Conditioning of and Algorithms for Image Reconstruction from Irregular Frequency Domain Samples

by

Benjamin Choong Lee

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2010

Doctoral Committee:

Professor Andrew E. Yagle, Chair
Professor Jeffrey A. Fessler
Professor Douglas C. Noll
Associate Professor Gregory H. Wakefield

To my parents.

ACKNOWLEDGEMENTS

There are many individuals whom without I would not have been able to complete my dissertation. First and foremost I would like to thank Professor Andrew Yagle for his guidance, kindness, and patience as my advisor throughout the years. I am grateful especially for his attentiveness and unrelenting support during the busy periods of each of my degree milestones.

I would like to thank my dissertation committee members, starting with Professor Jeffrey Fessler who has been helpful in providing conscientious and expert feedback on my dissertation and for providing me with real-world data with which to test my methods. Professor Gregory Wakefield has provided me poignant feedback and guidance through Quals I, Quals II, my proposal, and my defense, as well as when I was his teaching assistant for a semester. Professor Douglas Noll has generously set aside time for me to meet with him to discuss helpful ideas on real-world applications. I am also thankful for having taken courses from each of my committee members which have contributed to my basic knowledge from which I was able to perform my research.

I also would like to thank Dr. Edward Ficaro who has financially supported me since the beginning of my degree and has provided me with applications research and industry work experience. His mentorship and encouragement during my Ph.D. study has been invaluable in my completion of my degree.

I would like to thank Professor Adam Alessio at the University of Washington for the use of the actual CT data provided to me by the way of Professor Fessler.

I am also grateful for the employment opportunities over the years at the University of Michigan from the College of Engineering Department where I was an Engineering 101 Graduate Student Instructor (GSI) my first few years, the Department of Nuclear Medicine and Radiology for the summer internships, the Engineering Graduate Student Mentor (GSM) program complementing my GSI for a year, and from INVIA Medical Imaging Solutions in Ann Arbor from the last half of my years of my degree completion to the present. I also appreciate the incredible support and celebration from my coworkers at INVIA.

The love and unconditional support from my parents, sisters, brothers-in-law, and my nieces has provided me with the emotional support that has sustained me through out my studies here. I am thankful for all that my family has done for me.

Finally, my friends here at Michigan have made all the difference in providing camaraderie, a social outlet, and many great memories. I thank Paul, Johnson, Paul, Kurt, Ipek, Faisal, Dongsook, Margaret, Eric, Richard, Rob, Nick, Chad, and Jeff for their continued friendships.

TABLE OF CONTENTS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Conditioning of and Algorithms for Image Reconstruction from Irregular Frequency Domain Samples

by

Benjamin Choong Lee

Chair: Andrew E. Yagle

The problem of reconstructing an image from irregular samples of its 2-D DTFT arises in synthetic aperture radar (SAR), magnetic resonance imaging (MRI), computed tomography (CT), limited angle tomography, and 2-D filter design. The problem of determining a configuration of a limited number of 2-D DTFT samples also arises in magnetic resonance spectroscopic imaging (MRSI) and 3-D MRI.

This work first focuses on the selection of the measurement data. Since there is no 2-D Lagrange interpolation formula, sufficient conditions for the uniqueness and conditioning of the reconstruction problem are both not apparent. Kronecker substitutions, such as the Good-Thomas FFT, the helical scan FFT, and the 45° rotated support, unwrap the 2-D problem into a 1-D problem, resulting in uniqueness and insights into the problem conditioning. The variance of distances between the adjacent unwrapped 1-D DTFT samples was developed as a sensitivity measure to quickly and accurately estimate of the condition number of the system matrix. A well-conditioned configuration of DTFT samples, restricted to radial lines in CT or spirals

in MRI, is found by simulated annealing with the variance sensitivity measure as the objective function. The preconditioned conjugate gradient method reconstructs the 1-D solution that is then rewrapped to a 2-D image. In unrestricted cases, 2-D DTFT configurations like a regular hexagonal pattern can be unwrapped to uniformly-spaced and perfectly conditioned 1-D configurations and quickly solved using an inverse 1-D DFT.

The next focus is on developing fast reconstruction algorithms. A non-iterative DFT-based method of reconstructing an image is presented, by first masking the 2-D DTFT samples with the frequency response of a filter that is zeroed at the unknown 2-D DFT locations, and then quickly deconvolving the filtered image using three 2-D DFTs. The masking filter needs to be precomputed only once per DTFT configuration. A divide-and-conquer image reconstruction method is also presented using subband decomposition and Gabor filters to solve smaller subband problems, leading to a quick unaliased low-resolution image or later to be recombined into the full solution. All methods are applied to actual CT data resulting in faster reconstructions than POCS and FBP with equivalent errors.

CHAPTER I

Introduction

1.1 Problem Overview

Image reconstruction is the process of forming an image from measurement data, typically a transformation of the image information. The problem is to reconstruct a discrete image from some number of values of its 2-D Discrete-Time Fourier Transform (DTFT) measurements. Note that the measured set of DTFT values do not necessarily represent *all* points on a 2-D grid, nor are they necessarily values on a *rectangular* grid.

If 2-D DTFT values are in fact known on a rectangular grid, then the problem can easily be solved using the 2-D Discrete Fourier Transform (DFT). However, this requires knowledge of *all* 2-D DFT values on this grid. In many applications though, only *some* of the values on this grid are known. For example, in synthetic aperture radar (SAR) this is known on several different arcs of points in the Fourier domain as shown in [1]. No matter how fine the rectangular grid, the 2-D DFT cannot be used because not all of the points are known. In several different forms of medical imaging (e.g., magnetic resonance imaging (MRI) and X-ray compute tomography (CT)) in [2], the DTFT is known on a polar raster of points in the Fourier domain. Discretized filtered back-projection can be used, but this is only an approximation. Also, slices must be taken over 360 degrees; otherwise we have the limited-angle tomography

(LAT) problem in [3], in which the 2-D DTFT is known only in a bow-tie region. In spiral MRI, the values on a spiral trajectory are interpolated to a regular Cartesian grid using gridding. In 2-D filter design, the problem is to determine a 2-D FIR filter having a prescribed frequency response at some points in the 2-D frequency plane [4]. Again, this requires computing the image from some of its 2-D DTFT values.

In certain applications, such as MRI and SAR, the image to reconstruct is complex-valued and hence the reconstruction method should be generalized to reconstruct complex-valued objects and should not rely on constraints such as real-valued or non-negativity, which significantly extends the range of applications.

Sufficient conditions on the 2-D DTFT values for the unique reconstruction of the image are not immediately clear, since there is no 2-D Lagrange interpolation formula. We would like to know how large the set of measured 2-D DTFT values must be to reconstruct a unique image. In general, unique reconstruction of an $M_1 \times M_2$ image requires more than $M_1 M_2$ values of its 2-D DTFT.

We are also interested in determining how the selection of known DTFT locations affects the *conditioning* of the reconstruction problem. The conditioning of the problem is the sensitivity of the reconstructed image to perturbations in the 2D DTFT data, where perturbations may be due to measurement noise. Poor conditioning means that the unregularized solution is of little practical interest, since there is always noise in the data, and this will greatly affect the solution. In this case, regularization must be more drastic at the cost of biasing the solution. Additionally, poor conditioning results in slow convergence rates of iterative reconstruction methods such as conjugate gradient; we would like to know this in advance in order to choose other configurations of DTFT locations.

These concerns are even more emphasized in 3-D imaging where the criteria for choosing the most time-efficient 3-D k-space trajectory is not clear, such as in 3-D MRI where echo planar (EP) and stack of spiral (SOS) are common approaches

extended from 2-D and in Magnetic Resonance Spectroscopic Imaging (MRSI) where the third dimension measures the chemical shift to identify the concentration of water or fat.

Lastly we are interested in applying an algorithm to reconstruct the image from its 2-D DTFT values at the chosen locations, that is fast and if iterative it should converge quickly. As one of the disadvantages of iterative methods are long computation times, a method where the iterative computations are performed off-line once and then applied to any measurement data in one non-iterative step is desired. For large reconstruction problems when the computation is slow or memory is limited, breaking up those problems into smaller problems is also desirable.

Therefore the following issues are of particular interest:

1. What DTFT locations are sufficient to ensure that the image is uniquely determined? Some results have been obtained previously in [5],[6],[7],and [8];
2. How does the conditioning of the reconstruction problem depend on the DTFT locations? Often there is some choice in choosing these locations.
3. How can the image be reconstructed from its DTFT values? Conjugate gradient methods were applied directly in [9]; are there less memory intensive and faster computational methods?

1.2 Previous Work

In [5] and [6], sufficient conditions for uniqueness were obtained for 2-D DTFT locations $\{\omega_{1,k}, \omega_{2,k}\}$ lying on lines of the form $\omega_2 = \alpha\omega_1 + \beta$. A much simpler interpretation of those results involve the discrete Radon transform. Similar results were presented in [7] and [8], which also involve unwrapping the 2-D problem into a 1-D problem.

However, in [6], [7], and [8], little attention was paid to the *conditioning* of the

problem, which is one of the focuses of this dissertation. It is stated in [5] that examining the conditioning of the 2-D problem using the Lagrange interpolation formula as done in 1-D cannot be done for the 2-D problem since there is no 2-D Lagrange interpolation formula.

As for the reconstruction process, the straightforward method to reconstruct an image given some of its 2-D DTFT samples is to solve the possibly over-determined system of linear equations written as a matrix using lexicographic ordering of the image shown in [9]. However, there are some difficulties with this approach. Not all choices of the 2-D DTFT samples will lead to a non-singular system, as will be shown in Section 2.4. Sufficient conditions for the selection of 2D DTFT values for unique reconstruction of the image are not immediately clear.

An image can also be reconstructed directly using the inverse 2-D DFT of an interpolated regular grid of 2-D DFT samples. The non-uniform Fast Fourier Transform (FFT) has been developed in [10] and in [11] which in the latter is optimal in the min-max sense and can be used for this purpose. In [12], sampling density compensation for nonuniform sampling in MRI prior to interpolation is computed iteratively with only knowledge of the sampling locations and with out knowledge of the sampling trajectories. Though, we would like to try to reconstruct an image from only known DTFT samples without interpolation.

Another reconstruction approach is to use Projection Onto Convex Sets (POCS) introduced in [13]. POCS alternately projects onto the spatial domain (imposing finite $M \times M$ support) and onto the DTFT domain (imposing the known DTFT values). While this algorithm is guaranteed to converge, there are several difficulties:

1. Convergence in general requires thousands of iterations, and algorithms cannot be parallelized in iteration;
2. The DTFT must be computed at all of the given locations at each iteration; even using a pruned DFT requires a large amount of computation;

3. Roundoff error in the FFT over thousands of iterations may lead to problems in poorly conditioned problems, since the inverse FFT is not an exact inverse to the FFT in a finite-precision environment.

In [9] it is noted that, "since the computational cost of the POCS method is several orders of magnitude higher than other methods, and it provides only poor rate of convergence, it is not included in our comparison" (slightly edited). Hence POCS is not considered further.

Yet another method that is extensively used in image reconstruction is the conjugate gradient (CG) method which solves symmetric positive-definite systems. Each iteration requires a non-uniform forward 2-D DFT. Although the conjugate gradient method is iterative like POCS, CG is guaranteed to converge to the solution in N iterations for an $N \times N$ system matrix [14]. In most cases CG converges to within an acceptable error tolerance in less than N iterations.

The CG method can be made to converge faster by preconditioning the system which involves premultiplying both sides of the system equation with the inverse of a preconditioner matrix. A good preconditioner is one that is similar to the system matrix so as to best diagonalize the preconditioned system matrix.

1.3 Contributions of This Thesis

In [8] the Good-Thomas FFT is used to unwrap the 2-D problem into a 1-D problem which allows the determination of sufficient conditions for uniqueness of the solution. The Lagrange interpolation equation then reconstructs the 1-D signal. My contributions build upon [8] by answering the questions on conditioning and reconstruction methods posed in the above Section 1.1. The new contributions are listed below and followed up in detail in following chapters.

1. In [15], I showed how the Lagrange interpolation formula coefficients are related

to the condition number of the system matrix. With further evaluation of this relationship, I derive a closed-form equation for an upper bound on the condition number which empirically shows a 0.9 correlation with the condition number itself. In the just-determined case, this upper bound, computed in $O(N^2)$ time, can be used to select a set of DTFT values that leads to a well-conditioned system. A full explanation is found in Chapter III.

2. Seeing how the upper bound on the condition number was a function of distances between frequency locations in the denominator, I use the insight that clustering of DTFT locations result in poor conditioning to propose the "variance of distances between adjacent frequency locations" as a sensitivity measure. Because the variance sensitivity measure is not derived directly from the Lagrange interpolation equation, the variance sensitivity measure, computed in $O(N)$ time, could be used to select well-conditioned DTFT values in the over-determined case. This new sensitivity measure is useful considering that the just-determined case is usually poorly conditioned. The variance sensitivity measure has a 0.8 correlation with the condition number in one over-determined case. A full explanation is found in Chapter III.
3. I used the variance sensitivity measure as the cost function in the simulated annealing algorithm to determine a well-conditioned configuration of DTFT values. The low-variance configuration results in reconstructed images with fewer iterations required to reach the same residual tolerance in the conjugate gradient reconstruction method due to improved conditioning of the system matrix. A full explanation is found in Chapter III.
4. I correct and use the new 45° rotated-support reconstruction method of odd image size which makes a Kronecker substitution in the 2-D z-transform instead of the Good-Thomas FFT to unwrap the 2-D problem into a 1-D one. This is an improvement over the Good-Thomas FFT unwrapping because the rotated-

support method eliminates 2/3 the bands of zeros in the solution which reduces the overall 1-D signal support. Thus the dimensions of the system matrix to solve is smaller. A full explanation is found in Chapter III.

5. In the 1-D unwrapped method, I show that when there are no constraints on the DTFT locations such as intersections with radial projection lines or spiral trajectories, "perfect conditioning," where the condition number of the system matrix is 1 and hence measurement noise is not amplified in the solution, can be achieved by uniformly distributing the 1-D DTFT samples. Though seemingly trivial in 1-D, depending the arbitrary size of the measurement set, the wrapped 2-D frequency configuration produce regular but non-rectangular patterns. A full explanation is found in Chapter III.
6. I introduce a new non-iterative DFT-based method that reconstructs an image from deconvolving a precomputed filter from a filtered DFT of the data, done quickly using FFTs. Though the filter is precomputed iteratively it only needs to be done once per frequency configuration and can be applied repeatedly to different data. A full explanation is found in Chapter IV.
7. In the non-iterative DFT-based method, I precompute the filter which applies the finite-support constraint as done in the POCS method but converges faster than POCS by formulating a new perspective of the finite-support constraint as a regularization term and solving it using the preconditioned conjugate gradient method. A full explanation is found in Chapter IV.
8. I introduce a new divide-and-conquer image reconstruction method using subband decomposition. A large image reconstruction is divided into smaller problems by splitting the irregular frequency data into subbands by projecting the data onto a modified Gabor logon (or 2-D Gaussian kernel). After solving the smaller problem the solution is deconvolved from the Gabor logon and the 2-D DFT of the solution is combined with the others. Initial results plagued with

banding artifacts were improved with finite-support padding. A full explanation is found in Chapter V.

9. In the divide-and-conquer subband decomposition method, I empirically present the advantages of varying the amount of regularization or the over-determining factor specifically for each subproblem with decreased error or reduced computation time, respectively. A full explanation is found in Chapter V.
10. Lastly, I apply the 1-D unwrapped conditioning, non-iterative, and divide-and-conquer methods to actual CT sinogram data, provided by Adam M. Alessio of the University of Washington. The fanbeam CT data with 888 detector bins and 820 views is converted to parallel beam projections and transformed into radial 2-D DTFT samples. Images were reconstructed from the actual CT data at sizes as large as 889×889, with some zero-padded images reconstructed at 1690×1690 in just a few seconds. Full explanations are found in the Results sections of Chapters III-V.

1.4 Organization of This Thesis

Chapter 2 describes the problem statement by providing the mathematical background, uniqueness requirements, and conditioning of a system of linear equations. The problem is precisely formulated mathematically. Chapter 3 discusses the conditioning approach in detail by reformulating the 2D problem into a 1D for which conditioning is better understood. A sensitivity measure is developed to estimate conditioning and used in simulated annealing to arrive at a near-optimal global solution with numerical results. Additionally, the requirements for perfect conditioning is presented. Chapter 4 introduces a fast non-iterative approach based on prefiltering the Fourier data then solving a deconvolution problem, where the advantage is that most of the computations are performed in precomputing the filter. An accurate method of precomputing the filter using the conjugate gradient method is also pre-

sented. Chapter 5 explains how we can separate a large image reconstruction problem into smaller subproblems with a divide-and-conquer approach by using subband decomposition. In Chapters 3 through 5, each reconstruction method is applied on both simulated and actual CT data. Lastly, Chapter 6 summarizes the results the various reconstruction methods presented, evaluates their specific merits, and recommends future areas of further research.

CHAPTER II

Problem Statement

2.1 Background

The problem is to reconstruct an $M_1 \times M_2$ discrete image $x(i_1, i_2)$ from some N values of its 2-D Discrete-Time Fourier Transform

$$X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}) = \sum_{i_1=0}^{M_1-1} \sum_{i_2=0}^{M_2-1} x(i_1, i_2) e^{-j(i_1\omega_{1,k}+i_2\omega_{2,k})}. \tag{2.1}$$

The values $x(i_1, i_2)$ can be complex such as in MRI and SAR or real such as in CT. Note that $\{X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}), k = 1, \ldots, N\}$ do not necessarily represent *all* points on a 2-D grid, nor are they necessarily values on a *rectangular* grid.

Throughout this work, we formulate the problem with discrete measurements with discrete unknown image samples or we assume a "bed of nails" representation of the image as a lattice of 2-D impulses weighted by $x(i_1, i_2)$. Other image models (e.g. pixels in which the image is constant in a small square) quickly reduce to this problem, since the lattice of impulses is convolved with a known point-spread function, which corresponds to multiplication in the 2-D DTFT domain.

If $X(e^{j\omega_1}, e^{j\omega_2})$ is known on a rectangular grid such as $X(e^{j2\pi k_1/M_1}, e^{j2\pi k_2/M_2})$, then

the problem obviously can be solved using the 2-D Discrete Fourier Transform.

$$X(e^{j2\pi k_1/M_1}, e^{j2\pi k_2/M_2}) = \sum_{i_1=0}^{M_1-1} \sum_{i_2=0}^{M_2-1} x(i_1, i_2) e^{-j2\pi(\frac{i_1 k_1}{M_1} + \frac{i_2 k_2}{M_2})}. \tag{2.2}$$

However, this requires knowledge of *all* $X(e^{j\omega_1}, e^{j\omega_2})$ values on this grid. In many applications though, only *some* of the values on this grid are known. While interpolation can be used to resample the frequency values to a rectangular lattice, this necessarily involves some approximation and some computation.

In reconstructing $x(i_1, i_2)$, several issues are of interest. First, we would like to know some sets of locations $\{\omega_{1,k}, \omega_{2,k}\}$ which are sufficient to ensure that $x(i_1, i_2)$ is uniquely determined. More specifically, we ask how many DTFT values, and in what patterns, are needed for a unique reconstruction. In the specific cases where $x(i_1, i_2)$ is assumed to be real, then DTFT values can be collected in complex conjugate pairs.

Second, we are interested in determining how the selection of the known DTFT locations $\{\omega_{1,k}, \omega_{2,k}\}$ affects the *conditioning* of the reconstruction problem. The conditioning of the problem is the sensitivity of the reconstructed $x(i_1, i_2)$ to perturbations in the data $\{X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}), k = 1, \ldots, N\}$, where perturbations may be due to measurement noise. *There is a need for a computationally quick sensitivity measure to evaluate the relative conditioning of the problems arising due to various configurations of* $\{\omega_{1,k}, \omega_{2,k}\}$.

Third, we need to determine which algorithm should be used to reconstruct $x(i_1, i_2)$ from its DTFT values at locations $\{\omega_{1,k}, \omega_{2,k}\}$. This algorithm should be fast, and if it is iterative like the conjugate gradient method, it should have fast convergence. Another desired attribute is the ability to initialize it with a quick-to-compute estimate if iterative.

In each of these issues, we also explore new approaches that arise when we constrain the set of 2-D DTFT samples to gridded subset of $N \times N2 - DDFT$.

2.2 Problem Formulation

Most of this section is review, but is necessary background for what follows. For parts of the discussion of the 2-D case, when "M" is used we assume a square image support, where $M = M_1 = M_2$.

2.2.1 2-D Problem

The 2-D reconstruction of an $M \times M$ image $x(i_1, i_2)$ given $\{X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}), k = 1, \ldots, N\}$ may be solved written as a matrix using lexicographic ordering of $x(i_1, i_2)$ in (2.1) or specifically column-wise unwrapping. This possibly over-determined problem can be formulated as the 2-D non-square Vandermonde system of equations,

$$\hat{A}x = b \tag{2.3}$$

$$\hat{A} = \begin{bmatrix} 1 & z_{1,1} & \cdots & z_{1,1}^{M-1} & z_{1,1}^{0} z_{2,1}^{1} & \cdots & z_{1,1}^{i_1} z_{2,1}^{i_2} & \cdots & z_{1,1}^{M-1} z_{2,1}^{M-1} \\ 1 & z_{1,2} & \cdots & z_{1,2}^{M-1} & z_{1,2}^{0} z_{2,2}^{1} & \cdots & z_{1,2}^{i_1} z_{2,2}^{i_2} & \cdots & z_{1,2}^{M-1} z_{2,2}^{M-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 1 & z_{1,N} & \cdots & z_{1,N}^{M-1} & z_{1,N}^{0} z_{2,N}^{1} & \cdots & z_{1,N}^{i_1} z_{2,N}^{i_2} & \cdots & z_{1,N}^{M-1} z_{2,N}^{M-1} \end{bmatrix}$$

$$x = [x(0,0) \ldots x(M-1,0) x(0,1) \ldots x(M-1,M-1)]^T$$

$$b = [X(z_{1,k}^*, z_{2,k}^*) \ldots X(z_{1,N}^*, z_{2,N}^*)]^T$$

where $z_{1,k} = e^{-j\omega_{1,k}}$ and $z_{2,k} = e^{-j\omega_{2,k}}$. The least-squares solution is computed by solving the 2-D normal equation,

$$(\hat{A}^H \hat{A})x = \hat{A}^H b \tag{2.4}$$

where the system matrix $\hat{A}^H\hat{A}$ is Toeplitz-block-Toeplitz (TBT). Toeplitz blocks of uniform sizes form a block-Toeplitz matrix. Note that the hat notation above $\hat{A}$ is used to avoid confusion with the 1-D non-square Vandermonde matrix A in (2.6) below.

However, it is not initially apparent how large N must be to reconstruct a unique $x(i_1, i_2)$. In general, unique reconstruction of an $M_1 \times M_2$ image requires more than $M_1 M_2$ values of its 2-D DTFT. Though, sufficient conditions for uniqueness in the 2-D problem have been obtained in [5], [6], [7], and [8], which all involve unwrapping the 2-D problem in to a 1-D problem, which will be discussed in further detail in Section 3.2.

2.2.2 Just-Determined 1-D Problem

The 1-D problem involves reconstructing a length M signal $\{x(n), n = 0, \ldots, M-1\}$ from M DTFT values $\{X(e^{j\omega_k}), k = 1, \ldots, M\}$. This is often called the "non-uniform DFT."

A "just-determined" problem is one where M DTFT values are sufficient to reconstruct a unique signal of length M, can be formulated as the solution of the Vandermonde system of equations

$$Vx = b \tag{2.5}$$

$$V = \begin{bmatrix} 1 & z_1 & z_1^2 & \cdots & z_1^{M-1} \\ 1 & z_2 & z_2^2 & \cdots & z_2^{M-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & z_M & z_M^2 & \cdots & z_M^{M-1} \end{bmatrix}$$

$$x = [x(0) \dots x(M-1)]^T, \quad b = [X(z_1^*) \dots X(z_M^*)]^T$$

where $z_k = e^{-j\omega_k}$.

2.2.3 Over-Determined 1-D Problem

Unfortunately, the just-determined problem is usually so poorly conditioned that it is necessary to consider the over-determined case, where the 1-D problem is to reconstruct the length M signal $\{x(n), n = 0, \dots, M-1\}$ from N DTFT values $\{X(e^{j\omega_k}), k = 1, \dots, N\}$, where $N > M$.

This over-determined problem can be formulated as the 1-D non-square Vandermonde system of equations

$$Ax = b \tag{2.6}$$

$$A = \begin{bmatrix} 1 & z_1 & z_1^2 & \cdots & z_1^{M-1} \\ 1 & z_2 & z_2^2 & \cdots & z_2^{M-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & z_N & z_N^2 & \cdots & z_N^{M-1} \end{bmatrix}$$

$$x = [x(0) \dots x(M-1)]^T, \quad b = [X(z_1^*) \dots X(z_N^*)]^T$$

where $z_k = e^{-j\omega_k}$. If the observed values $X(e^{j\omega_k})$ are the true values $\bar{X}(e^{j\omega_k})$ corrupted by zero-mean additive white Gaussian noise with noise variance σ^2 as in,

$$Ax = b = \bar{b} + n \tag{2.7}$$

$$\bar{b} = [\bar{X}(z_1^*) \dots \bar{X}(z_N^*)]^T, \quad n = [n(0) \dots n(N)]^T \quad n \sim \mathcal{N}(0, \sigma^2)$$

then the maximum likelihood estimate of the $x(n)$ is the solution to

$$(A^H A)x = A^H b \tag{2.8}$$

which is also the best linear unbiased estimator (BLUE).

Since the (i,k)th element $(A^H A)_{i,k}$ of the $M \times M$ matrix $A^H A$ is

$$(A^H A)_{i,k} = \sum_{\ell=1}^{M} A_{\ell,i}^H A_{\ell,k} = \sum_{\ell=1}^{M} e^{-j\omega_\ell (i-k)} \tag{2.9}$$

and $A^H A$ is a symmetric Toeplitz matrix. Thus the normal equation in (2.8) can be solved quickly in $O(M \log^2 M)$ operations.

2.2.4 Regularization

When the over-determined system of linear equation, $Ax = b$, is ill-conditioned, small perturbations in b will greatly perturb x. Regularization modifies the problem to improve the conditioning. In the case of the normal equations, $A^H Ax = A^H b$, Tikhonov regularization replaces minimization of the residual error, $\|Ax - b\|^2$, with minimization of

$$\|Ax - b\|^2 + \lambda^2 \|x\|^2. \tag{2.10}$$

The coefficient λ balances the trade-off between minimizing the residual error and limiting the size of x. A general regularization term is presented in Section 4.4 that can limit 2nd-order differences for example. The new minimization replaces the old system with

$$(A^H A + \lambda^2 I)x = A^H b \tag{2.11}$$

and now A need not be over-determined. However, in improving the conditioning and reducing the variance of the solution, regularization also biases the solution, which is an undesirable side-effect.

2.3 Uniqueness

Sufficient conditions on $\{X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}), k = 1, \ldots, N\}$ for the unique reconstruction $x(i_1, i_2)$ without regularization as in 2.6 are not immediately clear. We would like to know how large N must be to reconstruct a unique $x(i_1, i_2)$. In general, unique reconstruction of an $M_1 \times M_2$ image requires more than $M_1 M_2$ values of its 2-D DTFT and not all choices of $\{X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}), k = 1, \ldots, N\}$ will lead to a non-singular system.

For example, consider the two images

$$\begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix} \quad \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}. \tag{2.12}$$

Their respective zero-padded 5×4-point 2-D DFTs are

$$\begin{bmatrix} 8 & 4 - i4 & 0 & 4 + i4 \\ 5.2 - i3.8 & 1.0 - i6.2 & -1.4 - i1.9 & 2.9 + i0.5 \\ 0.8 - i2.4 & -2.0 - i4.0 & -3.6 - i1.2 & -0.8 + i0.4 \\ 0.8 + i2.4 & -0.8 - i0.4 & -3.6 + i1.2 & -2.0 + i4.0 \\ 5.2 + i3.8 & 2.9 - i0.5 & -1.4 + i1.9 & 5.2 + i3.8 \end{bmatrix}$$

$$\begin{bmatrix} 8 & 4 - i4 & 0 & 4 + i4 \\ 5.2 - i3.8 & 0.7 - i4.5 & 0 & 4.5 + i0.7 \\ 0.8 - i2.4 & -0.8 - i1.6 & 0 & 1.6 - i0.8 \\ 0.8 + i2.4 & 1.6 - i0.8 & 0 & -0.8 + i1.6 \\ 5.2 + i3.8 & 4.5 - i0.7 & 0 & 0.7 + i4.5 \end{bmatrix} \tag{2.13}$$

shown with rounded values, and they agree at 8 of the 20 frequency points along the left-most column and the top-most row. Hence, although it might seem as though only four frequency points should be necessary to reconstruct a 2×2 image uniquely,

in fact even *eight* points of the 5×4-point 2-D DFT are not enough to ensure unique reconstruction. (All four points of the 2×2-point 2-D DFT would suffice.) Reconstruction using the matching 8 frequency points leads to an over-determined but singular system.

2.4 Conditioning

We are also interested in determining how the selection of known DTFT locations $\{\omega_{1,k}, \omega_{2,k}\}$ affects the *conditioning* of the reconstruction problem.

The conditioning of the problem is the sensitivity of the reconstructed $x(i_1, i_2)$ to perturbations in the data $\{X(e^{j\omega_{1,k}}, e^{j\omega_{2,k}}), k = 1, \ldots, N\}$, where perturbations may be due to measurement noise.

Consider the following perturbed linear system seen in [14]:

$$Ax(\epsilon) = b + \epsilon f \tag{2.14}$$

where $x(0) = x \in \mathbf{R}^M$, $\epsilon \in \mathbf{R}$, and $f \in \mathbf{R}^M$. When measuring sensitivity, we would like to observe how much the solution vector $x(\epsilon)$ deviates from x when perturbing the data vector b with ϵf. The relative error in $x(\epsilon)$ is upper-bounded by the relative error of the data times the condition number. Hence the condition number $\kappa_p(A)$ defined in (3.41) quantifies the sensitivity as shown:

$$\frac{\|x(\epsilon) - x\|_p}{\|x\|_p} \leq \kappa_p(A)|\epsilon|\frac{\|f\|_p}{\|b\|_p} \tag{2.15}$$

For example, observe the various configurations of the frequency locations on a 11×13 grid in Fig. 2.1 chosen for reconstruction of a 51-pixel-support image[1]. Each subfigure represents the 2-D discrete frequency domain, where the black and

[1]The $M \times M = 4 \times 4$ slanted image has a 1-D support of 51.

