varepsilon\Delta s + \frac{\Delta s^2}{2} \frac{\partial^2 h}{\partial u^2}(t, \alpha_{i+1/2}, \varepsilon) \right| \leq \Delta\alpha/2$$

and therefore

$$\varepsilon \leq \frac{1}{2\Delta s} \left(\Delta\alpha - \Delta s^2 M \right) \tag{2.26}$$

where

$$M = \sup_{t' \in (0, \Delta s)} \left| \frac{\partial^2 h}{\partial u^2}(t', \alpha_{i+1/2}, \varepsilon) \right|.$$

The remaining bounds follow by a similar argument. □

If $\Delta\alpha - \Delta s^2 M < 0$ then the analysis of Theorem 2.5 will not hold for our estimates on the errors in the linear interpolation. If this condition is violated the interpolation in α may not satisfy

$$\begin{aligned} \alpha_i < h(\pm\Delta s, \alpha_{i+1/2}, \varepsilon) < \alpha_{i+1} \\ \alpha_i < \nu(\pm\varepsilon\Delta s, \alpha_{i+1/2}) < \alpha_{i+1}. \end{aligned} \tag{2.27}$$

If the bounds in (2.27) do not hold then the terms $h(\pm\Delta s, \alpha_{i+1/2}, \varepsilon)$ or $\nu(\pm\varepsilon\Delta s, \alpha_{i+1/2})$ will be located more than one detector width apart. This suggests that the calculations of d_α or d_s will be less accurate than if (2.27) is satisfied.

Now we shall demonstrate that M5 is a second order difference scheme. We begin by evaluating the errors in the difference schemes without considering the effects of the linear interpolation in α.

Lemma 2.8. *(Error for difference scheme d_s without interpolation) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. Then*

$$\begin{aligned} &\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) + \varepsilon\frac{\partial g}{\partial \alpha}(s_k,\alpha_{i+1/2}) - \frac{g(s+\Delta s, h(\Delta s,\alpha,\varepsilon)) - g(s-\Delta s, h(-\Delta s,\alpha,\varepsilon))}{2\Delta s} \\ &\quad = \frac{\Delta s^2}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2})\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) \\ &\qquad + \frac{\Delta s^2}{2}(\varepsilon-1)^2\varepsilon\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2}) \\ &\qquad + \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha\partial s}(s_k,\alpha_{i+1/2}) - \varepsilon^3\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \\ &\qquad - \varepsilon^2\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2}) - \varepsilon\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2}) - \frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}) \\ &\qquad + O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3). \end{aligned}$$

Proof. We shall derive the error estimates for d_s without incorporating the linear interpolation in α. We begin with Taylor series in both s and α for the terms in d_s

$$\begin{aligned}
&g(s_{k+1}, h(\Delta s, \alpha_{i+1/2}, \varepsilon)) \\
&\quad = g(s_k, \alpha_{i+1/2}) + (h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2}) \frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) \\
&\qquad + \Delta s \frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}) + \frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2}{2} \frac{\partial^2 g}{\partial \alpha^2}(s_k, \alpha_{i+1/2}) \\
&\qquad + (h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2}) \Delta s \frac{\partial^2 g}{\partial \alpha \partial s}(s_k, \alpha_{i+1/2}) \\
&\qquad + \frac{\Delta s^2}{2} \frac{\partial^2 g}{\partial s^2}(s_k, \alpha_{i+1/2}) + \frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^3}{6} \frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_{i+1/2}) \\
&\qquad + \frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2}{2} \Delta s \frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_k, \alpha_{i+1/2}) \\
&\qquad + \frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})}{2} \Delta s^2 \frac{\partial^3 g}{\partial \alpha \partial s^2}(s_k, \alpha_{i+1/2}) + \frac{\Delta s^3}{6} \frac{\partial^3 g}{\partial s^3}(s_k, \alpha_{i+1/2}) \\
&\qquad + O(\Delta\alpha^4 + \Delta\alpha^3 \Delta s + \Delta\alpha^2 \Delta s^2 + \Delta\alpha \Delta s^3 + \Delta s^4)
\end{aligned} \tag{2.28}$$

and likewise

$$\begin{aligned}
&g(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) \\
&\quad = g(s_k, \alpha_{i+1/2}) + (h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2}) \frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) \\
&\qquad - \Delta s \frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}) + \frac{(h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2}{2} \frac{\partial^2 g}{\partial \alpha^2}(s_k, \alpha_{i+1/2}) \\
&\qquad - (h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2}) \Delta s \frac{\partial^2 g}{\partial \alpha \partial s}(s_k, \alpha_{i+1/2}) \\
&\qquad + \frac{\Delta s^2}{2} \frac{\partial^2 g}{\partial s^2}(s_k, \alpha_{i+1/2}) + \frac{(h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^3}{6} \frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_{i+1/2}) \\
&\qquad - \frac{(h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2}{2} \Delta s \frac{\partial^3 g}{\partial \alpha^2 \partial s}(s_k, \alpha_{i+1/2}) \\
&\qquad + \frac{(h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2}{2} \Delta s^2 \frac{\partial^3 g}{\partial \alpha \partial s^2}(s_k, \alpha_{i+1/2}) - \frac{\Delta s^3}{6} \frac{\partial^3 g}{\partial s^3}(s_k, \alpha_{i+1/2}) \\
&\qquad + O(\Delta\alpha^4 + \Delta\alpha^3 \Delta s + \Delta\alpha^2 \Delta s^2 + \Delta\alpha \Delta s^3 + \Delta s^4).
\end{aligned} \tag{2.29}$$

By Lemma 2.6 we know

$$\begin{aligned} h(\Delta s, \alpha_{i+1/2}, \varepsilon) &= \alpha_{i+1/2} + \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) \\ &\quad - \frac{\Delta s^3}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2}) + O(\Delta s^4) \\ h(-\Delta s, \alpha_{i+1/2}, \varepsilon) &= \alpha_{i+1/2} - \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) \\ &\quad + \frac{\Delta s^3}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2}) + O(\Delta s^4). \end{aligned} \tag{2.30}$$

The first order derivative terms in $d_s(s_k, \alpha_{i+1/2})$ given by (2.28) and (2.29) are

$$\begin{aligned} &\frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2}) - (h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2}))}{2\Delta s}\frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) \\ &\quad = -\frac{\Delta s^2}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2})\frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) + O(\Delta s^3) \\ &\qquad + \varepsilon\frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) \end{aligned}$$

and

$$\frac{(\Delta s + \Delta s)}{2\Delta s}\frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}) = \frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}).$$

The second order derivatives terms are

$$\begin{aligned} &\frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2 - (h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})^2}{4\Delta s}\frac{\partial^2 g}{\partial \alpha^2}(s_k, \alpha_{i+1/2}) \\ &\quad = -\frac{\Delta s^2}{2}(\varepsilon-1)^2\varepsilon\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha^2}(s_k, \alpha_{i+1/2}) + O(\Delta s^4), \end{aligned}$$

along with

$$\begin{aligned} &\frac{(h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})\Delta s + (h(-\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_{i+1/2})\Delta s}{2\Delta s}\frac{\partial^2 g}{\partial \alpha \partial s}(s_k, \alpha_{i+1/2}) \\ &\quad = -\frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha \partial s}(s_k, \alpha_{i+1/2}) + O(\Delta s^4) \end{aligned}$$

and

$$\frac{\Delta s^2 - \Delta s^2}{4\Delta s}\frac{\partial^2 g}{\partial s^2}(s_k, \alpha_{i+1/2}) = 0.$$

The third order derivative terms are

$$\frac{(h(\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})^3-(h(-\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})$$
$$=\varepsilon^3\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+O(\Delta s^3),$$

with

$$\frac{(h(\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})^2\Delta s+(h(-\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})^2\Delta s}{4\Delta s}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})$$
$$=\varepsilon^2\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})+O(\Delta s^4),$$

along with

$$\frac{(h(\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})\Delta s^2-(h(-\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})\Delta s^2}{4\Delta s}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2})$$
$$=\varepsilon\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2})+O(\Delta s^4)$$

and

$$\frac{\Delta s^3+\Delta s^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})=\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}).$$

Thus

$$\begin{aligned}
&\frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2})+\varepsilon\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})\\
&\quad-\frac{g(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon))-g(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon))}{2\Delta s}\\
&=\frac{\Delta s^2}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2})\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})\\
&\quad+\frac{\Delta s^2}{2}(\varepsilon-1)^2\varepsilon\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\
&\quad+\frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha\partial s}(s_k,\alpha_{i+1/2})-\varepsilon^3\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad-\varepsilon^2\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})-\varepsilon\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2})-\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2})\\
&\quad+O(\Delta\alpha^3+\Delta\alpha^2\Delta s+\Delta\alpha\Delta s^2+\Delta s^3).
\end{aligned}\tag{2.31}$$

□

Lemma 2.9. *(Error for difference scheme d_α without interpolation) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. Then*

$$\begin{aligned}\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) &- \frac{g(s,\nu(\varepsilon\Delta s,\alpha)) - g(s,\nu(-\varepsilon\Delta s,\alpha))}{2\varepsilon\Delta s}\\ &= \varepsilon^2\frac{\Delta s^2}{2}\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2}) + \varepsilon^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\ &\quad - \varepsilon^2\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) + O(\Delta s^3).\end{aligned}$$

Proof. We take the Taylor series of d_α in α

$$\begin{aligned}g(s_k,\nu(\varepsilon\Delta s,\alpha_{i+1/2})) &= g(s_k,\alpha_{i+1/2}) + (\nu(\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_{i+1/2})\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})\\ &\quad + \frac{(\nu(\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_{i+1/2})^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\ &\quad + \frac{(\nu(\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_{i+1/2})^3}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^4)\end{aligned} \tag{2.32}$$

and

$$\begin{aligned}g(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2})) &= g(s_k,\alpha_{i+1/2}) + (\nu(-\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_{i+1/2})\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})\\ &\quad + \frac{(\nu(-\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_{i+1/2})^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\ &\quad + \frac{(\nu(-\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_{i+1/2})^3}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^4).\end{aligned} \tag{2.33}$$

As shown in Lemma 2.6

$$\begin{aligned}\nu(\varepsilon\Delta s,\alpha_{i+1/2}) &= \alpha_{i+1/2} + \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \frac{(\varepsilon\Delta s)^3}{2} + O(\Delta s^4)\\ \nu(-\varepsilon\Delta s,\alpha_{i+1/2}) &= \alpha_{i+1/2} - \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) + \frac{(\varepsilon\Delta s)^3}{2} + O(\Delta s^4).\end{aligned}$$

Next we substitute our Taylor expansion for ν into our approximations of (2.32) and (2.33) and calculate the error terms of d_α

$$\begin{aligned}
&\frac{\nu(\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2}-(\nu(-\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2})}{2\varepsilon\Delta s}\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})\\
&\quad=\left(1-\varepsilon^2\frac{\Delta s^2}{2}\right)\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})+O(\Delta s^3)\\
&\frac{(\nu(\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2})^2-(\nu(-\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2})^2}{4\varepsilon\Delta s}\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\
&\quad=-\varepsilon^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})+O(\Delta s^3)\\
&\frac{(\nu(\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2})^3-(\nu(-\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2})^3}{12\varepsilon\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad=\varepsilon^2\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+O(\Delta s^3).
\end{aligned}$$

Therefore by

$$\begin{aligned}
&\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})-\frac{g(s_k,\nu(\varepsilon\Delta s,\alpha_{i+1/2}))-g(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2}))}{2\varepsilon\Delta s}\\
&\quad=\varepsilon^2\frac{\Delta s^2}{2}\frac{\partial g}{\partial\alpha}(s_k,\alpha_{i+1/2})+\frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\
&\qquad-\varepsilon^2\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+O(\Delta s^3).
\end{aligned}$$

□

Our next results are for differentiating the errors from using linear interpolation in α to calculate g at the fan angles given by h and ν. The proofs of the next two results depend on Lemma 2.7. We begin with interpolation in d_s.

Lemma 2.10. *(Error for linear interpolation in d_s) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k=k\Delta s$ and $\alpha_i=i\Delta\alpha$. If*

$$0<\varepsilon\le\frac{\Delta\alpha-\Delta s^2M}{2\Delta s}$$

where

$$M=\sup_{t'\in(0,\Delta s)}\left|\frac{\partial^2 h}{\partial u^2}(t',\alpha_{i+1/2},\varepsilon)\right|$$

then

$$\begin{aligned}
&\frac{g(s+\Delta s, h(\Delta s,\alpha,\varepsilon)) - g(s-\Delta s, h(-\Delta s,\alpha,\varepsilon))}{2\Delta s} - d_s(s_k,\alpha_{i+1/2},\varepsilon)\\
&= \left(-\frac{\Delta\alpha^2}{8} + \varepsilon^2\frac{\Delta s^2}{2}\right)\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2}) + \left(-\varepsilon\frac{\Delta\alpha^2}{24} + \varepsilon^3\frac{\Delta s^2}{6}\right)\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad - \varepsilon(1-\varepsilon)^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\
&\quad + O(\Delta\alpha^3 + \Delta\alpha^2\Delta s + \Delta\alpha\Delta s^2 + \Delta s^3).
\end{aligned}$$

Proof. Recall

$$d_s(s_k,\alpha_{i+1/2},\varepsilon) = \frac{Ig(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon)) - Ig(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon))}{2\Delta s}.$$

and let

$$\begin{aligned}
Ig(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon)) &= (1-c)g(s_{k+1},\alpha_i) + cg(s_{k+1},\alpha_{i+1})\\
Ig(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon)) &= (1-c_1)g(s_{k-1},\alpha_i) + c_1 g(s_{k-1},\alpha_{i+1})
\end{aligned}$$

where

$$c = \frac{h(\Delta s,\alpha_{i+1/2},\varepsilon) - \alpha_i}{\Delta\alpha} \tag{2.34}$$

$$c_1 = \frac{h(-\Delta s,\alpha_{i+1/2},\varepsilon) - \alpha_i}{\Delta\alpha}. \tag{2.35}$$

Now we shall focus on the effects of the linear interpolation in α. By Lemmas 2.7 and 2.1 we know $\alpha_i < h(\pm\Delta s,\alpha_{i+1/2},\varepsilon) < \alpha_{i+1}$

$$\begin{aligned}
&g(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon)) - (1-c)g(s_{k+1},\alpha_i) - cg(s_{k+1},\alpha_{i+1})\\
&\quad = -(1-c)c\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon))\\
&\qquad - (2c^3 - 3c^2 + c)\frac{\Delta\alpha^3}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon)) + O(\Delta\alpha^4)
\end{aligned} \tag{2.36}$$

and

$$\begin{aligned} & g(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) - (1-c_1)g(s_{k-1}, \alpha_i) - c_1 g(s_{k-1}, \alpha_{i+1}) \\ & \quad = -(1-c_1)c_1 \frac{\Delta\alpha^2}{2} \frac{\partial^2 g}{\partial \alpha^2}(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) \\ & \qquad - (2c_1^3 - 3c_1^2 + c_1)\frac{\Delta\alpha^3}{6}\frac{\partial^3 g}{\partial \alpha^3}(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) + O(\Delta\alpha^4). \end{aligned} \tag{2.37}$$

Our goal is to estimate

$$\begin{aligned} & \frac{1}{2\Delta s}[g(s_{k+1}, h(\Delta s, \alpha_{i+1/2}, \varepsilon)) - g(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon))] \\ & \quad - \frac{1}{2\Delta s}[(1-c)g(s_{k+1}, \alpha_i) + cg(s_{k+1}, \alpha_{i+1}) - (1-c_1)g(s_{k-1}, \alpha_i) - c_1 g(s_{k-1}, \alpha_{i+1})]. \end{aligned} \tag{2.38}$$

Consider the following term

$$-\left((1-c)c - (1-c_1)c_1\right)\frac{\Delta\alpha^2}{4\Delta s}\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) \tag{2.39}$$

and suppose we estimate the difference of the product of the interpolation weights. From Lemma 2.6 we have

$$\begin{aligned} c\Delta\alpha &= h(\Delta s, \alpha_{i+1/2}, \varepsilon) - \alpha_i \\ &= \alpha_{i+1/2} + \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) - \alpha_i + O(\Delta s^3) \\ &= \frac{\Delta\alpha}{2} + \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) + O(\Delta\alpha^3) \\ (1-c)\Delta\alpha &= \frac{\Delta\alpha - h(\Delta s, \alpha_{i+1/2}, \varepsilon)) + \alpha_i}{\Delta\alpha}\Delta\alpha \\ &= \alpha_{i+1} - h(\Delta s, \alpha_{i+1/2}, \varepsilon) \\ &= \frac{\Delta\alpha}{2} - \varepsilon\Delta s + \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) + O(\Delta s^3) \end{aligned} \tag{2.40}$$

and

$$\begin{aligned} c_1\Delta\alpha &= \alpha_{i+1/2} - \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) - \alpha_i + O(\Delta s^3) \\ &= \frac{\Delta\alpha}{2} - \varepsilon\Delta s - \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) + O(\Delta s^3) \\ (1-c_1)\Delta\alpha &= \alpha_{i+1} - \alpha_{i+1/2} + \varepsilon\Delta s + \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) + O(\Delta s^3) \\ &= \frac{\Delta\alpha}{2} + \varepsilon\Delta s + \frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2}) + O(\Delta s^3). \end{aligned} \tag{2.41}$$

We can then express the terms $(1-c)c\Delta\alpha^2$ and $(1-c_1)c_1\Delta\alpha^2$ up to higher order terms as

$$\begin{aligned} (1-c)c\Delta\alpha^2 &= \frac{\Delta\alpha^2}{4} - \varepsilon^2\Delta s^2 + \varepsilon(1-\varepsilon)^2\Delta s^3\tan(\alpha_{i+1/2}) \\ (1-c_1)c_1\Delta\alpha^2 &= \frac{\Delta\alpha^2}{4} - \varepsilon^2\Delta s^2 - \varepsilon(1-\varepsilon)^2\Delta s^3\tan(\alpha_{i+1/2}). \end{aligned} \tag{2.42}$$

The above calculation on the interpolations weights gives us the estimate

$$\begin{aligned} &-\frac{1}{2\Delta s}\left((1-c)c-(1-c_1)c_1\right)\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) \\ &= -\varepsilon(1-\varepsilon)^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)) + O(\Delta\alpha^3) \\ &= -\varepsilon(1-\varepsilon)^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k, \alpha_{i+1/2}) + O(\Delta\alpha^3 + \Delta\alpha\Delta s^2 + \Delta s^3). \end{aligned} \tag{2.43}$$

The next term we consider is

$$\begin{aligned} &-(2c^3-3c^2+c)\frac{\Delta\alpha^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_{k+1}, h(\Delta s, \alpha_{i+1/2}, \varepsilon)) \\ &\quad+(2c_1^3-3c_1^2+c_1)\frac{\Delta\alpha^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_{k-1}, h(-\Delta s, \alpha_{i+1/2}, \varepsilon)). \end{aligned} \tag{2.44}$$

The term (2.44) can be rewritten as

$$-\left((2c^3-3c^2+c)-(2c_1^3-3c_1^2+c_1)\right)\frac{\Delta\alpha^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k, \alpha_{i+1/2})$$

by Lemma 2.7 and because $\Delta\alpha^3/\Delta s$ is a second order term. By (2.40) we have

$$\begin{aligned} &-\left((2c^3-3c^2+c)-(2c_1^3-3c_1^2+c_1)\right)\frac{\Delta\alpha^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \\ &\quad=\left(\varepsilon\frac{\Delta\alpha^2}{12}-\varepsilon^3\frac{\Delta s^2}{3}\right)\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+O(\Delta s^3). \end{aligned} \tag{2.45}$$

Next we shall estimate the error in

$$-(1-c)c\frac{\Delta\alpha^2}{2}\left(\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon))-\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon))\right). \tag{2.46}$$

We compute the Taylor series in s and α

$$\begin{aligned} &\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon)) \\ &\quad=\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})+(h(\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \\ &\qquad+\Delta s\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})+\frac{(h(\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})^2}{2}\frac{\partial^4 g}{\partial\alpha^4}(s_k,\alpha_{i+1/2}) \\ &\qquad+(h(\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})\Delta s\frac{\partial^4 g}{\partial\alpha^3\partial s}(s_k,\alpha_{i+1/2}) \\ &\qquad+\frac{\Delta s^2}{2}\frac{\partial^4 g}{\partial\alpha^2\partial s^2}(s_k,\alpha_{i+1/2}) \\ &\qquad+O(\Delta\alpha^3+\Delta\alpha^2\Delta s+\Delta\alpha\Delta s^2+\Delta s^3) \end{aligned} \tag{2.47}$$

and also

$$\begin{aligned} &\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon)) \\ &\quad=\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})+(h(-\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \\ &\qquad-\Delta s\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})+\frac{(h(-\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})^2}{2}\frac{\partial^4 g}{\partial\alpha^4}(s_k,\alpha_{i+1/2}) \\ &\qquad-(h(-\Delta s,\alpha_{i+1/2},\varepsilon)-\alpha_{i+1/2})\Delta s\frac{\partial^4 g}{\partial\alpha^3\partial s}(s_k,\alpha_{i+1/2}) \\ &\qquad+\frac{\Delta s^2}{2}\frac{\partial^4 g}{\partial\alpha^2\partial s^2}(s_k,\alpha_{i+1/2}) \\ &\qquad+O(\Delta\alpha^3+\Delta\alpha^2\Delta s+\Delta\alpha\Delta s^2+\Delta s^3). \end{aligned} \tag{2.48}$$

We know from Lemma 2.8 that the only first order terms in $\Delta\alpha$ and Δs in the difference of (2.47) and (2.48) are

$$\begin{matrix} 2\Delta s\dfrac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2}) \\ 2\Delta s\varepsilon\dfrac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2}) \end{matrix} \tag{2.49}$$

and using the estimates from (2.42) we conclude

$$\begin{aligned} &-(1-c)c\frac{\Delta\alpha^2}{4\Delta s}\left(\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon))-\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon))\right)\\ &\quad=\left(-\frac{\Delta\alpha^2}{8}+\varepsilon^2\frac{\Delta s^2}{2}\right)\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})+\left(-\frac{\Delta\alpha^2}{8}+\varepsilon^2\frac{\Delta s^2}{2}\right)\varepsilon\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\ &\qquad+O(\Delta\alpha^3+\Delta\alpha^2\Delta s+\Delta\alpha\Delta s^2+\Delta s^3). \end{aligned} \tag{2.50}$$

We can use (2.43), (2.45) and (2.50) to determine

$$\begin{aligned} &\frac{1}{2\Delta s}[g(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon))-g(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon))]\\ &\quad-\frac{1}{2\Delta s}[-(1-c)g(s_{k+1},\alpha_i)+cg(s_{k+1},\alpha_{i+1})+(1-c_1)g(s_{k-1},\alpha_i)-c_1g(s_{k-1},\alpha_{i+1})]\\ &=\frac{1}{2\Delta s}[(1-c)c\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_{k+1},h(\Delta s,\alpha_{i+1/2},\varepsilon))\\ &\quad-(1-c_1)c_1\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_{k-1},h(-\Delta s,\alpha_{i+1/2},\varepsilon))]\\ &=\left(-\frac{\Delta\alpha^2}{8}+\varepsilon^2\frac{\Delta s^2}{2}\right)\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})+\left(-\varepsilon\frac{\Delta\alpha^2}{24}+\varepsilon^3\frac{\Delta s^2}{6}\right)\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\ &\quad-\varepsilon(1-\varepsilon)^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})\\ &\quad+O(\Delta\alpha^3+\Delta\alpha^2\Delta s+\Delta\alpha\Delta s^2+\Delta s^3). \end{aligned}$$

□

The effects of the interpolation in α for the term d_α is the last result that we need to complete our numerical analysis of M5.