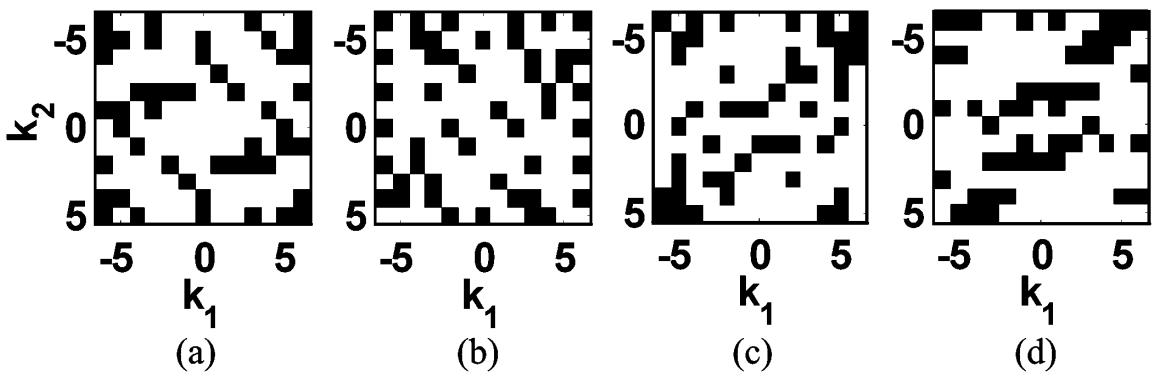


Figure 2.1: Various 2-D frequency configurations.

white pixels denote selected (approximately just-determined where $N \approx 51$) and unselected frequency locations, respectively. The reader is asked to guess which 2-D configurations form system matrices with the lowest condition numbers. Unless the construction of the system matrix is explicitly determined, it is not apparent which configurations pose poorly conditioned problems.

The condition numbers are computed from system matrices constructed as unwrapped 1-D problems explained in detail in Section 3.2. The answers are that the configurations in Fig. 2.1a and 2.1b form system matrices with the lowest condition numbers on the order of 10^2. Configurations in Fig. 2.1c and 2.1d form system matrices with the highest condition numbers on the order of 10^{14}, as shown in Fig. 3.1. This is an important clue to the relative conditioning of the reconstruction problem from these locations.

2.5 Reconstruction Algorithm

The 1-D problem can be solved either in closed-form for the just-determined case but is commonly solved iteratively for the over-determined case.

2.5.1 Lagrange Interpolation Formula

The just-determined problem can be solved in closed-form by using the *Lagrange interpolation formula*

$$X(u) = \sum_{k=1}^{M} X(z_k) \prod_{\substack{\ell=1 \\ \ell \neq k}}^{M} \frac{u - z_\ell}{z_k - z_\ell} \tag{2.16}$$

where $z_k = e^{-j\omega_k}$, and $X(z_k^*)$ are the known DTFT values. Evaluating (2.16) at $\{u = e^{-j2\pi k/M}, k = 0, \ldots, M-1\}$ followed by an inverse DFT results in $x(n)$.

We note that (2.16) is computationally inefficient; a better procedure is to write $X(u)$ in its Newton representation

$$X(u) = c_0 + c_1(u - z_1) + c_2(u - z_1)(u - z_2) + \ldots \tag{2.17}$$

Then setting $u = z_1, z_2, \ldots$ in succession allows $c_0, c_1, \ldots$ to be computed recursively. While the direct implementation of (2.16) has complexity $O(M^3)$, this recursive implementation has a complexity of $O(M^2)$. A still better procedure is to use the Chinese Remainder Theorem as a divide-and-conquer algorithm in [16]. This requires only $O(M \log^2 M)$ operations.

A still better procedure is to use the Chinese Remainder Theorem. To do this, we partition the interpolation problem $X(z)|_{z=z_k} = X(z_k), k = 1, ..., M$, M even, into two problems

$$X_1(z)|_{z=z_k} = X_1(z_k), k = 1, ..., \frac{M}{2} \tag{2.18}$$

$$X_2(z)|_{z=z_k} = X_2(z_k), k = \frac{M}{2} + 1, ..., M, \tag{2.19}$$

solve each subproblem separately, and then combine them using

$$X(z) = X_1(z)Y_1(z) \prod_{k=\frac{M}{2}+1}^{M} (z - z_k) \tag{2.20}$$

$$+ \quad X_2(z)Y_2(z) \prod_{k=1}^{\frac{M}{2}} (z - z_k) \mod \left(\prod_{k=1}^{M} (z - z_k) \right) \tag{2.21}$$

where $Y_i(z)$ solve

$$Y_1(z) \prod_{k=\frac{M}{2}+1}^{M} (z - z_k) \equiv 1 \mod \left(\prod_{k=1}^{\frac{M}{2}} (z - z_k) \right) \tag{2.22}$$

$$Y_2(z) \prod_{k=1}^{\frac{M}{2}} (z - z_k) \equiv 1 \mod \left(\prod_{k=\frac{M}{2}+1}^{M} (z - z_k) \right) \tag{2.23}$$

For more details see [16].

Even though the Lagrange interpolation formula (2.16) cannot solve the over-determined case, it provides considerable insight into the conditioning (sensitivity of the solution to the perturbations of the data) of the problem. The sensitivity as a function of the frequencies ω_k is dominated by the denominator $\prod_{\ell=1,\ell\neq k}^{M}(z_k - z_\ell)$. In Sections 3.4.3 and 3.4.5, we observe that the greater the departure of the ω_k from a uniform distribution on the interval $[0, 2\pi)$, the greater the sensitivity. Both close spacings and wide gaps between successive ω_k values will tent to increase sensitivity.

2.5.2 Conjugate Gradient

The conjugate gradient method iteratively solves the system $Ax = b$, where A is a symmetric positive definite matrix of size $M \times M$, by minimizing the quadratic form

$$f(x) = \frac{1}{2} x^T A x - b^T x - c \tag{2.24}$$

which is shaped like a paraboloid bowl when x is two-dimensional [17]. The minimum is located at the solution x, where the gradient of (2.24) is zero. We begin solving for x by selecting an initial guess vector $x_{(0)}$. Then for each iteration i, we create a search direction, $d_{(i)}$, to which we follow to bring us to our next iteration of $x_{(i+1)}$. Search directions are constructed from the Gram-Schmidt conjugation of the residuals. A residual,

$$r_{(i)} = b - Ax_{(i)} = -f'(x_{(i)}), \tag{2.25}$$

quantifies how far we are from the correct value of b but more importantly is the direction of the steepest descent. Each new residual is orthogonal, $\{r_{(i)}^T r_{(j)} = 0, i \neq j\}$, to all previous search directions and residuals. Furthermore, each new search direction is A-conjugate (or A-orthogonal), $\{d_{(i)}^T A d_{(j)} = 0, i \neq j\}$, to all previous search directions and residuals. The CG method is as follows:

$$\begin{aligned}
d_{(0)} = r_{(0)} &= b - Ax_{(0)} && (2.26)\\
\alpha_{(i)} &= \frac{r_{(i)}^T r_{(i)}}{d_{(i)}^T A d_{(i)}} && (2.27)\\
x_{(i+1)} &= x_{(i)} + \alpha_{(i)} d_{(i)} && (2.28)\\
r_{(i+1)} &= r_{(i)} - \alpha_{(i)} A d_{(i)} && (2.29)\\
\beta_{(i+1)} &= \frac{r_{(i+1)}^T r_{(i+1)}}{r_{(i)}^T r_{(i)}} && (2.30)\\
d_{(i+1)} &= r_{(i+1)} - \beta_{(i+1)} d_{(i)} && (2.31)
\end{aligned}$$

Since the search directions are A-orthogonal, the union of each new search direction to the previous search directions produces a Krylov subspace, created by repeatedly multiplying a matrix to a vector. The solution x is reached in M iterations [14] when all M search directions span the entire M-dimensional vector space. However, when M is large it may not be feasible to run M iterations. In practice, the iterations are terminated when the residual norm reaches zero within an error tolerance or when

a maximum number of iterations have been performed. The number of iterations is affected by convergence rate, which depends on the condition number of A.

The complexity of the conjugate gradient method is dominated by the matrix-vector multiplication, Ax, computed in $O(M^2)$. In [14] it is shown that if A is Toeplitz, then the complexity is reduced to $O(M \log M)$ by extending A into a circulant matrix C that has first column vector

$$c = [a_0 \quad a_1 \quad ... \quad a_{M-1} \quad \hat{a} \quad a_{-(M-1)} \quad ... \quad a_{-1}]^T \tag{2.32}$$

where $\{a_{i-j}\}$ are the entries of A and $\hat{a}$ can be any value. The modified matrix-vector multiplication is

$$C\hat{x} = \hat{y} \tag{2.33}$$

$$C = \begin{bmatrix} A & B \\ B & A \end{bmatrix}, \quad \hat{x} = \begin{bmatrix} x \\ 0 \end{bmatrix}, \quad \hat{y} = \begin{bmatrix} y \\ Bx \end{bmatrix}$$

where B is a $M \times M$ Toeplitz matrix with first column vector $b_c = [\hat{a}, a_{-(M-1)}, \ldots, a_{-1}]^T$ and first row vector $b_r = [\hat{a}, a_{M-1}, \ldots, a_1]$. By using the fact that a circulant matrix A_{circ} of size $M \times M$ is diagonalized by a M-point DFT matrix F_M as shown

$$A_{circ} = F_M^{-1} diag(F_M \cdot a_{circ}) F_M \tag{2.34}$$

where a_{circ} is the first column vector of A_{circ}, then (2.33) becomes

$$\hat{y} = F_{2M}^{-1} diag(F_{2M} \cdot c) F_{2M} \hat{x} \tag{2.35}$$

$$y = [\hat{y}_0, \ldots, \hat{y}_{M-1}]^T \tag{2.36}$$

and the computations involved are two FFTs, one vector multiplication, and one

inverse FFT. The matrix-vector multiplication in (2.35) can only be used as a forward projection since $\hat{x}$ is given and cannot be used to non-iteratively solve the system $Ax = b$ since the lower portion of the data vector $\hat{y}$ consisting of Bx includes solution vector x.

The $N \times N$ system matrix $A^H A$ in the normal equation in (2.8) is a real symmetric Toeplitz matrix and also a symmetric positive definite matrix. Therefore, we can use the conjugate gradient method to compute our solution in at most N iterations. A disadvantage of solving $A^H A$ is that its condition number is the square the condition number of A and so convergence is slower when solved using iterative methods.

For completeness, if the CG method is used to solve the 2-D problem in (2.3) then the matrix-vector multiplication with the Toeplitz-block-Toeplitz system matrix $A^H A$ can be performed in a similar manner but this time using the 2-D FFT.

2.5.3 Preconditioning

The CG method can be made to converge faster by preconditioning the system. In the precondition conjugate gradient (PCG) method, we solve the modified system,

$$P^{-1}Ax = P^{-1}b \tag{2.37}$$

where P is the preconditioner. The condition number of $P^{-1}A$ determines the effectiveness of the preconditioner. A good preconditioner is one that is as similar to A as possible so as to best diagonalize $P^{-1}A$ and one that has a low computational cost inverting itself in the operation, $P^{-1}b$.

For Toeplitz systems, we use a circulant preconditioner, where a Toeplitz matrix is converted into a circulant matrix of the same size. Specifically, in [18], T. Chan's preconditioner minimizes

$$\|P - A\|_F \tag{2.38}$$

over all circulant matrices P. This optimal preconditioner is constructed by replacing the k^{th} diagonal with weighted averages of the k^{th} and $(k-M)^{th}$ diagonals. Let A be an $M \times M$ Toeplitz matrix and P is T. Chan's circulant preconditioner. The first column elements of P, $\{p_i\}$, is formed from the elements of A, $\{a_{i-j}\}$, as follows,

$$p_i = \begin{cases} a_i, & i = 0 \\ \frac{M-i}{M} a_i + \frac{i}{M} a_{-(M-i)}, & 1 \leq i \leq M-1 \end{cases} \tag{2.39}$$

For example, if A is a 4×4 Toeplitz matrix then P is its preconditioner as shown:

$$A = \begin{bmatrix} a_0 & a_{-1} & a_{-2} & a_{-3} \\ a_1 & a_0 & a_{-1} & a_{-2} \\ a_2 & a_1 & a_0 & a_{-1} \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix} \quad P = \begin{bmatrix} a_0 & p_3 & p_2 & p_1 \\ p_1 & a_0 & p_3 & p_2 \\ p_2 & p_1 & a_0 & p_3 \\ p_3 & p_2 & p_1 & a_0 \end{bmatrix} \tag{2.40}$$

where

$$p_1 = \frac{3}{4} a_1 + \frac{1}{4} a_{-3}, \quad p_2 = \tfrac{2}{4} a_2 + \tfrac{2}{4} a_{-2}, \quad p_3 = \frac{1}{4} a_3 + \frac{3}{4} a_{-1}. \tag{2.41}$$

Notice that if A is Hermitian then P is also Hermitian: $p_1 = p_3^*$. Any circulant preconditioner can be inverted by diagonalizing it by DFT matrices as shown in (2.34) in $O(M \log M)$ operations.

Again for completeness in the 2-D problem, preconditioner of a TBT matrix is constructed by converting each Toeplitz block into circulant blocks again using T. Chan's method and then the block-Toeplitz structure is also converted into block-circulant. The circulant-block-circulant (CBC) preconditioner is easily diagonalized and inverted efficiently by the 2-D FFT as stated in [19].

CHAPTER III

Conditioning Approach and Results

3.1 Introduction

Since there is no 2-D Lagrange interpolation, sufficient conditions for the uniqueness and conditioning of the reconstruction problem are both not apparent. Using any of the unwrapping procedures presented below, the 2-D problem is transformed into a 1-D problem, from which uniqueness and, more importantly, insights into the problem conditioning result. We propose several sensitivity measures, which aid in determining a configuration of frequency values that produces a well-conditioned problem. The sensitivity measures are analyzed on their accuracy of estimating the conditioning and on their computational speed. The image is then reconstructed by solving the well-conditioned problem using either the Lagrange interpolation formula or the conjugate gradient method with less sensitivity to noise and faster convergence in the latter case. [1]

3.2 Unwrapping from 2-D to 1-D

Here we introduce the methods in which we unwrap the 2-D problem to solve the reconstruction problem in 1-D. Then we evaluate the conditioning of the problem in

[1] This chapter is based on our published work in [15].

1-D with the use of the Lagrange interpolation formula.

In [5] and [6] the 2-D DTFT is assumed to be given along contours of the form $\omega_2 = \alpha\omega_1 + \beta$. A contour of this form is really a straight line in unwrapped $\omega_1 \times \omega_2$ space, which is periodic with period 2π in both directions. Hence topology is toroidal; a line wraps around from one side of the square $[0, 2\pi) \times [0, 2\pi)$ to the other side and keeps going.

The results are easily interpreted as a single slice of the Radon transform,

$$g(\omega, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x_c(s,t)\delta(\omega - s\cos\theta - t\sin\theta)dsdt \tag{3.1}$$

in continuous space of the impulse bed image $x_c(s,t)$ where the impulse values correspond to the values of the discrete image as such: $x_c(\delta_s i_1, \delta_t, i_2) = x(i_1, i_2)$. Then recall from [1] that $X(\omega, 0)$ is the 1-D DFT of the projections of $x(i_1, i_2)$ taken in the vertical direction. Now rotate this slice of the 2-D DTFT from the ω_1-axis to some angle θ to this axis. This slice of the 2-D DTFT is the 1-D DTFT of the projections of $x_c(s,t)$ taken at an angle $\theta + \pi/2$ to the ω_1-axis. And unless θ is such that any two impulses belong on a line that has an angle $\theta + \pi/2$, each impulse will not sum and will remain separate in these projections. Hence knowledge of any slice of the 2-D DTFT at angle θ will, except for those special values of θ, determine $x(i_1, i_2)$ as values of a 1-D signal. Thus the 2-D problem has been transformed into a 1-D problem.

Note that if $\tan(\theta) = \dfrac{1}{M}$ or M the projections will simply be $x(i_1, i_2)$ unwrapped by rows or columns (lexicographic ordering). Hence any M^2 values of the 2-D DTFT along this slice will uniquely determine $x(i_1, i_2)$. Other angles of θ will produce projections, a 1-D signal, that are rows or columns of $x(i_1, i_2)$ separated by bands of zeros, so that more values of the 2-D DTFT along the slice are needed. Even if the projections consist of sums of values of various $x(i_1, i_2)$, recovery of $x(i_1, i_2)$ from its

2-D DTFT on several slices may still be possible, [3] and [20].

For the following unwrapping procedures, we provide an interpretation as a discrete Radon transform as describted in [20] as well as a Kronecker substitution in a 2-D z-transform.

3.2.1 Good-Thomas FFT

The Good-Thomas FFT transforms the 2-D problem into a 1-D problem as done in [8]. The Good-Thomas FFT, described in [21], maps an N_1N_2-point 1-D DFT to an $N_1 \times N_2$-point 2-D DFT, and vice-versa, using a residue number system (RNS) mapping. This requires that N_1 and N_2 be relatively prime. The computational savings comes from the recursive manner in which the Good-Thomas FFT is reapplied along each row and column of the newly mapped 2-D image.

The RNS mapping $x(i) \rightarrow x(i_1, i_2)$ is defined by the Good-Thomas shuffling equations

$$i_1 = i \qquad (\text{mod } N_1) \tag{3.2}$$

$$i_2 = i \qquad (\text{mod } N_2). \tag{3.3}$$

and $X(k_1, k_2) \rightarrow X(k)$ is defined by

$$k = N_2k_1 + N_1k_2 \qquad (\text{mod } N_1N_2) \tag{3.4}$$

The reverse RNS mapping $x(i_1, i_2) \rightarrow x(i)$ is defined using the Chinese remainder theorem, which states that there exist integers L_1 and L_2 such that

$$i = i_1L_2N_2 + i_2L_1N_1 \qquad (\text{mod } N_1N_2) \tag{3.5}$$

where L_1 and L_2 satisfy

$$L_1 N_1 + L_2 N_2 = 1. \tag{3.6}$$

Then $X(k) \rightarrow X(k_1, k_2)$ is defined by

$$k_1 = L_2 k \qquad (\text{mod } N_1) \tag{3.7}$$

$$k_2 = L_1 k \qquad (\text{mod } N_2) \tag{3.8}$$

$$\tag{3.9}$$

Here are two separate examples of the Good-Thomas frequency-domain mapping between 2-D and 1-D for when $N_1 = 3$ and $N_2 = 4$. The 1D solution values

$$x(i) = \left\{ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12 \right\}. \tag{3.10}$$

are mapped to the 2-D solution values

$$x(i_1, i_2) = \begin{Bmatrix} 1 & 10 & 7 & 4 \\ 5 & 2 & 11 & 8 \\ 9 & 6 & 3 & 12 \end{Bmatrix}. \tag{3.11}$$

The 2-D DFT values

$$Y(k_1, k_2) = \begin{Bmatrix} 1 & 4 & 7 & 10 \\ 5 & 8 & 11 & 2 \\ 9 & 12 & 3 & 6 \end{Bmatrix} \tag{3.12}$$

are mapped to the 1-D DFT values

$$Y(k) = \left\{ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10\ 11\ 12 \right\}. \tag{3.13}$$

Notice the wrapping in (3.11) and unwrapping in (3.12) occur in diagonal and anti-diagonal directions, respectively, specific for this example where $N_2 = N_1 + 1$.

The Good-Thomas FFT can also be viewed as a discrete Radon transform at angle θ where $\tan(\theta) = \frac{N_1}{N_2}$ or $\frac{N_2}{N_1}$. We can use the Good-Thomas FFT to unwrap the 2-D DFT of $x(i_1, i_2)$ into the 1-D DFT of a signal $x(i)$. This immediately leads to the uniqueness results of [8]: Knowledge of the $(N-1) \times N$-point 2-D DFT of an $M \times M$ image at any $2MN$ different frequency points is sufficient to specify the image uniquely. If the image support is slanted, where each successive column of the image is shifted down one position, then only MN different frequency points are required.

In this paper, we will set $N_2 = N_1 + 1$, which make N_1 and N_2 relatively prime. We select N_1 such that the locations $(\omega_{1,k}, \omega_{2,k})$ of all the known 2-D DTFT values $\{X(e^{\omega_{1,k}}, e^{\omega_{2,k}}), 1 \leq k \leq N\}$ fit a rectangular grid of frequency locations $\{(2\pi k_1/N_1, 2\pi k_2/N_2), 1 \leq k_1 \leq N_1, 1 \leq k_2 \leq N_2\}$. In [8] it is stated that choosing sufficiently large N_1 and N_2 will allow the choice of any $(\omega_{1,k}, \omega_{2,k})$ to an arbitrarily-small nearest-neighbor approximation, where in practice $(\omega_{1,k}, \omega_{2,k})$ is often defined on a rectangular lattice anyway, and that we only need some of the values on this lattice. Now the 2-D DTFT samples can be mapped to 1-D DTFT samples using (3.4). At this point the sensitivity of the 1-D problem can be measured for various frequency configurations as done in Section 3.4. Once a configuration with acceptable sensitivity is found, the 1-D signal is reconstructed as done in Section 3.3 and then mapped to a 2-D image using (3.2) and (3.3).

Another perspective on the unwrapping is to define the 2-D z-transform

$$X(z_1, z_2) = \sum_{i_1=0}^{M-1} \sum_{i_1=0}^{M-1} x(i_1, i_2) z_1^{i_1} z_2^{i_2} \tag{3.14}$$

and make the following Kronecker substitution. Letting $z_1 = w_2^N, z_2 = w_1^{-N}$ and evaluating on the unit circle at $w = e^{j2\pi k/(N_1 N_2)}$ leads to the Good-Thomas FFT.

3.2.2 45° Rotated-Support Kronecker Substitution

In Section 3.4 a variance-based sensitivity metric is presented to measure the conditioning of the problem. Then the image reconstruction is performed by an explicit reconstruction algorithm based on the Good-Thomas FFT explained in [8]. However the solution requires solving a large system matrix with a size twice the number of the image values due to zero-padding and the wrapping nature of the Good-Thomas FFT method. Therefore we present a newer problem formulation with a 45° rotated image-support. This new rotated formulation can also be considered as a DTFT generalization of the Good-Thomas FFT with rotated support as well.

3.2.2.1 Problem Statement

The goal again is to reconstruct an $M \times M$ discrete image $x(i_1, i_2)$ from M^2 values of its 2-D DTFT as was shown in (2.1). We make the following assumptions throughout:

1. $x(i_1, i_2) \neq 0$ only for:

$$i_1 + i_2 \equiv 0 \pmod 2 \tag{3.15}$$

$$|i_1 + i_2| \leq M - 1 \tag{3.16}$$

$$|i_1 - i_2| \leq M - 1 \tag{3.17}$$

   where M is odd. The image support is seen to be an $M \times M$ square rotated 45° (in MRI or CT we can choose basis functions on any locations);

2. $X(e^{j\omega_1}, e^{j\omega_2})$ is known for any M^2 points that satisfy

$$e^{j\omega_1(M+1)} = e^{j\omega_2(M-1)} \tag{3.18}$$

and consequently lie on the diagonal lines

$$\omega_2 = \frac{M+1}{M-1}\omega_1 + \frac{2\pi}{M-1}k \tag{3.19}$$

where $k = 0, 1, -1, 2, -2 \ldots$.

3.2.2.2 Small Example

For clarity this procedure is illustrated on a specific example. Consider the problem of reconstructing the 3×3 rotated-by-45° image

$$\begin{bmatrix} 0 & 0 & a_{-4} & 0 & 0 \\ 0 & a_{-1} & 0 & a_{-3} & 0 \\ a_2 & 0 & a_0 & 0 & a_{-2} \\ 0 & a_3 & 0 & a_1 & 0 \\ 0 & 0 & a_4 & 0 & 0 \end{bmatrix} \tag{3.20}$$

from 9 DTFT values $X(e^{j\omega_1}, e^{j\omega_2})$ on the diagonal lines

$$\omega_2 = 2\omega_1 + \pi k \tag{3.21}$$

where $k = -1, 0$ taken from (3.19).

Treat the center of the image (3.20) as the origin for coordinates, and take its 2-D z-transform. This gives

$$\begin{aligned} X(z_1, z_2) = a_{-4} \cdot z_2^2 + a_{-3} \cdot z_1 z_2 + a_{-2} \cdot z_1^2 \\ + a_{-1} \cdot z_1^{-1} z_2 + a_0 + a_1 \cdot z_1 z_2^{-1} \\ + a_2 \cdot z_1^{-2} + a_3 \cdot z_1^{-1} z_2^{-1} + a_4 \cdot z_2^{-2}. \end{aligned} \tag{3.22}$$

Now substitute $z_1 = z$ and $z_2 = z^2$:

$$\begin{aligned} X(z, z^2) = a_{-4}z^4 + a_{-3}z^3 + a_{-2}z^2 \\ + a_{-1}z^1 + a_0 z^0 + a_1 z^{-1} \\ + a_2 z^{-2} + a_3 z^{-3} + a_4 z^{-4}. \end{aligned} \tag{3.23}$$

$X(z, z^2)$ is clearly z^{-4} times a polynomial of degree 8, which has 9 coefficients. Hence it can be reconstructed uniquely from its values at any 9 distinct points (values of z).

The values of z are related to values of z_1 and z_2 by

$$z = \frac{z_2}{z_1} = \frac{e^{j\omega_2}}{e^{j\omega_1}} = e^{j(\omega_2 - \omega_1)} = e^{j\omega} \tag{3.24}$$

where $z_1 = e^{j\omega_1}$ and $z_2 = e^{j\omega_2}$.

We will derive an explicit formula for reconstructing uniquely the image from these DTFT values.

3.2.2.3 General Case

The general case of an $M \times M$ rotated-by-45° image should be evident. The following changes from the above example need to be made (M must be odd):

1. The substitution into the z-transform $X(z_1, z_2)$ of the image that unwraps the 2-D problem into a 1-D one is now

$$z_1 = z^{(M-1)/2}; \quad z_2 = z^{(M+1)/2} \tag{3.25}$$

which leads to

$$X(z^{\frac{M-1}{2}}, z^{\frac{M+1}{2}}) = \sum_{i_1} \sum_{i_2} x(i_1, i_2) z^{-(\frac{M-1}{2} i_1 + \frac{M+1}{2} i_2)}. \tag{3.26}$$

The 1-D z-transform equivalent is

$$X(z) = \sum_{i=-\frac{M^2-1}{2}}^{\frac{M^2-1}{2}} x(i) z^{-i} \tag{3.27}$$

where the 1-D time index

$$i = \frac{M-1}{2} i_1 + \frac{M+1}{2} i_2 \tag{3.28}$$

in terms of i_2 and i_2 relate the 1-D signal $x(i)$ with $x(i_1, i_2)$;

2. The data are now M^2 DTFT values $X(e^{j\omega_1}, e^{j\omega_2})$ that satisfy

$$e^{j\omega_1(M+1)} = e^{j\omega_2(M-1)} \tag{3.29}$$

a consequence of (3.25). These DTFT values lie on parallel diagonal lines

$$\omega_2 = \frac{M+1}{M-1} \omega_1 + \frac{2\pi}{M-1} k \tag{3.30}$$

where $k = 0, 1, -1, 2, -2 \ldots$;

3. The 2-D DTFT values now map to 1-D as

$$z = \frac{z_2}{z_1} = \frac{e^{j\omega_2}}{e^{j\omega_1}} = e^{j(\omega_2 - \omega_1)} = e^{j\omega} \tag{3.31}$$

where $z_1 = e^{j\omega_1}$ and $z_2 = e^{j\omega_2}$.

3.2.2.4 Even Image Size with Correction Factor

An image size where M is even may be used as was originally proposed with the factor $e^{j\pi\ell}$ to correct for non-unique 2-D to 1-D mapping of DTFT samples

$$X(e^{j\hat{\omega}_1}, e^{j\hat{\omega}_2}) = X(e^{j\omega_1}, e^{j\omega_2})e^{j\pi\ell}. \tag{3.32}$$

where if $\omega_1 \times \omega_2 \in [-\pi, \pi) \times [-\pi, \pi)$ then ℓ denotes how many times ω_1 has wrapped around. The integer ℓ also denotes which diagonal line segment (ω_1, ω_2) lies on

$$\ell = \frac{1}{2\pi}(\omega_2(M-1) - \omega_1(M+1)). \tag{3.33}$$

The following sections assume the problem statement with the odd image size.

3.2.2.5 Interpretation

The substitution (3.25) can be regarded as the transformation

$$\frac{y}{x} = z = z_1; \quad yx = z^M = z_2. \tag{3.34}$$

The substitution $z_2 = z^M$ for unwrapping a 2-D signal to a 1-D signal is the Kronecker substitution. It unwraps the 2-D signal column-by-column.

On the unit circle

$$x = e^{j\omega_x}; y = e^{j\omega_y}; z_1 = e^{j\omega_{z_1}}; z_2 = e^{j\omega_{z_2}} \tag{3.35}$$

the transformation (3.34) becomes the 45° rotation

$$\begin{bmatrix} \omega_{z_2} \\ \omega_{z_1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix}. \tag{3.36}$$

Since the image is rotated 45°, it makes sense to rotate the frequency axes by 45° as well.

Even if we knew $X(e^{j\omega_1}, e^{j\omega_2})$ for all of these values of k_1 and k_2, we would still not uniquely determine the 3×3 example image in (3.20), since $(M+1) \times (M-1) = M^2 - 1 = 8$ values are insufficient to reconstruct uniquely a signal of length $M^2 = 9$. But if we know *some* of these values for some constant c_0, and *some* of these values for another constant c_1, and so on, so that the total number of distinct known values is 9, then we can uniquely reconstruct $I(z^{2/2}, z^{4/2})$. In general we can choose any M^2 distinct $X(e^{j\omega_1}, e^{j\omega_2})$ values that lie on the lattice to reconstruct $I(z^{\frac{M-1}{2}}, z^{\frac{M+1}{2}})$.

It is also possible to interpret the transformation (3.34) using the discrete Radon transform. Using the projection-slice theorem, the projections along lines orthogonal to a line with slope $\frac{M+1}{M-1}$ are the inverse DFT of a slice in the 2-D DTFT plane at that slope. It is straightforward to confirm that these projections form the unwrapped 1-D signal (e.g., unwrapping the 2-D image (3.20) to the 1-D signal (3.23)). The slice in the 2-D DTFT plane at slope $\frac{M+1}{M-1}$ can be confirmed to define the diagonal lines (3.30).