Lemma 2.11. *(Error for linear interpolation in d_α) Let $g(s,\alpha)$ be C^4 on $\mathbb{R}\times[-\pi,\pi)$, $s_k = k\Delta s$ and $\alpha_i = i\Delta\alpha$. If*

$$0 < \varepsilon \le \frac{\Delta\alpha - \Delta s^2 M}{2\Delta s}$$

where

$$M = \sup_{t'\in(0,\varepsilon\Delta s)} \left|\frac{\partial^2\nu}{\partial u^2}(t',\alpha_{i+1/2})\right|$$

then

$$\begin{aligned}
&\frac{g(s,\nu(\varepsilon\Delta s,\alpha_{i+1/2})) - g(s,\nu(-\varepsilon\Delta s,\alpha_{i+1/2}))}{2\varepsilon\Delta s} - d_\alpha(s_k,\alpha_{i+1/2},\varepsilon)\\
&= \left(-\frac{\Delta\alpha^2}{24} + \varepsilon^2\frac{\Delta s^2}{6}\right)\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad - \frac{\varepsilon^2\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2}) + O(\Delta\alpha^3 + \Delta s^3).
\end{aligned}$$

Proof. Now we shall focus on the effects of the linear interpolation in α. Recall

$$d_\alpha(s_k,\alpha_{i+1/2},\varepsilon) = \frac{Ig(s_k,\nu(\varepsilon\Delta s,\alpha_{i+1/2})) - Ig(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2}))}{2\varepsilon\Delta s}.$$

and let

$$\begin{aligned}
Ig(s,\nu(\varepsilon\Delta s,\alpha)) &= (1-w)g(s_k,\alpha_i) + wg(s_k,\alpha_{i+1})\\
Ig(s,\nu(-\varepsilon\Delta s,\alpha)) &= (1-w_1)g(s_k,\alpha_i) + w_1 g(s_k,\alpha_{i+1})
\end{aligned}$$

where

$$\begin{aligned}
w &= \frac{\nu(\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_i}{\Delta\alpha}\\
w_1 &= \frac{\nu(-\varepsilon\Delta s,\alpha_{i+1/2}) - \alpha_i}{\Delta\alpha}.
\end{aligned} \tag{2.51}$$

By Lemmas 2.7 and 2.1 we know $\alpha_i < \nu(\pm\varepsilon\Delta s, \alpha_{i+1/2}) < \alpha_{i+1}$ and therefore

$$\begin{aligned} &g(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) - (1-w)g(s_k, \alpha_i) - wg(s_k, \alpha_{i+1}) \\ &\quad = -(1-w)w\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) \\ &\qquad - (2w^3 - 3w^2 + w)\frac{\Delta\alpha^3}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) + O(\Delta\alpha^4) \end{aligned} \tag{2.52}$$

and

$$\begin{aligned} &g(s_k, \nu(-\varepsilon\Delta s, \alpha_{i-1/2})) - (1-w_1)g(s_k, \alpha_i) - w_1 g(s_k, \alpha_{i+1}) \\ &\quad = -(1-w_1)w_1\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial\alpha^2}(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2})) \\ &\qquad - (2w_1^3 - 3w_1^2 + w_1)\frac{\Delta\alpha^3}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2})) + O(\Delta\alpha^4). \end{aligned} \tag{2.53}$$

Our goal is to estimate

$$\begin{aligned} &\frac{1}{2\varepsilon\Delta s}[g(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) - g(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2}))] \\ &\quad - \frac{1}{2\varepsilon\Delta s}[(1-w)g(s_k, \alpha_i) + wg(s_k, \alpha_{i+1}) - (1-w_1)g(s_k, \alpha_i) - w_1 g(s_k, \alpha_{i+1})]. \end{aligned} \tag{2.54}$$

Consider the following term

$$-((1-w)w - (1-w_1)w_1)\frac{\Delta\alpha^2}{4\varepsilon\Delta s}\frac{\partial^2 g}{\partial\alpha^2}(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2})) \tag{2.55}$$

and suppose we estimate the difference of the product of the interpolation weights. From Lemma 2.6 we have

$$\begin{aligned} w\Delta\alpha &= \nu(\varepsilon\Delta s, \alpha_{i+1/2}) - \alpha_i \\ &= \alpha_{i+1/2} + \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \alpha_i + O(\Delta s^3) \\ &= \frac{\Delta\alpha}{2} + \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) + O(\Delta s^3) \\ (1-w)\Delta\alpha &= \frac{\Delta\alpha - \nu(\varepsilon\Delta s, \alpha_{i+1/2}) + \alpha_i}{\Delta\alpha}\Delta\alpha \\ &= \alpha_{i+1} - \nu(\varepsilon\Delta s, \alpha_{i+1/2}) \\ &= \frac{\Delta\alpha}{2} - \varepsilon\Delta s + \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) + O(\Delta s^3) \end{aligned} \tag{2.56}$$

and

$$\begin{aligned} w_1\Delta\alpha &= \alpha_{i+1/2} - \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \alpha_i + O(\Delta s^3) \\ &= \frac{\Delta\alpha}{2} - \varepsilon\Delta s - \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) + O(s^3) \\ (1-w_1)\Delta\alpha &= \alpha_{i+1} - \alpha_{i+1/2} + \varepsilon\Delta s + \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \alpha_i + O(\Delta s^3) \\ &= \frac{\Delta\alpha}{2} + \varepsilon\Delta s + \frac{(\varepsilon\Delta s)^2}{2}\tan(\alpha_{i+1/2}) - \alpha_i + O(\Delta s^3). \end{aligned} \tag{2.57}$$

We can then express the terms $(1-w)w\Delta\alpha^2$ and $(1-w_1)w_1\Delta\alpha^2$ up to higher order terms as

$$\begin{aligned} (1-w)w\Delta\alpha^2 &= \frac{\Delta\alpha^2}{4} - \varepsilon^2\Delta s^2 + (\varepsilon\Delta s)^3\tan(\alpha_{i+1/2}) \\ (1-w_1)w_1\Delta\alpha^2 &= \frac{\Delta\alpha^2}{4} - \varepsilon^2\Delta s^2 - (\varepsilon\Delta s)^3\tan(\alpha_{i+1/2}). \end{aligned} \tag{2.58}$$

The above calculation on the interpolations weights gives us the estimate

$$\begin{aligned}
&-\left((1-w_1)w_1-(1-w)w\right)\frac{\Delta\alpha^2}{4\varepsilon\Delta s}\frac{\partial^2 g}{\partial\alpha^2}(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2}))\\
&\quad=-\frac{\varepsilon^2\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2}))+O(\Delta s^3)\\
&\quad=-\frac{\varepsilon^2\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})+O(\Delta\alpha\Delta s^2+\Delta s^3).
\end{aligned}\tag{2.59}$$

The next term we consider is

$$\begin{aligned}
&-(2w^3-3w^2+w)\frac{\Delta\alpha^3}{12\varepsilon\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\nu(\varepsilon\Delta s,\alpha_{i+1/2}))\\
&\quad+(2w_1^3-3w_1^2+w_1)\frac{\Delta\alpha^3}{12\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2})).
\end{aligned}\tag{2.60}$$

The term (2.60) can be rewritten as

$$-\left((2w^3-3w^2+w)-(2w_1^3-3w_1^2+w_1)\right)\frac{\Delta\alpha^3}{12\varepsilon\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})$$

by Lemma 2.7 and because $\Delta\alpha^3/\Delta s$ is a second order term. By (2.56) we have

$$\begin{aligned}
&-\left((2w^3-3w^2+w)-(2w_1^3-3w_1^2+w_1)\right)\frac{\Delta\alpha^3}{12\varepsilon\Delta s}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad=\left(\frac{\Delta\alpha^2}{12}-\varepsilon^2\frac{\Delta s^2}{3}\right)\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+O(\Delta\alpha^3).
\end{aligned}\tag{2.61}$$

Next we shall estimate the error in

$$-(1-w)w\frac{\Delta\alpha^2}{2}\left(\frac{\partial^2 g}{\partial\alpha^2}(s_k,\nu(\varepsilon\Delta s,\alpha_{i+1/2}))-\frac{\partial^2 g}{\partial\alpha^2}(s_k,\nu(-\varepsilon\Delta s,\alpha_{i+1/2}))\right).\tag{2.62}$$

We compute the Taylor series in α

$$\begin{aligned}
\frac{\partial^2 g}{\partial\alpha^2}(s_k,\nu(\varepsilon\Delta s,\alpha_{i+1/2}))&=\frac{\partial^2 g}{\partial\alpha^2}(s_k,\alpha_{i+1/2})+(\nu(\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2})\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad+\frac{(\nu(\varepsilon\Delta s,\alpha_{i+1/2})-\alpha_{i+1/2}))^2}{2}\frac{\partial^4 g}{\partial\alpha^4}(s_k,\alpha_{i+1/2})+O(\Delta\alpha^3)
\end{aligned}\tag{2.63}$$

and also

$$\begin{aligned}
&\frac{\partial^2 g}{\partial \alpha^2}(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2})) \\
&\quad = \frac{\partial^2 g}{\partial \alpha^2}(s_k, \alpha_{i+1/2}) + (\nu(-\varepsilon\Delta s, \alpha_{i+1/2}) - \alpha_{i+1/2})\frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_{i+1/2}) \\
&\qquad + \frac{(\nu(-\varepsilon\Delta s, \alpha_{i+1/2}) - \alpha_{i+1/2}))^2}{2}\frac{\partial^4 g}{\partial \alpha^4}(s_k, \alpha_{i+1/2}) + O(\Delta\alpha^3).
\end{aligned} \tag{2.64}$$

We know from Lemma 2.9 that the only first order term in Δs and $\Delta\alpha$ in the difference of (2.63) and (2.64) is

$$2\varepsilon\Delta s\frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_{i+1/2})$$

and using the estimates from (2.58) we conclude

$$\begin{aligned}
&-\frac{1}{2\varepsilon\Delta s}(1-w)w\frac{\Delta\alpha^2}{2}\left(\frac{\partial^2 g}{\partial \alpha^2}(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) - \frac{\partial^2 g}{\partial \alpha^2}(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2}))\right) \\
&\quad = \left(-\frac{\Delta\alpha^2}{8} + \varepsilon^2\frac{\Delta s^2}{2}\right)\frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_{i+1/2}) + O(\Delta s^3).
\end{aligned} \tag{2.65}$$

We can use (2.59), (2.61) and (2.65) to determine

$$\begin{aligned}
&\frac{1}{2\varepsilon\Delta s}[g(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) - g(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2}))] \\
&\quad - \frac{1}{2\varepsilon\Delta s}[(1-w)g(s_k, \alpha_i) + wg(s_k, \alpha_{i+1}) - (1-w_1)g(s_k, \alpha_i) - w_1 g(s_k, \alpha_{i+1})] \\
&= \frac{1}{2\varepsilon\Delta s}[-(1-w)w\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial \alpha^2}(s_k, \nu(\varepsilon\Delta s, \alpha_{i+1/2})) \\
&\quad + (1-w_1)w_1\frac{\Delta\alpha^2}{2}\frac{\partial^2 g}{\partial \alpha^2}(s_k, \nu(-\varepsilon\Delta s, \alpha_{i+1/2}))] \\
&= \left(-\frac{\Delta\alpha^2}{24} + \varepsilon^2\frac{\Delta s^2}{6}\right)\frac{\partial^3 g}{\partial \alpha^3}(s_k, \alpha_{i+1/2}) \\
&\quad - \frac{\varepsilon^2\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha^2}(s_k, \alpha_{i+1/2}) + O(\Delta\alpha^3 + \Delta s^3).
\end{aligned}$$

□

We can now prove our main result for M5 by a sum of the error terms in Lemmas 2.8, 2.9, 2.10 and 2.11.

Proof Theorem 2.5. We collect all of the error terms from Lemmas 2.8, 2.9, 2.10 and 2.11 to estimate

$$\begin{aligned}
&\frac{\partial g}{\partial \alpha}(s_k,\alpha_{i+1/2}) + \frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) - d_s(s_k,\alpha_{i+1/2}) - (1-\varepsilon)d_\alpha(s_k,\alpha_{i+1/2})\\
&\quad = \frac{\partial g}{\partial s}(s_k,\alpha_{i+1/2}) + \varepsilon\frac{\partial g}{\partial \alpha}(s_k,\alpha_{i+1/2}) - \frac{g(s+\Delta s,h(\Delta s,\alpha,\varepsilon)) - g(s-\Delta s,h(-\Delta s,\alpha,\varepsilon))}{2\Delta s}\\
&\qquad + \frac{g(s+\Delta s,h(\Delta s,\alpha,\varepsilon)) - g(s-\Delta s,h(-\Delta s,\alpha,\varepsilon))}{2\Delta s} - d_s(s_k,\alpha_{i+1/2},\varepsilon)\\
&\qquad + (1-\varepsilon)\bigg(\frac{\partial g}{\partial \alpha}(s_k,\alpha_{i+1/2}) - \frac{g(s,\nu(\varepsilon\Delta s,\alpha)) - g(s,\nu(-\varepsilon\Delta s,\alpha))}{2\varepsilon\Delta s}\\
&\qquad + \frac{g(s,\nu(\varepsilon\Delta s,\alpha)) - g(s,\nu(-\varepsilon\Delta s,\alpha))}{2\varepsilon\Delta s} - d_\alpha(s_k,\alpha_{i+1/2})\bigg).
\end{aligned}$$

The first order derivative term in our estimate is

$$\begin{aligned}
&\frac{\Delta s^2}{4}(\varepsilon-1)^2(1+\varepsilon+(\varepsilon-1)\cos(2\alpha_{i+1/2}))\sec^2(\alpha_{i+1/2})\frac{\partial g}{\partial \alpha}(s_k,\alpha_{i+1/2})\\
&\quad + (1-\varepsilon)\varepsilon^2\frac{\Delta s^2}{2}\frac{\partial g}{\partial \alpha}(s_k,\alpha_{i+1/2}).
\end{aligned}$$

The second order derivative terms in our estimate consist of

$$\begin{aligned}
&\frac{\Delta s^2}{2}(\varepsilon-1)^2\varepsilon\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha^2}(s_k,\alpha_{i+1/2})\\
&\quad - \frac{\Delta s^2}{2}(\varepsilon-1)^2\varepsilon\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha^2}(s_k,\alpha_{i+1/2})\\
&\quad + (1-\varepsilon)\varepsilon^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha^2}(s_k,\alpha_{i+1/2})\\
&\quad - (1-\varepsilon)\varepsilon^2\frac{\Delta s^2}{2}\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha^2}(s_k,\alpha_{i+1/2}) = 0
\end{aligned}$$

and

$$\frac{\Delta s^2}{2}(\varepsilon-1)^2\tan(\alpha_{i+1/2})\frac{\partial^2 g}{\partial \alpha \partial s}(s_k,\alpha_{i+1/2}).$$

The third order derivative terms in our estimate are

$$\begin{aligned}
&-\varepsilon^3\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+\varepsilon^3\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad-(1-\varepsilon)\varepsilon^2\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})+(1-\varepsilon)\varepsilon^2\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&\quad\left(-\varepsilon\frac{\Delta\alpha^2}{24}-(1-\varepsilon)\frac{\Delta\alpha^2}{24}\right)\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})\\
&=-\frac{\Delta\alpha^2}{24}\frac{\partial^3 g}{\partial\alpha^3}(s_k,\alpha_{i+1/2})
\end{aligned}$$

with

$$\begin{aligned}
&-\varepsilon^2\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})+\varepsilon^2\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})-\frac{\Delta\alpha^2}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})\\
&\quad=-\frac{\Delta\alpha^2}{8}\frac{\partial^3 g}{\partial\alpha^2\partial s}(s_k,\alpha_{i+1/2})
\end{aligned}$$

along with

$$-\varepsilon\frac{\Delta s^2}{2}\frac{\partial^3 g}{\partial\alpha\partial s^2}(s_k,\alpha_{i+1/2}),$$

and

$$-\frac{\Delta s^2}{6}\frac{\partial^3 g}{\partial s^3}(s_k,\alpha_{i+1/2}).$$

□

This analysis of M5 provides new insight into the behavior of the method. The theory that we presented here suggests that the accuracy of the method depends on the size of α. Error terms E_1 and E_2 depend on trigonometric functions that are unbounded as α approaches $\pm\pi/2$. None of the other methods have such behavior in their error terms. This suggests that M5 will have lower accuracy in reconstructions where the support of the function lies near the source trajectory.

Let us now compare M4 to M5 with regards to the error terms we derived. M5 has all of the error terms as M4 in addition to E_1, E_2, E_5 and E_6. Thus in the upper bound for the error in M5 is higher than that of M4. It is possible that sum of error terms found in M5 provide a smaller error than found in M4. Our numerical experiments in Section 2.4 will explore this idea further.

The error term

$$E_5 = -\varepsilon \frac{\Delta s^2}{2} \frac{\partial^3 g}{\partial \alpha \partial s^2}(s_k, \alpha_{i+1/2})$$

is of particular interest in our analysis of M5. The term appears in Lemma 2.8, a Lemma estimating the difference scheme d_s without considering interpolation in α. For $\varepsilon \geq 1/4$ the term E_5 is as large or larger than the error term involving the same partial derivative in the M3's error estimate. This error term is the same error term we identified as providing the better results of M4 compared to M3 and it does not depend on the method of interpolation used in α.

2.4 Numerical Results

For each experiment we have $R = 3$, $D = 6$ and we reconstruct inside the unit circle over a grid of 256 by 256 pixels. We have P source positions per turn $\Delta s = 2\pi/P$ and Q x-rays measured uniformly over the angular range $[-\pi/2, \pi/2]$. That is we have $2q$ discrete angular measurements for $\alpha_l = l\Delta\alpha$ for $l = -q, \ldots, q-1$ with $\Delta\alpha = \pi/Q$ and $\varepsilon = \Delta\alpha/(4\Delta s)$ unless otherwise stated.

We let $I_\pi(\mathbf{x}) = [\gamma - \theta, \gamma + \theta]$ where $\mathbf{x}$ has polar coordinates (r, γ) and $\theta = \arccos(r/R)$. These are the orthogonal-long π-lines from Section 4.3. We perform linear interpolation in the backprojection and discretize our filtering kernel according to [8]. Further details on the numerical implementation are found in Section 6.3.

We will use the following function to construct mathematical phantom for our numerical test

$$b_m(\mathbf{x}) = \left(1 - |\mathbf{x}|^2\right)_+^m = \begin{cases} \left(1 - |\mathbf{x}|^2\right)^m & \text{if } 1 - |\mathbf{x}|^2 \geq 0 \\ 0 & \text{if } 1 - |\mathbf{x}|^2 < 0 \end{cases} \tag{2.66}$$

Here $m = 3$ or 0 is a parameter that controls the smoothness of the function. Our object function $f(\mathbf{x})$ is given by

$$f(\mathbf{x}) = b_m(T(\mathbf{x} - \mathbf{x}_0)) \tag{2.67}$$

with

$$T = \begin{bmatrix} \cos(\psi)/u & \sin(\psi)/u & 0 \\ -\sin(\psi)/v & \cos(\psi)/v & 0 \\ 0 & 0 & 1/w \end{bmatrix}.$$

P	M5 l_2 error	$O(P^{-\alpha})$	M4 l_2 error	$O(P^{-\alpha})$
128	0.020372	—	0.019527	—
256	0.0051527	1.9832	0.0049305	1.9857
512	0.0012901	1.9978	0.0012337	1.9987
1024	0.00032242	2.0005	0.00030825	2.0008
2048	8.0547×10^{-5}	2.0011	7.6998×10^{-5}	2.0012
4096	2.0134×10^{-5}	2.0002	1.9247×10^{-5}	2.0002

Table 2.1: Convergence study for difference schemes M4 and M5 for the phantom (2.67).

One function f we image is given by

$$f(\mathbf{x}) = \frac{J_1(b_0|\mathbf{x}-\mathbf{x}_0|)}{b_0|\mathbf{x}-\mathbf{x}_0|}, \quad \mathbf{x} \in \mathbb{R}^2, \quad b_0 = 100, \quad \mathbf{x}_0 = (0.4, 0.7) \tag{2.68}$$

where J_1 denotes the Bessel function of the first kind of order one. The function is strictly bandlimited for frequencies less than the cut-off frequency b_0. We also reconstruct (2.67) with $u = .35, v = .25, \psi = 25\pi/180$ and $m = 0$ or 3. The Shepp-Logan phantom is given as a superposition of multiple copies of (2.67).

The first experiment is a convergence study to verify our theory in Section 2.2 and 2.3. We let $P = 2^j$ source positions and $q = 2^{j-2}$. We see in Table 2.1 that for $j = 7, \ldots, 12$ both methods M4 and M5 exhibit second order convergence as predicted by Theorem 2.5 and Lemma 2.4. Each experiment in the convergence study has near identical results for M4 and M5.

We also present the reconstructions of the classical Shepp-Logan phantom and of the smooth function (2.67) as shown in Figures 2.4 and 2.5 respectively. In each case we have identical image quality and relative errors. This occurs even for the reconstructions of (2.68), show in Figure 2.3, with both standard data and interpolated data calculated on a denser grid. The interpolation used is known to be successful for high accuracy reconstructions [7]. Both methods suffer from large errors for the reconstructions from data without interpolation. In each case we see the sensitivity of the parameter ε to the resolution of the reconstruction. When $\varepsilon = \Delta\alpha/(4\Delta s)$ the reconstructions are comparable with M4 and for $\varepsilon = \Delta\alpha/\Delta s$ the loss in resolution is significant in all cases except for the strictly bandlimited phantom. For the bandlimited phantom the relative l_2 error is smallest for $\varepsilon = \Delta\alpha/\Delta s$ but not by a significant margin.

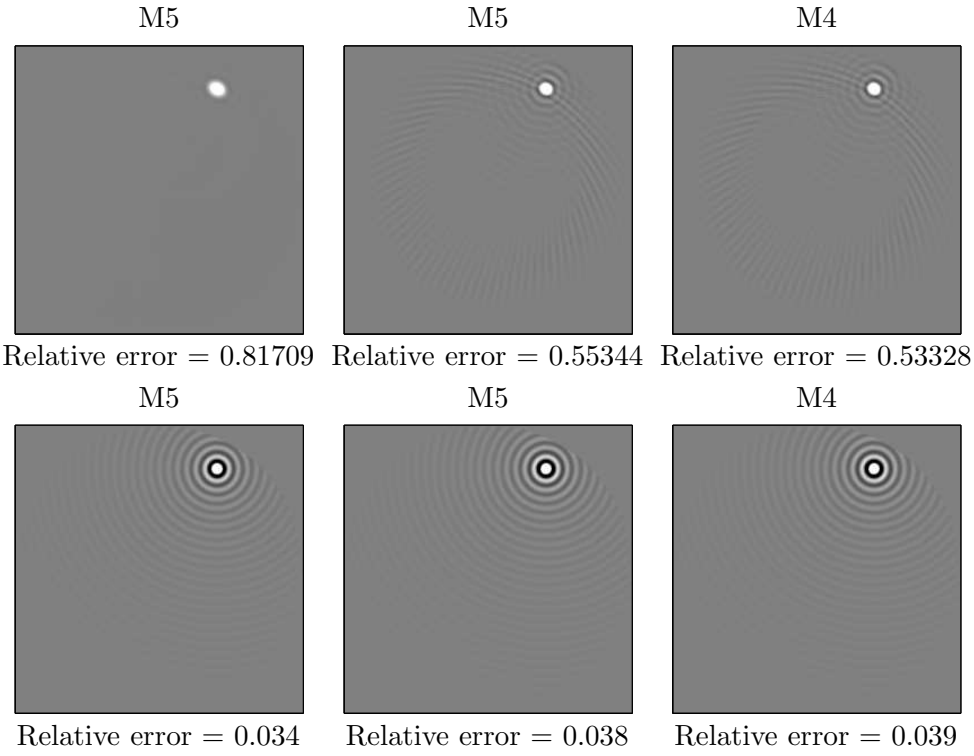

Figure 2.3: **Top Row:** Reconstructions of function (2.68). **Bottom Row:** Reconstructions of function (2.68) using interpolated data. **Left Column:** $\varepsilon = \Delta\alpha/\Delta s$. **Center Column:** $\varepsilon = \Delta\alpha/(4\Delta s)$.

Recall from Theorem 2.5 that the error estimate for M5 depends on $\tan(\alpha)$. Suppose that $R = 1.05$. Then the error terms of M5 will be larger than in the previous experiments. As we see in Table 2.2 the reconstruction from M5 has a larger l_2 relative error than the reconstruction from M4 for (2.67). This experiment validates our error estimates in Theorem 2.5. Thus the smaller source radius does effect the accuracy of M5 and does not greatly effect the accuracy of M4. We note that for the Shepp-Logan phantom the results of M4 and M5 are identical and don't differ greatly from the $R = 3$ experiments.

We see in Figure 2.6 that a reconstruction by (1.13) with M3 and with 64 projections produces a reconstruction that appears as an average of two rotated reconstructions. The rotated image is more pronounced with M3 and $P = 32$. The reconstruction with

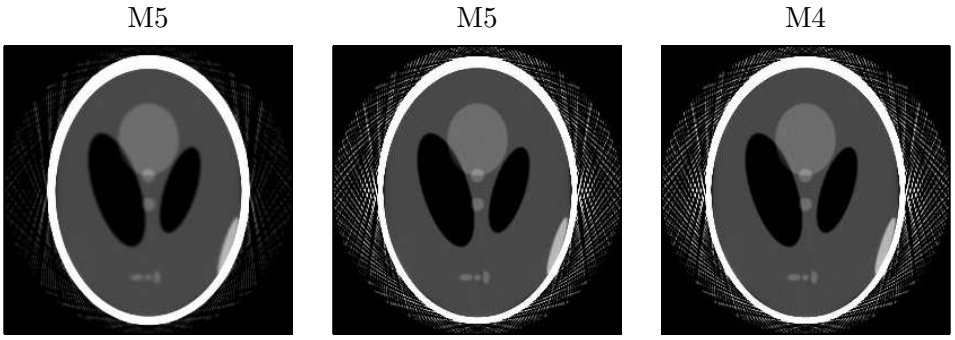

Figure 2.4: Reconstructions of the Shepp-Logan phantom. **Left Column:** $\varepsilon = \Delta\alpha/\Delta s$. **Center Column:** $\varepsilon = \Delta\alpha/(4\Delta s)$.