3.2.3 Helical Scan FFT

As mentioned in Section 3.2.2.5 the rotated image support also rotates the frequency domain which allows for points on a rectangular lattice to fit on the near 45° diagonal lines. However the unrotated image support where the lines of available DTFT samples are slightly diagonal with a slope of M requires a different Kronecker substitution of

$$z_1 = z^1; \quad z_2 = z^M \tag{3.37}$$

into the 2-D z-transform in (3.14) which leads to the 1-D time index of

$$i = i_1 + Mi_2 \tag{3.38}$$

and the M^2 DTFT values satisfy

$$e^{j\omega_1 M} = e^{j\omega_2} \tag{3.39}$$

on the unit circle as a consequence of (3.37).

These DTFT values lie on parallel lines

$$\omega_2 = M\omega_1 + 2\pi k \tag{3.40}$$

where $k = 0, 1, -1, 2, -2 \ldots$. This modification is essentially the unrotated version of the prior method but is related to the "helical scan" FFT in [22]. This can be viewed as taking the projection of a impulse bed image at angle $\arctan(1/M)$ which is now a 1-D signal at sampling period $1/M$ over a field-of-view (FOV) of M, with M^2 impulses. Using the projection-slice theorem, the 1-D FT of the projection is another impulse train with frequency delta of $1/M$, but normalized sampling frequency is M. In 2-D though, this wraps around. This is also the $\tan\theta = 1/M$ or M case of the discrete Radon transform as it is noted in [6].

3.3 1-D Reconstruction

This "just-determined" problem, where M values, DFT samples in the Good-Thomas FFT unwrapped case or DTFT values in the rotated support and helical scan FFT cases, are sufficient to reconstruct a unique signal of length M, can be formulated as a solution of a Vandermonde system of equations $Vx = b$ in (2.5). We can then uniquely reconstruct $I(z^{\frac{M-1}{2}}, z^{\frac{M+1}{2}})$ in closed-form using the Lagrange interpolation formula in (2.16)

When the system $Vx = b$ is ill-conditioned, regularization, as formulated in (2.11), can modify the problem to improve the conditioning. However, in reducing the vari-

ance of the solution regularization also biases the solution, which is an undesirable side-effect.

As the just-determined problem in general is poorly conditioned, it is necessary to consider the over-determined case. This over-determined problem can be formulated as the non-square Vandermonde system of equations $Ax = b$ in (2.6) and solve in normal equation form in (2.8). The system matrix $A^H A$, a symmetric and positive definite matrix, can use the conjugate gradient method to compute the solution.

3.4 Sensitivity Measure

We desire a sensitivity measure that accurately estimates the conditioning of a system constructed from some frequency values $\{X(e^{j\omega_k}), 1 \leq k \leq N\}$. While the accuracy criterion is paramount, we desire a low-complexity measure because it may be calculated multiple times, once for each frequency configuration to be evaluated. This allows the relative improvement in adding frequency values to be evaluated quickly, to see if it is worth the cost of, say, adding a sensor to produce that extra value. We illustrate this in Section 3.6.1.2 and 3.6.1.3.

3.4.1 Condition Number

One choice for measuring the sensitivity of the Vandermonde system (2.5) is the condition number

$$\kappa_p(A) = \|A\|_p \cdot \|A^{-1}\|_p \tag{3.41}$$

where $p = 1, 2, \ldots$ and A is square and invertible.

When A is Hermitian and the ℓ_2-norm is used where $p = 2$ in (3.41), the condition number is equal to the ratio of the maximum to the minimum eigenvalue magnitudes of A. These eigenvalues can be computed using the power and inverse power methods. Unfortunately, these often take thousands of iterations to converge (see [14]).

We would like to develop a simple procedure for evaluating the relative conditioning of various configurations of given 2-D DTFT samples without having to compute the computationally expensive condition number of the system matrix.

3.4.2 Extended Circulant Matrix Condition Number

For over-determined problems, we can try to relate the condition number of either a Toeplitz or Toeplitz-block-Toeplitz (TBT) system matrix to the condition number of a circulant matrix that has been modified from the original matrix. The motivation for this method is that computing the condition number of a circulant matrix is very quick (see below).

We illustrate the procedure with a specific example. Let A be a Hermitian TBT matrix as shown:

$$A = \left[\begin{array}{cc|cc} a_0 & a_1 & a_3 & a_4 \\ a_1^* & a_0 & a_2 & a_3 \\ \hline a_3^* & a_2^* & a_0 & a_1 \\ a_4^* & a_3^* & a_1^* & a_0 \end{array}\right] \tag{3.42}$$

We insert rows and columns between the Toeplitz blocks in A to create a Toeplitz matrix B:

$$B = \left[\begin{array}{cc|c|cc} a_0 & a_1 & a_2 & a_3 & a_4 \\ a_1^* & a_0 & a_1 & a_2 & a_3 \\ \hline a_2^* & a_1^* & a_0 & a_1 & a_2 \\ \hline a_3^* & a_2^* & a_1^* & a_0 & a_1 \\ a_4^* & a_3^* & a_2^* & a_1^* & a_0 \end{array}\right] \tag{3.43}$$

Finally, the columns of B are extended maintaining conjugate symmetry to create

a circulant matrix C:

$$C = \left[\begin{array}{cccc|ccc} a_0 & \cdots & a_3 & a_4 & a_4^* & \cdots & a_1^* \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_3^* & \cdots & a_0 & a_1 & a_2 & \cdots & a_4^* \\ a_4^* & \cdots & a_1^* & a_0 & a_1 & \cdots & a_4 \\ \hline a_4 & \cdots & a_2^* & a_1^* & a_0 & \cdots & a_3 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & \cdots & a_4 & a_4^* & a_3^* & \cdots & a_0 \end{array}\right] \tag{3.44}$$

where $c = [a_0, a_1, a_2, a_3, a_4, a_4^*, a_3^*, a_2^*, a_1^*]^H$ is the first column of C. In [14] it is shown that eigen-decomposition of a circulant matrix produces to an N-point DFT matrix, F_N, as its matrix of eigenvectors and the DFT of c as its eigenvalues, as such:

$$C = F_N^{-1} \operatorname{diag}(F_N c) F_N \tag{3.45}$$

We can compute the condition number of C as the ratio of the maximum and minimum magnitudes of its eigenvalues, which can be quickly computed by taking the FFT of c, which has a complexity of $O(N \log N)$.

Now, we must establish a relationship between the condition numbers of the TBT and circulant matrices using the Cauchy interlace theorem as explained in [14] and [23], which states that if a row-column pair is deleted from a square Hermitian matrix, then the eigenvalues of the resulting matrix alternate with those of the original one. Continuing with the previous example, let A and C have the eigenvalues $\alpha_1 \leq \ldots \leq \alpha_M$ and $\gamma_1 \leq \ldots \leq \gamma_N$, respectively, where $M < N$. The following more precisely restates the Cauchy interlace theorem:

$$\gamma_k \leq \alpha_k \leq \gamma_{k+N-M}, \quad k = 1, \ldots, M \tag{3.46}$$

Shown below is a graphical representation of the pyramid structure of the interlacing eigenvalues. The $\{\beta_k\}$ are the eigenvalues of the $(M+1)\times(M+1)$ supermatrix of A and $\{\hat{\beta}_k\}$ are the eigenvalues of the $(N-1)\times(N-1)$ submatrix of C.

$$\begin{array}{ccccccccccccccccc}
 & & & \alpha_1 & & \alpha_2 & & \cdots & & \alpha_{M-1} & & \alpha_M & & & & & \\
 & & \beta_1 & & \beta_2 & & \beta_3 & & \cdots & & \beta_M & & \beta_{M+1} & & & & \\
 & & & & \vdots & & & & & & & \vdots & & & & & \\
 & \hat{\beta}_1 & & \hat{\beta}_2 & & \hat{\beta}_3 & & \hat{\beta}_4 & & \cdots & & \hat{\beta}_{N-2} & & \hat{\beta}_{N-1} & & & \\
\gamma_1 & & \gamma_2 & & \gamma_3 & & \gamma_4 & & \gamma_5 & & \cdots & & \gamma_{N-1} & & \gamma_N
\end{array} \tag{3.47}$$

Letting $k = 1$ and $k = M$ in (3.46), we specify bounds on the largest and smallest eigenvalues of A in terms of $\{\gamma_k\}$,

$$\gamma_1 \leq \quad \alpha_1 \quad \leq \gamma_{1+N-M} \tag{3.48}$$

$$\gamma_M \leq \quad \alpha_M \quad \leq \gamma_N. \tag{3.49}$$

Since A and C are both Hermitian, their condition numbers are $\kappa_2(A) = \dfrac{\max_k |\alpha_k|}{\min_k |\alpha_k|}$ and $\kappa_2(C) = \dfrac{\max_k |\gamma_k|}{\min_k |\gamma_k|}$, respectively. Matrix A is formed from a cross-product $V^H V$ where V has full column rank. Therefore, A is positive definite, so all of the eigenvalues of A are positive. This results in $\kappa_2(A) = \dfrac{\alpha_M}{\alpha_1}$.

If all of the eigenvalues of C are also positive (equivalently $\gamma_1 > 0$) then $\kappa_2(C) = \dfrac{\gamma_N}{\gamma_1}$. Rearranging the inequalities

$$\gamma_1 \leq \alpha_1, \quad \alpha_M \leq \gamma_N \tag{3.50}$$

from (3.49), we can state that if C is positive definite then the condition number of

C is an upper bound on the condition number of A,

$$\kappa_2(A) = \frac{\alpha_M}{\alpha_1} \leq \frac{\gamma_N}{\gamma_1} = \kappa_2(C). \tag{3.51}$$

However, if the lowest eigenvalue of C is negative then $\min_k |\gamma_k|$ is not necessarily γ_1 and the condition number of C no longer serves as an upper bound on the condition number of A.

We illustrate this point with a specific counter-example. Let

$$A_{4\times4} = \left[\begin{array}{cc|cc} 5 & 4 & 1 & -1 \\ 4 & 5 & 3 & 1 \\ \hline 1 & 3 & 5 & 4 \\ -1 & 1 & 4 & 5 \end{array}\right] C_{9\times9}(v),\ v = \left[\begin{array}{ccccc|cccc} 5 & 4 & 3 & 1 & -1 & -1 & 1 & 3 & 4 \end{array}\right] \tag{3.52}$$

with respective eigenvalues,

$$\begin{array}{llll} \alpha_1 = 0.39, & \alpha_2 = 0.61, & \alpha_3 = 7.61, & \alpha_4 = 11.39, \\ \gamma_1 = -1.78, & \gamma_2 = -1.78, & \gamma_3 = 0.73, & \gamma_4 = 0.73, \\ \gamma_5 = 1.00, & \gamma_6 = 1.00, & \gamma_7 = 13.05, & \gamma_8 = 13.05, \quad \gamma_9 = 19.00. \end{array} \tag{3.53}$$

Notice that $\gamma_3 = 0.73$ is the smallest absolute value of C, not $\gamma_1 = -1.78$ and so $\kappa_2(C) = \dfrac{|\gamma_9|}{|\gamma_3|}$, instead of $\dfrac{|\gamma_9|}{|\gamma_1|}$. In fact $\kappa_2(A) = 28.86 \geq \kappa_2(C) = 25.98$. This counter-example is evidence that $\kappa_2(C)$ is not an upper-bound for $\kappa_2(A)$.

Alternatively, we attempt to establish a lower bound on $\kappa_2(A)$ by using the inequalities

$$\alpha_1 \leq \gamma_{1+N-M}, \quad \gamma_M \leq \alpha_M. \tag{3.54}$$

Since α_1 is positive, γ_{1+N-M} is also positive. Rearranging the inequalities in (3.54)

results in

$$\kappa_2(A) = \frac{\alpha_M}{\alpha_1} \geq \frac{\gamma_N}{\gamma_{1+N-M}} = \kappa_2(C). \tag{3.55}$$

The restriction on N is that $1 + N - M \leq M$ or,

$$M \leq N \leq 2M - 1, \tag{3.56}$$

because $\kappa_2(A)$ must be greater than or equal to 1, and because γ_M may cross over γ_{1+N-M} and become negative. However, a condition that restricts the dimensions of the extended circulant matrix to be less than twice the dimension of the original Toeplitz, mosaic-Toeplitz, or TBT matrix makes this lower bound unreliable. One example in which this occurs is for the matrices in (3.52) where A has dimensions 4×4 and C has dimensions 9×9.

Other scenarios to consider is that C could be singular in situations where the minimum eigenvalue of C, γ_1 is 0. This situation arises when the diagonalization of C produces 0 values, particularly any value value of $F_N c$ in (3.45), leading to a 0 determinant.

With this in mind, we also can enforce that we only have positive eigenvalues as presented in this modified circulant matrix

$$\hat{C} \triangleq F_N^{-1} \operatorname{diag}(\max(F_N c, \epsilon)) F_N \tag{3.57}$$

which is still circulant and also "extended" from A, where ϵ is a small non-zero number which should be less than α_1.

Although the eigenvalues of the extended circulant matrix can quickly be computed, the proposed bound measures do not properly bound the condition number of the system matrix and is an inaccurate sensitivity measure. Otherwise further

investigation is required of the proposed modification in (3.57). Therefore, we do not investigate this measure any further.

3.4.3 Energy of Lagrange Fundamental Polynomials

The problem of reconstructing $X(z)$ from M samples $\{X(k), k = 1, \ldots, M\}$ is formulated as the solution of the Vandermonde system of equations in (2.5), which can be solved using the Lagrange interpolation formula (2.16). As defined in [24], Lagrange's fundamental interpolating polynomials are

$$\pi_k(u) = \prod_{\substack{\ell=1 \\ \ell \neq k}}^{M} \frac{u - z_\ell}{z_k - z_\ell}. \tag{3.58}$$

The Lebesgue constant discussed in [25] uses (3.58) to compute a bound on the interpolation error as,

$$\Lambda_M(U) = \max_{-1 \leq u \leq 1} \sum_{k=1}^{M} \left| \prod_{\substack{\ell=1 \\ \ell \neq k}}^{M} \frac{u - z_\ell}{z_k - z_\ell} \right| \tag{3.59}$$

with omission of the frequency value $X(z_k)$. The Lebesgue constant has a lower bound, which is a function of M. However, a tight lower bound has yet to be found and determining the optimal set U for which the Lebesgue constant is smallest is an unsolved problem. Therefore, we do not further investigate this measure either.

However, the Lagrange interpolation formula (2.16) can be shown as a closed-form computation of the inverse of the Vandermonde system matrix V, which can lead us to an exact expression of the condition number. In the inverse DFT equation

$$x(n) = \frac{1}{M} \sum_{m=0}^{M-1} X(u_m) u_m^n \tag{3.60}$$

where $u_m = e^{\frac{j2\pi m}{M}}$. Substituting the uniformly distributed 1-D DTFT values $X(u_m)$

with the Lagrange interpolation equation in (2.16) results in

$$\begin{aligned} x(n) &= \frac{1}{M}\sum_{m=0}^{M-1}\left(\sum_{k=1}^{M}X(z_k)\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{u_m - z_\ell}{z_k - z_\ell}\right)u_m^n \\ &= \sum_{k=1}^{M}\left[\frac{1}{M}\sum_{m=0}^{M-1}\left(\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{u_m - z_\ell}{z_k - z_\ell}\right)u_m^n\right]X(z_k) \\ &= \sum_{k=1}^{M}V_{n+1,k}^{-1}X(z_k) \end{aligned} \tag{3.61}$$

where $0 \leq n \leq M-1$, and $z_k = e^{j\omega_k}$. Then the elements of the inverse Vandermonde matrix,

$$V_{n+1,k}^{-1} = \frac{1}{M}\sum_{m=0}^{M-1}\left(\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{u_m - z_\ell}{z_k - z_\ell}\right)u_m^n, \tag{3.62}$$

are the inverse DFT values of the Lagrange fundamental interpolating polynomials evaluated at uniformly spaced samples on the unit circle.

The advantage of (3.62) is that the sensitivity of reconstruction of any specific signal value $x(n)$ to variations in any specific data point $X(z_k)$ can be computed directly and explicitly. This amounts to computing any specific entry of V^{-1} in (3.62). Thus the reliability of reconstruction of specific $x(n)$ can be computed. In particular, this raises the possibility of selecting frequency points so that a specific $x(n)$ is relatively insensitive to noise (reliable reconstruction of a specific region of interest).

An overall measure of the sensitivity of the inverse problem is the sum of squared magnitudes of all elements of V^{-1}. By Parseval's theorem, the energy of $\{V_{n,k}^{-1}\}$ and the energy of its DFT are related by:

$$\sum_{n=1}^{M}|V_{n,k}^{-1}|^2 = \frac{1}{M}\sum_{m=1}^{M}\left|\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{u_m - z_\ell}{z_k - z_\ell}\right|^2 \tag{3.63}$$

The energy (or sum of squared magnitudes) of the Lagrange fundamental polynomial samples are related to the condition number in the following manner. The squared Frobenius norm of V is

$$\|V\|_F^2 = \sum_{k=1}^{M}\sum_{n=1}^{M} |V_{n,k}|^2 = M^2, \tag{3.64}$$

whereas

$$\|V^{-1}\|_F^2 = \sum_{k=1}^{M}\sum_{n=1}^{M} |V_{n,k}^{-1}|^2 = \frac{1}{M}\sum_{k=1}^{M}\sum_{m=1}^{M}\left|\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{u_m - z_\ell}{z_k - z_\ell}\right|^2. \tag{3.65}$$

Then the squared condition number of V associated with the Frobenius norm is

$$\kappa_F(V)^2 = \|V\|_F^2\|V^{-1}\|_F^2 = M\sum_{m=1}^{M}\sum_{k=1}^{M}\left|\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{u_m - z_\ell}{z_k - z_\ell}\right|^2 \tag{3.66}$$

where $u_m = e^{\frac{j2\pi m}{M}}$ and $z_k = e^{j\omega_k}$. Hence the squared condition number of the Vandermonde matrix V, using the Frobenius norm, is equal to the energy of the Lagrange fundamental polynomial samples (to a factor of M).

Further evaluation of (3.66) leads to the following:

$$\begin{aligned}\kappa_F(V)^2 &= M\sum_{m=1}^{M}\sum_{k=1}^{M}\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{|e^{\frac{j2\pi m}{M}} - e^{j\omega_\ell}|^2}{|e^{j\omega_k} - e^{j\omega_\ell}|^2}\\ &= M\sum_{m=1}^{M}\sum_{k=1}^{M}\prod_{\substack{\ell=1\\ \ell\neq k}}^{M}\frac{1-\cos(\frac{2\pi m}{M} - \omega_\ell)}{1-\cos(\omega_k - \omega_\ell)}\end{aligned} \tag{3.67}$$

Considering that

$$\frac{1-\cos(\theta)}{2} \in [0,1], \quad \forall\ \theta \in \Re, \tag{3.68}$$

we show that (3.67) is bounded:

$$\kappa_F(V)^2 \geq M^2 \sum_{m=1}^{M} \prod_{\substack{\ell=1 \\ \ell \neq k}}^{M} \frac{1 - \cos(\frac{2\pi m}{M} - \omega_\ell)}{2} \tag{3.69}$$

$$\kappa_F(V)^2 \leq M^2 \sum_{k=1}^{M} \prod_{\substack{\ell=1 \\ \ell \neq k}}^{M} \frac{2}{1 - \cos(\omega_k - \omega_\ell)}. \tag{3.70}$$

If any of the quantities $|\omega_k - \omega_\ell|$ in (3.70) decrease towards zero, the upper-bound on $\kappa_F(V)^2$ approaches infinity. We will use this insight in developing an appropriate sensitivity measure.

Note that the condition number associated with the ℓ_2-norm is

$$\kappa_2(V) = \|V\|_2 \|V^{-1}\|_2 = \frac{\sigma_{max}(V)}{\sigma_{min}(V)} \tag{3.71}$$

where $\sigma_{max}(V)$ is the maximum singular value of V, i.e., the square root of the maximum eigenvalue of $V^H V$. Although $\kappa_2(V) \neq \kappa_F(V)$, they are related, since

$$\|V\|_2 \leq \|V\|_F \leq \sqrt{M}\|V\|_2 \tag{3.72}$$

as shown in [14]. Hence the energy of the Lagrange fundamental polynomial samples is "closely" related to the ℓ_2-norm condition number of the Vandermonde system matrix to a factor of $\sqrt{(M)}$. Equation (3.66) holds true only for the just-determined case because the use of the Lagrange interpolation formula (2.16) in its derivation.

3.4.4 Approximations to Energy of Lagrange Fundamental Polynomials

To reduce the complexity of the upper-bound computation to $O(M)$, one approximation computes only the adjacent frequency differences as such:

$$\begin{aligned} \text{AdjNeighbUpperBoundApprox} &= M^2 \sum_{k=1}^{M} \prod_{\ell=k+1}^{M} \frac{2}{1-\cos(\omega_k - \omega_\ell)} \\ &= M^2 \sum_{k=1}^{M} \frac{2}{1-\cos(\omega_k - \omega_{k+1})}. \end{aligned} \tag{3.73}$$

where $\omega_1 < \omega_2 < \ldots < \omega_{M+1} = \omega_1 + 2\pi$. The other multiplicative terms that measure non-adjacent frequency differences do contribute to the condition number as empirical tests show in Section 3.4.6.

With this in mind a different approach is computing the products with just $2L$ terms instead of $M-1$ as in $\prod_{\substack{\ell=k-L \\ \ell\neq k}}^{k+L} \frac{2}{1-\cos(\omega_k-\omega_\ell)}$ where these terms measure distances from ω_k to its $2L$ closest neighbors. The multiplicative terms left out tend to be larger since they are much further from ω_k. This k is found to be one that produces the largest estimate of $\prod_{\substack{\ell=1 \\ \ell\neq k}}^{M} \frac{2}{1-\cos(\omega_k-\omega_\ell)}$ which only took $O(LM)$. Then we can compute the kth term fully in $O(M)$. The product $\prod_{\substack{\ell=1 \\ \ell\neq k}}^{M} \frac{2}{1-\cos(\omega_k-\omega_\ell)}$ becomes large when there is a clustering of frequency values around ω_k and the kth product term is much greater than any other term so the other terms are negligible when performing the final sum. Hence the other products are not computed. If L, the "search width" is set to M then we have the exact upper-bound expression computed in $O(M^2)$. If L is set to some constant then we have $O(N)$ time but may not scale well as M increases since the number of clustering frequencies may also increase. If L is arbitrarily set between 1 and M such as $\log_2(M)$ then a balance can be struck between accuracy and complexity.

$$k_{max} = \overset{argmax}{k} \prod_{\substack{\ell=k-L \\ \ell \neq k}}^{k+L} \frac{2}{1-\cos(\omega_k - \omega_\ell)}$$

$$\text{MaxSearchUpperBoundApprox} = M^2 \prod_{\substack{\ell=1 \\ \ell \neq k}}^{M} \frac{2}{1-\cos(\omega_{k_{max}} - \omega_\ell)} \tag{3.74}$$

3.4.5 Variance of Distances between Frequency Locations

We can take another approach to reduce the complexity of the upper-bound computation to $O(M)$. Examination of $\prod_{\ell=1,\ell\neq k}^{M}(1-\cos(\omega_k - \omega_\ell))$, the denominator of the Frobenius condition number (3.67) and of its upper-bound (3.70), shows that the configuration of the frequency locations $\{\omega_k, 0 \leq k \leq M-1\}$ significantly affects the sensitivity of the problem. Two DTFT samples with locations ω_w and ω_ℓ that are very close will cause the upper-bound on the condition number to become very large. To reduce this upper-bound, the minimum distance between any two frequency locations $\{\omega_k, 0 \leq k \leq M-1\}$ must be maximized which leads to an equally spaced distribution of $\{\omega_k, 0 \leq k \leq M-1\}$. This insight, that increased sensitivity is a result of a departure from a uniform distribution of frequency locations, relates the variance of distances between adjacent frequency locations to the condition number.

The variance of distances between adjacent frequency locations or the "variance sensitivity measure" in the 1-D unwrapped problem,

$$\text{VarianceMeasure} = \frac{1}{M}\sum_{k=1}^{M}(|\omega_{k+1} - \omega_k| - \mu)^2 \tag{3.75}$$

$$\mu = \frac{1}{M}\sum_{k=1}^{M}|\omega_{k+1} - \omega_k|,$$

$$\omega_1 < \omega_2 < \ldots < \omega_{M+1},\ \omega_{M+1} = \omega_1 + 2\pi,$$

is a good candidate for a measure of sensitivity. Note that the use of the scale

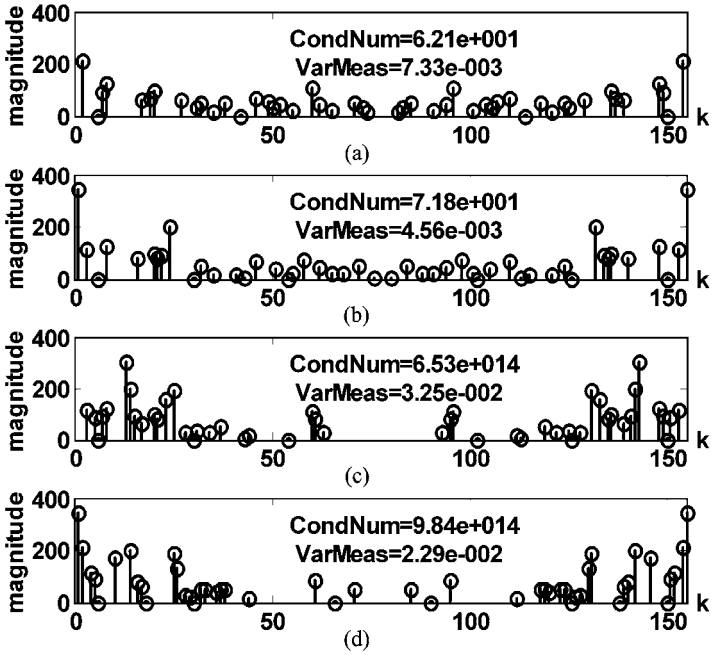


Figure 3.1: Various unwrapped 1-D frequency configurations.

factor $\frac{1}{M}$ instead of $\frac{1}{M-1}$ is intentional because we are computing the variance of a complete set of frequency location differences instead of samples of a random variable. The advantage of this measure is that its algorithm has a complexity of $O(M)$ and is applicable in both the just-determined and over-determined cases.

In the previous example of various 2-D frequency configurations in Fig. 2.1, clustering or wide gaps in between frequency locations are indicators of a poorly conditioned problem. This characteristic can be better seen once they are unwrapped into 1-D using the Good-Thomas FFT as shown in Fig. 3.1. The frequency locations in Fig. 3.1a and 3.1b are close to uniformly distributed with low variance measure values. In contrast, the frequency locations in Fig. 3.1c and 3.1d include large gaps and clustering with high variance measure values. The configurations with low and high variance measure values also have low and high condition numbers, respectively.

A variation on the variance measure involves incorporating back some of the nor-

malized non-adjacent distances leads to the following sensitivity measure,

$$\text{MultiNeighbVarianceMeasure} = \frac{1}{M}\sum_{\ell=1}^{L}\frac{1}{d(\ell,M)}\sum_{k=1}^{M}(|\omega_{k+\ell}-\omega_k|-\mu)^2 \qquad (3.76)$$

$$d(\ell,M) = \frac{\sin(\frac{\pi\ell}{M})}{\sin(\frac{\pi}{M})}$$

where d is the Cartesian distance between M uniformly distributed frequency locations on the complex unit circle, or the side length of an M-sided regular polygon, that normalizes the distance between skipped neighbors and ℓ is the integer index distance between frequency locations. The time complexity computing this measure is greater at $O(ML)$.

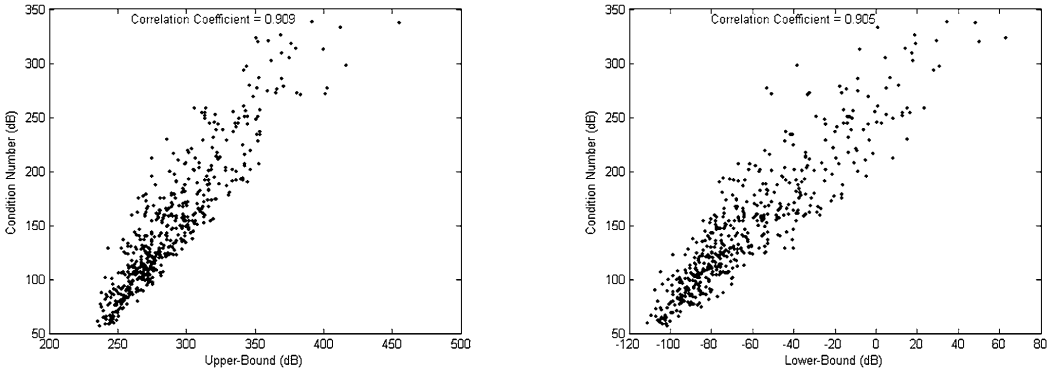
3.4.6 Performance of the Sensitivity Measure

The relationship between the condition number (3.67) and the proposed sensitivity measures are empirically verified by using their correlation coefficient.

We compute the condition number (using the ℓ_2-norm) and the sensitivity measure for 1000 system matrices like A in (2.6) over varying how much the problem is overdetermined as shown in Fig. 3.4. Matrix A solves for a solution vector of length 64 (or 4×4 in wrapped in 2D) and is constructed from exactly 64 up to an overdetermined 3×64 randomly chosen set of 1-D DTFT samples, unwrapped from a $20 \times 21(N_1/M = 5)$ 2-D grid using the Good-Thomas FFT. The condition number serves as the benchmark for a sensitivity measure.

The upper and lower bounds for the Frobenius condition number given in (3.70)-(3.70) correlate at 0.90 with the condition number as shown in Fig. 3.2. However they both require $O(M^2)$ computations compared to the $O(M)$ computations of the variance measure. Though $O(M^2)$ is still better than the $O(M^3)$ required in com-

puting the largest and smallest magnitude eigen values (or singular values) using the power iteration (or singular value decomposition).



(a) Upper-bound (b) Lower-bound

Figure 3.2: Scatter plots between condition number and bounds.

The "AdjNeighbUpperBoundApprox" measure of $O(M)$ had a correlation coefficient of only 0.33. When L was arbitrarily set to $\log_2(M)$ then the "MaxSearchUpperBoundApprox" measure $O(M \log M)$ performed better with a correlation coefficient of 0.83 as shown in Fig.3.3.

The relationship between the condition number and the variance sensitivity measure (3.75) in particular is based on the insight that both measures increase when the distances between DTFT samples decrease. The variance sensitivity measure has

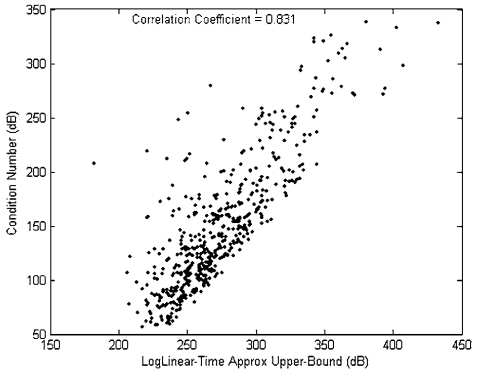


Figure 3.3: Scatter plot between condition number and "max search" upper-bound approximation.

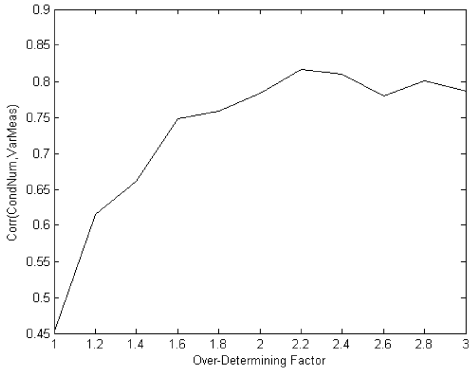


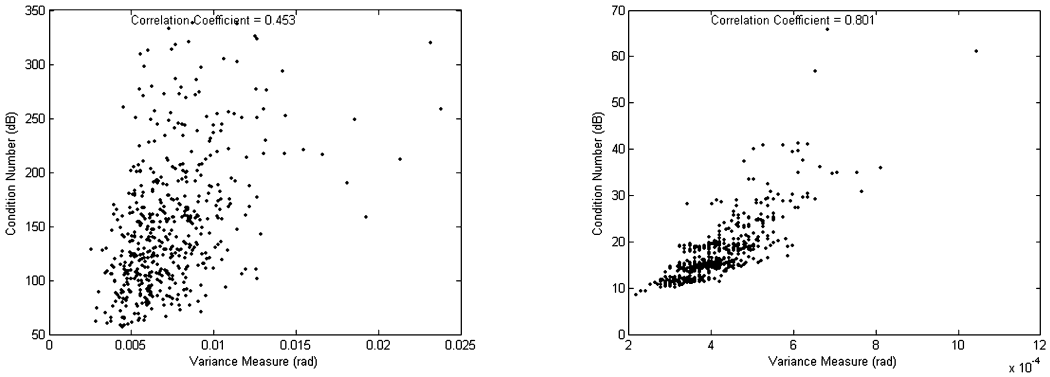
Figure 3.4: Correlation coefficient between condition number and variance sensitivity measure versus over-determining factor.

a 0.45 correlation with the condition number in the just-determined case shown in the scatter plot in Fig. 3.5. However in the 3-times overdetermined case the correlation rises to 0.80. Fig. 3.4 shows that the correlation plateaus at about 2 times over-determining.

Since the just-determined case is usually so poorly-conditioned as mentioned before, we focus on the over-determined case. Because the overdetermined problem is an entirely different system of equations, the measures derived from the Lagrange interpolation formula do not work well. The variance measure performs far better in correlating with the condition number in the over-determined case and has a low time complexity. Therefore the variance measure is our sensitivity estimate of choice for the remainder this work.

The extension of the variance measure, "MultiNeighbVarianceMeasure," has empirically shown some increases in the correlation coefficient commensurate with the additional cost.

We just showed empirically the variance measure strongly correlates with the condition number. However when the condition number is impractical to compute for large system matrices we can show the relationship between the condition number and the performance of the conjugate gradient method which then relates to the



(a) Just-determined (b) Over-determined

Figure 3.5: Scatter plots between condition number and variance sensitivity measure.

variance measure.

From [26] we know the residual error reduction per PCG iteration is approximately $(\sqrt{\kappa}-1)/(\sqrt{\kappa}+1)$, where $\kappa = \kappa_2(A)$ is the condition number. The total residual error

$$\epsilon \leq 2e^{\frac{-2i}{\sqrt{\kappa}}} \tag{3.77}$$

over i iterations has an upper bound that is a function of the condition number. The relative residual of configurations with lower variance measures decrease faster than the relative residual of the configurations with higher variance measures. A smaller relative residual $\epsilon = |Ax_{(i)} - b|/|b|$ then tends to lead to a smaller MSE of the solution x.

Conversely it can be shown that if the PCG relative residual tolerance ϵ is fixed, then a frequency configuration with a lower variance measure will require fewer iterations for convergence of the PCG method than a configuration with a higher variance measure. Solving (3.77) for the number of iterations i we get

$$i \leq \frac{1}{2}\sqrt{\kappa}\ln\left(\frac{2}{\epsilon}\right). \tag{3.78}$$

Therefore a frequency configuration with a lower variance measure requires fewer

iterations, typically a consequence of better conditioning.

3.5 Frequency Selection

The sufficient number of DTFT values needed for image reconstruction is not always immediately clear as shown in [8]. In the case of the 45° rotated problem we can use M^2 DTFT values (or greater for the over-determined case) while using a sensitivity measure to select the DTFT locations that produce a well-conditioned system. The DTFT locations are chosen anywhere on the diagonal lines in (3.30)

The variance sensitivity measure presented in [15] can be used to relatively find a frequency configuration that produce the best-conditioned system. This relatively best-conditioned system may not need regularization. Therefore, in some situations we can improve the conditioning of the system without the unwanted biasing of regularization.

The variance sensitivity measure quantifies the departure from a uniform distribution of DTFT locations, which in turn is related to the condition number. The variance of distances between adjacent DTFT locations in the 1-D unwrapped problem in (3.75) correlates well with the condition number and has a complexity of $O(N)$ as stated in [15].

For large problems an exhaustive search of frequency configurations that lead to well-conditioned systems is impractical. We use the method of simulated annealing explained in Section 3.6.2 with the variance measure as its cost function to find a near optimal set of DTFT locations, the estimate of the global minimum embedded in many local minima.

In the computed tomography (CT) case, the DTFT locations are selected at the intersections of the diagonal lines (3.30) and the radial projection lines. The angles and radial sampling are varied in the simulated annealing process to produce an near-optimal frequency configuration with the lowest variance measure.

3.5.1 Simulated Annealing

The variance sensitivity measure can be used to relatively find frequency configurations that produce well-conditioned systems. For large problems an exhaustive search is impractical. We use the method of simulated annealing with the variance measure as its cost function to find a set of frequency locations with a variance measure close to the global minimum. In [27], a cost function is evaluated from an initial discrete set of parameters. Then a candidate parameter set is created by changing a few of the parameters and the cost function is recomputed. If the cost decreases then the candidate set is accepted as the new optimal set. If the cost increases then the candidate set is still accepted based on the Boltzmann probability distribution,

$$P(\Delta E) = \exp(-\Delta E/k_B T), \tag{3.79}$$

where ΔE is the change in cost, k_B is the Boltzmann constant, and T is the temperature. For our purposes we treat $k_B T = \hat{T}$ as one variable. Several candidate sets are proposed at the current temperature until thermal equilibrium is reached. At that point then the temperature is lowered again until an acceptable cost is reached.

3.5.2 Fixed Configurations with Additional Samples

In the case where the frequency configuration is fixed and the DTFT samples have already been acquired, we would like to improve the conditioning of the problem by adding a few additional DTFT samples. We must determine the locations of these extra DTFT samples that reduce the variance measure, thus reducing the condition number of the system matrix. When placing each additional DTFT sample we can quickly recompute the variance measure to determine which additional location reduces the variance measure. Typically the best locations to reduce the variance sensitivity measure will be inside large gaps.

3.5.3 Perfect Conditioning

So far we have assumed the locations of the frequency samples are constrained to both (a) the geometry of the physical system such as radial lines in CT or a spiral trajectory in certain cases of MRI, and (b) the unwrapping rules such as a regular rectangular grid of the Good-Thomas FFT or the diagonal lines of the Kronecker substitution methods. The geometry of the sampling is then adjusted so that it intersects with the wrapped grid or lines. In this section, we explore the cases where each location in a set of arbitrary 2D DTFT samples can be chosen independently from each other. Arbitrary frequency selection over the 2-D freq domain is well suited for MRSI and 3D MRI.

An overview of the problem is to reconstruct an $M \times M$ image from frequency data with locations on a restricted diagonal lines uniformly distributed in 1-D. A 1-D unwrapping procedure determines the diagonal lines. The frequency locations are determined so that the 1-D case is perfectly conditioned, meaning the condition number equals exactly 1. Then the frequency values are mapped to 1-D and solved very quickly since the system matrix is now orthogonal and the matrix-vector multiplication is computed at FFT speed. Next the 1-D solution, which may have bands of zeros removed, is rewrapped into 2-D. If perfectly conditioned then there is no noise amplification.

The idea here is that we enforce a 1-D frequency location set that is uniformly spaced so that the condition number is always 1. In such cases, the variance sensitivity measure will be zero. This requires that we use enough frequency samples to ensure uniqueness. Otherwise we may reduce the solution support by removing any bands of zeros from the 1-D solution support. Empirical evidence shows that matrix is still well conditioned with bands of zeros removed. If we can ensure a condition number of 1 then there is no need for a sensitivity measure. However, additional frequencies cannot be added to the existing set. Instead a new uniformly spaced set must be

used.

In [28], Gao and Reeves state that the minimum sampling density is the area of the region of support. In our method after unwrapping in 1D that is the area of the rectangular support. Typically this is a helically wrapped grid in the frequency domain, has a conditioning of 1, and is computed fast, but this is the same for a 2-D FFT. The benefit of our method is that you can add a few more frequency samples and the 2-D frequency sampling grid updates. Though since this method does not amplify noise due to a condition number of 1, over-determining can not improve already perfect conditioning. Reduction in error is from frequency samples used.

The unwrapping procedure used is the helical scan FFT described in [22] and earlier in Section 3.2.3.

A few observations and consequences of the perfectly conditioned image reconstruction using regular non-rectangular 2-D DTFT samples are:

- The 2-D rectangular grid is not the only system with condition number of 1.
- Given an $M \times M$ image support, this method allows any overdetermined number of frequencies, but on M diagonal lines at angle $\arctan(1/M)$.
- The image can be padded to $N \times N$ which results in flatter diagonal lines at angle $\arctan(1/N)$ and closer N diagonal lines at $1/N$ apart.
- Voronoi cells created from the frequency locations are uniform, with zero variance in their area.
- When applying the variance sensitivity measure on 1-D diagonal wrapped line with no restrictions, the uniform distribution minimizes the variance measure.
- MRSI or 3D MRI are ideal applications where there're no restrictions in k-space but each sample is expensive, time-wise, to acquire.
- The three advantages are:

    1. No noise amplification

2. Fast 1-D FFT computation

3. Arbitrary number of samples

This last point is makes the perfectly conditioned method useful since normally, a 2-D DFT grid on $N \times N$ is used on a $M \times M$ image. To get an to an overdetermined system you need to increase DFT size from $N \times N$ to $(N+1) \times (N+1)$ which requires $2N+1$ extra samples (for example if $N = 32$ then $2N+1 = 65$). Then we can solve for inverse $(N+1) \times (N+1)$ inverse 2-D FFT, instead of the general method of solving normal equations using the conjugate gradient method for example.

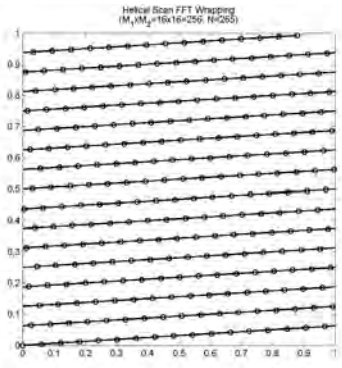
However, in our method we can increase the number of total points by an *arbitrary number* of extra points on a $N \times N$ grid and use the inverse 1-D FFT. If we know it is padded then we solve for just $(N+1) \times M - (N+1-M)$ samples. Though the result of uniformly distributed samples in 1-D is trivial, it becomes less so when wrapped to 2-D in a regular yet non-rectangular pattern.

A 16×16 example is shown in Figure 3.6, where the image is not padded and data is overdetermined by a factor of 1.037 with no noise. The theoretical system matrix is then of size 265×256 with a condition number of 1. The Voronoi cell patterns in Fig. 3.6(c) are hexagons with an over-determining factor of 1+18/512=1.037, in Fig. 3.6(d) are rhombuses with a factor of 1, in Fig. 3.6(e) are squares with a factor of 1+1/256, in Fig. 3.6(f) are snake-skin shape with a factor of 1+4/256.

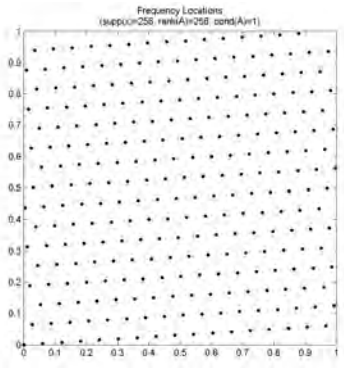
The hexagonal sampling pattern in Fig. 3.6(c) is of particular interest for the reasons that it is more efficient in sampling circularly band-limited signals [29] over rectangular patterns. In fast MRI applications such as MRI angiography and cardiology, a single-short hexagonal trajectory is easier on gradient field generators than a square spiral Fourier transform imaging while not requiring gridding methodologies as in the case for ordinary spiral imaging. In terms of computing the inverse Fourier transform of hexagonal Fourier samples, in [30] it's stated that many other hexagonal FFT algorithms are either computationally expensive, require twiddle factors,

must be interpolated to a rectangular grid in the end, or require insertion of zeros to fit to a 2D grid at the cost of aliasing. The method of helical scan FFT 2-D to 1-D unwrapping of a perfectly conditioned uniform sampling pattern followed by the standard inverse 1-D FFT lacks many of these difficulties.

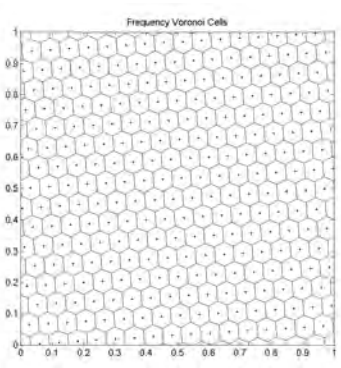
The small example shows a perfect reconstruction with 9.8×10^{-13} RMSE in 1D in Figure 3.7. The wrapped 2-D image not shown is of a cropped simulated cross-sectional head.



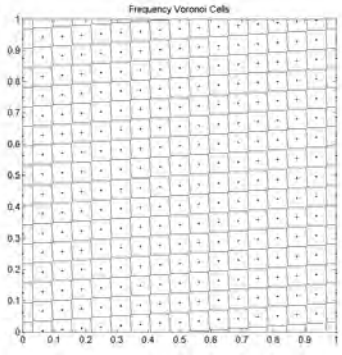
(a) Helical scan FFT wrapping



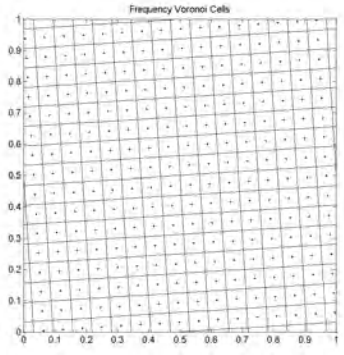
(b) Perfectly conditioned frequency locations



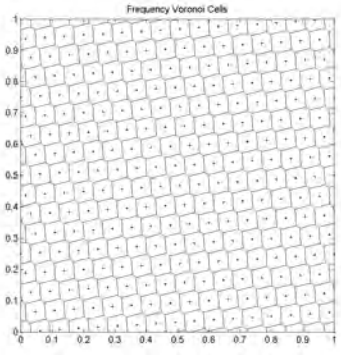
(c) Hexagonal Voronoi diagram of frequency locations



(d) Rhombus Voronoi diagram of frequency locations

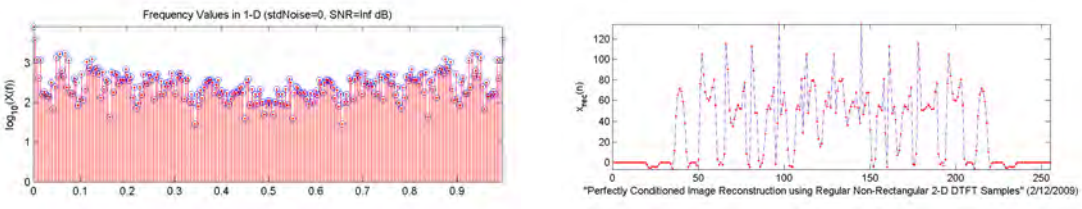


(e) Square Voronoi diagram of frequency locations



(f) "Snake-skin" Voronoi diagram of frequency locations

Figure 3.6: Perfectly conditioned 1-D frequency location wrapped to 2-D.



(a) Unwrapped 1-D DTFT values (b) Unwrapped 1-D reconstructed solution

Figure 3.7: Perfectly conditioned 1-D data and solution.

3.6 Results

3.6.1 Good-Thomas FFT: Simulation Results

The image to reconstruct has slanted support shown in Fig. 3.8a with $M_1 \times M_2 = 64 \times 64 = 4096$ real image values. We use a slanted image support because it leads to a shorter 1-D support length than one for a rectangular image support due to the diagonal unwrapping of the Good-Thomas FFT shown in the example in Fig. 3.9a. The data values are computed using the 2-D DTFT equation and are selected from a computed tomography (CT) pattern consisting of 256 radial slices with 256 points per slice shown in Fig. 3.8b. After choosing a large $N_1 = 6M$, these polar gridded 2-D DTFT locations are fitted to an $N_1 \times N_2 = 384 \times 385 = 147840$ rectangular grid of 2-D DFT locations using linear interpolation as we have done. This requires that the reconstructed 2-D image is zeropadded to 384×385. The example in Fig. 3.9a shows how this is done. No noise was explicitly added to the data since using interpolation to fit samples at $\{\omega_{1,k}, \omega_{2,k}\}$ already contributes to the noise. The settings for the limited angle tomography (LAT) pattern are the same except for where the slices are restricted to $[-\frac{\pi}{3}, \frac{\pi}{3}]$ as shown in Fig. 3.8c.

In [8], a slanted image has an unwrapped 1-D support length of at most $M_1 N_2$ as seen in the example in Fig. 3.9c. The 2-D zeropadded image is unwrapped to produce a 1-D signal with a support length of $M_1 N_2 = 24640$, which is the same number of DTFT values required for a unique reconstruction of 1-D signal stored in

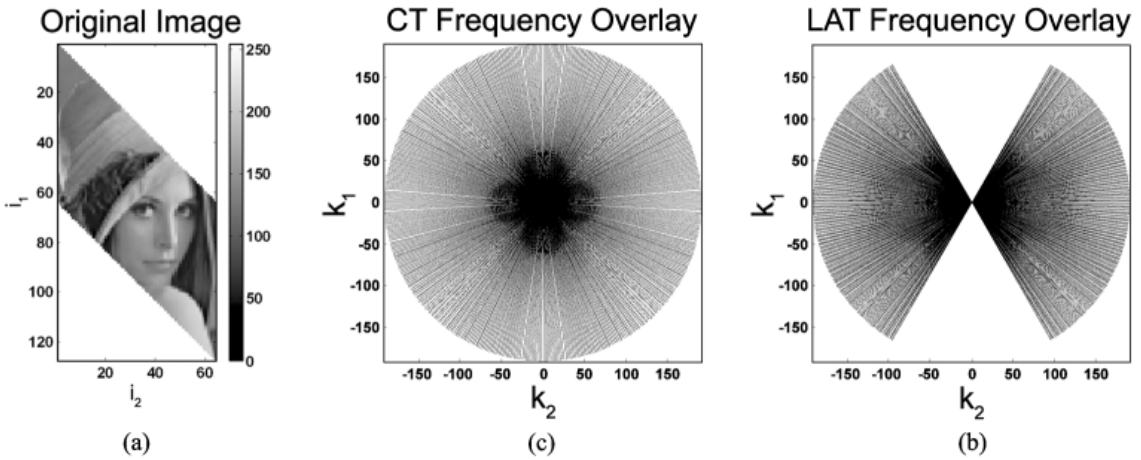


Figure 3.8: Original slanted-support image and computed tomography and limited angle tomography frequency configurations.

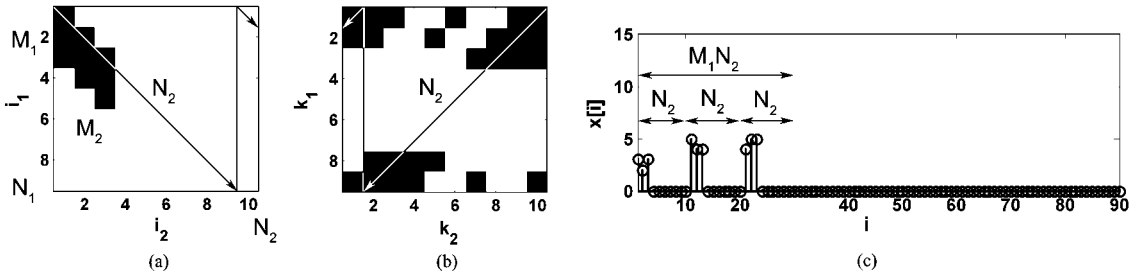


Figure 3.9: 2-D image slanted support and unwrapped 1-D support examples.

solution vector x. We chose a just-determined $M_1N_2 = 24640$ number of 2-D DTFT samples selected from either the CT or LAT pattern. These 2-D DTFT samples make up the data vector b. Because we know the image is real, the 2-D DTFT samples are selected in complex conjugate pairs. Though the method also applies to complex-valued images.

The system matrix A in (2.6) is constructed from 24640 selected unwrapped 1-D frequency locations out of 78674 available samples for the CT case and out of 67614 available samples for the LAT case. Simulated annealing with the variance sensitivity measure as the objective function is used to relatively find frequency configurations that produce well-conditioned systems. The annealing schedule we use is as follows:

$\hat{T}_{initial} = 1 \times 10^{-10}$, $\hat{T}_{final} = 1 \times 10^{-20}$, $\hat{T}_{decay} = 0.9$, $N_{trials} = 100$ candidate trials per temperature, and $N_{swap} = 1$ number of frequency samples swapped per trial.

With the use of the variance sensitivity measure, we are able to construct the best-conditioned system out of the many possible systems. This relatively best conditioned system may not need regularization compared to the other choices. Therefore, in some situations we have improved the conditioning of a system without the unwanted biasing side effect of regularization. For cases where all of the possible frequency configurations produce ill-conditioned systems, regularization must be used yet the variance measure is still useful in identifying the least ill-conditioned system. However for these simulations regularization is not used.

The image is reconstructed iteratively by solving the normal equation (2.8) using the preconditioned conjugate gradient method (2.31) with a circulant preconditioner (2.39).

3.6.1.1 Variable Frequency Configurations

For the case where we have the freedom to choose the 24640 2-D DTFT samples from the CT pattern and LAT pattern, we first select a random set of frequency samples and designate them as the "High Variance" configurations. Then we use the variance measure in simulated annealing to arrive at the "Low Variance" configurations. The "High Variance" configurations are solved with a maximum of 35 PCG iterations with a relative residual of 2.63×10^{-3} for the CT case and 3.66×10^{-3} for the LAT case. The "Low Variance" systems are then solved to reach the same relative residual value in hopefully fewer PCG iterations for each case.

The "High Variance" and "Low Variance" frequency configurations for both the CT and LAT cases are shown in Fig. 3.10. Notice the "Low Variance" configurations clump less near the origin and the samples are more uniformly distributed. The mean squared errors (MSE) of each reconstructed image in Fig. 3.11 are on the same order

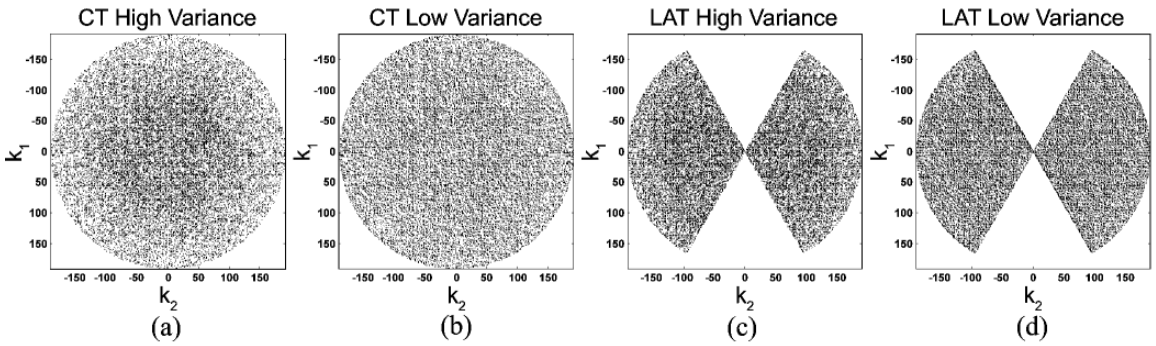


Figure 3.10: Variable CT and LAT frequency configurations.

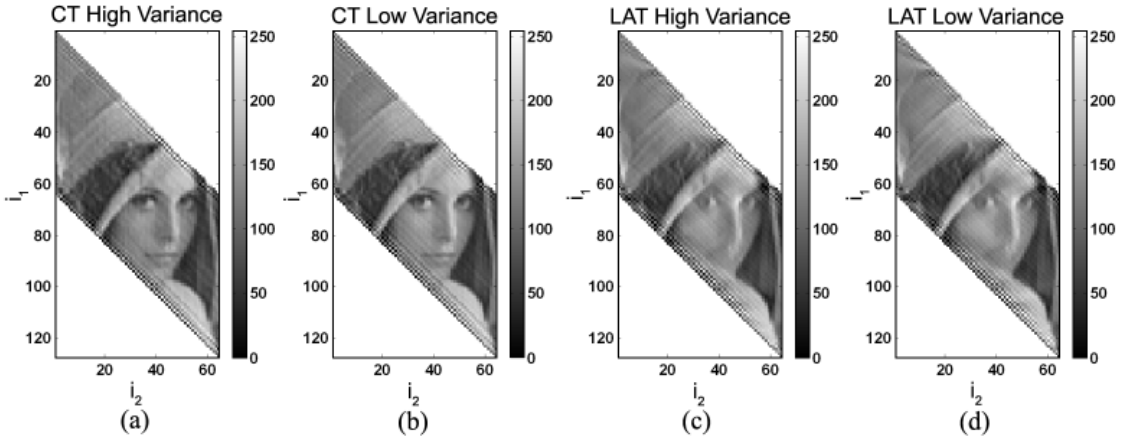


Figure 3.11: Reconstructed images from Good-Thomas unwrapped variable configurations.

of magnitude which is expected since they were solved for the same relative residual. However the number of iterations for the "Low Variance" systems are clearly less as seen in Table 3.1. Fig. 3.12 shows the relative residuals of both configurations for each iteration for each case. The relative residual of the "Low Variance" configurations decrease faster than the relative residual of the "High Variance" configurations.

3.6.1.2 Fixed Configurations with Restricted Additional Samples

The reconstruction uses the "High Variance" frequency configuration in Section 3.6.1.1 as the fixed configuration but with 100 additional DTFT samples in complex conjugate pairs. The total number of DTFT samples used then becomes 24740.

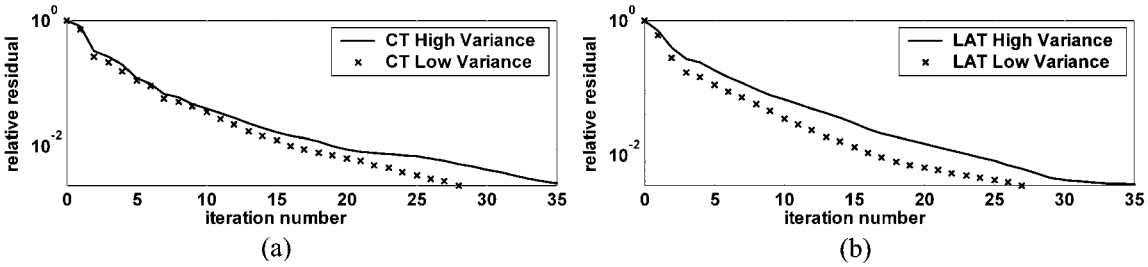


Figure 3.12: Relative residuals of variable configuration reconstructions per PCG iteration.

Table 3.1: Good-Thomas Reconstruction Data

Frequency Configuration	Variance Measure	Mean Squared Error	PCG Iterations
CT High Variance	3.36×10^{-7}	35.1	35
CT Low Variance	2.80×10^{-7}	36.5	28
LAT High Variance	8.33×10^{-7}	36.6	35
LAT Low Variance	7.84×10^{-7}	35.6	27
Add CT Random	3.35×10^{-7}	35.1	35
Add CT Gap	3.15×10^{-7}	32.5	34
Add Any Random	3.24×10^{-7}	35.1	35
Add Any Gap	2.57×10^{-7}	33.0	33

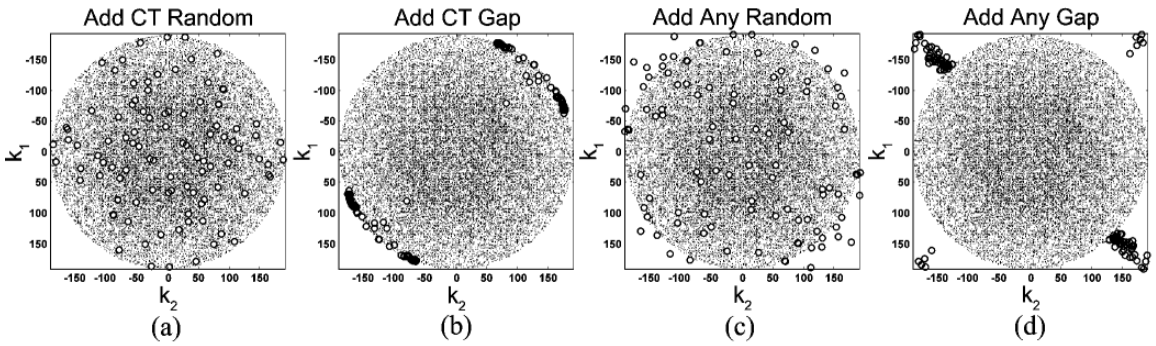


Figure 3.13: Fixed frequency configurations with additional samples.