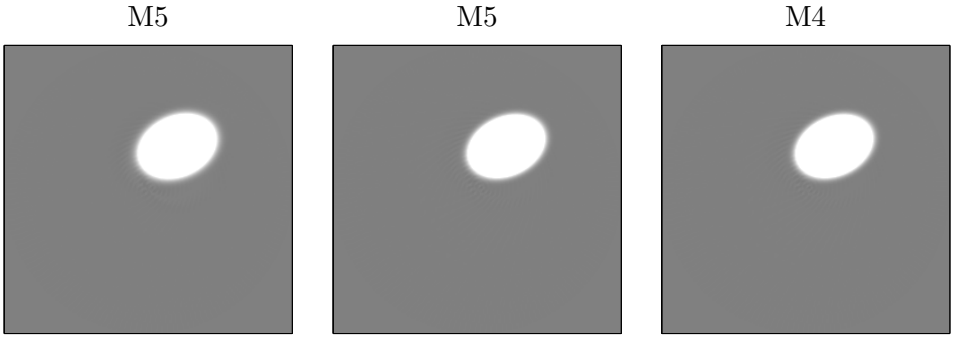

Figure 2.5: Reconstructions of the smooth function (2.67). **Left Column:** $\varepsilon = \Delta\alpha/\Delta s$. **Center Column:** $\varepsilon = \Delta\alpha/(4\Delta s)$.

(1.13) and M4 produces an image with better resolution. This supports our claim that M3 is the average of two slightly rotated data sets.

Our next experiment investigates the claim that the partial derivative with respect to source positions is unfavorable for reconstructions. Suppose we perform a reconstruction of f_s, given by (2.25), with $2P$ or $P/2$ with Q fixed. We will then calculate f_α, given by (2.24), with P and Q. Under the hypothesis that the derivative with respect to s is undesirable the reconstructions from the sparser and denser sampling in s will have nontrivial variations in the relative l_2 error in the reconstructions.

$f(\mathbf{x})$	M5 l_2 error	M4 l_2 error
Shepp-Logan	0.19753	0.19699
Equation (2.67)	0.0024100	0.001765

Table 2.2: Reconstructions errors small radius $R = 1.04$. For the Shepp-Logan phantom $P = 256$ and $Q = 503$ and for (2.67) $P = 175$ $Q = 311$.

Shepp-Logan	$P = 256$	$P = 256, 128$	$P = 256, 512$
Relative error	0.26195	0.26155	0.26206
Equation (2.67)	$P = 175$	$P = 175, 87$	$P = 175, 350$
Relative error	0.018455	0.017625	0.018668

Table 2.3: Reconstructions errors for over and under sampling in P in (2.25). The term (2.24) is calculated over the lattice P and Q. Here $P = p_\alpha, p_s$ implies f_s is calculated with $P = p_s$ and f_α with $P = p_\alpha$. For the Shepp-Logan phantom $Q = 924$ and for (2.67) $Q = 305$.

The results of the experiments using M4 are shown in Table 2.3. For both the smooth function (2.67) and the Shepp-Logan phantom decreasing or increasing P in (2.25) does not lead to a significant change in the l_2 error. In fact the l_2 error for $P = 256, 128$ is lower than that of $P = 256$ and $P = 256, 512$. However if we view a cross Section of the Shepp-Logan phantom in Figure 2.7 we see that the reconstruction from $P = 256, 128$ has larger errors at the boundary of the interior of the skull than the reconstruction from $P = 256$. This is a classic example of a Gibb's effect caused by under sampling in P and is present in reconstructions from (1.12) with $P = 128$ and $Q = 942$. We mention this to highlight that there is no benefit in reducing the number of source positions where (2.25) is evaluated over. We do not have Gibb's effect with (2.67) because the function is sufficiently smooth. We conclude that the derivative with respect to s is not an undesirable term in the numerical discretization of (1.11) because it is not sensitive to perturbations in P.

2.5 Conclusions

We have presented the numerical analysis of Noo et al's approach for calculating the derivative in (1.11). Our numerical results demonstrate that M4 is a more robust method when compared to M5. We therefore suggest M4 as the ideal solution to the view

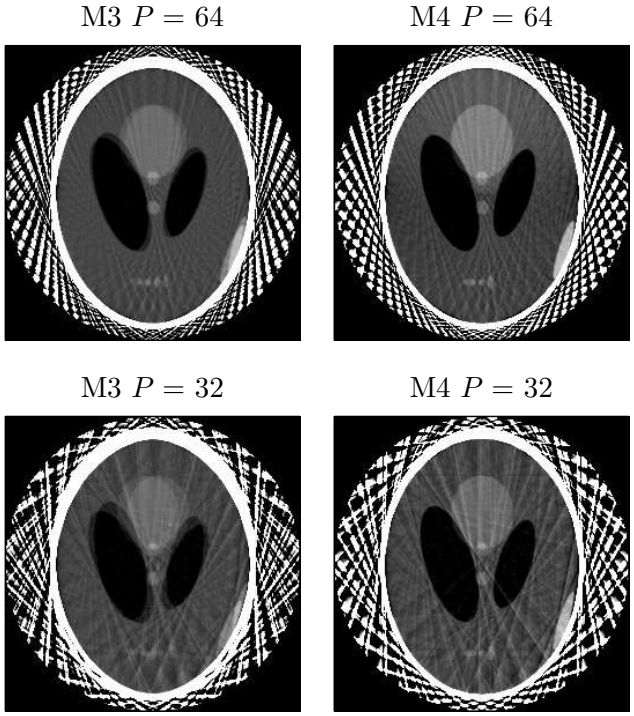

Figure 2.6: **Left:** Reconstruction with (1.13) with M3. **Right:** Reconstruction with (1.13) with M4. The image greyscale range is $[0, .07]$ with $P = 32$ or 64, $Q = 1183$, $R = 3$, and $D = 6$.

dependent derivative because of it simple derivation and implementation; neither of which depend on a free parameter ε.

Our analysis has provided a framework to compare direct methods with chain-rule based methods for the calculation of (2.1). We have provided conditions on ε so that the interpolation for M5 in α is between adjacent detector bins. We have also shown that the errors in M5 depend on α but this only matters for high accuracy reconstructions. We have also provided further reasoning to explain the need for a half detector shift in α in M5. Without the half detector shift in M5 the difference scheme in α, d_α, has step size $2\Delta\alpha$ for $\varepsilon = \Delta\alpha/(2\Delta s)$. The result of this is a loss of resolution compared to using M5 with a half detector shift in α.

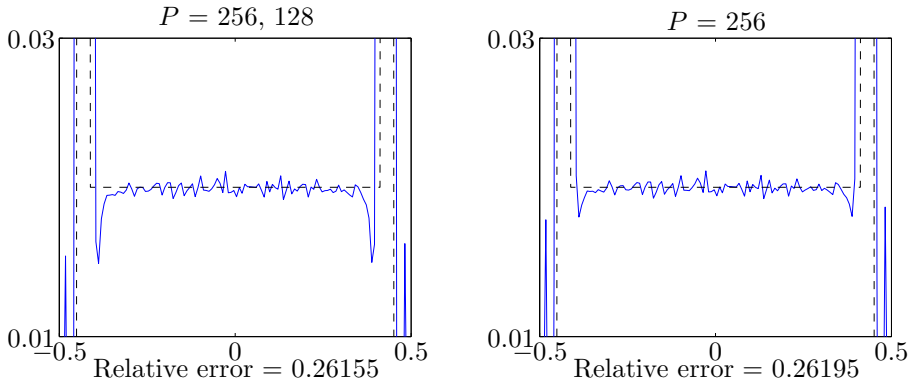

Figure 2.7: Cross Section $x_2 = -.702$ of the reconstructions of the Shepp-Logan phantom from Table 2.3.

We have also shown experimentally the derivative with respect to s is not a leading error term in the reconstructions and that the coarse sampling in P relative to Q is not a major problem for (2.1) in computed tomography.

3 Calculating π-Lines with Chips

Helical tomography was proposed as a scanning geometry that would be able to reconstruct entire volumes exactly. In helical tomography a patient lies on a moving table while the x-ray imaging device spins in a circle. Now clinical CT settings require that the table the patient lies on while the x-ray source and detector rotate around in a circle move at variable speed. With a constant table speed it is difficult to synchronize with biophysical events which occur during the scan [26].

The condition of a variable speed table requires a general reconstruction formula based on an existing exact reconstruction formula [13]. Katsevich's and Pan et al formulas have been shown in [23, 15, 30, 24] to hold for many scanning curves besides the helix of a constant pitch. Exact formulas we study rely heavily on π-lines for the reconstruction. In the generalized reconstruction formula it is necessary to have existence and uniqueness of π-lines for curves that behave as a helix in a local sense. Furthermore it is necessary to have a robust method to compute the π-lines and π-intervals for the admissible scanning trajectories. We shall develop such a method for calculating π-lines in this chapter.

Consider the following helical curve

$$\mathbf{y}(s) = (R\cos(s), R\sin(s), hs) \tag{3.1}$$

where R and h denote the radius and the pitch of the helix, respectively. We denote $S = \{(x, y, z) \in \mathbf{R}^3 : x^2 + y^2 < R^2\}$ as the helix cylinder of (3.1). Let $L(\mathbf{a}, \mathbf{b})$ denote the line passing through points $\mathbf{a}$ and $\mathbf{b}$.

Definition 3.1. *A π-line is a line segment, denoted $L(\mathbf{y}(s_b), \mathbf{y}(s_t))$, that intersects the helix at $\mathbf{y}(s_b)$ and $\mathbf{y}(s_t)$ such that $0 < s_t - s_b < 2\pi$.*

That is to say $\mathbf{y}(s_b(\mathbf{x}))$ and $\mathbf{y}(s_t(\mathbf{x}))$ are separated by no more than one helical turn. The name comes from the fact the π-line gives us a 180 degree view of the point we wish to reconstruct. See Figure 3.1. Most surprising is that every point inside the helix cylinder belongs to a unique π-line [3, 5].

Theorem 3.2. *Let* $\mathbf{y}(s)$ *be* (3.1) *and* S *be the helix cylinder. Then, any point* $\mathbf{x} \in S$ *is on a unique* π*-line.*

Proof. See [5, 11]. □

Definition 3.3. *Suppose* $\mathbf{x}$ *lies on a* π*-line* $L_\pi(\mathbf{x})$ *passing through* $\mathbf{y}(s_b(\mathbf{x}))$ *and* $\mathbf{y}(s_t(\mathbf{x}))$*. We call* $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$ *a parametric interval or a* π*-interval of* $\mathbf{x}$*.*

Three dimensional CT reconstruction formulas use the π-intervals to identify which source positions will be used for the reconstruction at a point $\mathbf{x}$. For example, Katsevich's formula (1.3) is of the form

$$f(\mathbf{x}) = \int_{I_\pi(\mathbf{x})} \rho(\mathbf{y}(s), \mathbf{x})\, ds \tag{3.2}$$

which is an integral over source positions from $I_\pi(\mathbf{x})$ with filtered and weighted x-ray data ρ. In practice we have knowledge of ρ from only a finite number of source positions and therefore we must approximate (3.2) by a numerical integration method. Previous work by [2, 11, 29] have led to non-linear equations which give the π-interval for any point within the helix.

In CT there are x-ray detectors which lie on a surface that rotates with the source position. We determine the integral along the line from $\mathbf{y}(s)$ to $\mathbf{x}$ by the measurement on the detector surface given by the stereographic projection of $\mathbf{x}$ onto the detector surface. There is a characterization of $s \in I_\pi(\mathbf{x})$ in terms of the native detector geometry. That is $s \in [s_b(\mathbf{x}), s_t(\mathbf{x})]$ if and only if the projection of $\mathbf{x}$ onto the detector surface lies within a region of the detector called the Tam-Danielsson window. For a proof see [16, 15]. The Tam-Danielsson window is the region on the detector surface bounded by the stereographic projection of the upper and lower turns of the source curve as shown in Figure 3.1.

Under ideal conditions this characterization would eliminate the need for calculating $s_b(\mathbf{x})$ and $s_t(\mathbf{x})$. However this characterization does not easily lead to a second order or higher numerical integration method. If the source curve is discretized by $s_k = k\Delta s$ then for each $\mathbf{x}$ there need not be a m and n such that $I_\pi(\mathbf{x}) = [m\Delta s, n\Delta s]$. Therefore interpolation in s is needed to accurately calculate (3.2) numerically [19]. This interpolation in s naturally depends on the distance of s_k to $s_b(\mathbf{x})$ and $s_t(\mathbf{x})$. The distance

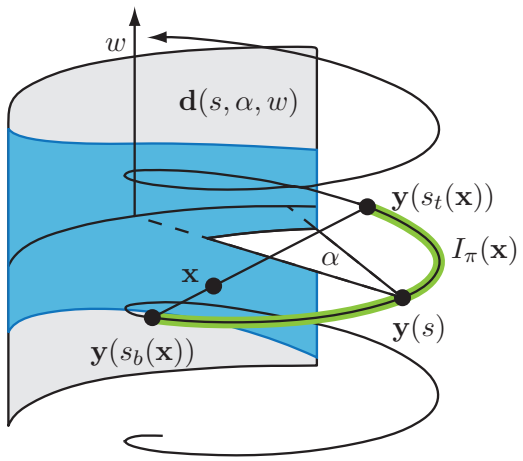

Figure 3.1: The Tam-Danielsson window is the shaded region of the detector surface $\mathbf{d}(s,\alpha,w)$. The π-line is shown as the line segment between $\mathbf{y}(s_b(\mathbf{x}))$ and $\mathbf{y}(s_t(\mathbf{x}))$.

cannot be easily calculated by determining the distance on the detector surface from the projection of $\mathbf{x}$ to the boundary of the Tam-Danielsson window. Further complicating the matter is that the shape of the Tam-Danielsson window need not be constant in s. This occurs for any helical source curve where the pitch or radius is allowed to vary with s. To overcome this obstacle we suggest calculating $s_b(\mathbf{x})$ and $s_t(\mathbf{x})$, before the reconstruction, by using the source curve's trajectory instead of relying on the location of the Tam-Danielsson window in the native detector geometry.

In Sections 3.1 and 3.2 we review and extend a decomposition of the helix cylinder into disjoint surfaces to generalize a method to compute π-intervals for many types of curves. Our proposed method in Section 3.3 is fast, handles any practical source trajectory, and does not require large overhead during computation. We conclude in Section 3.4 with a discussion on the speed of two competing reconstruction formulas for helical CT. The discussion in Section 3.4 is comparing the computational requirements of two different formulas when the reconstruction of the function occurs on a chip versus a plane.

3.1 Helical Chips

We now outline the algorithm developed by Izen [11] to calculate the π-lines of the helix. The method works on the idea of chips, sets of points in the helix with a common midpoint in their π-intervals.

Definition 3.4. *The chip anchored at* $\mathbf{y}(s)$*, denoted by* $C(s)$*, is the portion within the helix cylinder* S *of the union of all* π*-lines with endpoints equidistant along the helix from* $\mathbf{y}(s)$*. That is,*

$$C(s) = S \cap \{\mathbf{x} \in \mathbf{R}^3 | I_\pi(\mathbf{x}) = [s-\phi, s+\phi)], \phi \in (0,\pi)\} \tag{3.3}$$

and we call the line segment joining $\mathbf{y}(s)$ *to* $(0,0,hs)$ *the chip axis.*

A general formula for a chip is

$$C(s) = \{(1-t)\mathbf{y}(s-\alpha) + t\mathbf{y}(s+\alpha), 0 < t < 1, 0 < \alpha < \pi\}. \tag{3.4}$$

We shall first construct the chip $C(0)$ and show that it is given by

$$\left(x, y, \frac{hy\cos^{-1}(x/R)}{\sqrt{R^2-x^2}}\right) \in C(0) \tag{3.5}$$

where $x^2 + y^2 < R^2$. Suppose we project $C(0)$ onto the xy-plane as in Figure 3.2. The π-line of the point with projection (x,y) passes through the helix at $\mathbf{y}(-\alpha)$ and $\mathbf{y}(\alpha)$ where $\alpha = \cos^{-1}(x/R)$. The distance in the y coordinate between the two points on the helix is given by $R\sin(\alpha) - R\sin(-\alpha) = 2R\sin(\alpha)$ and the z coordinate varies from $-\alpha h$ to αh. Hence the π-line from $\mathbf{y}(-\alpha)$ to $\mathbf{y}(\alpha)$ at (x,y,z) satisfies

$$\begin{aligned} z &= (1-t)(-\alpha h) + t\alpha h \\ t &= \frac{y - \langle \mathbf{y}(-\alpha), \mathbf{e}_2\rangle}{\langle \mathbf{y}(\alpha), \mathbf{e}_2\rangle - \langle \mathbf{y}(-\alpha), \mathbf{e}_2\rangle} \end{aligned} \tag{3.6}$$

which simplifies to

$$z = \frac{hy\alpha}{R\sin(\alpha)} = \frac{hy\cos^{-1}(x/R)}{\sqrt{R^2-x^2}}. \tag{3.7}$$

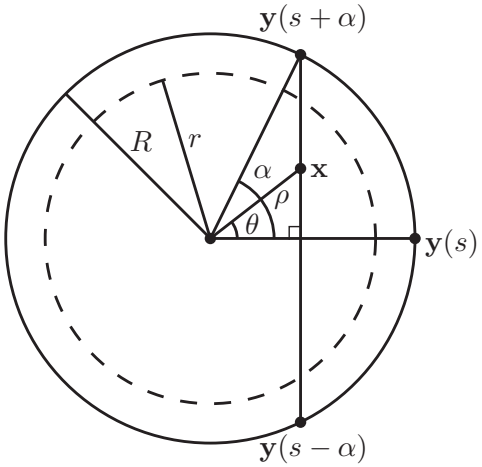

Figure 3.2: $C(s)$ projected onto the xy plane.

Suppose $0 \leq \rho < R$, $\theta \in [-\pi, \pi)$ is the angle in the xy plane measured with the chip axis and $(x, y) = (\rho\cos(\theta + s), \rho\sin(\theta + s))$. Then we have

$$\left(\rho\cos(\theta+s), \rho\sin(\theta+s), h\left(s + \frac{(\rho/R)\sin\theta\cos^{-1}\left((\rho/R)\cos\theta\right)}{\sqrt{1-(\rho/R)^2\cos^2\theta}}\right)\right) \in C(s). \tag{3.8}$$

The chip $C(s)$ is a rotated and translated copy of $C(0)$ and π-lines in each chip lie in parallel planes. These two conditions simplify identifying the chip $C(s)$ that contains a given point $\mathbf{x}$.

The goal is then to identify s and θ for each $\mathbf{x} \in S$. We follow the method Izen proposed in [11]. Once s and θ are identified it is a simple calculation to get $I_\pi(\mathbf{x})$. Izen showed to solve for the π-interval of $\mathbf{x}$ one computes

$$\beta = \gamma - z/h \mod 2\pi \tag{3.9}$$

$$\theta = g_\rho^{-1}(\beta) \tag{3.10}$$

$$s = \frac{z}{h} - \frac{\frac{\rho}{R}\sin\theta\cos^{-1}(\frac{\rho}{R}\cos\theta)}{\sqrt{1-(\frac{\rho}{R})^2\cos^2\theta}} \tag{3.11}$$

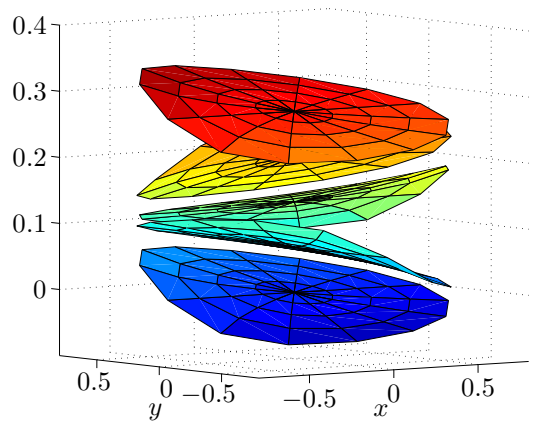

Figure 3.3: A stack of chips for $h = .274$ with $R = 1$ and $r = 3/4$.

where (ρ, γ, z) are the cylindrical coordinates of $\mathbf{x}$ and

$$g_\rho(\theta) = \theta - \frac{\frac{\rho}{R}\sin\theta\cos^{-1}(\frac{\rho}{R}\cos\theta)}{\sqrt{1-(\frac{\rho}{R})^2\cos^2\theta}}. \tag{3.12}$$

Now the π-interval of $\mathbf{x}$ is $I_\pi(\mathbf{x}) = [s-\alpha, s+\alpha]$ where $\alpha = \cos^{-1}(\frac{\rho}{R}\cos\theta)$. The following Newton method is then used to solve for θ

$$\theta_n = \theta_{n-1} - \frac{g_\rho(\theta_{n-1}) - \beta}{g'_\rho(\theta_{n-1})} \tag{3.13}$$

with

$$g'_\rho(\theta) = \frac{1-(\frac{\rho}{R})^2}{(1-(\frac{\rho}{R})^2\cos^2\theta)^{3/2}}\left(\sqrt{1-\left(\frac{\rho}{R}\right)^2\cos^2\theta} - \frac{\rho}{R}\cos\theta\cos^{-1}\left(\frac{\rho}{R}\cos\theta\right)\right). \tag{3.14}$$

3.2 Variable Pitch and Radius Helix

Now consider the following curve

$$\mathbf{y}(s) = (R\cos(s), R\sin(s), h(s)) \tag{3.15}$$

where R denotes the radius and $h(s)$ is a strictly monotone continuous function. Conditions for the existence and uniqueness of π-lines with a variable pitch helix have appeared in [26, 14]. Katsevich showed the π-lines exist if $h'(s) + h'''(s) > 0$ almost everywhere in s [14].

The goal is to extend the construction of a chip to a variable pitch helix and then to use the chips to indentify the π-intervals. We must first redefine the chip axis as the line from $\mathbf{y}(s)$ to $(0, 0, h(s))$. We can construct the chip $C(0)$ by utilizing the construction of (3.6) for the constant pitch helix

$$\begin{aligned} z &= (1-t)h(-\alpha) + th(\alpha) \\ t &= \frac{y - \langle \mathbf{y}(-\alpha), \mathbf{e}_2 \rangle}{\langle \mathbf{y}(\alpha), \mathbf{e}_2 \rangle - \langle \mathbf{y}(-\alpha), \mathbf{e}_2 \rangle}. \end{aligned} \tag{3.16}$$

This yields

$$t = \frac{y - R\sin(-\alpha)}{2R\sin(\alpha)} \tag{3.17}$$

and

$$\left(x, y, \frac{(h(\alpha) - h(-\alpha))y}{2\sqrt{R^2 - x^2}} + \frac{h(\alpha) + h(-\alpha)}{2} \right) \in C(0) \tag{3.18}$$

for $\alpha = \cos^{-1}(x/R)$. Suppose we use polar coordinates ρ, θ and let $\alpha = \cos^{-1}(\frac{\rho}{R}\cos\theta)$

$$\left(\rho\cos\theta, \rho\sin\theta, \frac{(\rho/R)\sin\theta(h(\alpha) - h(-\alpha))}{2\sqrt{1 - (\rho/R)^2\cos^2\theta}} + \frac{h(\alpha) + h(-\alpha)}{2} \right) \in C(0). \tag{3.19}$$

Suppose we have cylindrical coordinates ρ, γ, z for $\mathbf{x} \in S$. If θ is now the angle in the xy plane measured with respect to the chip axis as in Figure 3.2, $\theta + s = \gamma \mod 2\pi$, then

$$\begin{aligned} &\Bigg(\rho\cos(\theta + s), \rho\sin(\theta + s), \\ &\quad \frac{(\rho/R)\sin\theta(h(s+\alpha) - h(s-\alpha))}{2\sqrt{1 - (\rho/R)^2\cos^2\theta}} + \frac{h(s+\alpha) + h(s-\alpha)}{2} \Bigg) \in C(s). \end{aligned} \tag{3.20}$$

As with the standard helix, the π-lines in each chip remain in parallel planes orthogonal to the chip axis. Unlike the standard helix the π-lines do not necessarily intersect the chip axis.

Lemma 3.5. *Suppose* $\mathbf{y}(s) = (R\cos(s), R\sin(s), h(s))$ *where* $h(s)$ *is a strictly monotone continuous function. Then*

$$(\mathbf{y}(s+\alpha) - \mathbf{y}(s-\alpha)) \cdot ((0,0,h(s)) - \mathbf{y}(s)) = 0. \tag{3.21}$$

That is to say π*-lines in a chip are orthogonal to the chip axis.*

Next we shall study the chips of the helix with a variable pitch and radius,

$$\mathbf{y}(s) = (R(s)\cos(s), R(s)\sin(s), h(s)). \tag{3.22}$$

Conditions for existence and uniqueness of π-lines for the variable radius with a constant pitch are given in [25]. We construct $C(s)$ directly for each s by

$$\mathbf{z} = (x, y, (1-t)h(s-\alpha) + t\, h(s+\alpha)) \in C(s) \tag{3.23}$$

with α such that

$$\mathbf{z} = (1-t)\mathbf{y}(s-\alpha) + t\mathbf{y}(s+\alpha) \tag{3.24}$$

and

$$\begin{aligned} t &= \frac{y - \langle \mathbf{y}(s-\alpha), \mathbf{e}_2 \rangle}{\langle \mathbf{y}(s+\alpha), \mathbf{e}_2 \rangle - \langle \mathbf{y}(s-\alpha), \mathbf{e}_2 \rangle} \\ &= \frac{y - R(s-\alpha)\sin(s-\alpha)}{R(s+\alpha)\sin(s+\alpha) - R(s-\alpha)\sin(s-\alpha)}. \end{aligned} \tag{3.25}$$

As observed in Figure 3.4 the term α can no longer be determined geometrically because θ and ρ are independent of $R(s+\alpha)$. In the two previous cases we were able to relate α to the radius R. However R is no longer constant and we must identify the correct α such that (x, y) lies on the line between $\mathbf{y}(s-\alpha)$ and $\mathbf{y}(s+\alpha)$ projected onto the xy plane. The problem becomes no easier if we assume $h(s)$ is linear in (3.22). We see that much of elegance found in the construction of the original chips, such as $C(s)$ is a rotation and translation of $C(0)$, is lost as we generalize the behavior of the curve $\mathbf{y}(s)$ to act as a helix in a local sense. In particular the π-lines on a chip with a variable radius source curve are no longer in parallel planes.