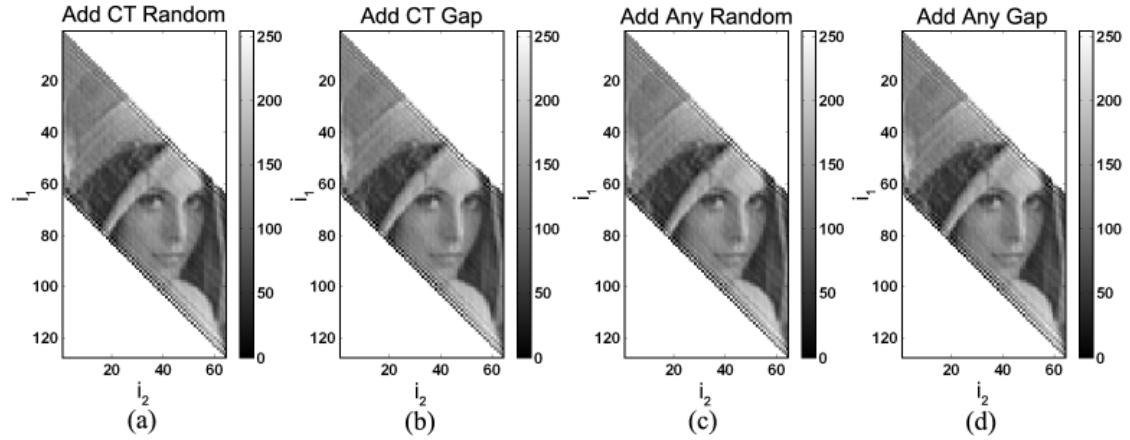


Figure 3.14: Reconstructed images from Good-Thomas unwrapped fixed configurations.

In the first scenario, each location of the additional DTFT sample pairs is selected from the CT pattern at random, marked as circles in Fig. 3.13a. This "Add CT Random" configuration has a variance measure of 3.35×10^{-7} radians, which is about the same as the variance measure of the "CT High Variance" configuration since large gaps still exist in the configuration. The MSE of the reconstructed image in Fig. 3.14a is not improved despite that we are reconstructing with more data values. The image is then reconstructed using PCG with 35 maximum iterations with a relative residual of 2.63×10^{-3}.

In the second scenario, each location of the additional DTFT sample pairs is selected from the CT pattern at locations within the largest gaps in the anti-diagonally

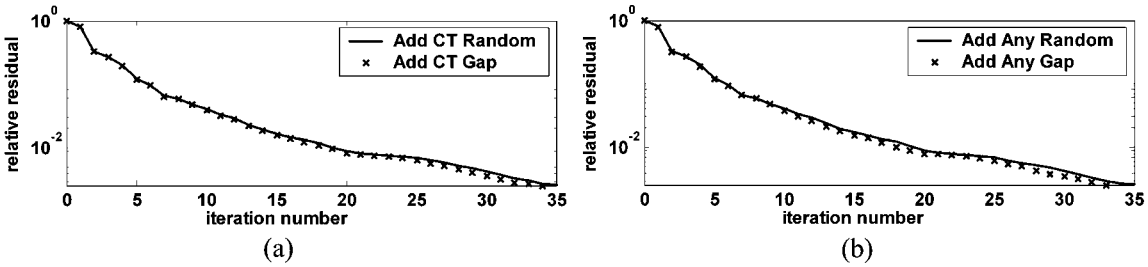


Figure 3.15: Relative residuals of fixed configuration reconstructions per PCG iteration.

unwrapped 1-D CT pattern. This is consistent with the insight developed in Section 3.4.5 that the absence of large gaps leads to smaller a variance measure value. This "Add CT Gap" configuration in Fig. 3.13b has a variance measure of 3.15×10^{-7} radians, which is lower than the "Add CT Random" configuration. Solving for the same relative residual tolerance, the PCG iterations and MSE of the reconstructed image in Fig. 3.14b are lower than that of the "Add CT Random" image.

Fig. 3.15a shows the PCG relative residuals per iteration of the two configurations with additional DTFT samples selected from the CT pattern. The relative residual of the "Add CT Gap" configuration decreases slightly faster than that of the "Add CT Random" configuration.

3.6.1.3 Fixed Configurations with Any Additional Samples

If we are able to select the additional DTFT samples from any location on the $N_1 \times N_2$ rectangular grid and not be restricted by the CT pattern, we can form a configuration with a much lower variance measure by filling in the larger gaps. In the first scenario, each location of the additional 100 DTFT samples in pair is selected from any location at random. This "Add Any Random" configuration in Fig. 3.13c has a variance measure of 3.24×10^{-7} radians and the reconstructed image in Fig. 3.14c has a MSE similar to that of the "CT High Variance" and "Add CT Random" configurations.

In the second scenario, each location of the additional DTFT sample pairs is selected from any location within the largest gaps in the unwrapped 1-D "CT High Variance" configuration. This "Add Any Gap" configuration in Fig. 3.13d has a variance measure of 2.57×10^{-7} radians, which is lower than the "Add Any Random" configuration even though the "Add Any Random" configuration appears to uniformly fill in 2-D gaps more than the "Add Any Gap" configuration does. The reason is not so obvious in 2-D but clearer in 1-D. Since the unwrapping of these frequency locations using the Good-Thomas FFT occurs anti-diagonally, the centers of the largest 1-D gaps line up along the diagonal of the 2-D frequency grid. Therefore uniformly filling in gaps in 2-D does not necessarily correspond to filling in most of the 1-D gaps. The MSE of the reconstructed image in Fig. 3.13d are lower than that of the "Added Any Random" configuration.

Fig. 3.15b shows the PCG relative residuals per iteration of the two configurations with any additional DTFT samples on the frequency grid. The relative residual of the "Add Any Gap" configuration decreases faster than that of the "Add Any Random" configuration.

3.6.2 Rotated Image Support: Simulation Results

The $M \times M = 127 \times 127$ image to reconstruct is shown in Fig. 3.16a and has a 45° rotated support as well as being rotated itself in Fig. 3.16b. This image is also zeropadded to $N \times N = (2M-1) \times (2M-1) = 253 \times 253$ to increase the number of diagonal lines in the frequency domain thus increasing the density of DTFT samples from which to choose. When the image is unwrapped diagonally into 1-D much of the head and tail of the signal are zeros, and therefore the signal support length is reduced to $N \times M = (2M-1) \times M = 32131$. Only the supported signal is shown rewrapped into 2-D in Fig. 3.16c.

In the frequency domain, $N-1 = 252$ equally-spaced diagonal lines fill a $[-\pi, \pi) \times$

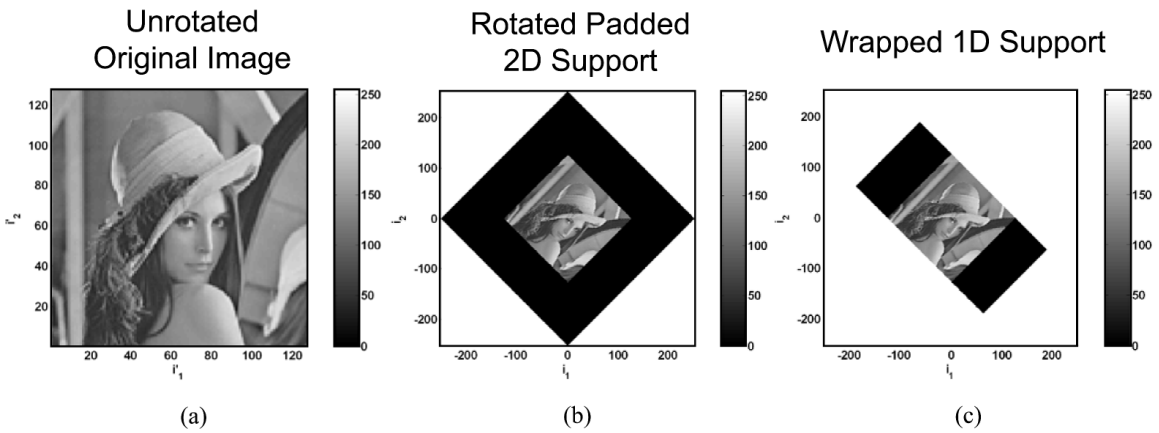


Figure 3.16: Original image with unrotated, rotated padded 2D, and rotated wrapped 1D support.

$[-\pi, \pi)$ area. These lattice lines intersect with radial projection lines used in CT. Because the DTFT samples are restricted to the diagonal lines, if either of the angles or the radii of the CT projection lines are independently chosen then the other parameters are automatically determined. For example in Fig. 3.17a, the uniform angles determine the radii at the intersections with the lattice lines, whereas in Fig. 3.17b, the uniform radii determine the angles at the intersections. Simulated annealing is used to adjust either the angles or the radii to find an optimal set of parameters with respect to minimizing the variance sensitivity measure. Once found, the optimal angles or radii can be fixed in CT systems. For example, 3rd generation CT scanners with rotate/rotate geometry can set their detectors to the optimal radial positions and then take projection data at variable angles.

There are four frequency configurations used in reconstructing the image: "CT Angle Uniform", "CT Angle Optimal", "CT Radius Uniform", "CT Radius Optimal". They differ in which parameter varies, "Angle" or "Radius", and how that parameter is arranged, "Uniform" or "Optimal". The "Optimal" modifier denotes the frequency configuration with the lowest variance measure value found from simulated annealing. The number of angles or projections and number of radii or radial samples is

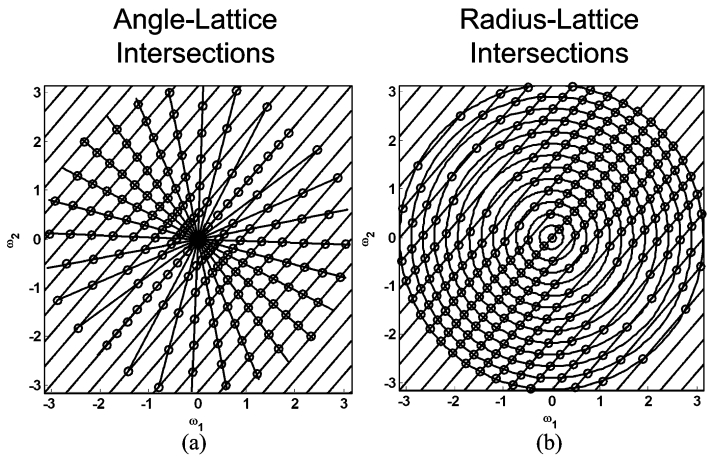


Figure 3.17: Lattice and radial projection intersections for rotated image support.

initially set to $N = 253$ each and adjusts its number and sometimes crops the highest frequencies to maintain the fixed number of DTFT samples. An over-determining number of $2 \times supportlength + 1 = 2NM + 1 = 64263$ DTFT samples are chosen in complex conjugate pairs to reconstruct only real images in this case and includes the single DC sample making the number odd. In the "Angle" configurations the angle which lines up with the middle lattice line is also always included and sampled at π/N over the full length.

After the DTFT locations are selected the DTFT values are computed using the 2-D DTFT equation (2.1). No noise is added to the DTFT values. The system of linear equations is set up with 64263 data values to solve for 33153 solution values. It is solved iteratively by PCG.

3.6.2.1 Variable Angle Configurations

The "CT Angle Uniform" configuration has DTFT samples taken from intersections between projection lines with angles uniformly spaced and the lattice shown in Fig. 3.18a. The "CT Angle Optimal" configuration shown in Fig. 3.18b uses simulated annealing on the former configuration and an initial configuration. The annealing schedule we use is as follows: $\hat{T}_{initial} = 10$, $\hat{T}_{final} = 1 \times 10^{-5}$, $\hat{T}_{decay} = 0.8$,

Table 3.2: Rotated Support Reconstruction Data

Frequency Configuration	Variance Measure	Root Mean Squared Error	PCG Iterations
CT Angle Uniform	61.7	2.23	185
CT Angle Optimal	30.3	0.87	122
CT Radial Uniform	52.4	16.23	678
CT Radial Optimal	40.4	9.71	297

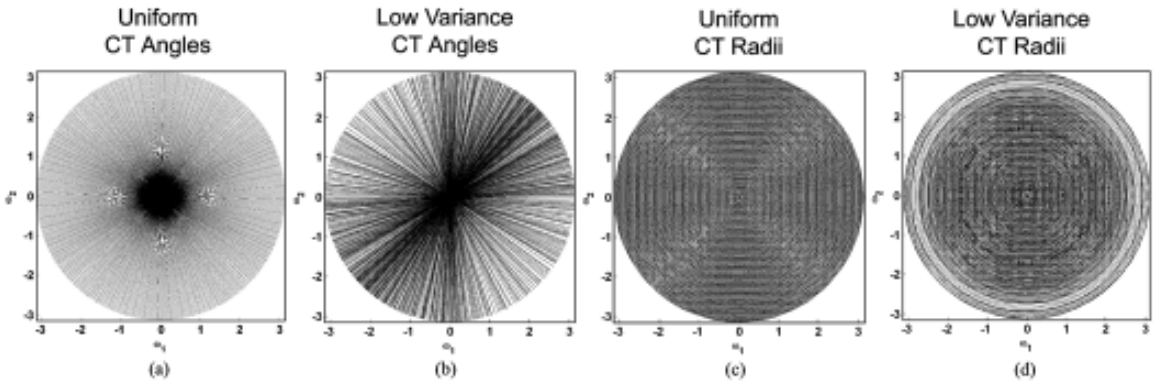


Figure 3.18: Computed tomography frequency configurations for rotated support.

$N_{trials} = 100$ candidate trials per temperature, and $N_{swap} = 10$ number of frequency samples swapped initially per trial. The N_{swap} decrements by 1 until it reaches 1 when no steps were taken in the current temperature. The PCG relative residual tolerance is set to 1×10^{-6}.

The "CT Angle Optimal" configuration has a variance measure of 30.3 which is about half that of "CT Angle Uniform" at 61.7 shown in Table 3.2. Likewise root mean squared error (RMSE) of the reconstructed images in Fig. 3.19a-b is lower for that of "CT Angle Optimal" with 0.87 compared to 2.23. More importantly "CT Angle Optimal" took 122 instead of 185 PCG iterations to reconstruct the image. The PCG relative residuals are shown in Fig. 3.20a.

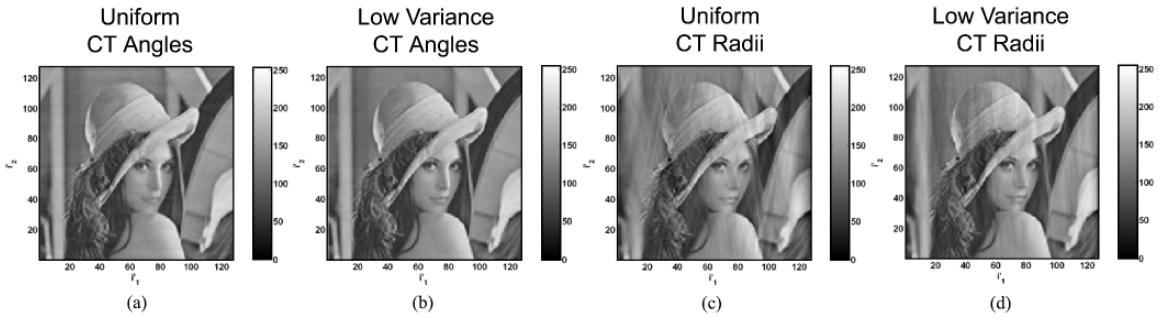


Figure 3.19: Unrotated reconstructed images from CT configurations.

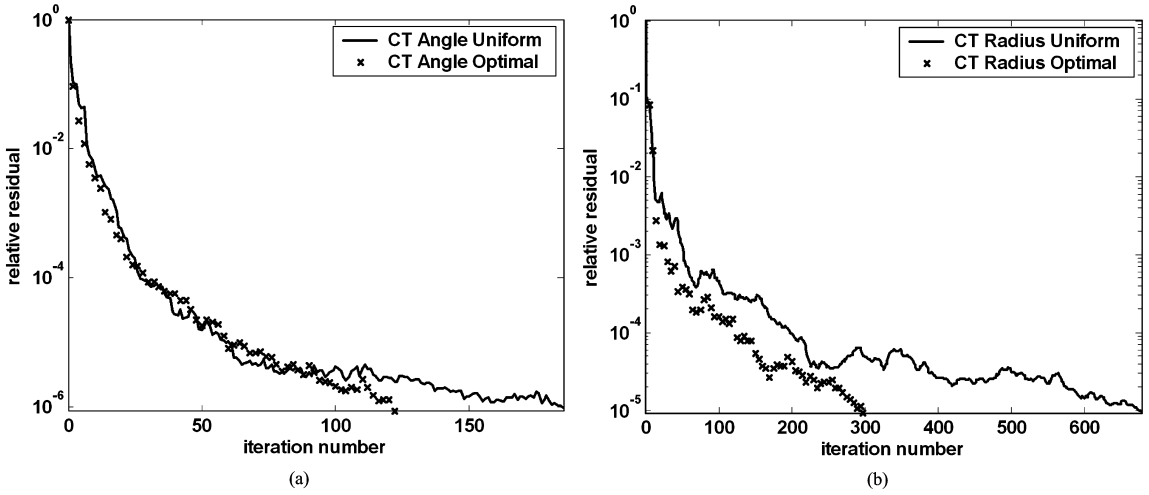


Figure 3.20: Relative residuals of CT configurations for rotated support per PCG iteration.

3.6.2.2 Variable Radius Configurations

The "CT Radius Uniform" configuration has DTFT samples taken from intersections between projection lines with radii uniformly spaced and the lattice shown in Fig. 3.18c. The "CT Radius Optimal" configuration shown in Fig. 3.18d uses simulated annealing on the former configuration and an initial configuration. The annealing schedule we use is as follows: $\hat{T}_{initial} = 1$, $\hat{T}_{final} = 1 \times 10^{-6}$, $\hat{T}_{decay} = 0.8$, $N_{trials} = 100$ candidate trials per temperature, and $N_{swap} = 5$ number of frequency samples swapped initially per trial. The N_{swap} decrements by 1 until it reaches 1 when no steps were taken in the current temperature. The PCG relative residual tolerance is set to $1 \times 10^{-}5$.

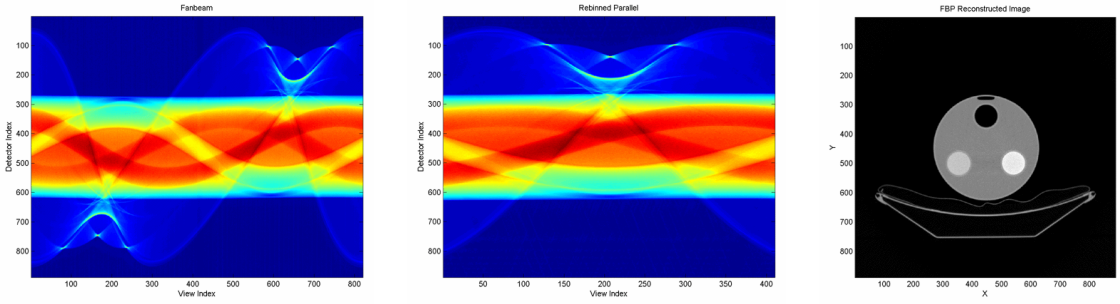
The "CT Radius Optimal" configuration has a variance measure of 40.4 which is just less than that of "CT Radius Uniform" at 52.4 shown in Table 3.2. The RMSE of the reconstructed images in Fig. 3.19c-d is lower for that of "CT Radius Optimal" with 9.71 instead of 16.23. Again more importantly "CT Radius Optimal" took 297 instead of 678 PCG iterations to reconstruct the image. The PCG relative residuals are shown in Fig. 3.20b.

3.6.3 Rotated Image Support: Actual CT Results

3.6.3.1 Data

Results implementing the 45° rotated support method using M as odd and on actual CT data, provided by Adam M. Alessio of the University of Washington, are presented. The Discrete-Time Fourier Transform (DTFT) locations are the intersections of radial projection lines and frequency supported diagonal lines. The DTFT values are computed from the 1-D DTFT of each projection rebinned into parallel beams from fanbeam data originally consisting of 820 views per 360° rotation and 888 detectors, shown in Fig. 3.21(a). The location from the distance from the source

to the detector is 949.075mm and the source to iso-center is 541mm. The detector spacing is 1.0239mm and the location of the iso-center is 444.75 in detector units which has a quarter-pixel shift to effectively double the radial resolution when combining views that are 180° apart. The rebinned parallel beam projections, shown in Fig. 3.21(b), have an equivalent 889 detectors spaced 0.9622 pixels with pixel size 0.5837mm, consisting of 410 views per 180° rotation, and is interpolated from the fanbeam projections using linear interpolation. A reference filtered back-projection (FBP) reconstructed image of size 889×889, same as the number of detector bins, of a physical phantom using the whole parallel beam sinogram is shown in Fig. 3.21(c).



(a) Fanbeam sinogram

(b) Rebinned parallel beam sinogram

(c) FBP reconstructed image from full sinogram

Figure 3.21: Sinograms and reference reconstructed image.

3.6.3.2 Small Reconstruction

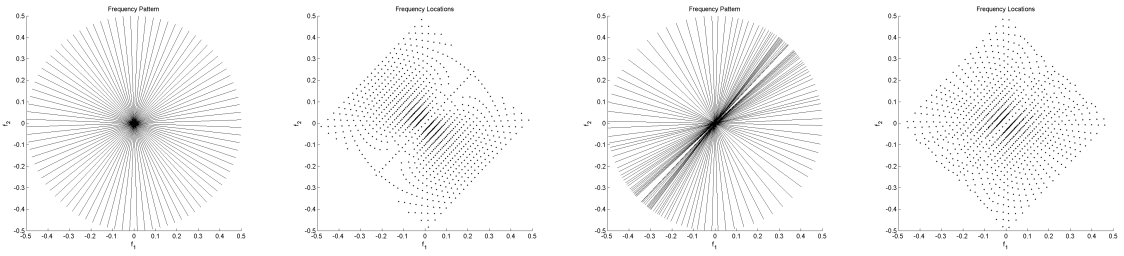
A small image $x(i_1, i_2)$ of size $M \times M = 31 \times 31$ with no zeropadding of the image support is reconstructed. The size of the solution vector (of the reconstructed image) is $M \times M = 961$ and the overdetermining factor is set to 1 initially. The 45° rotated image method which unwraps the 2-D DTFT to a 1-D DTFT using Kronecker substitution is solved using the preconditioned conjugate gradient (PCG) method with a stopping tolerance of 1×10^{-6} and a maximum of 50 iterations.

The data vector of the uniform-angle PCG reconstruction consists of 973 DTFT values, shown in Fig. 3.22(b), from the intersection of band-limited diagonally-

wrapped 1-D DTFT constraint lines and uniform-angle 1-D DTFT of 44 CT projection slices, shown in Fig. 3.22(a). The variance measure of the unwrapped 1-D DTFT locations is 27.7.

The data vector of the variable-angle PCG reconstruction consists of 969 DTFT values, shown in Fig. 3.22(d), from the intersection of band-limited diagonally-wrapped 1-D DTFT constraint lines and simulated annealing optimized variable-angle 1-D DTFT of 68 CT projection slices, shown in Fig. 3.22(c). The variance measure of the unwrapped 1-D DTFT locations is lower at 11.2.

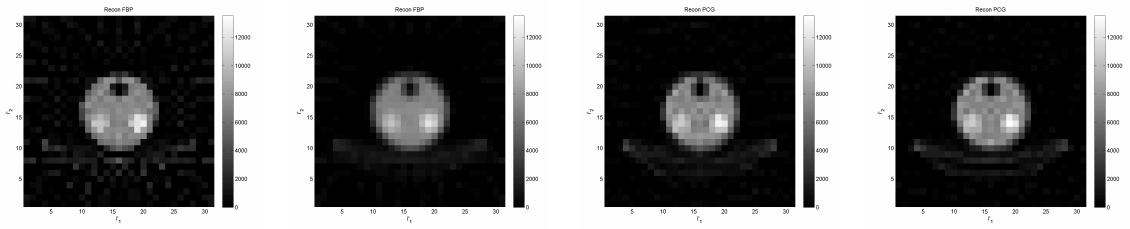
The PCG reconstructions are compared to two FBP reconstructions: one with projections filtered with a non-windowed ramp filter, and another with projections filtered with a Hann-windowed ramp filter. In both cases the data vector consists of 961 unique 2-D DTFT values from 16 uniformly spaced projections with 61 bins each.



(a) Uniform-angle projections (b) Uniform-angle intersecting DTFT locations (c) Variable-angle projections (d) Variable-angle intersecting DTFT locations

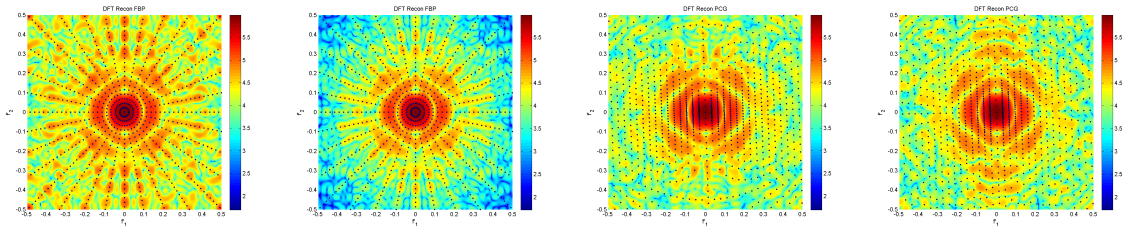
Figure 3.22: Frequency configurations for small (M=31) rotated-support reconstruction.

Fig. 3.23 show the 31×31 reconstructed images using a non-windowed FBP, windowed FBP, uniform-angle PCG, and variable-angle PCG. Fig. 3.24 shows the log magnitude of the 2-D DFTs of the reconstructed images with the locations of the unrotated 2-D DTFT locations overlayed. It is evident that the near-optimal variable-angle PCG reconstruction has a more uniform distribution of 2-D DTFT samples than the other methods.



(a) Non-windowed FBP image (b) Hann-windowed FBP image (c) Uniform-angle PCG image (d) Variable-angle PCG image

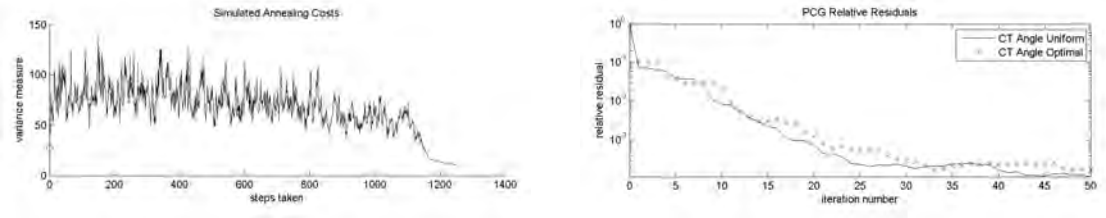
Figure 3.23: Unrotated reconstructed images for small (M=31) rotated-support reconstruction.



(a) DFT of non-windowed FBP image (b) DFT of Hann-windowed FBP image (c) DFT of uniform-angle PCG image (d) DFT of variable-angle PCG image

Figure 3.24: Log magnitudes of 2-D DFT of reconstructed images with unrotated DTFT locations for small (M=31) rotated-support reconstruction.

Fig. 3.25(a) shows the variance measure as the cost of the simulated annealing procedure which led to the near-optimal variable projections angles. Fig. 3.25(b) shows the PCG relative residuals of both the uniform-angle and variable-angle methods, which are similar.



(a) Variance measure vs. simulated annealing step

(b) PCG relative residuals vs. iteration

Figure 3.25: Simulated annealing optimization and PCG relative residual comparison for small (M=31) rotated-support reconstructions.

3.6.3.3 Large Reconstruction

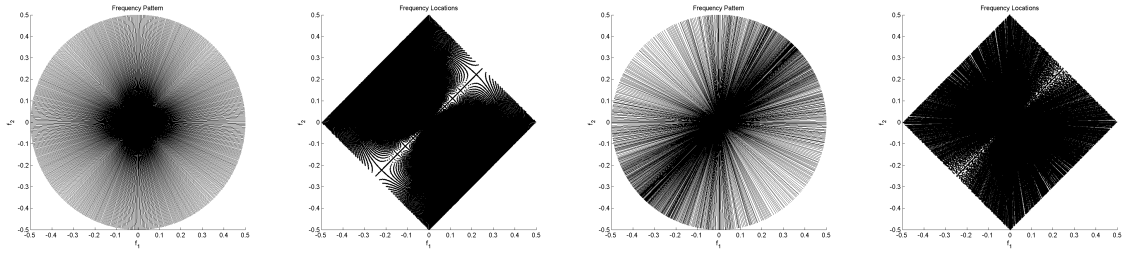
Now a larger image $x(i_1, i_2)$ of size $M \times M = 255 \times 255$ with no zeropadding of the image support is reconstructed. The size of the solution vector (of the reconstructed image) is $M \times M = 65,025$ and the overdetermining factor is set to 1 initially. The PCG method used with a stopping tolerance of 1×10^{-6} and a maximum of 1000 iterations.

The data vector of the uniform-angle PCG reconstruction consists of 65,137 DTFT values, shown in Fig. 3.26(b), from the intersection of band-limited diagonally-wrapped 1-D DTFT constraint lines and uniform-angle 1-D DTFT of 355 CT projection slices, shown in Fig. 3.26(a). The variance measure of the unwrapped 1-D DTFT locations is 57.8.

The data vector of the variable-angle PCG reconstruction consists of 65,095 DTFT values, shown in Fig. 3.26(d), from the intersection of band-limited diagonally-wrapped 1-D DTFT constraint lines and simulated annealing optimized variable-angle

1-D DTFT of 527 CT projection slices, shown in Fig. 3.26(c). The variance measure of the unwrapped 1-D DTFT locations is lower at 27.4.

The PCG reconstructions are compared to two FBP reconstructions: one with projections filtered with a non-windowed ramp filter, and another with projections filtered with a Hann-windowed ramp filter. In both cases the data vector consists of 65,025 unique 2-D DTFT values from 128 uniformly spaced projections with 509 bins each.