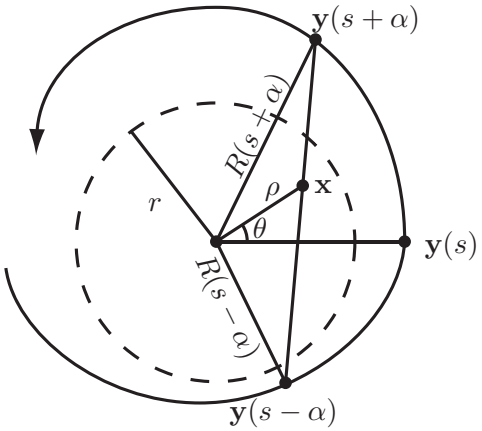

Figure 3.4: $C(s)$ projected onto the xy plane. For $\mathbf{y}(s) = (R(s)\cos s, R(s)\sin s, h(s))$.

To determine the chip $C(s)$ that contains $\mathbf{x}$ we shall define the distance between $C(s)$ and $\mathbf{x}$ as

$$|C(s) - \mathbf{x}| = \min_{\alpha} d(\mathbf{l}(s,\alpha), \mathbf{x}) \tag{3.26}$$

where $\mathbf{l}(s,\alpha)$ is the line segment from $\mathbf{y}(s+\alpha)$ to $\mathbf{y}(s-\alpha)$ and $d(\mathbf{l}(s,\alpha),\mathbf{x})$ is the distance from $\mathbf{l}$ to $\mathbf{x}$. We then find s such that $|C(s) - \mathbf{x}| = 0$.

3.3 Calculating π-Intervals for Variable Pitch and Radius Helix

We shall now demonstrate how to calculate π-intervals for variable pitch and variable radius helical scanning trajectories. In each case our method minimizes the distance from the point (x, y, z) to the chip $C(s)$ by varying the anchor point s. This numerical method is different than that of Izen's since he proposes solving for the angle θ of each point relative to the chip's axis and then identifies the chip's anchor point by equation (3.11).

First we shall show how to calculate the π-intervals of (3.15). Let ρ, γ, z be the cylindrical coordinates for $\mathbf{x} \in S$. We know $\gamma = s + \theta \mod 2\pi$ and

$$z = \frac{(\rho/R)\sin\theta(h(s+\alpha) - h(s-\alpha))}{2\sqrt{1-(\rho/R)^2\cos^2\theta}} + \frac{h(s+\alpha)+h(s-\alpha)}{2} \tag{3.27}$$

where $\alpha = \cos^{-1}(\frac{\rho}{R}\cos\theta)$. With $\theta = \gamma - s \mod 2\pi$, then we have

$$f(s) = z - \frac{(\rho/R)\sin\theta(h(s+\alpha) - h(s-\alpha))}{2\sqrt{1-(\rho/R)^2\cos^2\theta}} + \frac{h(s+\alpha)+h(s-\alpha)}{2}$$

and then solve

$$f(s) = 0. \tag{3.28}$$

Then the π-interval of $\mathbf{x}$ is $I_\pi(\mathbf{x}) = [s-\alpha, s+\alpha]$ where $\alpha = \cos^{-1}(\frac{\rho}{R}\cos\theta)$.

To calculate the π-intervals of the curve (3.22) we have a less direct approach than the variable pitch method. We no longer have an explicit description of each chip. Instead we must determine the surface of a chip at (x, y) by applying an optimization scheme. The scheme varies α such that the projection of the line from $\mathbf{y}(s-\alpha)$ to $\mathbf{y}(s+\alpha)$ in the xy plane passes through the point (x, y). Thus to calculate the chip that contains a point (x, y, z) we find the root of

$$\begin{aligned} f(s) &= z - (1-t)h(s-\alpha) + t\,h(s+\alpha) \\ t &= \frac{y - R(s-\alpha)\sin(s-\alpha)}{R(s+\alpha)\sin(s+\alpha) - R(s-\alpha)\sin(s-\alpha)} \end{aligned} \tag{3.29}$$

with respect to s. For each s we must calculate α given by the root of

$$\begin{aligned} D(\alpha) &= (x_2 - x_1)^\perp \cdot (x_0 - x_1) \\ x_0 &= (x, y) \\ x_1 &= (R(s+\alpha)\cos(s+\alpha), R(s+\alpha)\sin(s+\alpha)) \\ x_2 &= (R(s-\alpha)\cos(s-\alpha), R(s-\alpha)\sin(s-\alpha)). \end{aligned} \tag{3.30}$$

When $D(\alpha) = 0$ we have x_0 is on the line from x_1 to x_2. The π-interval is given by $[s-\alpha, s+\alpha]$.

3.4 Chips and π-Line Filtration Formulas

An alternative algorithm to Katsevich's cone-beam inversion formula has been developed by Pan et al [32]. To reconstruct a point $\mathbf{x} \in S$ the competing inversion formula (1.6), (1.7), (1.8), and (1.9) requires the backprojection of the points along the π-line of $\mathbf{x}$. The

formula then applies an inverse Hilbert transform to recover $f(\mathbf{x})$ along the π-line of $\mathbf{x}$. Pan et al's formula is a backprojection filtration, or BPF formula. Katsevich's formula is often implemented as FBP and therefore has the capability to reconstruct at a point $\mathbf{x}$ without having to reconstruct any other points.

Let us remark that it is possible to implement the π-line filtration formula as a FBP formula [20, 34]. Rewriting the method this way would reduce the dependency of reconstructing $f(\mathbf{x})$ along its π-lines and reduce the number of points to backproject. However this method is not advantageous because it places a much higher burden on the filtering in the local detector coordinates. Now for each source position the projection of the π-line of $\mathbf{x}$ onto the detector surface is filtered. The location of a π-line changes on the detector surface for each source position and hence the filtering curves change for each source position. This task seems daunting when compared to Katsevich's FBP formula. Here each filtering occurs along the same set of curves on the detector surface for every source position.

Suppose we wish to reconstruct in the set $V = [0, 1]^3$. One difficulty with the BPF formula has been finding a manageable set of π-lines that intersect V. We can discretize V over a uniform grid of dimensions M_x by M_y by M_z in x, y, and z and perform a reconstruction over the M_z M_x by M_y xy planes. This is a suitable method for Katsevich's 3D formula. However such a method would be wasteful with Pan's π-line filtration formula. This is because the choice of discretizing V is at conflict with the BPF formula's reconstruction along π-lines. We can perform P reconstructions in the xy plane but at the cost of making redundant calculations of $f(\mathbf{x})$. A π-line of $\mathbf{x}$ can intersect more than one of the P xy planes and hence be calculated more than once. What is needed is a clever way to fill V such that there is no redundant calculations by the BPF formula.

We propose decomposing the volume V into M_z chips. Each chip is disjoint and therefore we would have no redundant BPF calculations, that is calculating $f(\mathbf{x})$ along the π-line of $\mathbf{x}$ more than once. After the reconstruction is complete a simple tri-linear interpolation in the x_3 coordinate would yield $f(\mathbf{x})$ over the same coordinate system which works well with Katsevich's formula.

One common criticism of BPF algorithms is their speed is slower than that of the FBP methods [15]. If we were reconstructing a region contained in the support of the function then the BPF would require more backprojected points than the FBP. Suppose

# of Chips	FBP	BPF	# filterings for FBP	# filterings for BPF
1	232 sec	189 sec	188160	512
2	331 sec	275 sec	188160	1024
4	431 sec	386 sec	188160	2048
8	636 sec	615 sec	188160	4096
16	1031 sec	1063 sec	188160	8192
32	1837 sec	1960 sec	188160	16384

Table 3.1: Runtime of the Katsevich's FBP and Pan's BPF for the Shepp-Logan phantom over chips with anchor points uniformly spaced over $[-.53, .53]$. Reconstruction parameters $P = 896$ source positions per turn, $Q = 2070$ fan angles distributed over $[-\pi/2, \pi/2]$, $p = .274$, and 210 κ curves.

we want to reconstruct a volume containing the support of $f(\mathbf{x})$. We decompose our volume into chips and perform the reconstruction over each chip restricted to points in S. The number of backprojections required for each chip is the same whether the BPF or the FBP method is used. However the FBP has to perform many filtering steps for each source position. The BPF only needs to perform one filtering per π-line. Table 3.1 shows the runtime of the FBP and BPF reconstructions with 894 source positions per turn, 224 by 448 detector elements and with 210 filtering curves for the FBP method. The reconstruction is that of the Shepp-Logan phantom on the chips of size 512 by 512 pixels. As we can see the BPF algorithm has a significant speed advantage over the FBP when we reconstruct on 1 chip. However as the number of chips, uniformly spaced over $[-.53, .53]$, increases we have equal or faster runtime for the FBP over the BPF. As we increase the number of chips to reconstruct over the same interval the filtering cost for the FBP remains the same while the BPF's filtering costs increase with each chip. In each case the quality of the two reconstructions are identical visually and in the l_2 sense; cf. Figure 3.5. The overhead in our implementation of the inversion of the Hilbert transform reduces the performance of the BPF algorithm as the number of chips increases. We remark that our implementation of the BPF algorithm is not as accurate as the FBP method when reconstructing smooth phantoms. The cause of this problem is believed to be a first order numerical method to handle the singularities at the endpoints of integration in (1.9).

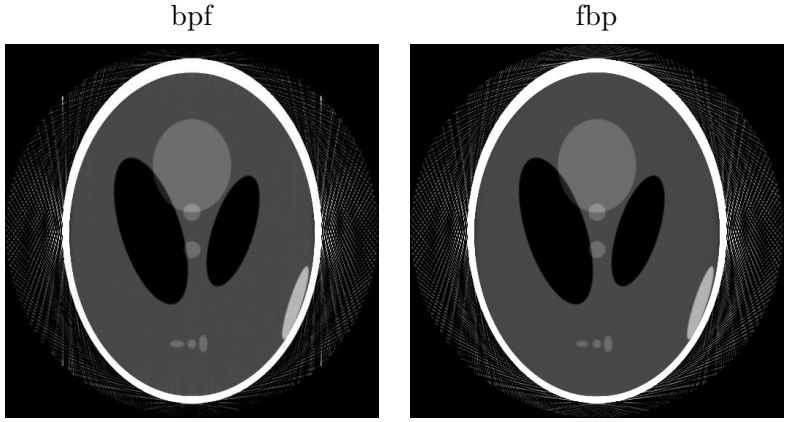

Figure 3.5: Reconstructions of Shepp-Logan phantom on the chip $C(0)$. In each case the relative l_2 error is 19%. The parameters are $P = 768$, $Q = 1774$, $R = 3$, $p = .274$ and 139 κ curves.

Another advantage to using chips in the BPF formula is a simplification of the numerical implementation. We can design our chips of size M by N so the points in the same column lie on the same π-line. Hence we would apply an inverse Hilbert transform along each column of every chip to get $f(\mathbf{x})$. Katsevich's formula, which applies its filtering before the backprojection, does not need such a coordinate system since all the calculations of $\mathbf{x}$ are done with respect to the local detector coordinates.

The implementation of the two methods is discussed in Section 6. The difficult step in the Katsevich code is rebinning, or change of coordinates from the w coordinate to the κ curves. The remaining steps in the algorithm follow naturally from the standard reconstruction formula. Pan's BPF formula has additional overhead in the backprojection step because projection data from $\mathbf{y}(s_b(\mathbf{x}))$ and $\mathbf{y}(s_t(\mathbf{x}))$ needs to be stored for the filtering. Furthermore the filtering in Pan's formula is more complicated than that of Katsevich's because of the of Hilbert inversion formula (1.7). The inversion formula has a singularity at the endpoints of the filtering and this makes our current implementation of the formula less accurate with smooth functions compared to Katsevich's formula. For discontinuous functions and sufficient projection data the two methods produce favorable results with an equal amount of resolution and accuracy as shown in Figure 3.5.

Regardless of which formula is used, the chips form a natural coordinate system for the general scanning curves. One can simply decompose the generalized helix cylinder

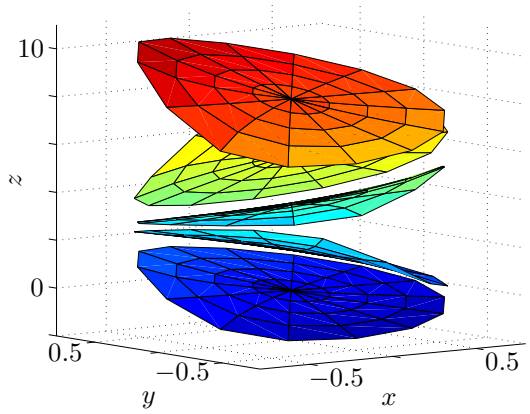

Figure 3.6: A stack of chips for $h(s) = \frac{.2740}{2\pi}s^2 - \frac{.2740}{2\pi} + s$ with $R = 1$ and $r = 3/4$.

into chips and reconstruct $f(\mathbf{x})$ on each chip. The chips provide an orderly method to group π-lines into sets which are helpful in implementing cone beam reconstruction formulas with general source curves. However, the chip surfaces have a drawback. As seen in Figure 3.6 the distance between two chips is not uniform. Therefore if we recover a function over many chips we cannot expect to know the function f over a uniform grid without using tri-linear interpolation.

4 Comet Tails

A new type of artifact is present in reconstructions from π-line dependent formulas. The artifact, called a comet tail artifact, is easy to illustrate with the reconstruction in Figure 4.1. Here a large artifact, originating from the reconstructed smooth function, resembles the appearance of a comet's tail. Comet tail artifacts have been found in reconstructions by formulas that allow the π-intervals to change with respect to $\mathbf{x}$. This includes, but is not restricted to, the 2D π-line formula (1.12) and Katsevich's 3D (1.5) formula. See [28] for a discussion on other types of artifacts in reconstructions by (1.5). The comet tail artifact discussed here is not to be confused with the comet tail artifact in ultrasound reconstructions [6]. The size of the error of the comet tail artifact is not large and it does not affect the rate of convergence of our algorithm. Our goal is to justify why the comet tail artifact occurs and to understand where it will occur.

In Section 4.1 we introduce a set called the region of backprojection that tells us which points have position s inside their π-intervals. We describe a heuristic principle for the appearance of the comet tail artifact in terms of the region of backprojection. Our claim is the comet tail artifact will appear strongest along the boundary of the region of backprojection. We derive properties of the region of backprojection that hold for fan-beam and helical scanning geometries.

Section 4.2 details the appearance of the comet tail artifact in helical tomography under our hypotheses. We observe that the region of backprojection and its boundary are collections of π-lines. We use disjoint surfaces called chips from Chapter 3 as a method to collect the π-lines that make up the region of backprojection for the helix. We then describe the comet tail artifact restricted to each chip. Our work determines that it is easier to describe the comet tail artifact on a chip rather than on a plane.

In Section 4.3 we construct the region of backprojection in 2D fan-beam tomography for special families of π-lines. In each case we have that the region of backprojection and its boundary is no longer made up of a collection of π-lines. A method to explicitly calculate a subset of the support of the comet tail artifact for the 2D π-formula (1.12) is presented and verified under our hypotheses about the comet tail's appearance.

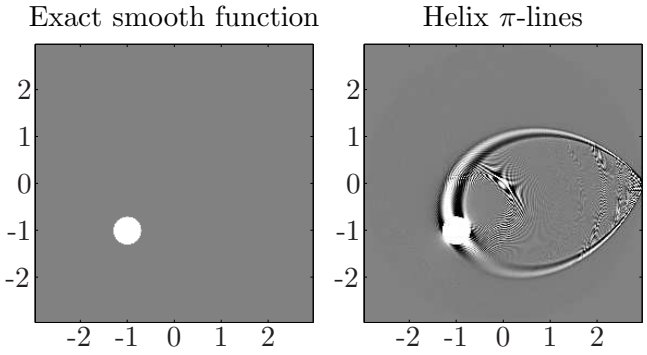

Figure 4.1: **Left:** Original smooth function. **Right:** Reconstruction by (1.12) with a comet tail artifact present. Reconstructed with $P = 2560$, $Q = 579$, $R = 3$, and $D = 6$. The image grayscale range is $[-8E - 5, 8E - 5]$. The reconstructed function is (2.67) with $\mathbf{x}_0 = (-1, -1), u = .3, v = .3, m = 3$ and $\psi = 0$.

The theory presented here proves useful in Section 5 after we observe comet tail artifacts in reconstructions where the location of the detector array for the given projection data is not known exactly.

Let $R = 1$ unless otherwise stated, define Ω to be the open unit ball in $\mathbb{R}^2$ and $S = \Omega \times \mathbb{R}$ be the helix cylinder in $\mathbb{R}^3$.

4.1 Region of Backprojection

The shape of the comet tail artifact is dictated by the π-lines. A distinctive feature of π-line reconstruction formulas is that for reconstruction at a point $\mathbf{x}$ only data from source positions from the π-interval of $\mathbf{x}$ are used. Viewed in another way, this means that data from a give source position s contributes only to the reconstruction of points in a certain region. This observation motivates the following definition.

Definition 4.1. *We call* $RBP(s) = \{\mathbf{x} \in \Omega \mid s \in I_\pi(\mathbf{x})\}$ *the region of backprojection of position* $\mathbf{y}(s)$. *Let* $\partial RBP(s)$ *be the boundary of* $RBP(s)$. *For 3 dimensions* $RBP(s) = \{\mathbf{x} \in S \mid s \in I_\pi(\mathbf{x})\}$. *We denote* $\partial' RBP(s)$ *as* $\partial RBP(s) \cap \Omega$ *in 2D and* $\partial RBP(s) \cap S$ *in 3D.*

A point $\mathbf{x}$ is in the region of backprojection if the current source position is in the point's π-interval. For 2π algorithms we don't have a notion of π-intervals and the

points where we reconstruct is the set Ω. The boundary of $RBP(s)$ is the feature of interest. Suppose $\mathbf{x}$ is inside Ω and in $\partial RBP(s)$ and that $s_b(\mathbf{x})$ and $s_t(\mathbf{x})$ are continuous. This implies that $f(\mathbf{x})$ will be reconstructed with data from source position s. There exists a point $\mathbf{x}'$ in any neighborhood of $\mathbf{x}$ outside of $RBP(s)$ and thus $f(\mathbf{x}')$ will not be reconstructed from data from source position s. Hence $\partial RBP(s)$ represents the cutoff of the points where the function receives a backprojection contribution from position s and contains points where s is an endpoint of their π-intervals. We shall give a better justification for this in Lemma 4.3.

The following definition is also central to describing the shape of the support of the comet tail artifact.

Definition 4.2. *A point* $\mathbf{x}$ *is in* $\Gamma_{\mathbf{x}_0}$ *if there exists* $s \in \mathbb{R}$ *such that* $\mathbf{x} \in \partial' RBP(s)$ *and* $\mathbf{x}$, $\mathbf{x}_0$ *and* $\mathbf{y}(s)$ *are collinear. Equivalently* $\mathbf{x}_0 \in L_\pi(\mathbf{x})$. *If* π*-lines are unique then* $L_\pi(\mathbf{x}) = L_\pi(\mathbf{x}_0)$.

Suppose the support of f is a small region around $\mathbf{x}_0$, ie f is an approximate δ function centered at $\mathbf{x}_0$. Then for a given s the filtered data $g * k$ will be large for the line connecting $\mathbf{x}_0$ to $\mathbf{y}(s)$ and a contribution to the artifact occurs at the intersection of this line with $\partial RBP(s)$. Alternatively we can define $\Gamma_{\mathbf{x}_0}$ as

$$\begin{aligned} \Gamma_{\mathbf{x}_0} &= \cup_s [\partial' RBP(s) \cap L(\mathbf{x}_0, \mathbf{y}(s))] \\ &= \{\mathbf{x} \mid \exists s, \mathbf{x} \in \partial' RBP(s) \text{ and } \mathbf{x}_0, \mathbf{x}, \text{ and } \mathbf{y}(s) \text{ are collinear}\}. \end{aligned} \tag{4.1}$$

Lemma 4.3. *Suppose* $\mathbf{x} \in \partial' RBP(s)$ *and* $\mathbf{y}(s) = (\cos(s), \sin(s))$ *or* $\mathbf{x} \in \partial RBP(s) \cap S$ *and* $\mathbf{y}(s) = (\cos(s), \sin(s), hs)$ *with* $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$. *If* $s_b(\mathbf{x})$ *and* $s_t(\mathbf{x})$ *are continuous at* $\mathbf{x}$, *then* $\mathbf{x} \in RBP(s)$ *and* $s = s_b(\mathbf{x})$ *or* $s = s_t(\mathbf{x})$.

Proof. We begin with the first statement. Suppose $\mathbf{x}_n \to \mathbf{x}$ and $\mathbf{x}_n \in RBP(s)$ for all n. Let $I_\pi(\mathbf{x}_n) = [b_n, t_n]$ and $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$. By the continuity of $I_\pi(\mathbf{x})$, we have $b_n \to s_b(\mathbf{x})$ and $t_n \to s_t(\mathbf{x})$ as $n \to \infty$. Now for each n we have $b_n \le s \le t_n$ for all n. Hence $s \in [s_b(\mathbf{x}), s_t(\mathbf{x})]$ and $\mathbf{x} \in RBP(s)$.

Suppose $\mathbf{x} \in \partial RBP(s)$. We know $s \in [s_b(\mathbf{x}), s_t(\mathbf{x})]$. Let U be an open neighborhood of $\mathbf{x}$. Then the set $U \backslash RBP(s)$ is nonempty. Let $\mathbf{x}_n \to \mathbf{x}$ and $\mathbf{x}_n \in U \backslash RBP(s)$ for all n. Let $I_\pi(\mathbf{x}_n) = [b_n, t_n]$. Then by construction $s \notin [b_n, t_n]$ for all n and we can construct a subsequence b_{n_k} such that $b_{n_k} > s$ or a subsequence t_{n_k} such that $t_{n_k} < s$.

Suppose $s < b_{n_k}$ for all $k \in \mathbb{N}$, then

$$s \leq \lim_{k\to\infty} b_{n_k} = s_b(\mathbf{x}). \tag{4.2}$$

We have $s \leq s_b(\mathbf{x})$ and $s \in [s_b(\mathbf{x}), s_t(\mathbf{x})]$. Therefore $s = s_b(\mathbf{x})$. The $s > t_{n_k}$ case follows by the same argument. Hence $s = s_b(\mathbf{x})$ or $s = s_t(\mathbf{x})$. □

Corollary 4.4. *Suppose* $\mathbf{x} \in \partial' RBP(s)$. *Then the line segment from* $\mathbf{x}$ *to* $\mathbf{y}(s)$ *is contained in the* π*-line of* $\mathbf{x}$.

Proof. By Lemma 4.3 we know if $\mathbf{x} \in \partial RBP(s)$ then $s = s_b(\mathbf{x})$ or $s = s_t(\mathbf{x})$ where $\mathbf{y}(s_b(\mathbf{x}))$ and $\mathbf{y}(s_t(\mathbf{x}))$ are points on the π-line through $\mathbf{x}$. □

Lemma 4.3 tells us that if $\mathbf{x}$ belongs to the boundary of $RBP(s)$, restricted to Ω or S, then s is an endpoint of the π-interval of $\mathbf{x}$. As we shall see in the next example, the converse is not true in the 2D case.

Example 4.5. *Suppose we have the following* π*-lines*

$$\begin{aligned} I_\pi(\mathbf{x}) &= [s_b(\mathbf{x}), 2\pi] \\ s_b(\mathbf{x}) &= \pi - 2\alpha^*(0, \mathbf{x}) \end{aligned} \tag{4.3}$$

where α^* *is defined in* (1.1).

The construction of the π-line at $\mathbf{x}$ follows from Figure 4.2. We take the line from $\mathbf{y}(0)$ through $\mathbf{x}$. To calculate $s_b(\mathbf{x})$ we use the local detector coordinate of $\mathbf{x}$ at $\mathbf{y}(0)$. Then $0 \in I_\pi(\mathbf{x})$ for all $\mathbf{x} \in \Omega$. Hence a point can be in the interior of $RBP(s)$ when $s = s_b(\mathbf{x})$ or $s = s_t(\mathbf{x})$.

4.2 Region of Backprojection for Helical Scanning Trajectories

It is difficult to analytically describe the region of backprojection for the helical π-lines in the plane $x_3 = 0$ for either the 2D or 3D formula. As seen in Figure 4.3, numerical results show that $RBP(s) \cap \{x_3 = 0\}$ changes its size as s varies and that it does not rotate with the source curve. To prove the following result we shall describe the region of backprojection on surfaces that are not parallel planes.

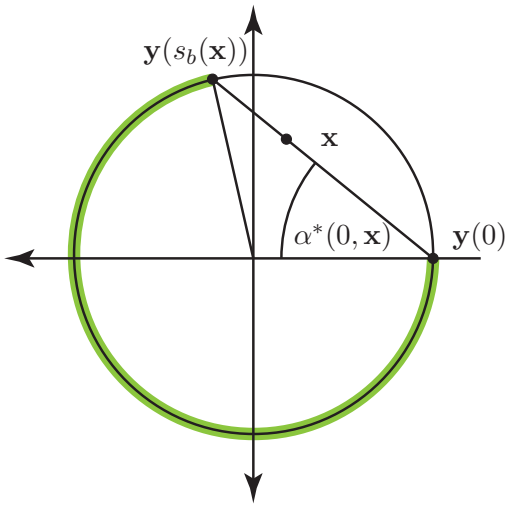

Figure 4.2: Construction of π-lines from Example 4.5.

Lemma 4.6. *Suppose* $\mathbf{y}(s) = (\cos s, \sin s, hs)$ *and* $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$, *then* $s_b(\mathbf{x})$ *and* $s_t(\mathbf{x})$ *are continuous functions for* $\mathbf{x} \in S$.

Proof. Let $\mathbf{x} \in C(s)$, $h = 1$, and suppose $\mathbf{x}_n \to \mathbf{x}$ where $\mathbf{x}$ has chip coordinates (s, ρ, θ) and $\mathbf{x}_n = (s_n, \rho_n, \theta_n)$ where

$$\mathbf{x} = \left(\rho\cos(\theta + s), \rho\sin(\theta + s), s + \frac{\rho\sin\theta\cos^{-1}(\rho\cos\theta)}{\sqrt{1 - \rho^2\cos^2\theta}} \right).$$

As $n \to \infty$, $(s_n, \rho_n, \theta_n) \to (s, \rho, \theta)$ provided $\rho, \rho_n < 1$ for all n.

Now $s_t(\mathbf{x}) = s + \alpha = s + \arccos(\rho\cos(\theta))$. Let $\mathbf{z}_n = (s, \rho_n, \theta_n) \in C(s)$. Then $|s_t(\mathbf{x}) - s_t(\mathbf{x}_n)| \leq |s_t(\mathbf{x}) - s_t(\mathbf{z}_n)| + |s_t(\mathbf{x}_n) - s_t(\mathbf{z}_n)| = |\arccos(\rho\cos(\theta)) - \arccos(\rho_n\cos(\theta_n))| + |s - s_n|$. As $n \to \infty$, $|s_t(\mathbf{x}) - s_t(\mathbf{x}_n)| \to 0$. □

Theorem 4.7. *Suppose* $\mathbf{y}(s) = (\cos s, \sin s, hs)$. *Then* $\mathbf{x} \in \partial' RBP(s)$ *if and only if* $s = s_b(\mathbf{x})$ *or* $s = s_t(\mathbf{x})$.

Theorem 4.7 tells us that $\partial RBP(s)$ intersected with S is the surface of all π-lines that contain $\mathbf{y}(s)$. We note that the $\partial' RBP(s)$ is the union of the lower and upper π-surfaces described in [21]. The proof of Theorem 4.7 depends on the uniqueness of π-lines in the

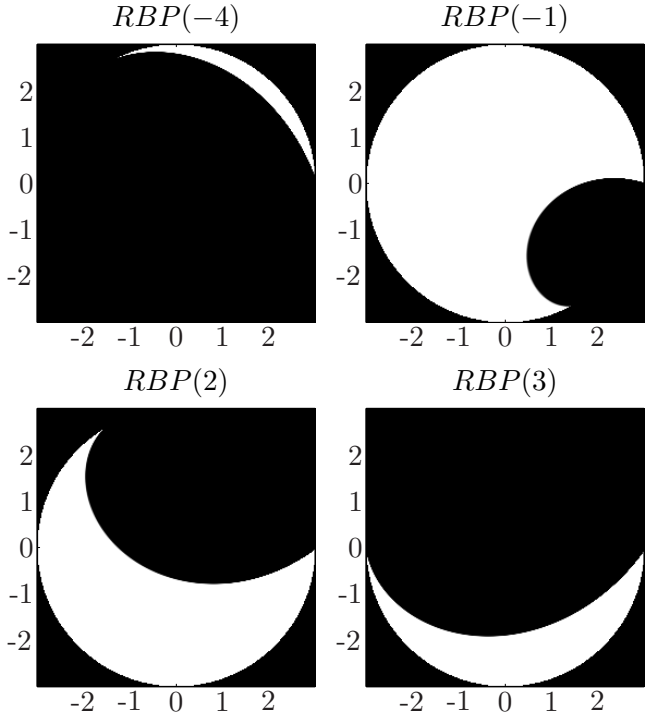

Figure 4.3: Region of backprojection for the 3D formula, white in color, for helical π-lines for the plane $x_3 = 0$ with $R = 3$.

helix by utilizing all points on a π-line have the same π-line. The results of Theorem 4.7 can be generalized to hold for π-lines and $RBP(s)$ with scanning trajectories where the π-lines exist and are unique. To simplify our work will we focus on collections of π-lines that lie in parallel planes. We use the subset of the helix cylinder discussed in Section 3.1 called a chip.

Lemma 4.8. *Let $C(t)$ be a chip anchored at t and suppose $\mathbf{y}(s) = (\cos s, \sin s, hs)$. Then $RBP(s) \cap C(t)$ is*

$$\{\mathbf{x} \in C(t) \mid I_\pi(\mathbf{x}) = [t - \alpha, t + \alpha], |t - s| \leq \alpha < \pi\}$$

and

$$\partial RBP(s) \cap C(t) = \{\mathbf{x} \in C(t) \mid I_\pi(\mathbf{x}) = [t - \alpha, t + \alpha], |t - s| = \alpha < \pi\}.$$

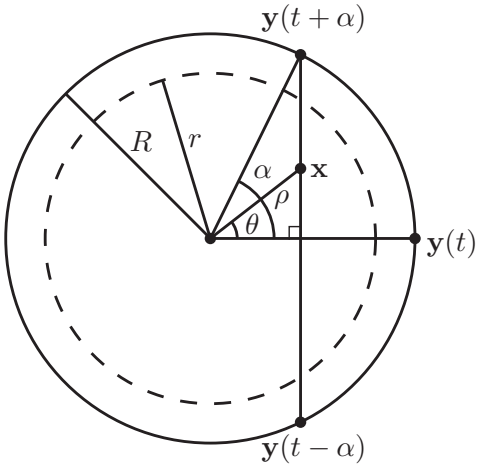

Figure 4.4: $C(t)$ projected onto the xy plane.

Proof. Observe from Figure 4.4

$$RBP(s)\cap C(t) = \{\mathbf{x} \in C(t) \mid s \in I_\pi(\mathbf{x})\} = \{\mathbf{x} \in C(t) \mid I_\pi(\mathbf{x}) = [t-\alpha, t+\alpha], |t-s| \leq \alpha < \pi\}.$$

Let U be an open neighborhood of $\mathbf{x} \in \partial RBP(s) \cap C(t)$. Then the set $U\backslash(RBP(s) \cap C(t))$ is nonempty. Let $\mathbf{x}_n \to \mathbf{x}$ and $\mathbf{x}_n \in U\backslash(RBP(s) \cap C(t))$ for all n where $I_\pi(\mathbf{x}_n) = [s_b(\mathbf{x}_n), s_t(\mathbf{x}_n)] = [t-\alpha_n, t+\alpha_n]$ with $\alpha_n < |t-s|$. By the continuity of s_b and s_t we know

$$\lim_{n\to\infty} s_b(\mathbf{x}_n) = s_b(\mathbf{x})$$
$$\lim_{n\to\infty} s_t(\mathbf{x}_n) = s_t(\mathbf{x})$$

and that $\mathbf{x} \in RBP(s)\cap C(t)$ by Lemma 4.3. Then $I_\pi(\mathbf{x}) = [t-\alpha, t+\alpha]$ with $\alpha = |t-s|$. □

Proof of Theorem 4.7. By Lemma 4.3 and Lemma 4.6 we know $\mathbf{x} \in \partial' RBP(s)$ implies $s = s_b(\mathbf{x})$ or $s = s_t(\mathbf{x})$.

Let $\mathbf{x} \in S$, then there exists a unique $t \in \mathbb{R}$ such that $\mathbf{x} \in C(t)$. Suppose $s = s_b(\mathbf{x})$ then the π-interval of $\mathbf{x}$ is centered at t, with $t > s$, and has length 2α where $\alpha = t - s$.

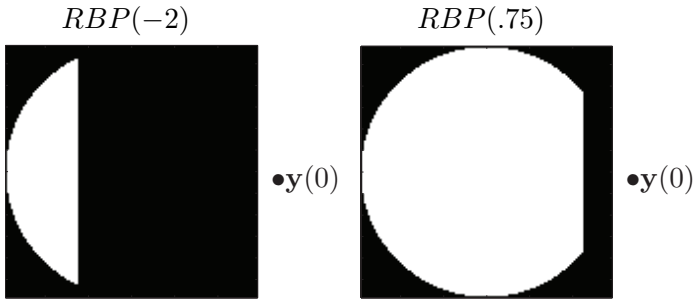

Figure 4.5: On each chip, $RBP(s)$ expands towards the anchor point and then retracts. Shown is $RBP(s) \cap C(0)$.

Thus $I_\pi(\mathbf{x}) = [t-\alpha, t+\alpha] = [s, s+2(t-s)] = [s, 2t-s]$. Therefore $\mathbf{x} \in \partial RBP(s) \cap C(t)$ by Lemma 4.8. The case $s = s_t(\mathbf{x})$ follows in a similar manner. □

The region of backprojection for the chip $C(t)$ expands from $\mathbf{y}(t-\pi)$ to the chip's anchor point $\mathbf{y}(t)$ and then retracts; cf. Figure 4.5. Suppose $s < t$, then the intersection of the boundary of the region of backprojection $RBP(s)$ with the chip $C(t)$ is the π-line from $\mathbf{y}(s)$ to $\mathbf{y}(2t-s)$. Recall that the helix cylinder can be decomposed into disjoint chips. It is now possible to describe the region of backprojection of the entire helix cylinder by the behavior of the $RBP(s)$ on any given chip. Then $RBP(s)$ for the helix cylinder is a rotated and translated copy of $RBP(0)$ because of the same behavior found in the chips. With our description of $RBP(s)$ and $\partial RBP(s)$ on a chip $C(t)$ we can now describe the support of the comet tail artifact on S.

Theorem 4.9. *Let* $\mathbf{y}(s) = (\cos s, \sin s, hs)$*, then* $\Gamma_{\mathbf{x}_0} = L_\pi(\mathbf{x}_0) \cap S$*.*

Proof. Let $\mathbf{x} \in S$. Then $\mathbf{x} \in \Gamma_{\mathbf{x}_0}$ occurs if and only if there exists s such that $\mathbf{x} \in \partial RBP(s)$ and $\mathbf{x} \in L(\mathbf{x}_0, \mathbf{y}(s))$. By Lemmas 4.3 and 4.6 $\mathbf{x} \in \Gamma_{\mathbf{x}_0}$ is equivalent to $s = s_b(\mathbf{x})$ or $s = s_t(\mathbf{x})$ and $\mathbf{x} \in L(\mathbf{x}_0, \mathbf{y}(s))$ for some s. By Corollary 4.4 this occurs if and only if $L(\mathbf{x}_0, \mathbf{y}(s)) = L_\pi(\mathbf{x})$. By the uniqueness of π-lines we have $\mathbf{x}_0 \in L_\pi(\mathbf{x})$ if and only if $\mathbf{x} \in L_\pi(\mathbf{x}_0)$. □

By Theorem 4.9 we have Γ is the set of all π-lines that pass through the support of the function. We see in Figure 4.6 that comet tail artifacts construction along the π-lines that pass through the support of f. Here the π-lines on the chip $C(.4)$ are orthogonal

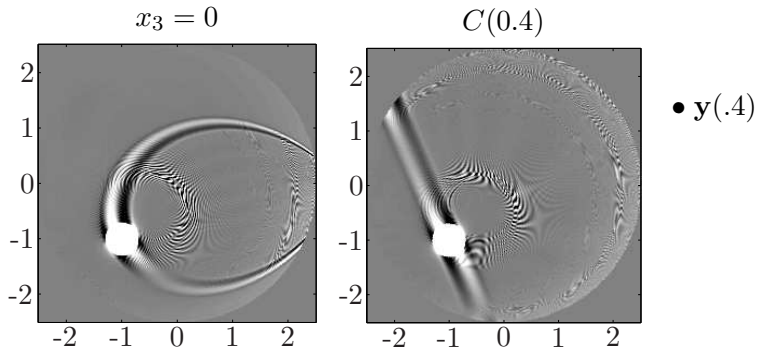

Figure 4.6: **Left:** 3D reconstruction with (1.5) in the plane $x_3 = 0$. **Right:** Same function as before but reconstructed on $C(.4)$. Reconstructed with $P = 2560$, $Q = 579$, $R = 3$, and $D = 6$ and $p = .274$. The image grayscale range is $[-1E-4, 1E-4]$. The reconstructed function is (2.67) with $\mathbf{x}_0 = (-1, -1, 0), u = .3, v = .3, w = 2, m = 3$ and $\psi = 0$.

to the chip axis that passes through the point $\mathbf{y}(.4)$ as shown in Lemma 3.5. Thus the support of the comet tail artifact shown in the plane $x_3 = 0$ is the collection of all π-lines that pass through the support of f and intersects the plane $x_3 = 0$.

4.3 Region of Backprojection for Fan-Beam Geometry

Every point inside Ω lies on a line that passes through the boundary of Ω twice. However the π-lines for a circle are not unique since there exists infinitely many such lines that intersect the scanning circle and $\mathbf{x}$.

For the 2D case we consider a few special families of π-lines. The first correspond to the π-intervals from a helix, modulo 2π, for the points that lie in the plane $x_3 = 0$. We remark that for points in the plane $x_3 = 0$ their π-intervals are independent of the helix pitch $h/(2\pi)$ [8]. See Section 3.3 for a method to calculate the helical π-lines.

Another type of π-lines we use are orthogonal-long π-lines. For the π-interval we take a line through $\mathbf{x}$ that is perpendicular to the line segment from the origin to $\mathbf{x}$. We then select the longest arc corresponding to the intersection of this line and the circle. Let $\mathbf{x}$ have polar coordinates (ρ, θ), then the orthogonal-long π-line interval is given by $s_b(\mathbf{x}) = \theta + \gamma$ and $s_t(\mathbf{x}) = \theta - \gamma$ where $\gamma = \arccos(\rho/R)$. We let $I_\pi(0) = [-\pi, \pi]$ and note

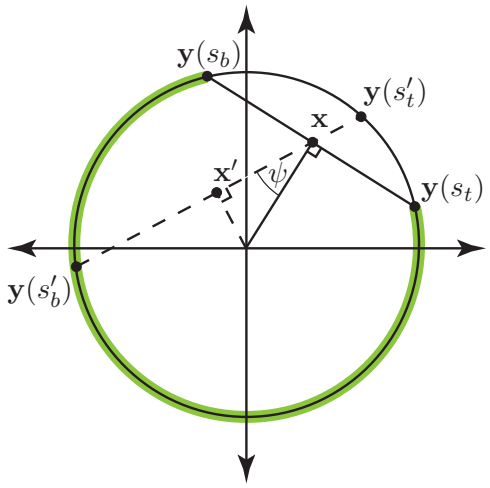

Figure 4.7: Orthogonal-long and tilted-long π-lines

that these π-lines do not vary continuously at the origin. If $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$ is the orthogonal-long π-interval then the orthogonal-short π-interval is $[s_t(\mathbf{x}), s_b(\mathbf{x})]$.

Our next family is tilted-long π-lines where the angle between the radial ray through $\mathbf{x}$ and the line through $\mathbf{x}$ have a fixed angle of intersection ψ. The tilted-long π-line of a point $\mathbf{x}$ is the orthogonal-long π-line for the point $\mathbf{x}'$ where $\mathbf{x}' = A\mathbf{x}$ and A is given by the matrix

$$A = \sin(\psi) \begin{bmatrix} \sin(\psi) & -\cos(\psi) \\ \cos(\psi) & \sin(\psi) \end{bmatrix} \tag{4.4}$$

with $I_\pi(\mathbf{0}) = [-\pi, \pi]$. We derive $A\mathbf{x}$ as the rotation of $\mathbf{x}$ by $\psi - \pi/2$ and dilated by $\sin(\psi)$. For $\psi = 0$ the matrix A is not defined as above because we don't have continuity of the orthogonal-long π-lines at the origin. We define the radial π-lines, $\psi = 0$, as the π-intervals $I_\pi(\mathbf{x}) = [\theta, \theta + \pi]$ with $I_\pi(\mathbf{0}) = [-\pi/2, \pi/2]$.

We now construct $RBP(s)$ for orthogonal and tilted π-lines in the fan-beam geometry. Our first result deals with nesting of orthogonal-long π-intervals.

Lemma 4.10. *Let* $\mathbf{x}$ *and* $\mathbf{y}$ *be points inside the unit circle, neither of them* $\mathbf{0}$*, with polar coordinates* $(\rho_\mathbf{x}, \theta)$ *and* $(\rho_\mathbf{y}, \theta)$ *respectively. Furthermore let* $I_\pi(\mathbf{x}) = [s_1, s_2]$ *and* $I_\pi(\mathbf{y}) = [t_1, t_2]$ *be the* π*-intervals corresponding to the orthogonal-long family of* π*-lines.*

If $\rho_{\mathbf{x}} < \rho_{\mathbf{y}}$, then $I_\pi(\mathbf{x}) \subset I_\pi(\mathbf{y})$. For the orthogonal-short π-lines we have $I_\pi(\mathbf{x}) \supset I_\pi(\mathbf{y})$.

Proof. We shall only focus on the orthogonal-long π-lines. The case of the orthogonal-short π-lines follows a similar argument. Let $\mathbf{x}$ have polar coordinates (ρ, θ). Recall that the orthogonal-long π-intervals are defined as $s_b(\mathbf{x}) = \theta + \gamma$ and $s_t(\mathbf{x}) = \theta - \gamma$ where $\gamma = \arccos(\rho/R)$.

The radial components satisfy $0 < \rho_{\mathbf{x}} < \rho_{\mathbf{y}} < 1$ and thus $\arccos(\rho_{\mathbf{x}}) > \arccos(\rho_{\mathbf{y}})$ since arccos is a decreasing function on $[0, 1]$. Hence $s_1 = \theta + \arccos(\rho_{\mathbf{x}}) \geq \theta + \arccos(\rho_{\mathbf{y}}) = t_1$ and $s_2 = \theta - \arccos(\rho_{\mathbf{x}}) \leq \theta - \arccos(\rho_{\mathbf{y}}) = t_2$. Thus $I_\pi(\mathbf{x}) \subset I_\pi(\mathbf{y})$. □

The following is an immediate consequence and is shown in Figure 4.8.

Corollary 4.11. *For the π-lines of the orthogonal-long type we have*

$$RBP(s) = \begin{cases} \Omega \cap \overline{D(s)^c} & s \in [-\pi/2, \pi/2] \\ \Omega \cap \overline{D(s)^c} \backslash \{\mathbf{0}\} & s \notin [-\pi/2, \pi/2] \end{cases} \tag{4.5}$$

where $D(s) = \{\mathbf{x} : |\mathbf{x} - \mathbf{c}| \leq 1/2\}$ and $\mathbf{c} = (\cos(s)/2, \sin(s)/2)$. For the orthogonal-short π-lines we have

$$RBP(s) = \begin{cases} \Omega \cap D(s) & s \in [-\pi/2, \pi/2] \\ \Omega \cap D(s) \backslash \{\mathbf{0}\} & s \notin [-\pi/2, \pi/2] \end{cases} \tag{4.6}$$

as shown in Figure 4.8.

Proof. Again, we shall only focus on the orthogonal-long π-lines. It suffices to show the result holds for $s = 0$ modulo 2π. Let $\mathbf{x} = (\rho, \theta)$ and suppose $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$.

If $\pi/2 < \theta < 3\pi/2$ then $\pi/2 < s_b(\mathbf{x}) < 2\pi$ and $s_t(\mathbf{x}) = \theta - \gamma$ where $\gamma = \arccos(\rho)$. In particular $s_t(\mathbf{x}) > 0$ since $\pi/2 < \theta < 3\pi/2$ and $\gamma \in [0, \pi/2]$. Hence $s \in [s_b(\mathbf{x}), s_t(\mathbf{x})]$.

Now let $0 \leq \theta \leq \pi/2$. Suppose $\mathbf{x}'$ is the point of the form (ρ', θ) with $I_\pi(\mathbf{x}') = [s_b(\mathbf{x}'), s_t(\mathbf{x}')]$ such that $s_t(\mathbf{x}') = 0$ which occurs if and only if $\cos(\theta) = \rho'$. Then $0 \in [s_b(\mathbf{x}'), s_t(\mathbf{x}')]$ and by Lemma 4.10 we have $0 \in [s_b(\mathbf{x}), s_t(\mathbf{x})]$ provided $\rho > \rho'$. If $\rho < \rho'$ then

$$\begin{aligned} s_t(\mathbf{x}) &= \theta - \arccos(\rho) \\ &= \arccos(\rho') - \arccos(\rho) \\ &< 0 \end{aligned} \tag{4.7}$$

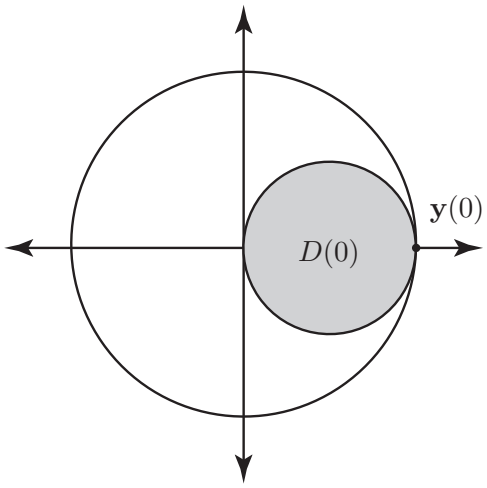

Figure 4.8: Region of backprojection for orthogonal-long π-lines is the region inside the curve $\mathbf{y}(s)$ and outside of $D(s)$.

since arccos is decreasing on $[0,1]$. Thus $0 \notin [s_b(\mathbf{x}), s_t(\mathbf{x})]$ if $\rho < \cos(\theta)$. A similar argument holds if $3\pi/2 \leq \theta < 0$. Thus $\{(\rho,\theta) \mid -\pi/2 \leq \theta \leq \pi/2, \cos(\theta) \leq \rho\} = D(s) \cap \Omega$. □

For each source position the points where we reconstruct using orthogonal-long π-lines in formula (1.12) lie in the region (4.5). This region is shown in Figure 4.8.

Corollary 4.12. *Consider* $\mathbf{y}(s) = (\cos s, \sin s)$ *with the* π*-lines of the tilted-long type. Let* $s \in [0, 2\pi)$*, then* $s \in I_\pi(\mathbf{x})$*, for all* $\mathbf{x} \in \Omega \cap \overline{D(s)^c}$ *where* $D(s) = \{\mathbf{x} \in \Omega \mid \|\mathbf{x} - A^{-1}c\| \leq \frac{1}{2\sin(\psi)}\}$*,* A *defined by* (4.4)*,* $c = (\cos(s)/2, \sin(s)/2)$*.*

Proof. Recall

$$A = \sin(\psi) \begin{bmatrix} \sin(\psi) & -\cos(\psi) \\ \cos(\psi) & \sin(\psi) \end{bmatrix}.$$

is the matrix that transforms orthogonal-long π-lines to tilted-long π-lines. We will show $\{\mathbf{x} \in \Omega \mid |\mathbf{x} - A^{-1}c| \leq \frac{1}{2\sin(\psi)}\} = \{\mathbf{x} \in \Omega \mid |A\mathbf{x} - c| \leq \frac{1}{2}\}$. By construction $\mathbf{x}$ is the point on the tilted π-line l such that l is the orthogonal π-line of $A\mathbf{x}$. By Corollary 4.11 we know $s \in I_\pi(\mathbf{x})$ if $\mathbf{x} \in \Omega \cap \{\mathbf{x} \in \Omega \mid |A\mathbf{x} - c| \geq \frac{1}{2}\}$. For A as above we have

$A^* = \sin^2(\psi)A^{-1}$. Now $|A\mathbf{x} - c| \leq \frac{1}{2}$ yields

$$\begin{aligned}\frac{1}{4} &\geq \langle A\mathbf{x} - c, A\mathbf{x} - c\rangle \\ &= \langle A\mathbf{x} - AA^{-1}c, A\mathbf{x} - c\rangle \\ &= \langle \mathbf{x} - A^{-1}c, A^*A\mathbf{x} - A^*c\rangle \\ &= \langle \mathbf{x} - A^{-1}c, \sin^2(\psi)\mathbf{x} - \sin^2(\psi)A^{-1}c\rangle \\ &= \sin^2(\psi)\langle \mathbf{x} - A^{-1}c, \mathbf{x} - A^{-1}c\rangle.\end{aligned}$$

Hence $|A\mathbf{x} - c| \leq \frac{1}{2}$ if and only if $|\mathbf{x} - A^{-1}c| \leq \frac{1}{2\sin(\psi)}$. Thus $\mathbf{x} \in \Omega \cap \{\mathbf{x} \mid |\mathbf{x} - A^{-1}c| \geq \frac{1}{2\sin(\psi)}\}$ implies $s \in I_\pi(\mathbf{x})$.