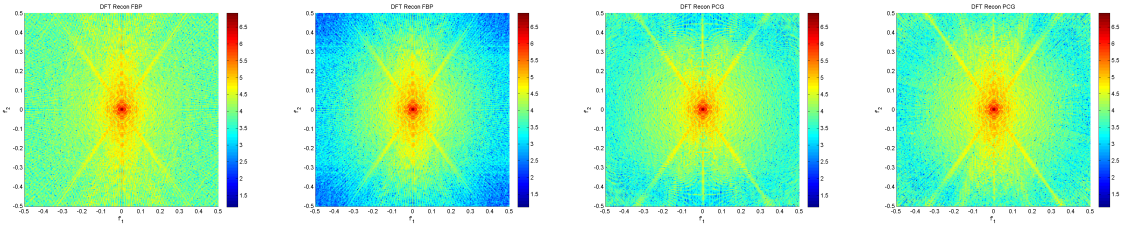


(a) Uniform-angle projections (b) Uniform-angle intersecting DTFT locations (c) Variable-angle projections (d) Variable-angle intersecting DTFT locations

Figure 3.26: Frequency configurations for large (M=255) rotated-support reconstruction.

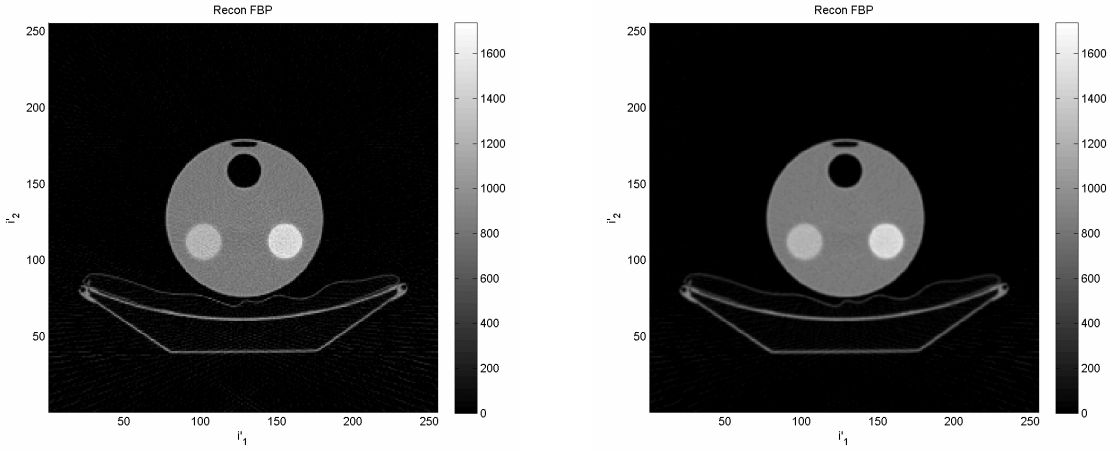
Fig. 3.28 show the 255×255 reconstructed images using a non-windowed FBP, windowed FBP, uniform-angle PCG, and variable-angle PCG. It is clear from the reconstructed images that the variable-angle reconstruction in Fig. 3.28(d) has sharper edge definition in the vertical direction than the uniform-angle reconstruction in Fig. 3.28(c). The PCG methods do exhibit some vertical waves most likely from sensitivity to noise in the data, which could be further remedied either with overdetermining with more frequency samples or with some mild regularization. The variable-angle PCG reconstruction exhibits no streaking artifacts of the non-windowed FBP reconstruction in Fig. 3.28(a) and has better edge-definition than the windowed FBP reconstruction in Fig. 3.28(b). Fig. 3.27 shows the log magnitude of the 2-D DFTs of the reconstructed images.

Fig. 3.29(a) shows the variance measure as the cost of the simulated annealing



(a) DFT of non-windowed FBP image (b) DFT of Hann-windowed FBP image (c) DFT of uniform-angle PCG image (d) DFT of variable-angle PCG image

Figure 3.27: Log magnitudes of 2-D DFT of reconstructed images with unrotated DTFT locations for (M=255) rotated-support large reconstruction.

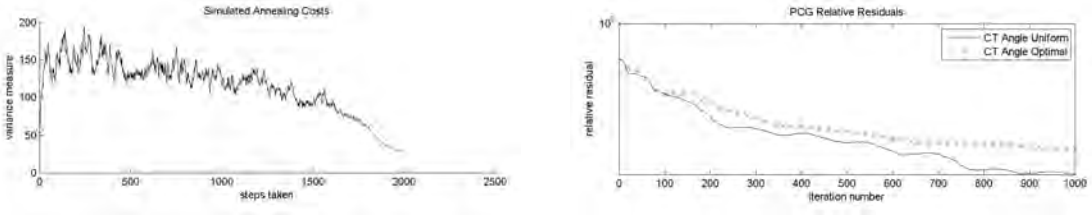


(a) Non-windowed FBP image (b) Hann-windowed FBP image

(c) Uniform-angle PCG image (d) Variable-angle PCG image

Figure 3.28: Reconstructed images for large (M=255) rotated-support reconstruction.

procedure which led to the near-optimal variable projections angles. Fig. 3.29(b) shows the PCG relative residuals of the uniform-angle and variable-angle methods. The variable-angle relative residual decreases slower per iteration even though the variable-angle reconstruction does show better vertical edge definition.



(a) Variance measure vs. simulated annealing step

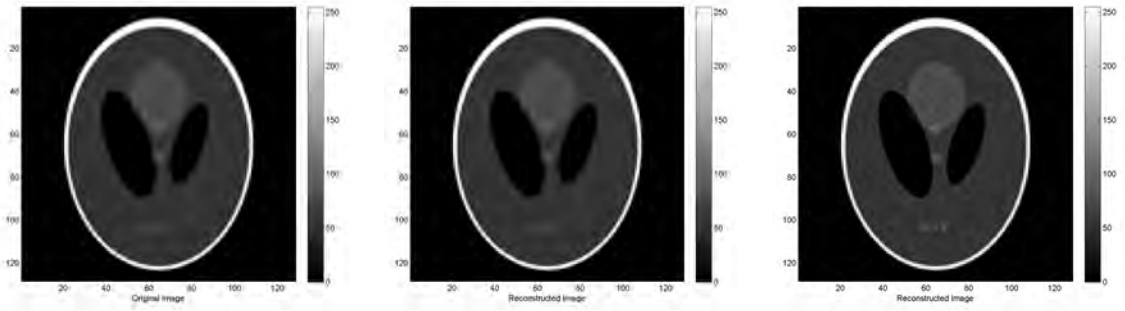
(b) PCG relative residuals vs. iteration

Figure 3.29: Simulated annealing optimization and PCG relative residual comparison for large (M=255) rotated-support reconstructions.

3.6.4 Perfect Conditioning: Simulation Results

The perfectly conditioned reconstruction method uses the helical scan FFT to unwrap the 2-D problem into a 1-D problem. The simulation results with two noise levels show how noise is not amplified.

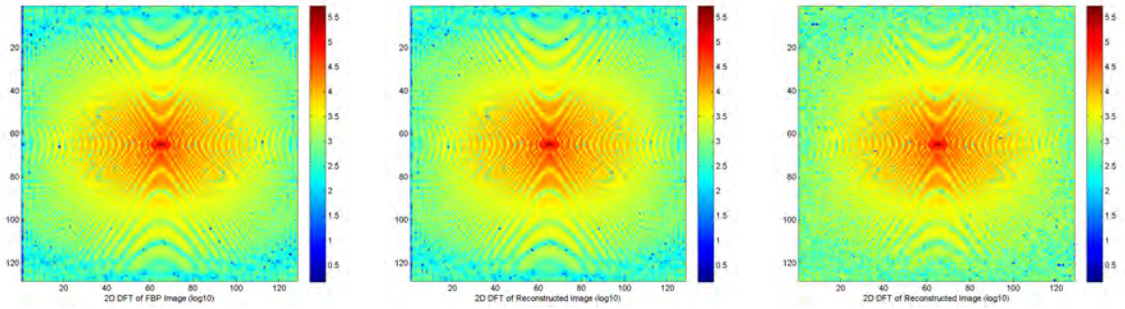
The simulation uses a Shepp-Logan phantom of size $M \times M = 128 \times 128$ no image padding but a overdetermining factor of 1.1 where the data vector size is then 18,022 and the solution vector size is 16,384. In Figure 3.30, the first case of low noise, the 2-D DTFT values are corrupted by additive white Gaussian noise with a standard deviation of 1×10^2 or an SNR of 30 dB. The reconstruction error is 0.5301 (RMSE). In the second case of high noise at 10 times the low noise case, the only difference is the data is corrupted with noise of 1×10^3 SD or an SNR of 10 dB. The reconstruction error is 5.301 (RMSE), which is exactly 10 times that of the low case case. This empirically upholds that the noise in the data is not amplified at all due to the implicit condition number of 1 of the system.



(a) Original image

(b) Noisy reconstruction (RMSE=0.5301)

(c) Noisier reconstruction (RMSE=5.301)



(d) 2-D DFT of original image (log10)

(e) 2-D DFT of noisy image ($\sigma_N = 1 \times 10^2$) (log10)

(f) 2-D DFT of noisier image ($\sigma_N = 1 \times 10^3$) (log10)

Figure 3.30: Perfectly conditioned noisy simulation reconstructions and their 2-D DFTs.

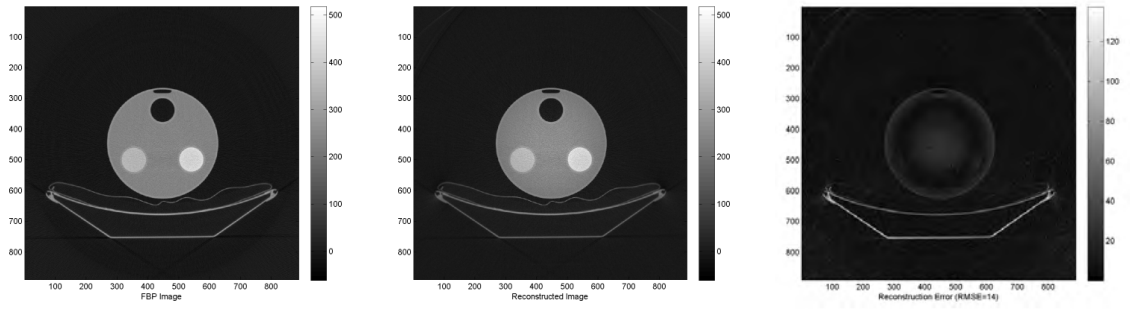
3.6.5 Perfect Conditioning: Actual CT Results

The perfectly conditioned method is performed on the same actual CT data to reconstruct an image of size $M \times M = 889 \times 889$ with an image padding factor of 1.9 to avoid image aliasing and a overdetermining factor of 1.7 where the data vector size is then 2,551,225 and the solution vector size is 790,321. The noise added to the frequency samples 1-D Fourier transformed in the radial direction from the sinogram data has a standard deviation of 1×10^4 or an SNR of 0.17 dB.

Since there is no true known image, for comparison purposes we reconstruct an reference image using filtered backprojection algorithm in Matlab called "iradon()" using linear interpolation from fanbeam to parallel beam and with a ramp or Ram-Lak filter. The sinogram data contains 364,490 data points with no noise added though. The actual CT data results compares the perfectly conditioned method with the FBP method, with regards to image quality and computation time.

In Figure 3.31, the perfectly conditioned reconstructed image can be seen with little artifacts. It differs from the FBP reconstruction by 13.76 RMSE in Figure 3.31(c). However whereas the FBP algorithm "iradon()" took 50 seconds to compute, the 1-D FFT of the perfectly conditioned method took only 1.5 seconds.

Figure 3.31(f) also shows the 2-D DFT of the difference between the images. Discrepancy in FBP and perfectly conditioned reconstruction is that the perfectly conditioned reconstruction method samples the low frequency region uniformly as well as the whole frequency plane but the FBP densely samples the low frequency region due to its polar coordinate sampling pattern. Relatively to FPB, the perfectly conditioned reconstruction method under-samples data in that region, relative to the FBP sampling. This is the explanation for the poorer reconstruction of the perfectly conditioned method.



(a) FBP image (b) Perfectly conditioned image (c) Difference image (RMSE=13.7)

(d) 2-D DFT of FBP image (log10) (e) 2-D DFT of perf. cond. image (log10) (f) 2-D DFT of difference image (log10)

Figure 3.31: Perfectly conditioned noisy simulation reconstructions and their 2-D DFTs.

CHAPTER IV

Non-Iterative Approach and Results

4.1 Introduction

The problem of reconstructing an image from irregular frequency samples is usually solved using an iterative algorithm, such as Projection Onto Convex Sets, or Conjugate Gradient applied to a linear system of equations with the image pixels as unknowns. However, these require many iterations and each iteration still requires three 2-D DFTs to implement convolution. Solving the normal equation, the condition number of $A^H A$ in (2.8) is the square of the condition number of A and the noise amplification will be enormous unless drastic regularization is used, even though the filtering effect of A^H on the data vector b in $A^H b$ may have reduce noise. Also the number of CG iterations increases roughly with the condition number even with the help of preconditioning.

We present a non-iterative reconstruction algorithm where computation speed gains are achieved from precomputing a filter which incorporates compact support assumptions on the solution image and requires the data to be fit on a DFT grid. The resulting solution can be used as an initialization for the conjugate gradient method in the previous iterative image reconstruction method.

4.1.1 Problem Statement

The goal is to reconstruct a possibly complex-valued $(M \times M)$ discrete image $x(i_1, i_2)$ from *some* of the values of its $(N \times N)$ 2-D DFT, where $N >> M$,

$$X(k_1, k_2) = \sum_{i_1=0}^{N-1} \sum_{i_2=0}^{N-1} x(i_1, i_2) e^{-j\frac{2\pi}{N^2}(i_1 k_1 + i_2 k_2)}. \tag{4.1}$$

We formulate the problem for square image support and DFT size; modification to the rectangular cases is trivial. We assume that the given DFT values are in complex conjugate pairs and that the frequency locations are already known.

4.1.2 New Approach

The approach used in this paper summarizes as follows (more details are provided in the next section):

- Precompute, for a given configuration of DFT samples, a filter which has zero magnitude response at all unknown DFT sample locations;
- Compute the (sparse) $(N \times N)$ inverse 2-D DFT of the data whose frequency content is known (and filtered) and set to zero elsewhere;
- Deconvolve the solution from this filtered image and the filter. This requires two $(M \times M)$ 2-D DFTs;
- The total computation is at most $N^2 \log_2 N + 2M^2 \log_2 M$, or less if a sparse 2-D DFT is used.

4.2 Filter Design

4.2.1 Filter Specification

Let Ω be the set of ordered pairs $\{(k_1, k_2)\}$ for which we know the $(N \times N)$ 2-D DFT $X(k_1, k_2)$ of the original image $x(i_1, i_2)$. Then define the 2-D filter with 2-D

impulse response $h(i_1, i_2)$ and frequency response $H(k_1, k_2)$ as

$$h(i_1, i_2) = 0 \quad for \quad N - M + 1 \leq i_1, i_2 \leq N - 1$$

$$H(k_1, k_2) = 0 \quad for \quad (k_1, k_2) \notin \Omega$$

$$H(k_1, k_2) = \sum_{i_1=0}^{N-1} \sum_{i_2=0}^{N-1} h(i_1, i_2) e^{-j\frac{2\pi}{N^2}(i_1 k_1 + i_2 k_2)} \tag{4.2}$$

Note that the nonzero values of $h(i_1, i_2)$ and $H(k_1, k_2)$ are known and determined (to an overall scale factor) by the above conditions.

The filter is *precomputed* for each configuration of known DFT values of interest. This is not a problem, since the frequency locations at which DFT values will be obtained are usually known in advance. So filters for all configurations of interest can be precomputed and stored. Storage is more efficient in the frequency domain, since fewer values of $H(k_1, k_2)$ than $h(i_1, i_2)$ are nonzero.

4.2.2 Filter Size

The size of the filter is determined as follows:

- $h(i_1, i_2)$ is $(N - M + 1) \times (N - M + 1)$;
- $h(i_1, i_2) = 0$ at $N^2 - (N - M + 1)^2$ points;
- $H(k_1, k_2) = 0$ at $(N - M + 1)^2$ points;
- $H(k_1, k_2)$ is arbitrary at $N^2 - (N - M + 1)^2$ points.

These numbers are specified as follows:

- $h(i_1, i_2) * x(i_1, i_2)$ must be $N \times N$;
- #unknowns=#equations in the linear system;
- $H(k_1, k_2)$ known at $N^2 - (N - M + 1)^2$ points.

Note that if $N >> M$, then

$$N^2 - (N - M + 1)^2 \approx 2NM = 2(\frac{N}{M})M^2 \tag{4.3}$$

so that the original problem must be overdetermined by a factor of $2N/M$. This is not unreasonable; the existence of a unique solution to the original problem may require more frequencies than the number of image pixels [8],[6],[7]. Overdetermination is required since the 2-D deconvolution into which the original problem is transformed is itself overdetermined.

The filter could be computed by solving a large linear system of equations, but POCS for this *off-line* computation requires less storage. POCS was used to determine the filters for the examples below. No non-uniqueness issues have been encountered in computing filters, but it is possible that the filters themselves must also be overdetermined. Since this means fewer frequencies would be required for the original problem, this could only help.

Alternatively, the filter could also be computed based on the "finite-support regularization" method explained in Section 4.3 which converges faster than POCS and was used to create the filter for the later examples.

4.2.3 Filter Convolution

Now consider the filtered signal

$$y(i_1, i_2) = \sum_{j_1=0}^{N-1} \sum_{j_2=0}^{N-1} h(j_1, j_2) x(i_1 - j_1, i_2 - j_2) \tag{4.4}$$

Then the $(N \times N)$ 2-D DFT $Y(k_1, k_2)$ of $y(i_1, i_2)$ is

$$Y(k_1, k_2) = \begin{cases} H(k_1, k_2)X(k_1, k_2) & for (k_1, k_2) \in \Omega; \\ 0 & for (k_1, k_2) \notin \Omega \end{cases} \tag{4.5}$$

Then we may compute $x(i_1, i_2)$ by deconvolution of the known $h(i_1, i_2)$ from the known $y(i_1, i_2)$ computed from the known $Y(k_1, k_2)$ using an $(N \times N)$ inverse 2-D DFT. Note that $Y(k_1, k_2)$ is sparse since most of the $(N \times N)$ 2-D DFT values are zero.

Deconvolution can be accomplished by computing the inverse 2-D DFT of the quotient of the 2-D DFTs of $y(i_1, i_2)$ and $h(i_1, i_2)$. However, the 2-D DFT values of $h(i_1, i_2)$ used for deconvolution must all be nonzero. If N is an integer multiple of M, then an $(M \times M)$ 2-D DFT must NOT be used here. Instead, a DFT of slightly different order is used, so different unit circle locations are sampled. Since $M << N$, this is not a problem computationally.

4.3 Filter Construction

In order to create the filter, we apply a finite-support regularization that places a norm penalty on the non-support spatial pixels and then solves the problem using the preconditioned conjugate gradient method with three 2-D DFTs per iteration yet still leading to a faster convergence to the solution than as done with POCS. This duality of this method is presented in [9] yet we present a different perspective.

4.3.1 Background

This method combines the use of a priori knowledge of finite support used in POCS to create the regularization term with quickly solving the least-squares TBT structured system in (2.3) using conjugate gradients. It is evident each of the previous approaches requires some form of regularization. Typical choices for the regularization term are the solution norm penalizing identity operator, roughness-penalizing difference operator, and edge-preserving non-quadratic penalty operator. Yet all of these choices make assumptions about the solution which may not be accurate.

What we do know for certain is that the object is finite support and hence the

image values outside of this square (or finite region) are known for certain (typically that of air). Essentially POCS, while making no assumptions about the image values in the finite support, makes this exact assumption on the values outside the finite support. However POCS is too strict in enforcing this a priori information. Values outside of the finite support are set to zero in the spatial domain while alternating setting measured frequency locations with noisy data values.

If we were to relax the restriction by allowing a tolerable prescribed variance of error in the image values outside of the finite support this would allow a better data-fit and hopefully lead to faster convergence. This is accomplished by a scaling factor on the regularization term which would trade-off term between the data-fit (or residual) and prior-fit (or regularization) terms of the objective function. Another way to look at this is we are weakening the assumption that the image values outside the finite support is deterministic and known to those values are random but with known means and covariances. The data-fit and prior-fit terms are then computed with weighted norms using their respective covariance matrices. This objective function consisting of both terms is minimized into in a modified normal equation form, from which the conjugate gradient method solves for the solution.

Concerns raised in the previous Section 4.1 about the poor conditioning of a system of linear equations approach and the added cost of preconditioning are mitigated. As we will see in the following sections, the preconditioners applied in our new method lower the condition number of the system matrix to a constant value as a function of the regularization parameter, the circulant or diagonal preconditioners are easily invertible, and the number of operations for matrix-vector multiplication with the preconditioner is at worst the same as two 2-D FFTs or at best of linear order. The regularization of finite spatial support is an easily verifiable assumption which only requires knowledge of the spatial values outside of the object of interest. As the 2-D DFT grid to which the frequency locations are sampled on increases the

number of the unknown spatial values and the number of measured frequency samples remains unchanged with only the computational cost of the 2-D FFT per PCG iteration increasing at $O(N^2 logN)$. Over-determining the problem does not increase the computational cost at all since the 2-D DFT operations are already operating on a full grid.

4.3.2 Objective Function

The problem can be formulated as a 2-D Vandermonde system of linear equations

$$Fx = b \tag{4.6}$$

where $F_{r,c} = \frac{1}{N} e^{-j\frac{2\pi}{N^2}(i_{1,c}k_{1,r}+i_{2,c}k_{2,r})}$, $x_c = x(i_{1,c}, i_{2,c})$, and $b_r = X(k_{1,r}, k_{2,r})$ are entries for the 2-D DFT matrix, solution vector, and data vector respectively, with lexicographically ordered spatial indices $(i_{1,c}, i_{2,c})$ and frequency indices $(k_{1,r}, k_{2,r})$ for $r = 1, ..., N^2$ and $c = 1, ..., N^2$.

If we do not know all N^2 values of b but only some $K < N^2$ values and since there are N^2 unknown pixel values in x, (4.6) is under-determined. Though, knowing that the image has $(M \times M)$ finite support, we have information about the $N^2 - M^2$ spatial values outside of the finite support that can be utilized by regularization.

Let Π be the set of $(N^2 - M^2)$ ordered pairs $\{(i_1, i_2)\}$ that are outside the $(M \times M)$ square finite support of the original image $x(i_1, i_2)$ and let Ω be the set of K ordered pairs $\{(k_1, k_2)\}$ for which the $(N \times N)$ 2-D DFT values $X(k_1, k_2)$ are known. Then the objective function includes a residual term of only the known 2-D DFT values and a Tikhonov regularization term including only the non-support spatial values as follows

$$\Phi(x) = \|J_\Omega Fx - b_\Omega\|^2 + \lambda^2 \|J_\Pi x - x_{0,\Pi}\|^2 \tag{4.7}$$

where J_Ω is a $(K \times N^2)$ binary matrix that selects only entries in resulting vector Fx for which the frequency locations are in Ω, b_Ω is a vector of known frequency values in Ω, λ is the regularization parameter, J_Π is a $((N^2 - M^2) \times N^2)$ binary matrix that selects only entries in x that are in Π somewhat like an irregular downsampling matrix, and $x_{0,\Pi}$ is a vector of known values of the padding non-support region. Letting $x_{0,\Pi} = \vec{0}$, the minimization of (4.7) by taking the derivative of $\Phi(x)$ with respect to x then setting it to zero and solving for x is

$$\hat{x} = \arg\min_x \Phi(x) = \left(F^H I_\Omega F + \lambda^2 I_\Pi\right)^{-1} F^H J_\Omega^H b_\Omega \tag{4.8}$$

where $I_\Omega = J_\Omega^H J_\Omega$ is a $(N^2 \times N^2)$ diagonal binary matrix when multiplied with a vector, zeros out entries with frequency locations not in Ω, $I_\Pi = J_\Pi^H J_\Pi$ similarly is a $(N^2 \times N^2)$ diagonal binary matrix that zeros out entries of a vector with spatial locations not in Π, and J_Ω^H is a $(N^2 \times K)$ irregular upsampling matrix that fills in entries with frequency locations not in Ω with zeros. In a modified normal equation form, (4.8) is

$$\left(F^H I_\Omega F + \lambda^2 I_\Pi\right) x = F^H J_\Omega^H b_\Omega \tag{4.9}$$

where $F^H J_\Omega^H b_\Omega$ can be viewed as an incomplete inverse 2-D DFT estimate for x, because the 2-D DFT matrix F we defined in (4.1) is orthonormal, its Hermitian transpose is equal to its inverse

$$F^H = F^{-1}. \tag{4.10}$$

It turns out that the set of locations of the unknown pixel values, Π, does not necessarily need to be square nor connected in 2-D, as long as the remaining locations in the field of view $N \times N$ have known pixel values.

4.3.3 Uniqueness

We know that the number of 2-D DFT samples $K \geq M^2$ is necessary but is $K = M^2$ sufficient? It is shown in [8] that M^2 samples of a $(N \times N)$ 2-D DFT are not enough to uniquely determine a $M \times M$ image. However in our case, we have added constraints where the non-support spatial values are known at $N^2 - M^2$ locations in Π in addition to the $K = M^2$ 2-D DFT samples resulting in N^2 knowns to N^2 unknowns in the padded solution x. We show that the use of $K = M^2$ 2-D DFT samples in b_Ω is sufficient for a unique solution to (4.9).

To show uniqueness of the solution to (4.9) is to show that the system matrix

$$A = F^H I_\Omega F + \lambda^2 I_\Pi \tag{4.11}$$

where $|\Omega| = K = M^2$ and $|\Pi| = N^2 - M^2$, has $\text{rank}(A) = N^2$ or is non-singular. Letting

$$C = F^H I_\Omega F \tag{4.12}$$

$$D = \lambda^2 I_\Pi \tag{4.13}$$

we can state that $\text{rank}(C) = \text{rank}(I_\Omega) = M^2$ because $F^H I_\Omega F = F^{-1} I_\Omega F$ is similarity transformation and $\text{rank}(D) = \text{rank}(I_\Pi) = N^2 - M^2$. In [31], it is shown that the rank of the sum of two matrices

$$\text{rank}(C + D) = \text{rank}(C) + \text{rank}(D) \tag{4.14}$$

is additive of their individual ranks, if and only if $R(C) \cap R(D) = \{0\}$ and $R(C^H) \cap R(D^H) = \{0\}$, where $R(C)$ denotes the row space of matrix C. Because C and D are Hermitian symmetric is it only necessary to show the former, that the row spaces of

C and D are disjoint. The rows of C

$$u_r(c) = \frac{1}{N} \sum_{(k_1,k_2)\in\Omega} e^{-j\frac{2\pi}{N^2}(k_1(i_{1,c}-i_{1,r})+k_2(i_{2,c}-i_{2,r}))} \tag{4.15}$$

are sums of 2-D complex exponentials, whereas the rows of D

$$v_r(c) = \begin{cases} \lambda^2\delta(c-r), & (i_{1,r}, i_{2,r}) \in \Pi \\ \vec{0}, & \text{otherwise} \end{cases} \tag{4.16}$$

are vectors of either scaled Kronecker delta values or zeros. Let $w \in (R(C) \cap R(D))$ then it follows that $w \in R(C)$ and $w \in R(D)$ or similarly w is a linear combination of $\{u_r\}_r$ and also a linear combination of $\{v_r\}_r$

$$w(c) = \sum_{m=1}^{N^2} \alpha_m u_m(c) = \sum_{n=1}^{N^2} \beta_n v_n(c) \tag{4.17}$$

for some complex scalars $\{\alpha_m\}_{m=1,\ldots,N^2}$ and $\{\beta_n\}_{n=1,\ldots,N^2}$. Substituting (4.15) and (4.16) into (4.17), grouping scalars, and eliminating zero vectors from the summation, we arrive at

$$\begin{aligned} w(c) &= \sum_{\substack{p=1 \\ (k_{1,p},k_{2,p})\in\Omega}}^{M^2} \hat{\alpha}_p z^{-(k_{1,p}i_{1,c}+k_{2,p}i_{2,c})} \\ &= \sum_{\substack{q=1 \\ (i_{1,q},i_{2,q})\in\Pi}}^{N^2-M^2} \hat{\beta}_q \lambda^2 \delta(c-q) \end{aligned} \tag{4.18}$$

where $\hat{\alpha}_p = \frac{1}{N}\sum_{m=1}^{N^2} \alpha_m z^{k_{1,p}i_{1,m}+k_{2,p}i_{2,m}}$, and $z = e^{j\frac{2\pi}{N^2}}$ is the primitive N^2th root of unity. For the case where $M^2 = 0$ or $M^2 = N^2$, trivially $w = \vec{0}$ which is in $\{0\}$. When $M^2 = 1$, w is a single complex exponential or sequence with up to $N^2 - 1$ non-zero values but at least 1 zero entry. Since no single complex exponential can represent

a finite sequence with a zero value, $\{\hat{\alpha}_p\}$ and $\{\hat{\beta}_q\}$ must all be zero. Similarly, for $M^2 > 1$, w as a linear combination of $N^2 - M^2$ distinct Kronecker deltas with M^2 zeros requires a linear combination of $M^2 + 1$ distinct complex exponentials to be create a finite sequence with M^2 zeros, yet only M^2 distinct complex exponentials are available. In other words a linear combination of M^2 distinct complex exponentials only has $M^2 - 1$ primitive N^2th roots of unity. Once again the choices for $\{\hat{\alpha}_p\}$ and $\{\hat{\beta}_q\}$ are trivially zeros. Therefore, $w = \vec{0}$ in all cases, and $R(C)$ and $R(D)$ are disjoint. Hence $\text{rank}(A) = \text{rank}(C + D) = M^2 + (N^2 - M^2) = N^2$, A is non-singular, and (4.9) has a unique solution.

4.3.4 Frequency Selection

In most cases the configuration of the K 2-D DFT samples will be restricted, for example along radial lines for CT and a spiral for MRI. When measurements of these samples are taken along analytical trajectories, the 2-D DTFT samples can be fit to a rectangular grid with less interpolation errors by simply increasing the 2-D DFT size N without an increase of K. Regardless of the size of the 2-D DFT we employ, the $K = M^2$ will still yield a unique solution, because the padded region with locations in Π just increases in size $N^2 - M^2$. The trade-off is that the computational cost of computing the two 2-D DFTs increases as well. However, if the 2-D FFT is employed the cost increases in $O(N^2 \log(N))$.

In Section 4.3.3, it was shown that setting the number of measured 2-D DFT samples to $K = M^2$ is sufficient for a unique solution of (4.9), yet in the presence of measurement noise it may be desirable to over-determine the problem by taking more data samples as such $M^2 < K \leq N^2$.

4.3.5 Optimization Algorithm

Since the system matrix in (4.9) is Hermitian symmetric and positive definite we employ the conjugate gradient (CG) method. After substituting (4.10) into (4.9), the data vector $F^{-1}J_{\Omega}^{H}b_{\Omega}$ is precomputed with one inverse 2-D FFT using (4.10). The matrix-vector multiplication of $(F^{H}I_{\Omega}F + \lambda^{2}I_{\Pi})v$ is split into two steps. The first term $F^{-1}I\Omega F$ requires a 2-D FFT, an element-wise masking operation, followed by an inverse 2-D FFT. The second term $\lambda^{2}I_{\Pi}$ is a simple scaled masking operation. Lastly a vector addition is performed. The approximate cost is $20N^{2}\log_{2}(N)$ per CG iteration (given a 2-D FFT consists of about $10N^{2}\log_{2}(N)$ operations). Because of some of the zero diagonal entries in I_{Ω}, sparse 2-D FFT routines with lower computational costs may be employed. The number of iterations can be predetermined or determined based on the value of the objective function in (4.7) at the current iteration. Lastly the reconstructed $N \times N$ solution x is lexicographically wrapped into 2-D then the padded region is cropped leaving the $M \times M$ reconstructed image.

4.3.6 Preconditioning

Preconditioning the system matrix in (4.9)

$$\begin{aligned} P^{-1}Ax &= P^{-1}b \\ P^{-1}(F^{H}I_{\Omega}F + \lambda^{2}I_{\Pi})x &= P^{-1}F^{H}J_{\Omega}^{H}b_{\Omega} \end{aligned} \tag{4.19}$$

where P is the preconditioning matrix, before using the conjugate gradient method can improve the convergence rate, by reducing the condition number of the preconditioned system matrix $P^{-1}A$. The trade-off is that preconditioning step is required per iteration, adding to the computational cost. Therefore a good preconditioner resembles the system matrix but is fast to perform a matrix-vector multiplication with its inverse P^{-1} and a vector.

One preconditioner for (4.9) is one where I_Ω is replaced by a scaled identity matrix as such

$$\begin{aligned} P &= F^H \alpha I F + \lambda^2 I_\Pi \\ &= \alpha I + \lambda^2 I_\Pi \end{aligned} \tag{4.20}$$

where

$$\alpha = \operatorname{tr}(I_\Omega)/\operatorname{tr}(I) = K/N^2. \tag{4.21}$$

The preconditioner P is also easily invertible because P is diagonal matrix with non-zero diagonal entries. The inverse of P in (4.20) is simply

$$P^{-1} = W_P \tag{4.22}$$

where

$$W_{P,i,j} = \begin{cases} 1/(\alpha + \lambda^2 I_{\Pi,i,i}), & i = j \\ 0, & \text{otherwise} \end{cases} \tag{4.23}$$

We can see in (4.20) as λ becomes larger P better approximates the heavily regularized system matrix in (4.19). The additional computational cost of preconditioning with P is $O(N^2)$, lower order than that of a 2-D FFT, and is not counted. The PCG method with preconditioner P is a $10N^2 log_2(N)$ operation per iteration, the same as that of the regular conjugate gradient method.

4.3.7 Asymptotically Infinite Regularization

Because we desire a regularized solution, we analyze the effect of taking the limit of λ as it approaches infinity, which is equivalently assuming values of x with locations

in the non-support region Π have 0 variances from the Bayesian perspective and are known with certainty. We examine the preconditioned normal equation in (4.19) with preconditioner P as such

$$W_P(F^H I_\Omega F + \lambda^2 I_\Pi)x \;=\; W_P F^H J_\Omega^H b_\Omega. \tag{4.24}$$

and we notice that if λ approaches infinity in the limit then the system matrix equivalent to A in (4.24) will also approach infinity when performing the Ax matrix-vector multiplication in the standard PCG method. The transformed PCG method on the other hand uses a symmetric preconditioned system matrix by Cholesky decomposing the preconditioner as such

$$P = EE^H \tag{4.25}$$

where the diagonal P easily decomposes with $E = E^H = \text{diag}((\sqrt{\alpha + \lambda^2 I_{\Pi,i,i}})_i)$. Notation $\text{diag}((v_i)_i)$ denotes a diagonal matrix with v_i diagonal entries of vector v. The transformed preconditioned normal equation as λ approaches infinity in the limit is

$$\left(\frac{1}{\alpha} I_{\Pi^c} F^H I_\Omega F I_{\Pi^c} + I_\Pi\right) y \;=\; \frac{1}{\sqrt{\alpha}} I_{\Pi^c} F^H J_\Omega^H b_\Omega \tag{4.26}$$

$$x \;=\; \frac{1}{\sqrt{\alpha}} I_{\Pi^c} y \tag{4.27}$$

where $I_{\Pi^c} = I - I_\Pi$ and Π^c is the complement set of Π or the set of image support locations. The transformed preconditioned conjugate gradient method with Cholesky decomposed preconditioner P is still $10N^2 log_2(N)$ operations per iteration, the same as that of the regular conjugate gradient method. So we get the benefit of preconditioning at a small cost of a few more $O(N^2)$ diagonal-matrix-vector multiplications.

4.3.8 Equivalence to POCS

The preconditioned finite support PCG method with an infinite λ in the limit is related to the POCS method. Both methods employ the same constraints of finite spatial support, however their optimization methods differ, leading to different convergence rates. We will show that solving (4.26) as a stationary iterative method, or a fixed-point iteration method, instead of a Krylov subspace method, like PCG, is equivalent to the POCS method re-expressed also as a stationary iterative method. The finite support PCG method can be viewed an improvement of the POCS method, one with a statistical model and a faster converging optimization method, while using the same finite support constraint of POCS as its regularization.

4.4 Noisy System Model

The presence of measurement noise in the 2-D DTFT samples needs to be considered. The data is assumed to be corrupted with additive white Gaussian complex noise $\eta(k_1, k_2)$, where $\eta(k_1, k_2) = \alpha(k_1, k_2)e^{j\beta(k_1,k_2)}$, $\alpha(k_1, k_2) \sim \mathcal{N}(0, \sigma_\eta)$, and $\beta(k_1, k_2) \sim \mathcal{U}(0, 2\pi)$. The effective system model is then

$$H(k_1, k_2)X(k_1, k_2) = \tilde{Y}(k_1, k_2) = Y(k_1, k_2) + \eta(k_1, k_2). \tag{4.28}$$

4.4.1 Regularization

Rewritten in matrix form, the noisy system model in (4.28) is

$$H_L X_L = \tilde{Y}_L = Y_L + \eta_L. \tag{4.29}$$

To alleviate the problems from potentially large magnitudes of the 2-D DFT of the inverse filter amplifying noise, we implement Tikhonov regularization that minimizes

the residual and regularization term in matrix form

$$\hat{X}_L = \arg\min_{X_L} \|H_L X_L - \tilde{Y}_L\|^2 + \lambda^2 \|C_L X_L\|^2 \tag{4.30}$$

where the noise vector η_L and solution vector $X_L$are both length L^2 column vectors of the lexicographically unwrapped noise $\eta(k_1, k_2)$ and $L \times L$ 2-D DFT of the solution $x(i_1, i_2)$, respectively.

Filter matrix H_L has size $L^2 \times L^2$ with the lexicographically unwrapped $L \times L$ 2-D DFT of the filter $h(i_1, i_2)$ along the diagonal. We set the Tikhonov matrix C_L in Fig. 4.1(a) as an $L^2 \times L^2$ matrix with the lexicographically unwrapped $L \times L$ 2-D DFT of

$$c(i_1, i_2) = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & -12 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{4.31}$$

along the diagonal. Function $c(i_1, i_2)$ is essentially a 2nd order difference operator and as such is a high-pass filter that penalizes high frequency components in the solution and assumes a smooth solution.

Data vector $\tilde{Y}_L$ is a length-L^2 column vector of the lexicographically unwrapped $L \times L$ 2-D DFT of the filtered data $\tilde{y}(i_1, i_2)$ in (4.53) and can be computed from $\tilde{Y}_N$, the length-N^2 column vector of the unwrapped $N \times N$ 2-D DFT version of $\tilde{Y}(k_1, k_2)$ in (4.52), as

$$\tilde{Y}_L = F_L I_{pad} F_N^{-1} \tilde{Y}_N \tag{4.32}$$

where I_{pad} is a $L^2 \times N^2$ matrix with standard basis column vectors e_r which pads in 2-D an unwrapped image of size $N \times N$ to $L \times L$. Matrix

$$F_N = \{e^{-j2\pi(i_1 k_1 + i_2 k_2)/N}\}_{r(k_1,k_2), c(i_1,i_2)} \tag{4.33}$$

is the 2-D DFT matrix of size $N^2 \times N^2$ where $r(k_1, k_2)$ and $c(i_1, i_2)$ are lexicographically ordering row and column functions based on the frequency and spatial indices (k_1, k_2) and (i_1, i_2).

The explicit solution to (4.30) is found by taking the derivative of both terms with respect to X_L, setting it to a zero vector, and solving for X_L as

$$\hat{X}_L = (H_L^* H_L + \lambda^2 C_L^* C_L)^{-1} H_L^* \tilde{Y}_L \tag{4.34}$$

which contains the matrix equivalent of the functional regularized inverse filter

$$G_{reg}(k_1, k_2) = \frac{H^*(k_1, k_2)}{|H(k_1, k_2)|^2 + \lambda^2 |C(k_1, k_2)|^2}. \tag{4.35}$$

The unwrapped spatial solution is computed as

$$\hat{x}_M = I_{crop} F_L^{-1} \hat{X}_L \tag{4.36}$$

where I_{crop} is a $M^2 \times L^2$ matrix with standard basis row vectors e_c which crops in 2-D an unwrapped image of padded size $L \times L$ to support size $M \times M$.

We can show the direct relationship of the solution $\hat{x}_M$ to the DFT of the noisy filtered image $\tilde{Y}_N$ in (4.37) or to DFT of the true image at K sampled locations X_K^{true} in (4.39), by successively substituting equations (4.52), (4.29), (4.32), (4.34), and (4.36).

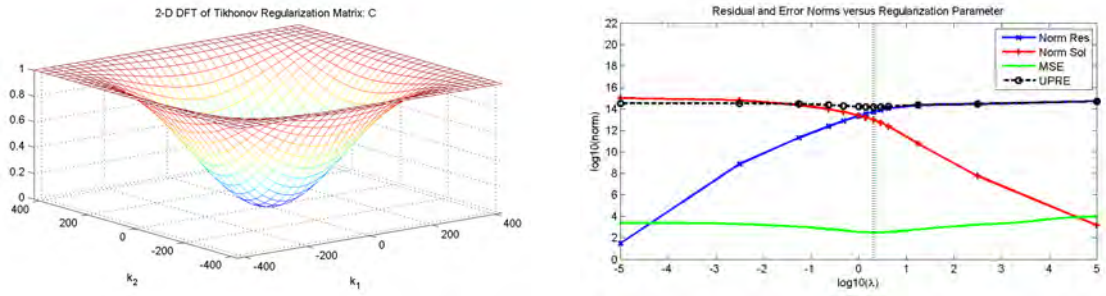
$$\hat{x}_M = I_{crop} F_L^{-1} \cdot (H_L^* H_L + \lambda^2 C_L^* C_L)^{-1} H_L^* \cdot F_L I_{pad} F_N^{-1} \cdot \tilde{Y}_N \tag{4.37}$$

$$= I_{crop} F_L^{-1} \cdot (H_L^* H_L + \lambda^2 C_L^* C_L)^{-1} H_L^* \cdot F_L I_{pad} F_N^{-1} \cdot H_N \cdot I_{sample} (X_K^{true} + \eta_K^{interp} + \eta_K^{meas}) \tag{4.38}$$

I_{sample} is a $N^2 \times K$ matrix of standard basis column vectors e_r that places the K

interpolated and measured 2-D DFT samples into a lexicographically unwrapped spiral frequency configuration. Column vectors η_K^{interp} and η_K^{meas} are the nearest-neighbor interpolation error and measurement noise vectors, respectively. Basically, the sequence of operations of (4.39) on the measured and interpolated 2-D DFT data are (from right to left multiplication): filtering, Fourier interpolation in the frequency domain, regularized deconvolution, and finally transformation into the spatial domain.

The solution vector in (4.34) can be computed very quickly because H_L and C_L are diagonals and the inverse operation of the diagonal matrix $H_L^* H_L + \lambda^2 C_L^* C_L$ reciprocates the diagonal entries and the remaining matrix multiplications are also with diagonal matrices. Therefore the computational complexity for $\hat{X}_L$ is $O(L^2)$ and can be cheaply computed repeatedly as done when iteratively selecting the regularization parameter in the next subsection. The resized DFT of the filtered image $\tilde{Y}_L$ in (4.32) is computed only once per image and in $O(L^2 \log L + N^2 \log N)$. The spatial solution $\hat{x}_M$ in (4.36) is also solved only once and in $O(L^2 \log L)$.



(a) 2-D DFT of 2nd order difference Tikhonov regularization matrix: C

(b) Residual and error norms versus regularization parameter

Figure 4.1: Tikhonov matrix and errors vs. regularization parameter using noisy data.

4.4.2 Regularization Parameter Selection

Regularization reduces noise amplification yet a problem that still remains is determining how much regularization is optimal. The regularization parameter λ effectively

sets the ratio between $r(\lambda) = \|H_L\hat{X}_L - \tilde{Y}_L\|^2$ which is the squared residual norm and $s(\lambda) = \|C_L\hat{X}_L\|^2$ which is the squared solution penalty norm as seen in Fig. 4.1(b). If λ is too small then the noise in the solution is not suppressed enough. Yet if λ is too large then considering C_L is a high-pass penalty the solution is over-smoothed.

Of the various regularization parameter selection methods implemented such as the L-curve method or the generalized cross validation (GCV) method, the unbiased predictive risk estimator (UPRE) resulted in the best success at selecting regularization parameter λ that closely matched the solution error norm function $\text{MSE}(\lambda) = \|e(\lambda)\|^2 = \|\hat{x}_L(i_1, i_2; \lambda) - x_L(i_1, i_2)\|^2$, where $x_L(i_1, i_2)$ is the unknown true image.

The unbiased predictive risk estimator in the general form as presented in [32] is

$$\text{UPRE}(\lambda) = \|H_L X_L - \tilde{Y}_L\|^2 + 2\sigma_\eta^2 tr(H_L(H_L^* H_L + \lambda^2 C_L^* C_L)^{-1} H_L^*) \tag{4.39}$$

assuming that the noise η_L vector consists of white noise samples. Using the fact that H_L and C_L are diagonal matrices, (4.39) simplifies to

$$\text{UPRE}(\lambda) = \|H_L\hat{X}_L - \tilde{Y}_L\|^2 + 2\sigma_\eta^2 \sum_{i=1}^{L^2} \frac{h_i^2}{h_i^2 + \lambda^2 c_i^2} \tag{4.40}$$

where h_i and c_i are the ith diagonal entries of H_L and C_L respectively.

The noise variance σ_η^2 is estimated using the fact that the DFT samples are measured at complex conjugate pair locations. Assuming the true image is real then the true filtered DFT samples exist in complex conjugate pairs as in

$$Y_L(k_1, k_2) = Y_L^*(\ell_1, \ell_2) \tag{4.41}$$

where $(\ell_1, \ell_2) = (L-1-k_1, L-1-k_2)$ is the complex conjugate location of (k_1, k_2). If we take the difference between the noisy filtered DFT samples $\tilde{Y}_L(k_1, k_2)$ in (4.29)

and its complex conjugate version and substitute with (4.41), the complex conjugates cancel one another as in

$$\begin{aligned}\tilde{Y}_L(k_1,k_2) - \tilde{Y}_L^*(\ell_1,\ell_2) &= Y_L(k_1,k_2) - Y_L^*(\ell_1,\ell_2) + \eta_L(k_1,k_2) - \eta_L^*(\ell_1,\ell_2) \\ &= Y_L(k_1,k_2) - Y_L(k_1,k_2) + \eta_L(k_1,k_2) - \eta_L^*(\ell_1,\ell_2) \\ &= \eta_L(k_1,k_2) - \eta_L^*(\ell_1,\ell_2). \end{aligned} \tag{4.42}$$

Then taking the variance of both sides in (4.42) and solving for the noise variance leads to

$$\sigma_\eta^2 = \frac{1}{2}\,\mathrm{Var}[\tilde{Y}_L(k_1,k_2) - \tilde{Y}_L^*(\ell_1,\ell_2)] \tag{4.43}$$

estimated by the sample variance

$$s_\eta^2 = \frac{1}{2(K/2-1)} \sum_{i=1}^{K/2} (\tilde{Y}_L(k_{1,i},k_{2,i}) - \tilde{Y}_L^*(\ell_{1,i},\ell_{2,i}) - m)^2 \tag{4.44}$$

$$m = \frac{1}{K/2} \sum_{i=1}^{K/2} \tilde{Y}_L(k_{1,i},k_{2,i}) - \tilde{Y}_L^*(\ell_{1,i},\ell_{2,i}) \tag{4.45}$$

where K again is the number of known frequency samples.

As shown in Fig. 4.1(b), the regularization parameter λ that minimizes the UPRE is found using bracketing and bisection search. The first iteration computes the UPRE at 3 locations, the bounds of $\lambda \in [10^{-5}, 10^5]$ and its midpoint, which if chosen properly, it brackets the minimum of the UPRE curve. Successive iterations of the regularization parameter selection computes 2 more UPRE values that further bisect the current two intervals, keep the 3 of the 5 sample points that refine the minimum bracket, and repeat the procedure until convergence. The cost of computing UPRE(λ) is fast because both terms in (4.40) and $\hat{X}_L$ each have computational complexities of $O(L^2)$. Therefore each iteration requires 2 UPRE values, each requiring 3 $O(L^2)$

operations, resulting in 6 $O(L^2)$ operations per iteration as shown in Table 4.2. The additional per-image $O(L^2 \log L)$ 2-D FFT operation in the regularization method comes from computation of the Tikhonov penalty matrix C_L. The estimation of noise variance σ_η^2 needs to be done only once under our noise model assumptions in $O(K)$.

4.5 Results

We begin with a tiny pedagogical example that illustrates the each step of this method with actual numerical values. Then various reconstruction methods are compared with our "fast non-iterative" method using a just-determined filter and an over-determined filter, both created with the POCS method. The practical considerations of non-gridded frequency data and noisy data are presented. Regularization described in the above section is applied in the noisy case. Then using the same precomputed filter, other images are quickly reconstructed from simulated data as well.

Then we compare the improvements of using a filter constructed from the finite-support regularization method over the POCS method again for the cases of a just-determined filter, and over-determined filter, and noisy data. Runtime comparisons of the single deconvolution step of just three 2D FFTs in the "fast non-iterative" method with the other reconstruction methods are presented.

Finally, images of size 256×256 are reconstructed from actual CT data using various methods including the filtered back-projection method are presented. Images of size 512×512 are reconstructed with low noise and high noise with regularization. Lastly, the images are reconstructed at the largest size of 888×888.

4.5.1 Small Example

4.5.1.1 Problem Specification

Consider the problem of reconstructing a 3×3 "image" from its frequencies marked with an x below:

$$\begin{bmatrix} x & * & * & * & x & * & * & * & x \\ * & x & * & * & x & * & * & x & * \\ * & * & x & * & x & * & x & * & * \\ * & * & * & x & x & x & * & * & * \\ x & x & x & x & x & x & x & x & x \\ * & * & * & x & x & x & * & * & * \\ * & * & x & * & x & * & x & * & * \\ * & x & * & * & x & * & * & x & * \\ x & * & * & * & x & * & * & * & x \end{bmatrix} \tag{4.46}$$

In this frequency sampling pattern:

- x denotes locations of known frequencies;
- $*$ denotes locations of unknown frequencies;
- The origin of the frequency plane is the center;
- The leftmost and rightmost columns are identical;
- The top and bottom rows are also identical;
- This polar raster is a type used in tomography;
- $N = 8; M = 3; N^2 - (N - M + 1)^2 = 28$;
- 28 values of (8×8) 2-D DFT are known;
- 36 values of (8×8) 2-D DFT are unknown.

This can be regarded as a tiny example of a tomography problem, in which an image is reconstructed from its projections at four angles.

4.5.1.2 Filter

The 6×6 filter $h(i_1, i_2)$ is specified as follows:

$$h(i_1, i_2) = 0 \quad for \quad 6 \leq i_1, i_2 \leq 7$$

$$H(k_1, k_2) = 0 \quad for \quad (k_1, k_2) \notin \Omega$$

$$H(k_1, k_2) = \sum_{i_1=0}^{7} \sum_{i_2=0}^{7} h(i_1, i_2) e^{-j\frac{2\pi}{64}(i_1 k_1 + i_2 k_2)} \tag{4.47}$$

Ω is the set of 28 locations marked with x above. These 36 linear equations in 36 unknowns can be solved either directly, by using POCS, or by using the CG method with finite-support regularization. The result (rounded off) is

$$h(i_1, i_2) = \begin{bmatrix} 1.1 & .79 & .56 & .56 & .79 & 1.1 \\ .79 & 1.3 & .84 & .84 & 1.3 & .79 \\ .56 & .84 & 1.4 & 1.4 & .84 & .56 \\ .56 & .84 & 1.4 & 1.4 & .84 & .56 \\ .79 & 1.3 & .84 & .84 & 1.3 & .79 \\ 1.1 & .79 & .56 & .56 & .79 & 1.1 \end{bmatrix} \tag{4.48}$$

Note that the filter could be specified using either:

- 36 nonzero values of $h(i_1, i_2)$;
- 28 nonzero values of $H(k_1, k_2)$.

For this tiny example there is no reason to prefer $H(k_1, k_2)$ specification. But for realistic examples the savings in storage can be substantial.

4.5.1.3 Solution

In particular, suppose we are given the (rounded) (8×8) 2-D DFT values $Y(k_1, 0 \leq k_2 \leq 4) =$

$$\begin{bmatrix} 45 & 36e^{-j0.9} & -6-15j & 7.5e^{j1.4} & 15 \\ 38e^{-j1.12} & 32e^{-j2.0} & * & * & * \\ -18-15j & * & -5+8j & * & * \\ 14e^{j1.9} & * & * & 2.5e^{j2.8} & * \\ 15 & * & * & * & 5 \\ 14e^{-j1.9} & * & * & 1.5e^{-j0.9} & * \\ -18+15j & * & 5+4j & * & * \\ 38e^{j1.12} & 30e^{j0.2} & * & * & * \end{bmatrix} \tag{4.49}$$

- $Y(0,0)$ is now in the upper left corner;
- The polar raster pattern is thus altered;
- $Y(k_1, 5 \leq k_2 \leq 7) = Y^*(8-k_1, 8-k_2)$;
- With conjugates, 28 of 64 values are known;
- The 36 unknowns are designated with $*$.

Now compute the (8×8) inverse 2-D DFT of

$$\begin{bmatrix} 45 & 36e^{-j0.9} & -6-15j & 7.5e^{j1.4} & 15 \\ 38e^{-j1.12} & 32e^{-j2.0} & 0 & 0 & 0 \\ -18-15j & 0 & -5+8j & 0 & 0 \\ 14e^{j1.9} & 0 & 0 & 2.5e^{j2.8} & 0 \\ 15 & 0 & 0 & 0 & 5 \\ 14e^{-j1.9} & 0 & 0 & 1.5e^{-j0.9} & 0 \\ -18+15j & 0 & 5+4j & 0 & 0 \\ 38e^{j1.12} & 30e^{j0.2} & 0 & 0 & 0 \end{bmatrix} \tag{4.50}$$

where again conjugate values have been omitted to save space. The inverse 2-D DFT itself is omitted here. Then deconvolving the filter $h(i_1, i_2)$ above from this inverse 2-D DFT results in the image

$$x(i_1, i_2) = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \tag{4.51}$$

Note that even for this tiny example, the computation of an (8×8) inverse 2-D DFT and a small deconvolution is significantly smaller than that of solving a linear system of equations with nine unknowns.

4.5.2 POCS Filter Simulation Results

4.5.2.1 Settings

Results implementing the method developed in this chapter are presented. The "head" image $x(i_1, i_2)$ has size $M \times M = 128 \times 128$. The Discrete Fourier Transform samples are measured along a square-inscribed spiral pattern with 60 revolutions and up to 16384 angular samples per revolution forced onto a $N \times N = 768 \times 768$ grid (where the ratio N/M is 6.00). The frequency values are sampled at discrete grid locations, loosely following an analytical spiral, as shown in Fig. 4.2(a), and thus are exact DFT values. The conjugate symmetric set of samples $X_{data}(k_1, k_2)$ shown in Fig. 4.2(b) is used in the reconstruction, assuming the solution is real. No noise is added to the DFT samples initially.

The filter $h(i_1, i_2)$ is $P \times P = (N - M + 1) \times (N - M + 1) = 641 \times 641$ constructed using projection onto convex sets (POCS) shown as its 2-D DFT in Fig. 4.2(c). The number of unknown values of $h(i_1, i_2)$ is $P^2 = (N - M + 1)^2 = 410,881$ and the number of zero 2-D DFT values of $h(i_1, i_2)$ is set to also be 410,881. The number of spiral revolutions is adjusted and the DFT samples with the highest frequency

locations are trimmed so that $h(i_1, i_2)$ can be solved as a just-determined system.

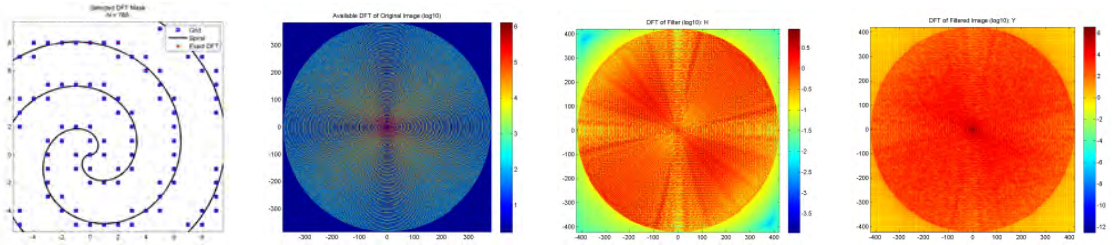
The incomplete 2-D DFT samples of the original image, $X_{incomp}(k_1, k_2)$ in Fig. 4.2(b), are multiplied with the 2-D DFT of the filter, $H(k_1, k_2)$ in Fig. 4.2(c), to produce the 2-D DFT of the filtered image, $Y(k_1, k_2)$ in Fig. 4.2(d), as

$$Y(k_1, k_2) = H(k_1, k_2) \cdot X_{incomp}(k_1, k_2). \tag{4.52}$$

This is equivalent to the original image $x(i_1, i_2)$ in Fig. 4.4(a) being convolved with filter $h(i_1, i_2)$ to produce a filtered image

$$y(i_1, i_2) = h(i_1, i_2) * *x(i_1, i_2). \tag{4.53}$$

Now the goal is to recover $x(i_1, i_2)$ by deconvolving $h(i_1, i_2)$ from $y(i_1, i_2)$, the inverse 2-D DFT of $Y(k_1, k_2)$.



(a) Spiral frequency sampling with conjugate symmetry (b) Spiral pattern masked 2-D DFT (c) Log DFT of filter: H (d) Log DFT of filtered image: $Y = X \cdot H$

Figure 4.2: Spiral frequency mask, POCS-generated filter, and filtered image.

The growth of the filter size $P \times P$ and the number of padded zeros in the filter $N^2 - P^2$ are shown in Fig. 4.3(a) and normalized by N^2 in Fig. 4.3(b). If the just-determined system for the filter requires the number of zeroed or unknown frequency samples equals the number of filter spatial coefficients or support size P^2, then equivalently the number of measured or known frequency samples $K = N^2 - P^2 = 2N(M-1) - (M-1)^2 = 178,943$ equals the number of zeros

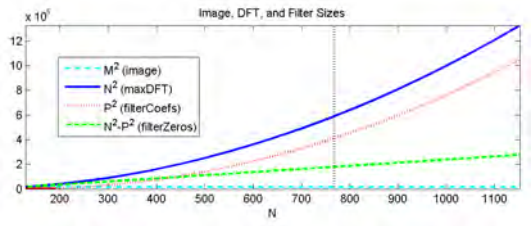
in the filter. The resulting known frequency density

$$\frac{K}{N^2} = 2(M-1) \cdot \frac{1}{N} - \frac{(M-1)^2}{N^2} \tag{4.54}$$

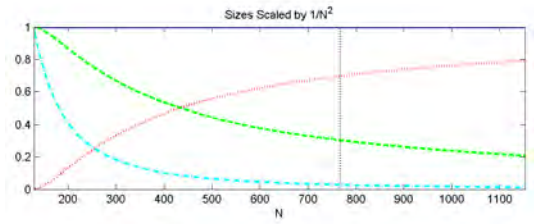
decreases dominantly with $1/N$ in Fig. 4.3(c). The overdetermining factor

$$\frac{K}{M^2} = \frac{2(M-1)}{M^2} \cdot N - \frac{(M-1)^2}{M^2} \tag{4.55}$$

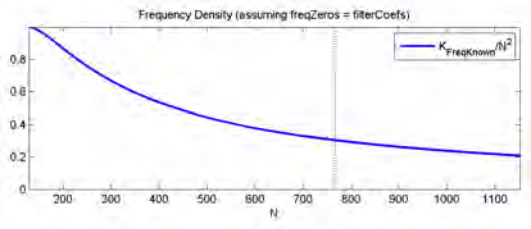
increases linearly with N in Fig. 4.3(d). Thus increasing the DFT size N allows a spiral configuration to have larger radial spacing and also lower nearest neighbor interpolation errors when approximating Discrete-Time Fourier Transform (DTFT) data to a DFT grid. In turn a larger number of frequency data samples are required thus increasing the data size to solution size ratio. In our experiment, setting $N = 768$, denoted by the vertical black dotted line in Fig. 4.3, leads to a frequency density of 30% DFT grid coverage and an overdetermining factor of 10.9.