If $\psi = 0$ then A and A^{-1} are not defined. However it is still easy to see that $D(0)$ is the right half of the plane rotated about the origin as s increases. □

The rotation of the $RBP(s)$ comes from a fundamental difference between the π-intervals from the helix in the plane x_3 and those of the orthogonal-long type. Every π-interval from the helix will contain the source position that intersects the $x_3 = 0$ plane since $\mathbf{y}(s_b(\mathbf{x}))$ is below the $x_3 = 0$ plane and $\mathbf{y}(s_t(\mathbf{x}))$ is above it. The orthogonal-long π-lines do not have this property, the intersection of all π-intervals is empty. However these two choices π-intervals share a feature not found in the collection of all helical π-lines. No two points lie on the same π-line for helical or orthogonal-long π-lines. If we do not restrict the helical π-lines to points in the plane $x_3 = 0$ then this is no longer the case. We showed in Section 4.2 that $\partial RBP(s)$ for the helix is a surface of π-lines. The $\partial RBP(s)$ for the orthogonal-long π-lines is a circle and is therefore not a union of π-lines.

For the orthogonal-long π-lines we have $RBP(s)$ is a convex set and therefore $\Gamma_{\mathbf{x}_0}$ has only one point $\mathbf{x}$ for every s such that $\mathbf{x}_0 \in L_\pi(\mathbf{x})$ and $\mathbf{x} \in \partial' RBP(s)$. We then end up with a curve γ that identifies the set $\Gamma_{\mathbf{x}_0}$. More precisely

$$\gamma(s, \mathbf{x}_0) = \{\mathbf{z} \in \partial' RBP(s) \mid \alpha^*(s, \mathbf{x}_0) = \alpha^*(s, \mathbf{z})\} = \partial' RBP(s) \cap L(\mathbf{x}_0, \mathbf{y}(s)) \qquad (4.8)$$

and by (4.1)

$$\Gamma_{\mathbf{x}_0} = \cup_s \gamma(s, \mathbf{x}).$$

Theorem 4.13. *Suppose* $\mathbf{y}(s) = (\cos s, \sin s)$. *Then for the* π*-lines of the orthogonal-long or short type we have* $\Gamma_{\mathbf{x}_0}$ *is a circle centered at* $\mathbf{x}_0/2$ *with radius* $|\mathbf{x}_0/2|$.

Theorem 4.13 predicts that the comet tail artifact for the orthogonal-long and short π-lines is a circle centered halfway between the object and the origin. In Figure 4.10 and Figure 4.1 we observe that the artifact changes with our choice of π-lines. The artifact for the orthogonal-long π-lines is located where our theory predicts.

Proof. We will use a geometric argument, that follows from Figure 4.9, to argue that $\Gamma_{\mathbf{x}_0}$ is a circle. Let $A = (x_0, y_0), C = (0, 0)$, and $G = \mathbf{y}(s)$. Let B be the midpoint between A and C and let D be the midpoint between C and E. Note that E is the point on the boundary of $D(s)$ and the line connecting A to the source position G. Let F be the center of $D(s)$.

The line segment CE is perpendicular to line segment AG since E lies on a circle centered between points C and G. We remark that AG is on the π-line of E and that E is the location of the artifact, ie $E = \gamma(s, \mathbf{x}_0)$. Triangle ACE is similar to triangle BCD. Furthermore triangle BCD is congruent to triangle BED. In particular we have that line segments BA, BC and BE have the same length.

We have shown that the distance between A and B is equal to the distance between E and B for all $s \in [0, 2\pi)$. Thus $\Gamma_{\mathbf{x}_0}$ is a circle centered at B with radius $d/2$.

□

The following result gives us a parametrization of Γ with respect to s.

Corollary 4.14. *Suppose* $\mathbf{y}(s) = (\cos s, \sin s)$. *Then for the* π*-lines of the orthogonal-long or short type we have*

$$\begin{aligned} \Gamma_{\mathbf{x}_0} &= \cup_s \gamma(s, \mathbf{x}_0) \\ \gamma(s, \mathbf{x}_0) &= R(s) z(s) \\ R(s) &= \begin{bmatrix} \cos(s) & -\sin(s) \\ \sin(s) & \cos(s) \end{bmatrix} \\ z(s, \mathbf{x}_0) &= \begin{bmatrix} \frac{1}{2}\cos(\pi - 2\alpha^*(s, \mathbf{x}_0)) + \frac{1}{2} \\ \frac{1}{2}\sin(\pi - 2\alpha^*(s, \mathbf{x}_0)) \end{bmatrix}. \end{aligned} \tag{4.9}$$

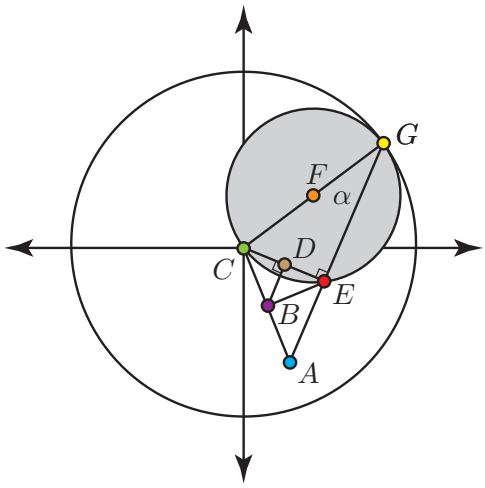

Figure 4.9: Diagram for proof of Theorem 4.13 with $A = \mathbf{x}_0, B = (x_0/2, y_0/2), C = \mathbf{0}$ and $G = \mathbf{y}(s)$.

where $\Gamma_{\mathbf{x}_0} = \cup_s \gamma(s, \mathbf{x}_0)$. *The term* α^* *is the curved detector coordinate defined by* (1.1).

Proof. The argument follows from Figure 4.9. We may view the curve γ in terms of the fan-beam detector coordinates. Let $s = 0$. We shall calculate the position of E along the boundary of $RBP(0)$. First we calculate $\angle EFG$, the angle between $\mathbf{x}_0$'s projection onto the boundary of $D(0)$, the center of $D(0)$, and $\mathbf{y}(0)$. Hence $\angle EFG = \pi - 2\alpha^*(0, \mathbf{x}_0)$ and the position of E in terms of $\partial' RBP(0)$ is given by

$$E = \begin{bmatrix} \frac{1}{2}\cos(\pi - 2\alpha^*(0, \mathbf{x}_0)) + \frac{1}{2} \\ \frac{1}{2}\sin(\pi - 2\alpha^*(0, \mathbf{x}_0)) \end{bmatrix}. \tag{4.10}$$

For $s \in (0, 2\pi)$ we exploit that $RBP(s)$ of the orthogonal-long π-lines rotates about the origin counterclockwise with respect to s. Now $\angle EFG$ is given by $\pi - 2\alpha^*(s, \mathbf{x}_0)$ and the position of $\partial RBP(s)$ is $R(s)\partial D(0)$ where

$$R(s) = \begin{bmatrix} \cos(s) & -\sin(s) \\ \sin(s) & \cos(s) \end{bmatrix} \tag{4.11}$$

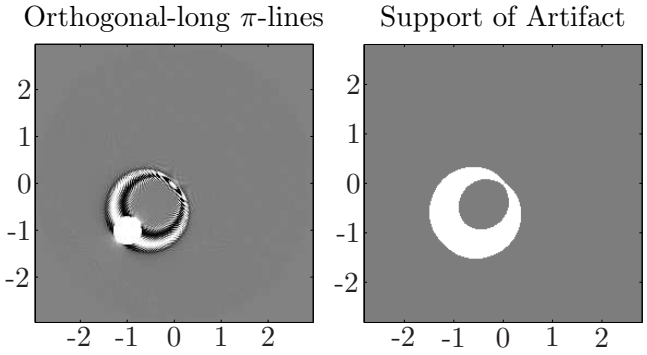

Figure 4.10: **Left:** Comet tail artifact for orthogonal-long π-lines Reconstructed with $P = 2560$, $Q = 668$, $R = 3$, and $D = 6$. The image grayscale range is $[-8E - 5, 8E - 5]$. The reconstructed function is (2.67) with $\mathbf{x}_0 = (-1, -1), u = .3, v = .3, m = 3$ and $\psi = 0$. **Right:** Support of comet tail artifact predicted by Theorem 4.13.

is a rotation matrix. Let $z = R^{-1}(s)E$. Next we calculate the position of z along the boundary of $R^{-1}(s)D(s)$, which is $\partial D(0)$,

$$z(s, \mathbf{x}_0) = \begin{bmatrix} \frac{1}{2}\cos(\pi - 2\alpha^*(s, \mathbf{x}_0)) + \frac{1}{2} \\ \frac{1}{2}\sin(\pi - 2\alpha^*(s, \mathbf{x}_0)) \end{bmatrix}. \tag{4.12}$$

Thus

$$\gamma(s, \mathbf{x}_0) = E = R(s)z(s, \mathbf{x}_0). \tag{4.13}$$

□

The change of coordinate method used to construct $\gamma(s, \mathbf{x}) = R(s)z(s, \mathbf{x})$ in the above proof allows us to determine where the comet tail artifact will appear when we use tilted-long π-lines. We need only use the description of $D(s)$ given to us by Corollary 4.3 and reparametrize z so that it corresponds to $\partial' RBP(s)$ of the tilted-long π-lines. The only drawback to this method is that it will not lead to a simple geometric argument that determines the shape of the comet tail artifact. Instead, it will give a parameterization of the artifact with respect to s.

This change of coordinates method found in Corollary 4.14 will work for any $RBP(s)$ that is convex, connected and has a fixed shape that rotates around the origin with

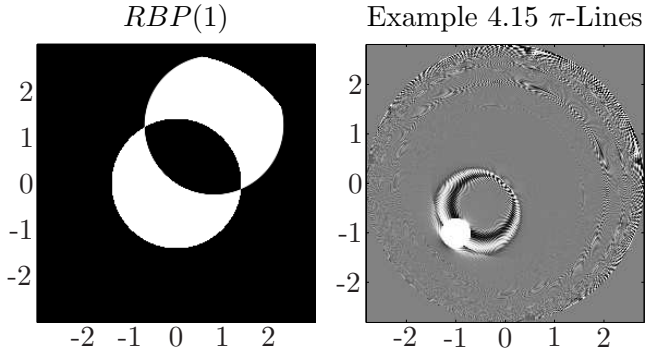

Figure 4.11: **Left:** Region of backprojection, white in color, for the π-lines given by Example 4.15 with $R = 3$ for points with $|\mathbf{x}| < 2.8$. **Right:** Reconstructed by (1.12) with $P = 2560$, $Q = 579$, $R = 3$, and $D = 6$. The image grayscale range is $[-8E - 5, 8E - 5]$. The reconstructed function is (2.67) with $\mathbf{x}_0 = (-1, -1), u = .3, v = .3, m = 3$ and $\psi = 0$.

respect to s. This class of region of backprojections is not limited to the ones we have described thus far. The next two sets of π-lines violate one of the conditions above.

Example 4.15. *Let $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$ correspond to π-lines of the orthogonal-long type . Consider the following set of π-lines*

$$I'_\pi(\mathbf{x}) = \begin{cases} [s_b(\mathbf{x}), s_t(\mathbf{x})] & \text{if } |\mathbf{x}| < \frac{R}{2} \\ [s_t(\mathbf{x}), s_b(\mathbf{x})] & \text{if } |\mathbf{x}| \geq \frac{R}{2} \end{cases} . \tag{4.14}$$

Then the $RBP(s)$ is $\Omega \cap \overline{D(s)^c}$ if $|\mathbf{x}| < \frac{R}{2}$ and is $\Omega \cap D(s)$ if $|\mathbf{x}| \geq \frac{R}{2}$. As seen in Figure 4.11, the $RBP(s)$ need not be connected and thus the construction of $\gamma(s, \mathbf{x})$ will not work as in Theorem 4.13. We also observe that the support of the comet tail artifact is the same as we see in Figure 4.10 with orthogonal-long π-lines. This occurs because Corollary 4.11 tells us the boundary of the region of backprojections for orthogonal-long and short π-lines is the same. However the values of comet tail artifact from Example 4.15 are clearly different from that of orthogonal-long..

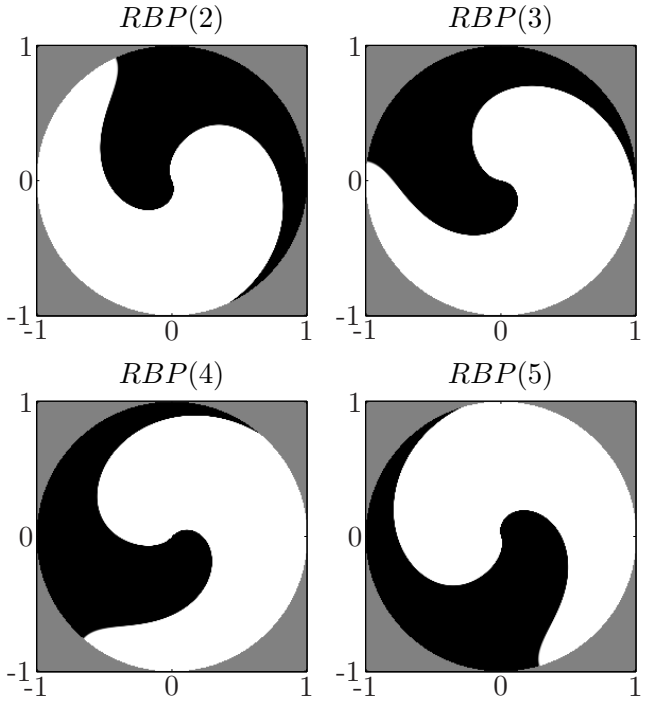

Figure 4.12: Region of backprojection, white in color, for the π-lines given by Example 4.16 with $R = 1$.

Example 4.16. *Let* $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$ *correspond to* π*-lines of the orthogonal-long type and* (ρ, θ) *be the polar coordinates of* $\mathbf{x}$*. Let*

$$I'_\pi(\mathbf{x}) = I_\pi(A\mathbf{x}) \tag{4.15}$$

where $\psi = (R - \rho)\pi$ *and*

$$A = \sin(\psi) \begin{bmatrix} \sin(\psi) & -\cos(\psi) \\ \cos(\psi) & \sin(\psi) \end{bmatrix}$$

As we see in Figure 4.12, $RBP(s)$ for 4.16 is not convex.

5 Aligning CT Data

A CT reconstruction requires two key pieces of information from the x-ray data g. The first is the value of the line integral from the position s in the direction α, $g(s, \alpha)$. The value of the angle of measurement α is as important as the measured value $g(s, \alpha)$ because errors in α can reduce image resolution. The angle α is given and we calculate $g(s, \alpha)$. In a x-ray machine the value $g(s, \alpha)$ is given by the measurements in the detector bin. The detector bin counts the number of photons that arrived since the last measurement at that detector bin was recorded. The photon count from the bin allows us to measure the loss of x-ray photons as they travel through the object and reach the detector. If $G_0(s, \alpha)$ is the number of photons emitted from the x-ray source towards the detector bin in direction α and $G_m(s, \alpha)$ is the measured photon count at that detector bin then

$$g(s, \alpha) = -\log\left(\frac{G_m(s, \alpha)}{G_0(s, \alpha)}\right). \tag{5.1}$$

The terms G_0 and G_m are recorded by the CT machine and estimate $g(s, \alpha)$. The values α are provided as well, but are not always stored with G_0 and G_m. We want to study the situation where $g(s, \alpha)$ is known and we only have approximate information about α. In practice each CT machine is calibrated using a calibration phantom. The measurements are saved and used for the following reconstructions. We shall see that our method proposed in Section 5.2 works for phantoms not intended for calibration.

A few assumptions about α can be made from the scanning acquisition pattern. First, the detectors are placed on a gantry that rotates around the patient. The location of the detector bin at angle α_i relative to the bin at α_j is fixed in s for all i and j. Furthermore we will assume that we know the angular spacing between all the adjacent detectors. For our purposes the spacing is uniform, ie $\Delta\alpha = \alpha_{i+1} - \alpha_i$ for all i.

We must locate the correct value of α_i for all i. Our assumptions on the fan angles reduces the problem to finding α_i for some i. We can then characterize our problem as identifying the detector bin in our measured data that corresponds to $\alpha = 0$. We call the detector bin located at $\alpha = 0$ the center of rotation of the projection data because,

in our coordinate system, the line with angle $\alpha = 0$ passes through the axis of rotation of $\mathbf{y}(s)$.

In Section 5.1 we introduce an error term to estimate the behavior of the reconstruction formulas when the x-ray data is uniformly shifted in the α coordinate. A method to correct the reconstructions from shifted data is presented for the fan-beam geometry in Section 5.2.

5.1 Shifted Data with Katsevich's 2D Formula

We are motivated by the reconstructions of a Siemen's calibration phantom by Katsevich's 2D formula (1.12) in Figure 5.1. We have 512 x-rays equally spaced over 40° from each source position. Here we have shifted the location of the x-ray detector bins to the right by either zero or one detector. The center of rotation is assumed to be the 256th detector bin. The effects of the shift are apparent as the image with a detector shift of zero in Figure 5.1 has a comet tail artifact originating from the densest block in the phantom, the block that is all white. A smaller comet tail artifact appears in the reconstruction with a shift of 1 detector bin. A reconstruction from the full scan formula (1.13) does not have any comet tail artifacts. Our theory from Section 4 implies that we should not expect to see any comet tail artifacts in the reconstruction because the region of backprojection for the full scan (1.13) is all points inside the scanning circle. Our goal is to verify that the comet tail artifact is present in the reconstructions from (1.12) with misaligned data and if we can use its presence to identify the center of rotation of the projection data $g(s, \alpha)$. Contrast this with the reconstructions in Figure 5.1 in the bottom row which lack any comet tail artifact. The reconstructions from (1.13) appear nominal with the reconstruction with a shift of zero having slightly smoother straight edges.

Suppose that our measured x-ray data has a uniform alignment error in the α coordinate. That is for some $h \in \mathbb{R}$ we have $g_h(s, \alpha) = g(s, \alpha + h)$. Let $f_h(\mathbf{x})$ be the reconstruction from g_h. The question is how does this error term propagate in the reconstruction of f_h from Katsevich's formula (1.12)? Assume that we have smooth data $g(s, \alpha)$. Then by a Taylor series with respect to α we have

$$G_h(s, \alpha) = g(s, \alpha) + h\frac{\partial g}{\partial \alpha}(s, \alpha) \tag{5.2}$$

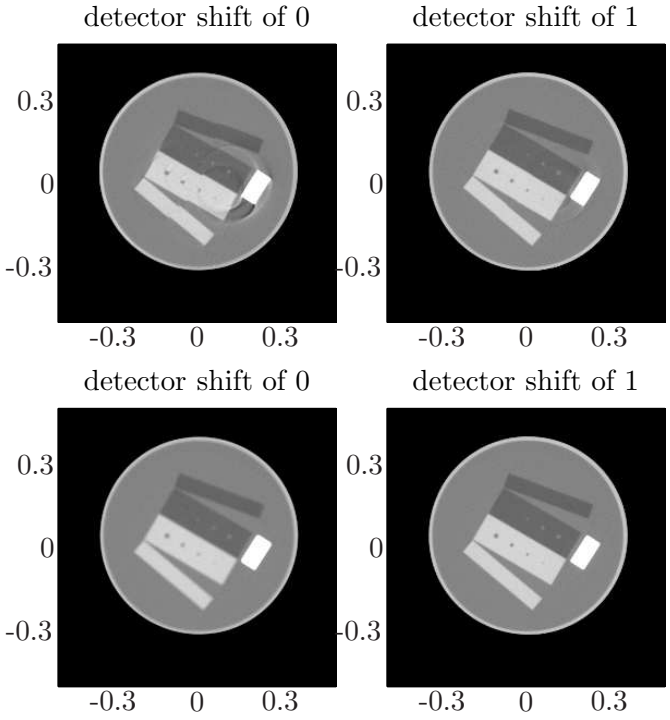

Figure 5.1: Calibration phantom from the 1980s. Shift in the data is relative to the 256th detector bin. **Top Row:** Reconstructions from π-line formula (1.12). **Bottom Row:** Reconstructions from 2π formula (1.13).

where $G_h(s,\alpha) \approx g_h(s,\alpha)$. The reconstruction of $G_h(s,\alpha)$ gives us

$$\begin{aligned} f_h(\mathbf{x}) &\approx \frac{1}{2\pi^2}\int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x}-\mathbf{y}(s)|}\int_0^{2\pi}\left(\frac{\partial G_h}{\partial s}+\frac{\partial G_h}{\partial \alpha}\right)k(\alpha^*-\alpha)\,d\alpha\,ds \\ &= \frac{1}{2\pi^2}\int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x}-\mathbf{y}(s)|}\Bigg[\int_0^{2\pi}\left(\frac{\partial}{\partial s}(g(s,\alpha)+h\frac{\partial g}{\partial \alpha}(s,\alpha))\right)k(\alpha^*-\alpha)\,d\alpha \\ &\quad +\int_0^{2\pi}\left(\frac{\partial}{\partial \alpha}(g(s,\alpha)+h\frac{\partial g}{\partial \alpha}(s,\alpha))\right)k(\alpha^*-\alpha)\,d\alpha\Bigg]ds. \end{aligned} \tag{5.3}$$

Now we have an error term

$$\begin{aligned} e(h) &= f(\mathbf{x}) - f_h(\mathbf{x}) \\ &\approx -\frac{h}{2\pi^2} \int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} \left(\frac{\partial}{\partial \alpha} \frac{\partial g}{\partial s}(s, \alpha) + \frac{\partial^2 g}{\partial \alpha^2}(s, \alpha) \right) \\ &\quad k(\alpha^* - \alpha)\, d\alpha\, ds. \end{aligned} \tag{5.4}$$

After taking an integration by parts with respect to α, we have

$$e(h) \approx -\frac{h}{2\pi^2} \int_{I_\pi(\mathbf{x})} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} \left(\frac{\partial g}{\partial s}(s, \alpha) + \frac{\partial g}{\partial \alpha}(s, \alpha) \right) k'(\alpha^* - \alpha)\, d\alpha\, ds. \tag{5.5}$$

seeing that the boundary term vanishes since g is the x-ray data of a compactly supported function. This gives us a new formula with the same structure as Katsevich's reconstruction formula (1.12) but with a different convolution kernel. To implement (5.5) it is necessary to discretize the new kernel. We follow our method from Section 6.1.2 by writing the kernel as

$$\frac{1}{\sin(\alpha)} = \frac{\alpha}{\alpha \sin(\alpha)} \tag{5.6}$$

and then use the following band-limited approximation for our convolution kernel

$$\frac{1}{\sin(\alpha)} \approx \frac{1 - \cos(b\alpha)}{\sin(\alpha)} \tag{5.7}$$

where b is our cut-off frequency. Differentiating with respect to α gives us our new kernel

$$k'(\alpha) \approx \frac{b \sin(b\alpha)}{\sin(\alpha)} - \frac{(1 - \cos(b\alpha)) \cos(\alpha)}{\sin^2(\alpha)} \tag{5.8}$$

since differentiating and bandlimiting a function commute. We must use a Taylor series to approximate the new kernel near 0.

Let us briefly study how accurate our approximation of the shifted data is. We let O be the mathematical phantom (2.67). We recover $f(\mathbf{x})$ with $P = 360$, $Q = 591$ and

h	$\|f_h - O\|$	$\|f_h + e(h) - O\|$	$\|f - O\|$
$\frac{\Delta\alpha}{8}$	0.021905	0.0026654	0.0049773
$\frac{\Delta\alpha}{4}$	0.043132	0.0011116	0.0049773
$\frac{\Delta\alpha}{2}$	0.085889	0.00078741	0.0049773
$\Delta\alpha$	0.17097	0.0049358	0.0049773

Table 5.1: Relative l_2 errors of reconstructions for shifted x-ray data. Here $f(\mathbf{x})$ and $f_h(\mathbf{x})$ are reconstructed by (1.12) with no shift in α and a shift of h respectively. The error term $e(h)$ is calculated by (5.5) and O is the function (2.67). For each reconstruction we have $P = 360$, $Q = 591$ and $R = 3$.

$R = 3$ on 256 by 256 grid. We then calculate the relative l_2 error

$$\|f - O\| = \left(\sum_{i,j} \frac{(f(x_i, y_j) - O(x_i, y_j))^2}{O(x_i, y_j)^2} \right)^{1/2}. \tag{5.9}$$

For this experiment we also reconstruct $f_h(\mathbf{x})$ with a shift of $h = 2^{-j}\Delta\alpha$ for $j = 0, 1, 2, 3$. As we can see in Table 5.1, error term $e(h)$, calculated by (5.5), does well as a local approximation for the errors introduced by shifted data. In each case the term $f_h(\mathbf{x})$ corrected by $e(h)$ has a lower l_2 error than the reconstruction of $f(\mathbf{x})$ from properly centered data.

Furthermore, as seen in Figure 5.2, the approximate error term for $e(h)$, (5.5), and $e(h)$ contain the comet tail artifacts found in the calibration phantom reconstruction. The images of (5.5) and $e(h)$ are very similar and thus validating the approximation used in (5.2). The placement of the comet tail artifact varies with our selection of $I_\pi(\mathbf{x})$ and the artifact appears in the locations predicted by Theorem 4.13. We note that the presence of the comet tail artifact from Katsevich's 2D formula (1.12) suggests that the center of rotation of the x-ray data is incorrectly calculated. The evidence for comet tail artifacts implying incorrectly calculated center of rotation is strengthened when the object function is discontinuous. Our experiments have shown the comet tail artifact discussed in Section 4 is not easily visible in well aligned projection data of discontinuous functions. We have observed that the comet tail artifact, from misaligned data, is of the same order of magnitude as the density values of the function and thus a leading source of error.

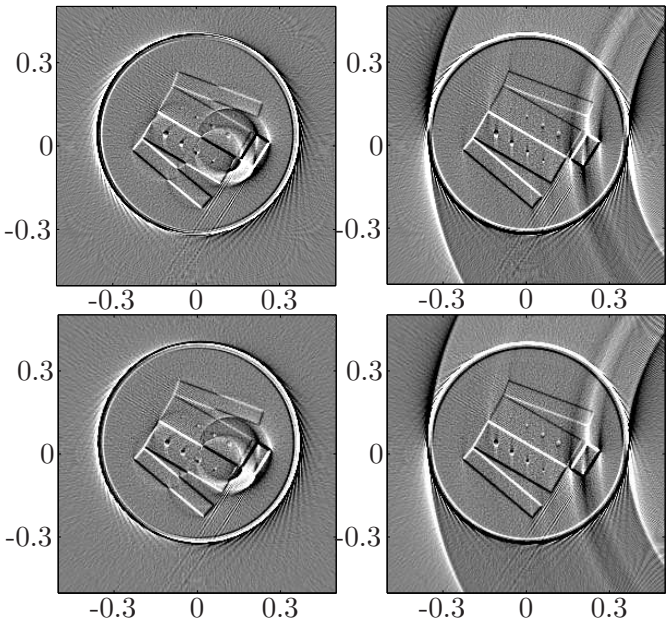

Figure 5.2: **Top Left:** Approximate error term (5.5) for $e(h)$ with orthogonal-long π-lines. **Top Right:** Approximate error term (5.5) for $e(h)$ with helical π-lines. **Bottom Left:** Actual error term $e(h)$ with orthogonal-long π-lines. **Bottom Right:** Actual error term $e(h)$ with helical π-lines.

5.2 Centering Fan-Beam Data

One method to calibrate fan-beam data is to use a phantom with small support and applying a nonlinear optimization method [9]. A common method to centering CT data involves a series of reconstructions with different values of h and then selecting the reconstruction that looks the best to the viewer. Typically, reconstructions from improperly centered data will have edges that are not well resolved. In the case of the calibration phantom, shown in Figure 5.1, the straight edges in the phantom have a kink along the longer sides of the rectangular objects. In Figure 5.2 the kink can also be seen in the error term for the orthogonal-long π-lines as a change in sign along the boundary of the block's edge. In Figure 5.3 we have (2.67) for $m = 0$ reconstructed by the full scan formula (1.13). A shift of 5 detector bins, for $Q = 591$, distributes the boundary of the object outside the support of the function but fails to introduce any comet tail artifacts.

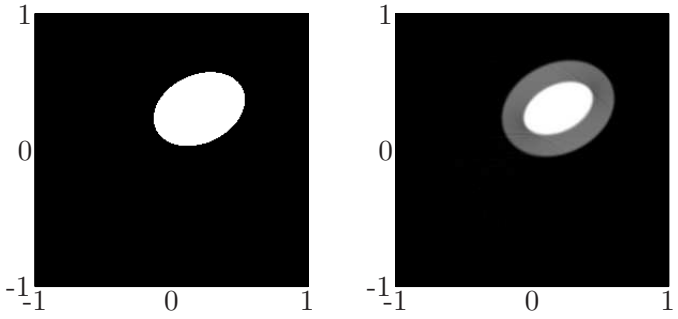

Figure 5.3: **Left:** Object function (2.67) with $m = 0$. **Right:** Reconstruction with $h = 5$ reconstructed with a full scan.

We now wish to exploit that formula (1.12) is more sensitive to the placement of the center of rotation. Ideally we wish to have an algorithm that can correctly place the center of rotation in our data provided we have a good idea of where to start from. We will assume that we know the proper center of rotation within a few detector widths. This is not unreasonable to assume because it is easy to tell if the selected choice of h is in the neighborhood of the exact value of h.

As observed in Table 5.1, we need to know the correct shift h for $e(h)$ to accurately remove the artifacts found in f_h. This condition makes $e(h)$ impractical for calculating the center of rotation because of its dependence on the unknown term h. However $e(h)$ is still useful for justifying the appearance of comet tail artifacts from formula (1.12) with a shift in x-ray data.

Our idea is that an ignored shift in α will reduce the resolution of f's boundaries and introduce comet tail artifacts. We introduce a penalty function

$$J_{tv}(h) = \int |\nabla f_h| \, dx. \tag{5.10}$$

The expectation is that $J_{tv}(h)$ will be smaller for f with well aligned edges and large otherwise.

We therefore study J_{tv} and its effectiveness at identifying the proper center of rotation in our x-ray data. We first present a plot of J_{tv} versus the shift in α to verify that minimizing J_{tv} will give us the correct center of rotation when there is no shift in α. Figure 5.4 shows that J_{tv} has a minimum and that the minimum corresponds to having

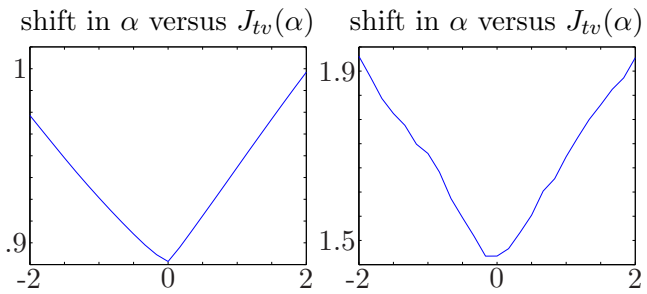

Figure 5.4: **Left:** For $f(\mathbf{x})$, $m = 3$, $h = -2\Delta\alpha, , \ldots 2\Delta\alpha$, **Right:** $m = 0$. Each reconstruction calculates $J_{tv}(h)$ with (1.12) in the plane $x_3 = .1$

no shift, $h = 0$. We can now expect the minimization of J_{tv} to give us an estimate to the proper shift in the data.

For each experiment we will assume $R = 3$, $D = 6$, 360 source positions per turn and 128 x-rays uniformly distributed across the unit circle, $Q = 591$. We reconstruct (2.67) with $u = .35, v = .25$ and $\psi = 25\pi/180$ and density 1.

The first step will identify a good initial guess for the shift in α variable. Let us assume $h = \pm\Delta\alpha$. We minimize J_{tv} numerically with the MATLAB function *fminbnd.m*. We discretize J_{tv} using central difference schemes in both coordinates

$$J_{tv}(h) \approx \sum_{i,j} \left(\left(\frac{f_h(x_{i+1}, y_j) - f_h(x_{i-1}, y_j)}{2\Delta x} \right)^2 + \left(\frac{f_h(x_i, y_{j+1}) - f_h(x_{i-1}, y_{j-1})}{2\Delta y} \right)^2 \right)^{1/2}. \tag{5.11}$$

To test the method we assume that the correct center of rotation in our x-ray data requires a shift of half of a detector to the left, $h_0 = -.5$, and the correct value to calculate is $h = 0$. Suppose $m = 3$, then our minimization method converges to $h = -.0061$ after 18 iterations. Thus we have calculated the proper shift in our x-ray data within 1/150th of a detector width. It is remarkable that our scheme is accurate to within a small fraction of a detector width.

Now consider $f(\mathbf{x})$ with $m = 0$. If we repeat the same experiment as above we now have a solution $h = .052$ or 1/20th of a detector width. The lesser degree of accuracy compared to the smooth case could be explained by the fact that the function is discontinuous. However if we examine the graph in Figure 5.4 we can conclude something different. Here we do not have a minimum value of J_{tv} when for the correct shift of $h = 0$.

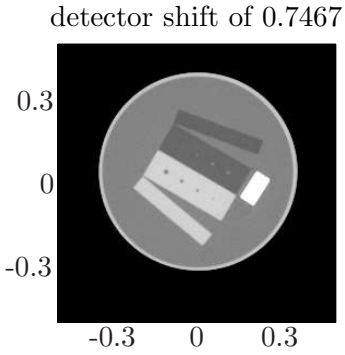

Figure 5.5: Centered data reconstruction for the calibration phantom.

We return to our original example, the calibration phantom. Our method yields $h = 0.7467$ of a detector width which is consistent with the shift of 3/4 of a detector provided with the projection data. We show the image reconstructed from $h = 0.7467$ in Figure 5.5 and immediately see a great reduction of the artifact from the previous reconstructions in Figure 5.1. This validates that our method can handle real x-ray data.

Our choice of the total variation semi-norm is not the only functional we have explored. We have found similar results as above with the following functional

$$J_1(h) = \int |f_h| \, dx. \tag{5.12}$$

6 Implementing π-Line Reconstruction Algorithms

Now we will now focus on the details for implementing π-line reconstruction formulas. We first begin by discussing the steps involved in the 3D formula. Then we discuss the 2D formula because much of the details follow directly from the 3D case. We conclude with the details for the π-line filtration 3D formula.

6.1 3D Katsevich Implementation

We scan an object that we assume to be have support in a cylinder of radius r centered at the point $(0,0,0)$ with the cylinder's axis of rotation in the z-direction. Our spatial discretization is $\mathbf{x}_{m,n,t} = (m\Delta x, n\Delta y, p\Delta z)$.

The source curve of the helix, $\mathbf{y}(s)$, has P sources per turn at a pitch of p and is discretized by $\mathbf{y}(s_k) = \left(R\cos(s_k), R\sin(s_k), \frac{ps_k}{2\pi}\right)^T$ for $s_k = k\Delta s$, where $\Delta s = \frac{2\pi}{P}$. The range of k is such that the smallest and largest π-interval endpoints, of the points where the function is to be construction, have three source positions before and after them.

We assume that we have angular discretization $\alpha_i = (i + shift_\alpha)\Delta\alpha$, for $i = -q_2 \dots q_2 - 1$, and $\Delta\alpha = \frac{\sin^{-1}(r/R)}{q_2}$. The w-component of the detector is discretized by $w_j = (j + shift_w)\Delta w$, for $k = -q_1 \dots q_1 - 1$, where $\Delta w = D\Delta\alpha$. The κ curves will be parameterized on the detector by α_i and the angle ψ whose discretization is $\psi_l = l\Delta\psi$, for $l = -M \dots M$.

Our algorithm is as follows. First we calculate the π-intervals for all the points where f is calculated. For each position s_k we acquire the projection data from $\mathbf{y}(s_{k+1})$ and calculate a derivative. The data is then corrected for the length of the line from $\mathbf{y}(s)$ to $d(s, \alpha_i, w_j)$. Next we interpolate or rebin the data from detector coordinates to the κ curves. The data is then filtered with a Hilbert kernel and rebinned from the κ curves to the regular detector coordinates. Finally the filtered data is then backprojected for all points in $RBP(s_k)$.

6.1.1 Numerical Difference Schemes

We require an effective discretization of the following derivative of our measured data $g(s, \alpha, w)$

$$g'(s, \alpha, w) = \frac{d}{ds} g(s, \alpha, w) = \frac{\partial g}{\partial s}(s, \alpha, w) + \frac{\partial g}{\partial \alpha}(s, \alpha, w). \tag{6.1}$$

We suggest the following discretization from Section 2.1

$$\begin{aligned} \frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}, w_j) &\approx \frac{1}{4\Delta s}[g(s_{k+1}, \alpha_i, w_j) - g(s_{k-1}, \alpha_i, w_j)] \\ &\quad + g(s_{k+1}, \alpha_{i+1}, w_j) - g(s_{k-1}, \alpha_{i+1}, w_j)] \\ \frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}, w_j) &\approx \frac{1}{\Delta \alpha}[g(s_k, \alpha_{i+1}, w_j) - g(s_k, \alpha_i, w_j)]. \end{aligned} \tag{6.2}$$

6.1.2 Filtering

It is necessary to interpolate equation (6.1) along the κ curves before we apply the convolution term $1/(\sin(\alpha^* - \alpha))$. That is we must perform what is called forward height rebinning, that is changing from coordinates in w to those in terms of the κ curves. The rebinning step is necessary because the filtering in equation (1.5) does not occur strictly in the w coordinate. Recall that Figure 1.4 shows κ curves that are not parallel to the α axis.

First we apply a length correction to the data

$$g'_{lc}(s, \alpha, w) = \frac{w}{\sqrt{w^2 + D^2}} g'(s, \alpha, w). \tag{6.3}$$

The rebinning is

$$g_\kappa(s, \alpha, \psi) = \frac{\partial g_{lc}}{\partial s}(s, \alpha, w_\kappa(\alpha, \psi)) + \frac{\partial g_{lc}}{\partial \alpha}(s, \alpha, w_\kappa(\alpha, \psi)) \tag{6.4}$$

where

$$w_\kappa(\alpha, \psi) = \frac{Dp}{2\pi R} \left(\psi \cos \alpha + \frac{\psi}{\tan \psi} \sin \alpha \right). \tag{6.5}$$

A Taylor series approximation for $w_\kappa(\alpha, 0)$ is necessary to avoid the singularity at zero. Recall that the choice of the difference scheme shifts α coordinate and it is necessary to take this shift into account when calculating the κ curves. This computation follows

directly from linear interpolation of (6.1) in the w coordinate. That is

$$g_\kappa(s,\alpha,\psi) \approx (1-c)g'_{lc}(s,\alpha,m') + c\ g'_{lc}(s,\alpha,m'+1)$$

where $m' = \lfloor t \rfloor$, $c' = t - \lfloor t \rfloor$, and $t = \lfloor w_\kappa(\alpha_i,\psi_l)/\Delta w - shift_w \rfloor$. It is worth noting that the κ curves are independent of the of source position s and thus the values c and m' are the same for all s. This observation allows us to calculate the values of c and m' once and save them in a table for reuse at each source position.

We now focus on the discretization of the term

$$\frac{1}{\sin(\alpha^* - \alpha)}.$$

We use the idea from [17] of expressing

$$\frac{1}{\sin(\alpha^* - \alpha)} = \frac{1}{\alpha^* - \alpha}\frac{(\alpha^* - \alpha)}{\sin(\alpha^* - \alpha)}$$

giving us a Hilbert transform divided by a sinc function. Next we use the following bandlimited approximation for the Hilbert transform kernel

$$\frac{1}{\alpha} \approx (2\pi)^{-1/2}\int_{-b}^{b}\left(-i\sqrt{\frac{\pi}{2}}\,\mathrm{sign}(\sigma)\right)e^{i\sigma\alpha}\,d\sigma = \frac{1-\cos(b\alpha)}{\alpha} \tag{6.6}$$

with $b < 1/(2\Delta\alpha)$ a user chosen cut-off frequency. The result is the approximation of $1/\sin(\alpha^* - \alpha)$ by $k(\alpha)$ where

$$k(\alpha) = \frac{\alpha}{\sin\alpha}\frac{1-\cos(b\alpha)}{\alpha} = \frac{1-\cos(b\alpha)}{\sin\alpha} \tag{6.7}$$

From the trapezoidal rule we end up with the following discretization

$$\begin{aligned} G_\kappa(s,i\Delta\alpha,\psi) &= \int_0^{2\pi} g_\kappa(s,i\Delta\alpha,\psi)k(i\Delta\alpha-\alpha)\,d\alpha \\ &\approx \Delta\alpha\sum_l g_\kappa(s,i\Delta\alpha,\psi)\ k((i-l)\Delta\alpha). \end{aligned} \tag{6.8}$$

We take care in sampling the kernel near zero by using a Taylor series approximation. We use Fourier transforms to speed up the convolution.

Suppose we calculate equation (6.1) on the shifted grid $(s_{k+1/2}, \alpha_{i+1/2}, w_j)$. The grid may be shifted to its original detector coordinates, $(s_{k+1/2}, \alpha_i, w_j)$, by the filtering kernel or the shift can be taken into account during the backprojection step. It was shown in [8] that it is beneficial to remove the shift in α by using a shifted filtering kernel.

It is necessary to return the filtered 2D data from the κ curve coordinates to the native detector coordinates to speed up the interpolation in the backprojection step. We follow the backward height rebinning method of Noo, Pack and Heucsher [19] and linearly interpolate from w_k to w using the κ curve of smallest ψ through w. That is

$$G(s, \alpha, w) \approx (1 - c(\alpha, w, \psi))G_\kappa(s, \alpha, \psi) + c(\alpha, w, \psi)G_\kappa(s, \alpha, \psi + 1). \tag{6.9}$$

In particular for $\alpha \geq 0$ we loop l from $-m$ to m until $w_\kappa(\alpha, \psi_l) \leq w_j < w_\kappa(\alpha, \psi_{l+1})$ and then let $c = (w_j - w_\kappa(\alpha, \psi_l))/(w_\kappa(\alpha, \psi_{l+1}) - w_\kappa(\alpha, \psi_l))$. For $\alpha < 0$ we now loop ψ_l from $-m$ to m until $w_\kappa(\alpha, \psi_{l-1}) \leq w_j < w_\kappa(\alpha, \psi_l)$ and then let $c = (w_j - w_\kappa(\alpha, \psi_{l-1}))/(w_\kappa(\alpha, \psi_l) - w_\kappa(\alpha, \psi_{l-1}))$. The weight calculation need only be done once and stored for future use. If a detector coordinate lies outside of the Tam-Danielsson window, $w_j > |w_k(\alpha, \psi_l)|$ for some l, it is not possible to use linear interpolation. One may overcome this complication by using nearest neighbor interpolation. Again, as with the forward height rebinning, the weights from the interpolation are independent of the source position s, and need only to be calculated once and then saved in a table. It is necessary to calculate the κ curves on the grid corresponding to the detector coordinates which result from applying the filtering kernel. For this calculation a Taylor series approximation for $w_\kappa(\alpha, 0)$ is necessary to avoid the singularity at zero.

6.1.3 Backprojection

The process of backprojection is as follows. From the current source position we stereographically project a point $\mathbf{x}$ we wish to reconstruct onto the detector. The projected point has local detector coordinates (α^*, w'). We then use bilinear interpolation to approximate the value at (α^*, w') of the filtered, measured data. Finally the interpolated value of (α^*, w') is weighted and added to the density map of $\mathbf{x}$. In essence, for each

point $\mathbf{x}$, we compute

$$\begin{aligned} f(\mathbf{x}) &\approx \frac{\Delta s}{2\pi}\sum_{j=0}^{N-1}\frac{\rho(s_k, I_\pi(\mathbf{x}))}{\bar{v}(s_k,\mathbf{x})}\cos(\alpha^*)G(s_k,\alpha^*(s,\mathbf{x}),w'(s,\mathbf{x})) \\ G(s_k,\alpha^*(s,\mathbf{x}),w'(s,\mathbf{x})) &\approx (1-c_1)(1-c_2)G(s_k,\alpha_{i'},w_{j'}) + c_1(1-c_2)G(s_k,\alpha_{i'+1},w_{j'}) \\ &\quad + (1-c_1)c_2G(s_k,\alpha_{i'},w_{j'+1}) + c_1c_2G(s_k,\alpha_{i'+1},w_{j'+1}) \end{aligned} \tag{6.10}$$

where $l' = \lfloor t_1 \rfloor, m' = \lfloor t_2 \rfloor$, $c_1 = t_1 - \lfloor t_1 \rfloor, c_2 = t_2 - \lfloor t_2 \rfloor$, $t_1 = \lfloor \alpha^*(s,\mathbf{x})/\Delta\alpha - shift_\alpha \rfloor$, and $t_2 = \lfloor w'(s,\mathbf{x})/\Delta w - shift_w \rfloor$. The detector coordinates are defined as follows

$$\begin{aligned} \bar{v}(s,\mathbf{x}) &= R - x\cos(s) - y\sin(s) \\ \alpha^*(s,\mathbf{x}) &= \arctan\left(\frac{1}{\bar{v}(s,\mathbf{x})}\left(-x\cos(s) + y\sin(s)\right)\right) \\ w'(s,\mathbf{x}) &= \frac{D\cos(\alpha^*(s,\mathbf{x}))}{\bar{v}(s,\mathbf{x})}\left(z - \frac{p}{2\pi}s\right) \end{aligned} \tag{6.11}$$

We now focus our discussion on the weight term $\rho(s, I_\pi(\mathbf{x}))$. The role of ρ is to implement the trapezoidal rule over $I_\pi(\mathbf{x})$ for all $\mathbf{x}$. We require that ρ is 1 in an open interval contained in $I_\pi(\mathbf{x})$, 0 outside an open interval containing $I_\pi(\mathbf{x})$ and $\rho(s_b(\mathbf{x}),\mathbf{x}) = \rho(s_b(\mathbf{x}),\mathbf{x}) = 1/2$. The discretization of s makes it impossible to have for all $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$, $s_k = s_b(\mathbf{x})$ for some $j \in \mathbb{N}$. It is therefore necessary to use a form of interpolation to faithfully implement the trapezoidal rule over $I_\pi(\mathbf{x})$.

The linear interpolation cutoff function is given by

$$\rho(s_k, I_\pi(\mathbf{x})) = \begin{cases} 0 & s_k \le s_b(\mathbf{x}) - \Delta s \\ \frac{1}{2}(\Delta s - (s_b(\mathbf{x}) - s_k)(1 + \frac{s_{k+1} - s_b(\mathbf{x})}{\Delta s})) & s_b(\mathbf{x}) - \Delta s \le s_k < s_b(\mathbf{x}) \\ \Delta s - \frac{(s_b(\mathbf{x}) - s_{k-1})^2}{2\Delta s} & s_b(\mathbf{x}) \le s_k < s_b(\mathbf{x}) + \Delta s \\ 1 & s_b(\mathbf{x}) + \Delta s \le s_k \le s_t(\mathbf{x}) - \Delta s \\ \Delta s - \frac{(s_{k+1} - s_t(\mathbf{x}))^2}{2\Delta s} & s_t(\mathbf{x}) - \Delta s < s_k \le s_t(\mathbf{x}) \\ \frac{1}{2}(\Delta s - (s_k - s_t(\mathbf{x}))(1 + \frac{s_t(\mathbf{x}) - s_{k-1}}{\Delta s})) & s_t(\mathbf{x}) < s_k < s_t(\mathbf{x}) + \Delta s \\ 0 & s_t(\mathbf{x}) + \Delta s \le s_k \end{cases} \tag{6.12}$$

and it is equivalent to equation (57) from [19].

6.2 3D π-Line Filtration Implementation

We use the same parameters as in Section 6.1. We reconstruct our function on rotated and translated copies of $C(0)$, the chip anchored at 0. The coordinate system for each chip $C(t)$ is relative to the chip's axis $\mathbf{y}(t)$ and $\mathbf{y}(t+\pi/2)$. Thus we reconstruct $f(\mathbf{x})$ on $C(t)$ with $C(t)$ discretize by $\mathbf{x}_{m,n,t} = (x'_m, y'_n, z_{m,n,t})$ where

$$z_{m,n,t} = \frac{hy_n \cos^{-1} x_m/R}{\sqrt{R^2 - x_m^2}} + \frac{tp}{2\pi} \tag{6.13}$$

and

$$\begin{bmatrix} \cos(t) & -\sin(t) \\ \sin(t) & \cos(t) \end{bmatrix} \begin{bmatrix} x_m \\ y_n \end{bmatrix} = \begin{bmatrix} x'_m \\ y'_n \end{bmatrix}. \tag{6.14}$$

Thus as we vary y' in each chip we traverse the π-line corresponding to the π-interval $[t-\alpha, t+\alpha]$ where $\alpha = \arccos(x_m/R)$.

The algorithm is broken down into the following steps. First we calculate the π-intervals for all the points where f is calculated. For each position s_k we aquire the projection data from $\mathbf{y}(s_{k+1})$ and calculate a derivative. The differentiated data is then backprojected on each chip $C(t)$. If $\mathbf{x} \in \partial RBP(s)$ then the projection data for $\mathbf{x}$ is saved. Once all of the backprojections have been completed for all source positions necessary, a Hilbert transform is applied to each π-line. In this case that implies we apply the transform strictly in the y coordinate of each chip. Trilinear interpolation is then used to interpolate from each chip to the cartesian grid.

6.2.1 Derivatives

Again we differentiate our data by (2.7)

$$G(s_k, \alpha_{i+1/2}, w_j) = \frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}, w_j) + \frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}, w_j). \tag{6.15}$$

6.2.2 Backprojection

For each chip we compute

$$Hf(\mathbf{x}_{m,n,t}) \approx \Delta s \sum_{j=0}^{N-1} \frac{\rho(s_k, I_\pi(\mathbf{x}_{m,n,t}))}{|\mathbf{x}_{m,n,t} - \mathbf{y}(s_k)|} G(s_k, \alpha^*(s, \mathbf{x}_{m,n,t}), w'(s, \mathbf{x}_{m,n,t})) \tag{6.16}$$

with

$$\begin{aligned} G(s_k, \alpha^*(s, \mathbf{x}_{m,n,t}), w'(s, \mathbf{x}_{m,n,t})) \approx\ & (1-c_1)(1-c_2)G(s_k, \alpha_{i'}, w_{j'}) \\ & + c_1(1-c_2)G(s_k, \alpha_{i'+1}, w_{j'}) \\ & + (1-c_1)c_2 G(s_k, \alpha_{i'}, w_{j'+1}) \\ & + c_1 c_2 G(s_k, \alpha_{i'+1}, w_{j'+1}) \end{aligned} \tag{6.17}$$

where $l' = \lfloor t_1 \rfloor, m' = \lfloor t_2 \rfloor$, $c_1 = t_1 - \lfloor t_1 \rfloor, c_2 = t_2 - \lfloor t_2 \rfloor$, and $t_1 = \lfloor \alpha^*(s, \mathbf{x}_{m,n,t})/\Delta\alpha - shift_\alpha \rfloor, t_2 = \lfloor w'(s, \mathbf{x}_{m,n,t})/\Delta w - shift_w \rfloor$. The terms α^* and w' are defined by (6.11). If s_k is near the endpoints of the π-interval of $\mathbf{x}_{m,n,t}$ then we store the value of s_k and $g(s_k, \alpha^*(s, \mathbf{x}_{m,n,t}), w'(s, \mathbf{x}_{m,n,t}))$ to later calculate the inverse Hilbert transform.

6.2.3 Calculating c

Recall that the inversion formula requires us to calculate

$$c = 2\frac{Df(\mathbf{y}(s_b(\mathbf{x})), \Theta(s_b(\mathbf{x}), \mathbf{x}))}{|\mathbf{y}(s_b(\mathbf{x})) - \mathbf{y}(s_t(\mathbf{x}))|}. \tag{6.18}$$

By observing that $Df(\mathbf{y}(s_b(\mathbf{x})), \Theta(s_b(\mathbf{x}), \mathbf{x})) = Df(\mathbf{y}(s_t(\mathbf{x})), \Theta(s_t(\mathbf{x}), \mathbf{x}))$ we can use an average of the two terms to get a more accurate approximation to c. We use linear interpolation in s to calculate $Df(\mathbf{y}(s_b(\mathbf{x})), \Theta(s_b(\mathbf{x}), \mathbf{x}))$ and $Df(\mathbf{y}(s_t(\mathbf{x})), \Theta(s_t(\mathbf{x}), \mathbf{x}))$.

Thus

$$\begin{aligned} C_t = & \frac{(1-c_b)g(s_i, \alpha^*(s_i, \mathbf{x}_{m,n,t}), w'(s_i, \mathbf{x}_{m,n,t}))}{|\mathbf{y}(s_b(\mathbf{x}_{m,n,t})) - \mathbf{y}(s_t(\mathbf{x}_{m,n,t}))|} \\ & + \frac{c_b g(s_{i+1}, \alpha^*(s_{i+1}, \mathbf{x}_{m,n,t}), w'(s_i, \mathbf{x}_{m,n,t}))}{|\mathbf{y}(s_b(\mathbf{x}_{m,n,t})) - \mathbf{y}(s_t(\mathbf{x}_{m,n,t}))|} \\ & + \frac{(1-c_t)g(s_{i'}, \alpha^*(s_{i'}, \mathbf{x}_{m,n,t}), w'(s_{i'+1}, \mathbf{x}_{m,n,t}))}{|\mathbf{y}(s_b(\mathbf{x}_{m,n,t})) - \mathbf{y}(s_t(\mathbf{x}_{m,n,t}))|} \\ & + \frac{c_t g(s_{i'+1}, \alpha^*(s_{i'+1}, \mathbf{x}_{m,n,t}), w'(s_{i'+1}, \mathbf{x}_{m,n,t}))}{|\mathbf{y}(s_b(\mathbf{x}_{m,n,t})) - \mathbf{y}(s_t(\mathbf{x}_{m,n,t}))|} \end{aligned} \tag{6.19}$$

with

$$\begin{aligned} c_b &= \frac{s_i - s_b(\mathbf{x}_{m,n,t})}{\Delta s} \\ c_t &= \frac{s_{i'} - s_t(\mathbf{x}_{m,n,t})}{\Delta s} \end{aligned} \tag{6.20}$$

and $s_i \le s_b(\mathbf{x}) \le s_i + \Delta s$ and $s_{i'} \le s_t(\mathbf{x}) \le s_{i'} + \Delta s$. The term C_t need only be calculated for one point $\mathbf{x}$ on the π-line because in the continuous case $Df(\mathbf{y}(s_b(\mathbf{x})), \Theta(s_b(\mathbf{x}), \mathbf{x}))$ is constant for the points on the same π-line as $\mathbf{x}$. In the discrete case the average over all points on the π-line provides a more stable approximation

$$c \approx \frac{1}{t} \sum_t C_t. \tag{6.21}$$

6.2.4 Filtering

Recall $Hf(\mathbf{x}_{m,n,t})$ is the Hilbert transform of $f(\mathbf{x})$ along the π-line of $\mathbf{x}$. Note $\mathbf{x}_{m,n,t} \in C(t)$ and as we vary n the point $\mathbf{x}_{m,n,t}$ traverses the π-line from $\mathbf{y}(t-\alpha)$ to $\mathbf{y}(t+\alpha)$ where $\alpha = \arccos(x_m/R)$. We shall apply the Hilbert inversion formula (1.9) in the y coordinate of each chip.

Suppose $I_\pi(\mathbf{x}_{m,n,t}) = [t-\alpha, t+\alpha]$. We determine a and b such that $|\pi(\mathbf{x}_{m,n,t}, a)| = |\pi(\mathbf{x}_{m,n,t}, b)| = r$ where

$$\pi(\mathbf{x}_{m,n,t}, T) = (1-T)\mathbf{y}(s_b(\mathbf{x}_{m,n,t})) + T\mathbf{y}(s_t(\mathbf{x}_{m,n,t})).$$

We can calculate a and b by using linear interpolation

$$\begin{aligned} a &= \frac{|(R\cos(s_b(\mathbf{x}_{m,n,t})), R\sin(s_b(\mathbf{x}_{m,n,t}))) - (R\cos(-\beta), R\sin(-\beta))|}{|(R\cos(s_b(\mathbf{x}_{m,n,t})), R\sin(s_b(\mathbf{x}_{m,n,t}))) - (R\cos(s_t(\mathbf{x}_{m,n,t})), R\sin(s_t(\mathbf{x}_{m,n,t})))|} \\ b &= \frac{|(R\cos(s_b(\mathbf{x}_{m,n,t})), R\sin(s_b(\mathbf{x}_{m,n,t}))) - (R\cos(\beta), R\sin(\beta))|}{|(R\cos(s_b(\mathbf{x}_{m,n,t})), R\sin(s_b(\mathbf{x}_{m,n,t}))) - (R\cos(s_t(\mathbf{x}_{m,n,t})), R\sin(s_t(\mathbf{x}_{m,n,t})))|} \\ \beta &= \arccos(x_m/r). \end{aligned} \tag{6.22}$$

To invert Hf we first compute

$$\begin{aligned} F(x, n\Delta y, z) &= \Delta y \sum_l \sqrt{(b-\tau)(\tau-a)} Hf(x, n\Delta y, z) W((n-l)\Delta y) \\ &\approx \int_a^b \frac{\sqrt{(b-\tau)(\tau-a)}}{n\Delta y - t'} Hf(x, n\Delta y, z)\, dt' \end{aligned} \tag{6.23}$$

with

$$\tau = \frac{n\Delta y - R\sin(s_b(\mathbf{x}_{m,n,t}))}{|(R\cos(s_b(\mathbf{x}_{m,n,t})), R\sin(s_b(\mathbf{x}_{m,n,t}))) - (R\cos(s_t(\mathbf{x}_{m,n,t})), R\sin(s_t(\mathbf{x}_{m,n,t})))|} \tag{6.24}$$

is such that $\pi(\mathbf{x}_{m,n,t}, \tau) = \mathbf{x}_{m,n,t}$. We utilize the following Hilbert kernel W

$$W(t) = \frac{1 - \cos(\pi t)}{\pi t} \tag{6.25}$$

with a shift of $-\Delta y/2$.

We then calculate

$$f(\mathbf{x}_{m,n,t}, \tau) = \frac{F(\mathbf{x}_{m,n,t}) + c}{2\pi\sqrt{(b-\tau)(\tau-a)}} \tag{6.26}$$

and thus we have reconstructed f for points on the chip $C(t)$ with the coordinate system on the chip given by $\mathbf{y}(t)$ and $\mathbf{y}(t + \pi/2)$.

6.3 2D Katsevich Implementation

We scan an object that we assume to be have support in a circle of radius r. The source curve of the circle, $\mathbf{y}(s)$, has P sources per turn and is discretized by $\mathbf{y}(s_k) =$

$(R\cos(s_k), R\sin(s_k))^T$ for $s_k = k\Delta s$, for $k = 0 \dots P$, where $\Delta s = \frac{2\pi}{P}$. We assume that we have angular discretization $\alpha_i = (l + shift_\alpha)\Delta\alpha$, for $l = -q \dots q-1$, and $\Delta\alpha = \frac{\sin^{-1}(r/R)}{q}$.

6.3.1 Derivatives

We implement the derivative (6.1) by M4

$$\begin{aligned}\frac{\partial g}{\partial s}(s_k, \alpha_{i+1/2}) &\approx \frac{1}{4\Delta s}[g(s_{k+1}, \alpha_i) - g(s_{k-1}, \alpha_i) \\ &\quad + g(s_{k+1}, \alpha_{i+1}) - g(s_{k-1}, \alpha_{i+1})] \\ \frac{\partial g}{\partial \alpha}(s_k, \alpha_{i+1/2}) &\approx \frac{1}{\Delta\alpha}[g(s_k, \alpha_{i+1}) - g(s_k, \alpha_i)].\end{aligned}$$

6.3.2 Filtering

We follow the idea from the 3D case and perform the following convolution

$$\begin{aligned}G(s, n\Delta\alpha) &= \int_0^{2\pi} g'(s, n\Delta\alpha) k(n\Delta\alpha - \alpha)\, d\alpha \\ &\approx \Delta\alpha \sum_l g'(s, l\Delta\alpha)\, k((n-l)\Delta\alpha).\end{aligned} \tag{6.27}$$

6.3.3 Backprojection

The process of the backprojection is similar to the 3D case. For each $\mathbf{x}$ in the circle of radius r it is necessary to compute

$$\begin{aligned}f(\mathbf{x}) &\approx \frac{\Delta s}{2\pi} \sum_{j=0}^{N-1} \frac{\rho(s_k, I_\pi(\mathbf{x}))}{\bar{v}(s_k, \mathbf{x})} G(s_k, \alpha^*(s, \mathbf{x})) \\ G(s_k, \alpha^*(s, \mathbf{x})) &\approx (1-c)G(s_k, \alpha_{i'}) + cG(s_k, \alpha_{i'+1})\end{aligned} \tag{6.28}$$

where $l' = \lfloor t \rfloor$, $c = t - \lfloor t \rfloor$and $t = \lfloor \alpha^*(s, \mathbf{x})/\Delta\alpha - shift_\alpha \rfloor$. The detector coordinates are defined as follows

$$\bar{v}(s, \mathbf{x}) = R - x\cos(s) - y\sin(s) \tag{6.29}$$

$$\alpha^*(s, \mathbf{x}) = \arctan\left(\frac{1}{\bar{v}(s, \mathbf{x})}(-x\cos(s) + y\sin(s))\right) \tag{6.30}$$

Unlike the 3D case, we would like to scan around the object only once. To that end we need to know for which s are in $I_\pi(\mathbf{x}) = [s_b(\mathbf{x}), s_t(\mathbf{x})]$ where $s_b(\mathbf{x})$ is not necessarily less than $s_t(\mathbf{x})$ modulo 2π. To over come this we apply a change of coordinates that maps the interval $[s_b(\mathbf{x}), s_t(\mathbf{x})]$ to $[s_b(\mathbf{x})', s_t(\mathbf{x})']$ where $s_b(\mathbf{x})' = \Delta s$ and $s_t(\mathbf{x})' = s_t(\mathbf{x}) - (s_b(\mathbf{x}) - \Delta s)$ along with $s' = s - (s_b(\mathbf{x}) - \Delta_s)$. This change of coordinates allows us to verify quickly if $0 < s_b(\mathbf{x})' \leq s' \leq s_t(\mathbf{x})' < 2\pi$.

7 Numerical Experiments

We conclude with a few experiments designed to investigate behavior of π-line algorithms. The first experiment is a convergence study to validate the accuracy of our implementation of formula (1.5). We perform reconstructions in the $x-y$ plane for the 3D formula and across a volume for the 3D reconstruction formula.

The second experiment looks at the accuracy of (1.5) when the helical pitch is large or small and the size of the detector elements is held constant.

7.1 Asymptotic Behavior

We shall investigate the order of convergence of the 3D Katsevich algorithm implementation. Our phantom is the smooth single ellipse and we will reconstruct a 256 by 256 image of the phantom in the plane $z = .1$. The function is (2.67) with $u = .35, v = .25, \psi = 25\pi/180$ and $m = 3$. We have a scanning geometry with a radius $R = 3$, a source to detector distance $D = 6$, and a helical pitch $P = .274$. The differentiation method for the 3D case is method M4. We double the number of detectors, filtering lines, and source positions for each experiment. From Table 7.1 we see that we have quadratic convergence. Next we will test reconstruction of the smooth single ellipse in the unit circle crossed with [0, .2]. Our image is a volume 256 by 256 by 26, and again we see that in Table 7.2 that the rate of convergence is quadratic.

P	q_2	q_1	M	l_2 error	$O(p^{-\alpha})$
128	32	4	5	0.097233	—
256	64	8	11	0.025816	1.9132
512	128	16	23	0.0055083	2.2286
1024	256	32	47	0.0013618	2.0161
2048	512	64	95	0.00035066	1.9574
4096	1024	128	191	8.6436×10^{-5}	2.0204

Table 7.1: 3D Katsevich $x-y$ slice reconstruction

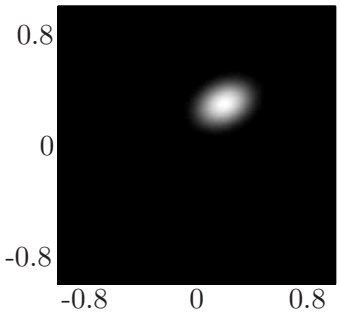

Figure 7.1: Slice $z = .1$ of the smooth single ellipsoid scaled between 0 and 1. The function is (2.67) with $u = .35, v = .25, \psi = 25\pi/180$ and $m = 3$.

P	q_2	q_1	M	l_2 error	$O(p^{-\alpha})$
128	32	4	5	0.11298	—
256	64	8	11	0.032469	1.7990
512	128	16	23	0.0062335	2.3809
1024	256	32	47	0.0016582	1.9104
2048	512	64	95	0.00040495	2.0338
4096	1024	128	191	0.00010053	2.0102

Table 7.2: 3D Katsevich volume reconstruction

7.2 Large and Small Helical Pitches

We now investigate the behavior of the pitch of the helix on the accuracy of the reconstruction. The motivation is to determine if a large pitch with a large number of detector rows is less accurate than a smaller pitch with a fewer number of detector rows. The pitches are such that 90 or 10 detector columns lie within the Tam-Danielsson window. For the case of the smooth single ellipse the errors differ slightly but no real advantage is realized by choosing one pitch over another. For the case of the Shepp-Logan phantom the relative l_2 errors are nearly identical and have the same visually quality.

P	# effective columns	relative l_2 error
0.27400	90	0.024622
0.027400	10	0.051670

Table 7.3: Large pitch and small pitch results for smooth single ellipse phantom. Scanning Parameters, $p = .274, .0274$, $r = 1$, $R = 3$, $D = 6$, $P = 256$ and $Q = 591$ and 23 κ curves. Reconstructed on 256 by 256 by 16 grid over $[-1, 1]^2 \times [0.2]$.

P	# effective columns	relative l_2 error
0.274	90	0.32916
0.0274	10	0.32996

Table 7.4: Large pitch and small pitch results for Shepp-Logan phantom. Scanning Parameters, $p = .274, .0274$, $r = 1$, $R = 3$, $D = 6$, $P = 256$ and $Q = 591$ and 23 κ curves. Reconstructed on 256 by 256 by 16 grid over $[-1, 1]^2 \times [-.1.1]$.

8 Conclusions

We have presented results that cover many areas of π-line reconstruction formulas. In each problem, we presented new methods to handle the challenges of recovering a function from its x-ray projection data. We have shown that in 2D that M4 and M5 have the same order of convergence and that M5 has more error terms than M4. To do this we required a decomposition of M5 in the local detector coordinates. Our analysis provides a method to compare direct methods with chain rule based methods for (2.1). Overall the performance of M4 is better than that of M5. We suggest using M4 because it is simpler to implement than M5 and the error terms of M4 do not depend on α. Furthermore we developed rational for the higher performance of M4 compared to M3.

We have presented a new method to compute the π-lines of general scanning trajectories by utilizing the notion of a chip. We also defined the region of backprojection. We have shown that the region of backprojection for the helix is easy to describe on each chip. In doing so we have simplified the description of the comet tail artifact on each chip and justified the artifact's appearance and location based on our hypothesis. We have shown the boundary of the region of backprojection identifies points $\mathbf{x}$ where s is the endpoint of $[s_b(\mathbf{x}), s_b(\mathbf{x})]$. We defined the region of backprojection for orthogonal-long and tilted long π-lines and showed that the comet tail artifact for orthogonal-long π-lines is a circle centered halfway between the point source function and the origin.

We demonstrated that the misalignment of data in helical and fan-beam π-line reconstruction formulas introduces a comet tail artifact. We developed an error term for the misaligned data in the 2D case. We proposed minimizing the reconstruction's TV norm to properly center the misaligned data. Our heuristic method worked for clinical and exact x-ray projection data.

We presented complete implementation notes on three π-line reconstruction formulas including Katsevich's and Pan's formulas. We demonstrated that our numerical implementation of Katsevich's 3D formula had second order convergence and that Katsevich's 3D formula is not sensitive to the pitch of the helix.

Bibliography

[1] K. E. Atkinson, *Introduction to Numerical Analysis*, Wiley, 1989.

[2] K. Champley, *Unpublished lecture notes.* 2006.

[3] P.-E. Danielsson, P. Edholm, J. Eriksson, and M. S. Magnusson, *Toward exact reconstruction for helical cone-beam scanning of long objects. a new detector arrangement and a new completeness condition*, in Proceedings of the 1997 International Meeting on Fully ThreeDimensional Image Reconstruction in Radiology and Nuclear Medicine, D. W. Townsend and P. E. Kinahan, eds., 1997, pp. 141–144.

[4] M. Defrise, F. Noo, R. Clackdoyle, and H. Kudo, *Truncated hilbert transform and image reconstruction from limited tomographic data*, Inverse Problems, 22 (2006), pp. 1037–1053.

[5] M. Defrise, F. Noo, and H. Kudo, *A solution to the long-object problem in helical cone-beam tomography*, Phys. Med. Biol., 45 (2000), pp. 623–43.

[6] V. Dogra and D. J. Rubens, *Ultrasound secrets*, Elsevier Health Sciences, 2003.

[7] A. Faridani, *Fan-beam tomography and sampling theory*, in The Radon Transform, Inverse Problems, and Tomography (Proceedings of Symposia in Applied Mathematics), G. Olafsson and E. T. Quinto, eds., vol. 63, Providence, Rhode Island, 2006, American Mathematical Society.

[8] A. Faridani, R. Hass, and D. C. Solmon, *Numerical and theoretical explorations in helical and fan-beam tomography*, J. Phys.: Conf. Ser., 124 (2008).

[9] G. Gullberg, B. Tsui, and C. Crawford, *Estimation of geometrical parameters for fan beam tomography*, Phys. Med. Biol, (1987), pp. 1581–1594.

[10] G. Herman and A. Naparstek, *Fast image reconstruction based on a radon inversion formula appropriate for rapidly collected data*, SIAM J. Appl. Math., (1977), pp. 511–533.

[11] S. H. Izen, *A fast algorithm to compute the π-line through points inside a helix cylinder*, Proc. Amer. Math. Soc., 135 (2007), pp. 269–276 (electronic).

[12] A. Katsevich, *Analysis of an exact inversion algorithm for spiral cone-beam ct*, Phys. Med. Biol., 47 (2002), pp. 2583–97.

[13] ——, *Improved exact fbp algorithm for spiral ct*, Adv. Appl. Math, 32 (2004), pp. 681–697.

[14] A. Katsevich, S. Basu, and J. Hsieh, *Exact filtered backprojection reconstruction for dynamic pitch helical cone beam computed tomography*, Phys. Med. Biol., 49 (2004), pp. 3089–3103.

[15] A. Katsevich and M. Kapralov, *Filtered backprojection inversion of the cone beam transform for a general class of curves*, SIAM Journal on Applied Mathematics, 68 (2007), p. 334.

[16] A. Katsevich and G. Lauritsch, *Filtered backprojection algorithms for spiral cone beam ct*, in Sampling, Wavelets, and Tomography (Applied and Numerical Harmonic Analysis), J. Benedetto and A. I. Zayed, eds., Birkhäuser, 2004, pp. 255–288.

[17] F. Natterer, *The mathematics of computerized tomography*, Society for Industrial and Applied Mathematics, 2001.

[18] F. Noo, S. Hoppe, F. Dennerlein, G. Lauritsch, and J. Hornegger, *A new scheme for view-dependent data differentiation in fan-beam and cone-beam computed tomography*, Phys. Med. Biol., 52 (2007), pp. 5393–5414.

[19] F. Noo, J. Pack, and D. Heuscher, *Exact helical reconstruction using native cone-beam geometries*, Phys. Med. Biol., 48 (2003), pp. 3787–3818.

[20] E. Y. Sidky, Y. Zou, and X. Pan, *Minimum data image reconstruction algorithms with shift-invariant filtering for helical, cone-beam ct*, Phys. Med. Biol., 50 (2005), pp. 1643–1657.

[21] H. Turbell and P.-E. Danielsson, *An improved pi-method for reconstruction from helical cone-beamprojections*, 1999 IEEE Nuclear Science Symposium, 1999. Conference Record, 2 (1999), pp. 865–868.

[22] H. K. Tuy, *An inversion formula for cone-beam reconstruction*, SIAM J. Appl. Math., 43 (1983), pp. 546–552.

[23] Y. Ye and G. Wang, *Filtered backprojection formula for exact image reconstruction from cone-beam data along a general scanning curve*, Med. Phys., 32 (2005), p. 42.

[24] Y. Ye, S. Zhao, H. Yu, and G. Wang, *A general exact reconstruction for cone-beam ct via backprojection-filtration*, IEEE transactions on medical imaging, 24 (2005), pp. 1190–1198.

[25] Y. Ye, J. Zhu, and G. Wang, *Geometric studies on variable radius spiral cone-beam scanning*, Medical Physics, (2004), pp. 1473–1480.

[26] ——, *Minimum detection windows, pi-line existence and uniqueness for helical cone-beam scanning of variable pitch*, Medical Physics, 31 (2004), pp. 566–572.

[27] J. You and G. L. Zeng, *Hilbert transform based fbp algorithm for fan-beam ct full and partial scans*, IEEE Transactions on Medical Imaging, 26 (2007), pp. 190–9.

[28] H. Yu and G. Wang, *Studies on artifacts of the katsevich algorithm for spiral cone-beam ct*, Proceedings of SPIE, 5535 (2004), p. 540.

[29] ——, *Studies on implementation of the katsevich algorithm for spiral cone-beam ct*, Journal of X-ray Science and Technology, 12 (2004), pp. 97–116.

[30] H. Yu, Y. Ye, S. Zhao, and G. Wang, *A backprojection-filtration algorithm for nonstandard spiral cone-beam ct with an n-pi-window*, Phys. Med. Biol., 50 (2005), pp. 2099–2111.

[31] A. Zamyatin, K. Taguchi, and M. D. Silver, *Practical hybrid convolution algorithm for helical ct reconstruction*, IEEE Transactions on Nuclear Science, 53 (2006), pp. 167–174.

[32] Y. Zou and X. Pan, *Exact image reconstruction on pi-lines from minimum data in helical cone-beam ct*, Phys. Med. Biol., 49 (2004), pp. 941–959.

[33] ——, *An extended data function and its generalized backprojection for image reconstruction in helical cone-beam ct*, Phys. Med. Biol., 49 (2004), pp. N383–N387.

[34] N. Zuo, D. Xia, Y. Zou, T. Jiang, and X.-C. Pan, *Chord-based image reconstruction in cone-beam ct with a curved detector*, Med. Phys., 33 (2006), pp. 3743–57.

APPENDIX

List of Symbols