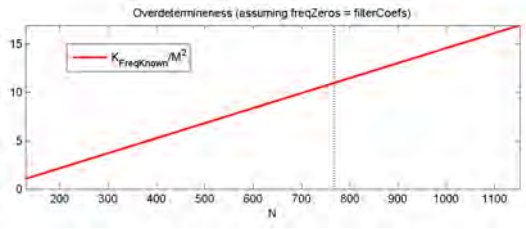
(a) Image, DFT, and filter sizes



(b) Sizes scaled by $1/N^2$



(c) Frequency density

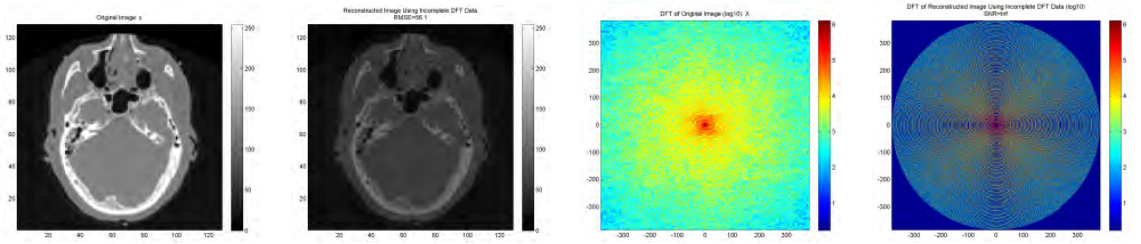


(d) Overdetermining factor

Figure 4.3: Solution, data, and filter sizes and ratios as functions of DFT size N (black dotted line denotes N = 768) for "fast non-iterative" method.

4.5.2.2 Inverse 2-D DFT of *Incomplete* Frequency Samples

For a baseline comparison, we can approximate the solution $x(i_1, i_2)$ by applying the inverse 2-D DFT to use only the known 2-D DFT samples while setting the unknown 2-D DFT samples to zero, which we will call the "incomplete-DFT" solution. Fig. 4.4(a)-4.4(b) compare the incomplete-DFT solution with a root mean square error (RMSE) of 56.1 with the original image. The RMSE is computed over the image support of size $M \times M$ only even though the reconstructed solution is of size $N \times N$. Fig. 4.4(c)-4.4(d) compare the original 2-D DFT with only the known 2-D DFT samples.



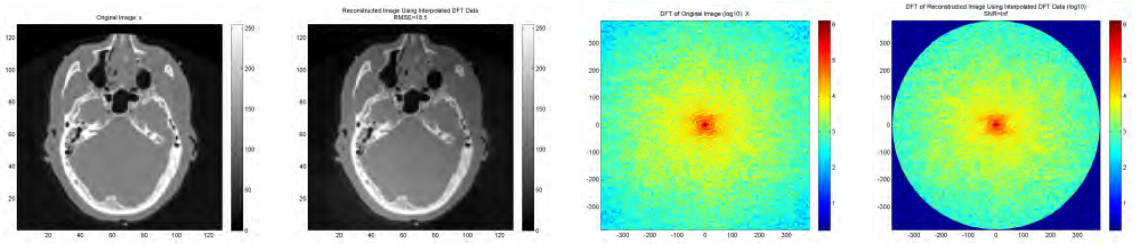
(a) Original image: x (b) Incomplete-DFT solution (c) Original 2-D DFT (d) Incomplete-DFT 2-D DFT

Figure 4.4: Original and incomplete-DFT solution 2-D DFTs in log scale and images with RMSE = 56.1.

4.5.2.3 Inverse 2-D DFT of *Interpolated* Frequency Samples

Linearly interpolating the missing 2-D DFT samples of the convex set of measured 2-D DFT samples leads to a more accurate "interpolated-DFT" solution. The "interpolated-DFT" performs a Delaunay triangulation which can be performed in $O(N^2 \log N)$, a triangle search procedure for each grid point, interpolation using barycentric coordinates, and finally an inverse 2-D FFT. Fig. 4.5(a)-4.5(b) compare the interpolated-DFT solution with a root mean square error of 18.5 with the original image. Fig. 4.5(c)-4.5(d) compare the original 2-D DFT with the interpolated 2-D DFT samples. The reconstructed image in Fig. 4.5(b) shows fading in intensity in

the upper right of the image, more apparent in later reconstructions. The cause of this artifact is that linear interpolation poorly estimates missing DFT samples at very low frequencies around the DC location, due to the high-spiking and non-piecewise nature and the non-symmetric Delaunay triangulation of the DC and its nearest DFT samples.

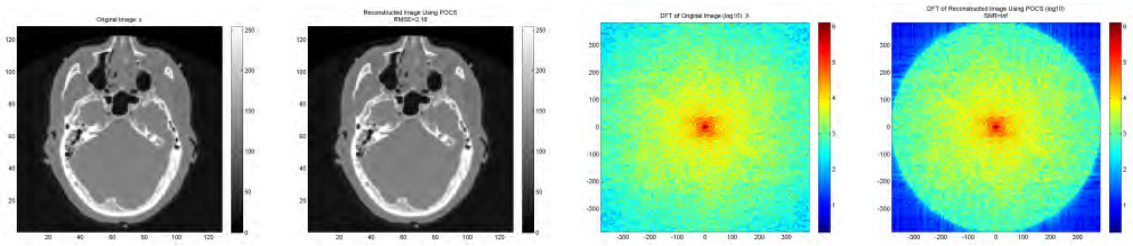


(a) Original image: x (b) Interpolated-DFT solution (c) Original 2-D DFT (d) Interpolated-DFT 2-D DFT

Figure 4.5: Original and interpolated-DFT solution 2-D DFTs in log scale and images with RMSE = 18.5.

4.5.2.4 Projection Onto Convex Sets

Projection onto convex sets (POCS) alternates applying constraints on the data in the spatial and frequency domains until the iterative solution converges. In the spatial domain the solution is assumed to be real and to have a finite support of $M \times M$. In the frequency domain the values of measured 2-D DFT samples are maintained. The POCS method requires two 2-D FFTs per iteration. The number of POCS iterations is set so that the RMSE of the POCS reconstructed at least image matches the RMSE of the "fast non-iterative" reconstructed image, introduced in the next subsection. Fig. 4.6(a)-4.6(b) compare the POCS solution with a root mean square error of 2.16 with the original image. Fig. 4.6(c)-4.6(d) compare the original 2-D DFT with the 2-D DFT of the POCS solution.



(a) Original image: x (b) POCS solution image (c) Original 2-D DFT (d) POCS solution 2-D DFT

Figure 4.6: Original and POCS solution image 2-D DFTs in log scale and images with RMSE = 2.16.

4.5.2.5 2-D DFT Based Inverse Filtering or "Fast Non-Iterative"

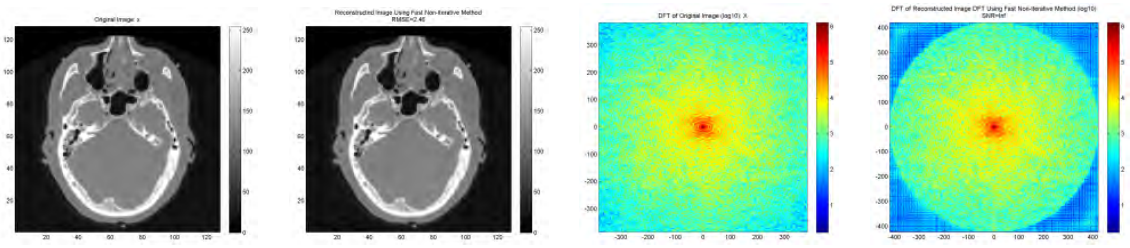
Now we solve the deconvolution problem in (4.52) by taking the inverse 2-D DFT of the quotient of the 2-D DFTs of $y(i_1, i_2)$ and $h(i_1, i_2)$, which is

$$\hat{x}(i_1, i_2) = \mathcal{F}_L^{-1}\left(\frac{\mathcal{F}_L(y(i_1, i_2))}{\mathcal{F}_L(h(i_1, i_2))}\right) = \mathcal{F}_L^{-1}(G(k_1, k_2) \cdot Y(k_1, k_2)), \tag{4.56}$$

where $\mathcal{F}_L(\cdot)$ is the 2-D $L \times L$ DFT operator and

$$G(k_1, k_2) = \frac{1}{H(k_1, k_2)} \tag{4.57}$$

is the inverse filter. We also name this the "fast non-iterative" method. In this case the size of the 2-D DFT is $L \times L = 844 \times 844$, slightly larger than $N \times N = 768 \times 768$. The "fast non-iterative" method requires N^2 multiplications in (4.52) to get $Y_{N \times N}(k_1, k_2)$, an inverse $N \times N$ 2-D FFT to get $y_{N \times N}(i_1, i_2)$, an $L \times L$ 2-D FFT to get $Y_{L \times L}(k_1, k_2)$, and finally L^2 multiplications with the inverse filter in (4.57) followed by an inverse $L \times L$ 2-D FFT to get $\hat{x}(i_1, i_2)$ in (4.56). Fig. 4.6(a)-4.6(b) compare the fast non-iterative solution with a root mean square error of 2.46 with the original image. Fig. 4.6(c)-4.6(d) compare the original 2-D DFT with the 2-D DFT of the fast non-iterative solution.



(a) Original image (b) Inverse filtered image (c) Original 2-D DFT (d) Inverse filtered 2-D DFT

Figure 4.7: Original and fast non-iterative inverse filtered solution 2-D DFTs in log scale and images with RMSE = 2.46.

4.5.2.6 Comparison of Reconstruction Methods

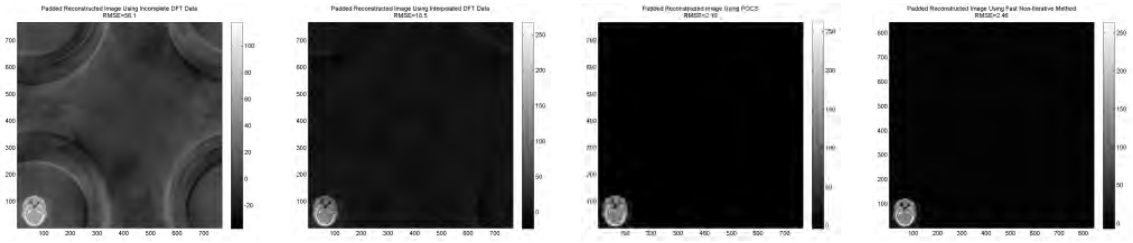
The RMSE of the fast non-iterative solution, as shown in Fig. 4.7(b), of 2.46 is lower than the RMSEs of the incomplete-DFT solution of 56.1, the interpolated-DFT solution of 18.5, and is intentionally close to the RMSE of the POCS solution of 2.16, summarized in Table 4.1. Observing the whole $N \times N$ reconstructed solution of the "incomplete-DFT" solution in Fig. 4.8(a), much of the energy in the image is spread across the $N \times N$ grid, which can be explained by the fact that the zeros in the Fourier domain between the spiral samples is equivalent to upsampling a denser DFT and thus getting distorted duplicates of the images offset radially outwards. The other padded reconstructed images in Fig. 4.8(b)-4.8(d) show relatively low energy in the non-image support regions.

The approximate computational costs of each reconstruction method are listed in Table 4.1 along with their RMSE versus cost plot in in Fig. 4.9. The "incomplete-DFT" method just performs an inverse 2-D FFT and thus is the same as the first iteration of the POCS method with the same computational cost. The "interpolated-DFT" and "fast non-iterative" methods each have lower RMSEs, the greater the number of required operations. The POCS method is iterated until its RMSE falls below that of the "fast non-iterative" method. In this case, POCS iterated 9 times, requiring 4.6 times more operations than the "fast non-iterative" method. If POCS

Table 4.1: Fast Non-Iterative Reconstruction Error

Reconstruction Method	Support RMSE	Number of Iterations (n)	Approximate Complexity	Approximate Operations
Incomplete-DFT	56.1	1	$N^2 \log_2 N$	5.7×10^6
Interpolated-DFT	18.5	1	$2N^2 \log_2 N$	1.1×10^7
POCS	2.16	9	$(2n-1)N^2 \log_2 N$	9.6×10^7
Inverse Filtered	2.46	1	$2L^2 \log_2 L + L^2$ $+N^2 \log_2 N + N^2$	2.1×10^7

were iterated only 2 times with approximately 2.8×10^7, slightly more than the "fast non-iterative" method's 2.1×10^7 approximate operations, the RMSE of the POCS reconstructed image of 17.7 is still over 7 times worse than that of the "fast non-iterative" reconstruction.



(a) Padded incomplete-DFT image (b) Padded interpolated-DFT image (c) Padded POCS image (d) Padded inverse filtered image

Figure 4.8: Size padded reconstructed image comparison against inverse filtered method.

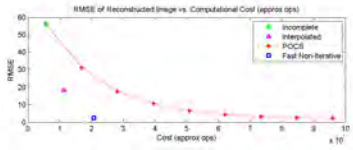
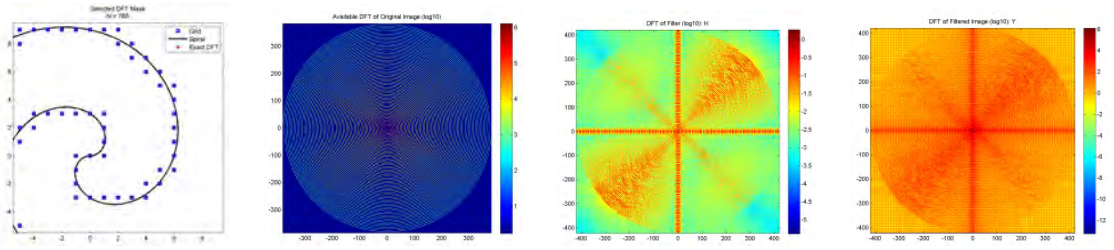


Figure 4.9: RMSE vs. approximate operations of reconstruction methods comparison against inverse filtered method.

4.5.2.7 Over-Determined Filter

If we want to change the filter requirement so that the number of zeroed or unknown frequency samples can be greater than the number of filter spatial coefficients or support size $P^2 = 410,881$, we achieve this by decreasing the number of frequency sampling spiral revolutions from 60 to 32 as shown in Fig. 4.10(a)-4.10(b). The number of zeroed or unknown frequency samples increases from 410,881 to 493,887, over-determining the filter by a ratio of 1.2. The number of known frequency samples K then decreases from 178,943 to 95,937. As a consequence, the known frequency density K/N^2 then decreases from 0.30 to 0.16 and the reconstructed image over-determining factor K/M^2 from 10.9 to 5.9. So therefore, if collecting data samples fewer than $N^2 - P^2 = 178,943$ is desired, then the filter $h(i_1, i_2)$ in the "fast non-iterative" method becomes over-determined for which there is no exact solution, and the filter will be computed as an approximation using POCS.

Fig. 4.10(c) and Fig. 4.10(d) show the DFT of the over-determined filter and the DFT of the associated filtered image, respectively. Notice in Fig. 4.10(c) how the DFT of the over-determined filter has large amplitudes in between the spiral arms along the vertical and horizontal axes, which may cause artifacts in the deconvolved images.



(a) Spiral frequency sampling with conjugate symmetry

(b) Spiral pattern masked 2-D DFT

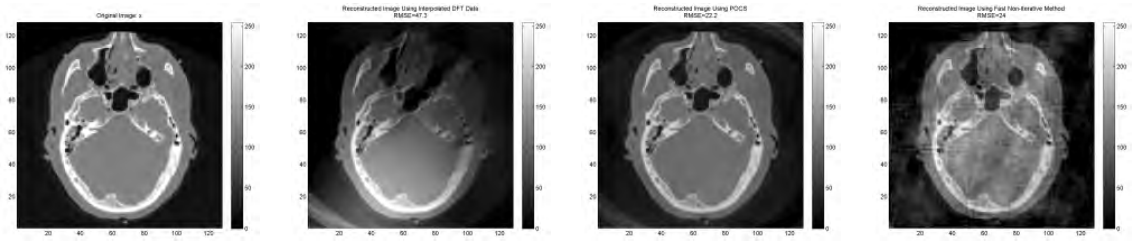
(c) Log DFT of over-determined filter: H

(d) Log DFT of over-determined-filtered image: $Y = X \cdot H$

Figure 4.10: Filter-over-determining spiral frequency mask, POCS-generated filter, and filtered image.

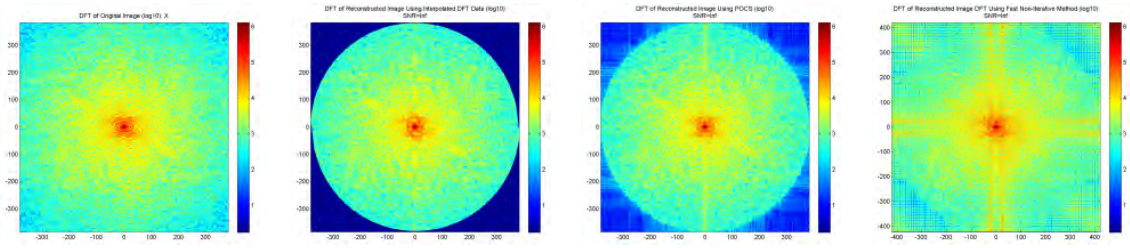
The reconstruction results, of using fewer DFT samples and for the "fast non-

iterative" method using an over-determined filter, are shown in Fig. 4.11 and Fig. 4.12. To conserve space in this paper, the figures of the "incomplete-DFT" case are not shown but their RMSEs and approximate costs are still presented in plot form. With fewer data samples to reconstruct the same sized image, all methods perform worse. The "interpolated-DFT" method has pronounced intensity fading in the corner. Even though the reconstructed images of POCS and "fast non-iterative" methods achieve similar RMSE's of 22.2 and 24.0 respectively, the latter has worse visual artifacts in Fig. 4.11(d). The 2-D DFT of the "fast non-iterative" image shows large streaks along the frequency axes, which is most likely a result of using an inexact over-determined filter $H(k_1, k_2)$ that does not attenuate the unknown DFT locations all the way to zero, as desired. The error versus operations in Fig. 4.13 shows that POCS iterated only 6 times, requiring 3 times more operations than the "fast non-iterative" method.



(a) Original image (b) Interpolated-DFT image, RMSE=47.3 (c) POCS solution image, RMSE=22.2 (d) Over-determined inverse filtered image, RMSE=24

Figure 4.11: Reconstructed images using fewer DFT samples and an **over-determined** filter in log scale and images compared against inverse filtered method.



(a) Original 2-D DFT (b) Interpolated DFT (c) POCS solution 2-D DFT (d) Over-determined inverse filtered 2-D DFT

Figure 4.12: 2-D DFTs of reconstructed images using fewer DFT samples and an **over-determined** filter in log scale and images compared against inverse filtered method.

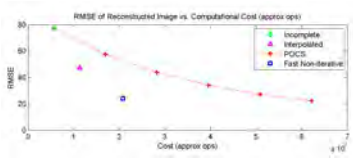
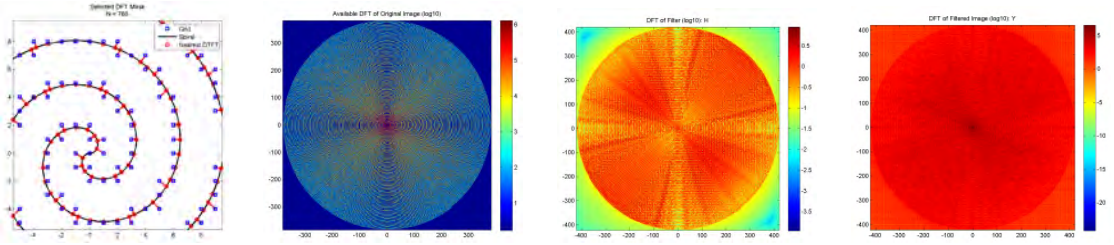


Figure 4.13: RMSE vs. approximate operations of reconstruction methods compared against inverse filtered method using fewer DFT samples and an over-determined filter.

4.5.2.8 Non-Gridded Frequency Data

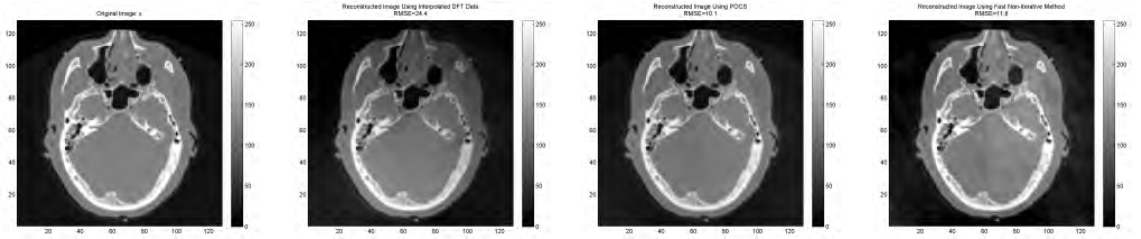
Practically, frequency data sampled on a DFT grid may not be possible. In such cases we can approximate the required DFT samples by using nearest neighbor interpolation from the acquired DTFT samples as shown in Fig. 4.14(a). This introduces some error in the DFT samples in Fig. 4.14(b) which can be perceived as additive noise. The effective signal-to-noise ratio (SNR) defined as $\text{SNR}(dB) = 20\log_{10}(\frac{\sigma_{signal}}{\sigma_{noise}})$ is 13.7 dB.

The POCS image with an RMSE of 10.1 requires 2.4 times as many operations as the "fast non-iterative" image with an RMSE of 11.8. Some artifacts are seen in the "fast non-iterative" image which can also be seen equivalently in its 2-D DFT in the vertical and horizontal high frequency locations. This suggests as the noise level increases, the inverse filtering will lead to an increasingly blown up solution.



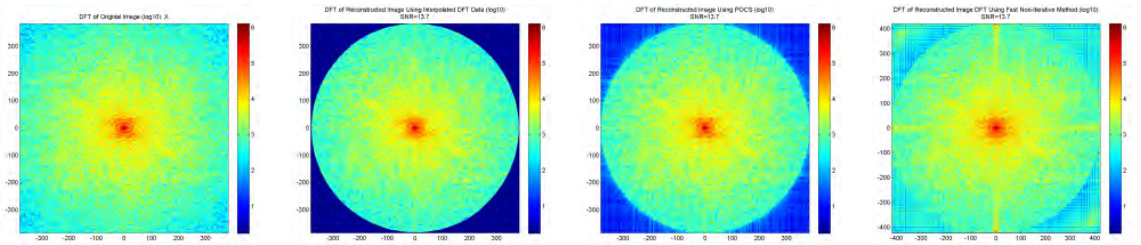
(a) Spiral frequency sampling with conjugate symmetry (b) Spiral pattern masked 2-D DFT (c) Log DFT of filter: H (d) Log DFT of image: $Y = X \cdot H$

Figure 4.14: Nearest-neighbor interpolated spiral frequency mask, POCS-generated filter, and filtered image.



(a) Original image (b) Interpolated-DFT image, RMSE=24.4 (c) POCS solution image, RMSE=10.1 (d) Inverse filtered image using nearest-neighbor data, RMSE=11.8

Figure 4.15: Reconstructed images using **nearest-neighbor** interpolated data in log scale and images compared against inverse filtered method.



(a) Original 2-D DFT (b) Interpolated DFT (c) POCS solution 2-D DFT (d) Inverse filtered 2-D DFT using nearest-neighbor interpolated data

Figure 4.16: 2-D DFTs of reconstructed images using **nearest-neighbor** interpolated data in log scale and images compared against inverse filtered method.

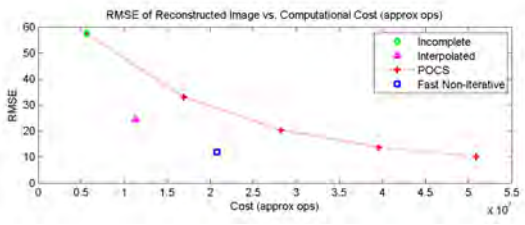
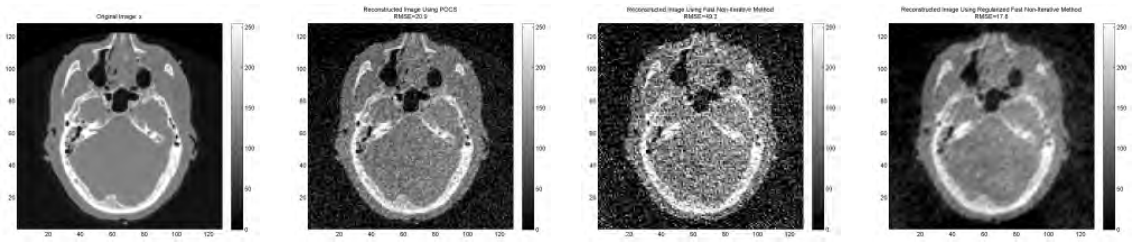


Figure 4.17: RMSE vs. approximate operations of reconstruction methods compared against inverse filtered method using nearest-neighbor interpolated data.

4.5.2.9 Noisy Reconstruction

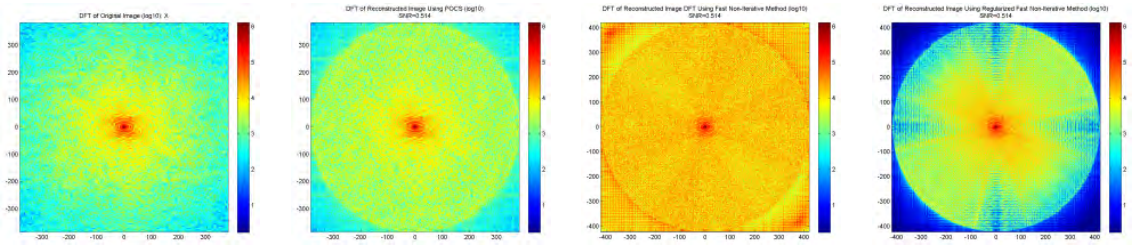
Another practical consideration is the presence of measurement noise in the 2-D DTFT samples. The data is corrupted with additive white Gaussian noise so that the SNR is 0.51 dB. Again the DFT samples used in the data vector are nearest-neighbor interpolated from the measured DTFT samples as done in the previous section. A just-determined filter H is used in the "fast non-iterative" method.

The POCS solution (with RMSE=20.9) in Fig. 4.18(b) performs better than that of the "fast non-iterative" solution in Fig. 4.18(c) in the presence of noise in terms of RMSE at the same number of computations or in terms of the number of computations at the same RMSE as shown in Fig. 4.20. The "fast non-iterative" reconstructed image is very noisy (with RMSE=49.3) and is more apparent in its 2-D DFT in Fig. 4.19(c), where the noise in high frequency locations have been amplified. The "incomplete-DFT" and "interpolated-DFT" figures were again removed to space space but their RMSEs and costs are shown in Fig. 4.20.



(a) Original image (b) POCS solution image, RMSE=20.9 (c) Inverse filtered image, RMSE=49.3 (d) Regularized inverse filtered image, RMSE=17.8

Figure 4.18: Reconstructed images using **noisy** data in log scale and images compared against inverse filtered method.



(a) Original 2-D DFT (b) POCS solution 2-D DFT (c) Inverse filtered 2-D DFT (d) Regularized inverse filtered 2-D DFT

Figure 4.19: 2-D DFTs of reconstructed images using **noisy** data in log scale and images compared against inverse filtered method.

4.5.2.10 Regularized Noisy Reconstruction

The UPRE estimated the optimal regularization parameter as $\lambda_{UPRE} = 2.1$ with an RMSE of 17.8 shown in Fig. 4.1(b) and in Table 4.2. The true optimal regularization parameter is $\lambda_{true} = 2.4$ with the lowest possible RMSE of 17.8, the same as the RMSE found using the UPRE.

The UPRE-estimated "regularized" solution achieved a much lower RMSE of 17.8 versus the "fast non-iterative" solution with an RMSE of 49.3 but using more operations shown in Fig. 4.20. The "regularized" solution also out-performed the POCS method in terms RMSE (of 20.9) at the same number of operations and lowest achievable RMSE up to 6.2×10^7 operations. Further iterations of the POCS solution would increasingly model noise and the RMSE would increase. However, even though the POCS reconstructed image has a larger RMSE than that of the "regularized" image, the POCS image retains more crisp edges in Fig. 4.18(b). The "regularized" image is less noisy and smoother with loss of sharp edges in Fig. 4.18(d). The reasons are more apparent in their 2-D DFTs in Fig. 4.19(b) and Fig. 4.19(d). Clearly the "regularized" method has much greater attenuated the high frequencies more than the POCS method. This is not surprising since the a priori information of each method differ. The POCS method assumes that the spatial domain image has a $M \times M$ finite support and zero everywhere else, which is a correct assumption. The "regularized" method assumes that the spatial domain image is smooth, which is not entirely correct, and with no information on the size of the support. On the other hand the POCS method assumes the 2-D DFT samples are true values without noise, whereas the "regularized" method does assume dominance of noise in the higher frequency locations.

At lower levels of noise, simulations have shown that with enough iterations the POCS method achieves a lower RMSE than that of the "regularized" method most likely due to the differing a priori information used. The "regularized" method can be

Table 4.2: Fast Non-Iterative Reconstruction Error using Noisy Data

Reconstruction Method	Support RMSE	Number of Iterations (n)	Approximate Complexity	Approximate Operations	Regularization Parameter (λ)
Incomplete-DFT	57.9	1	$N^2 \log_2 N$	5.7×10^6	n/a
Interpolated-DFT	30.1	1	$2N^2 \log_2 N$	1.1×10^7	n/a
POCS	20.9	6	$(2n-1)N^2 \log_2 N$	6.2×10^7	n/a
Fast Non-Iterative	49.3	1	$2L^2 \log_2 L + L^2$ $+N^2 \log_2 N + N^2$	2.1×10^7	0.0
Regularized	17.8	6	$3L^2 \log_2 L + (6n+3)L^2$ $+N^2 \log_2 N + N^2$	5.5×10^7	$2.1 = 10^{0.31}$
Best Regularized	17.8	n/a	n/a	n/a	$2.4 = 10^{0.38}$

implemented using a different Tikhonov matrix C_L for example finite spatial support. However C_L would no longer be a diagonal matrix because a finite spatial support masking in the spatial domain requires a convolution in the Fourier domain and C_L would become a full matrix or a block-banded matrix at best. The computational advantage of the "regularized" method would be lost in non-trivially inverting non-diagonal matrices in (4.34).

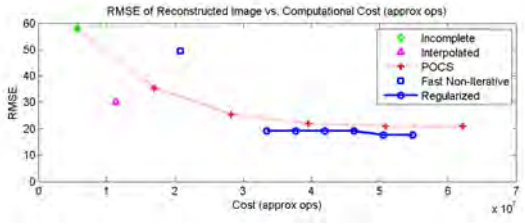


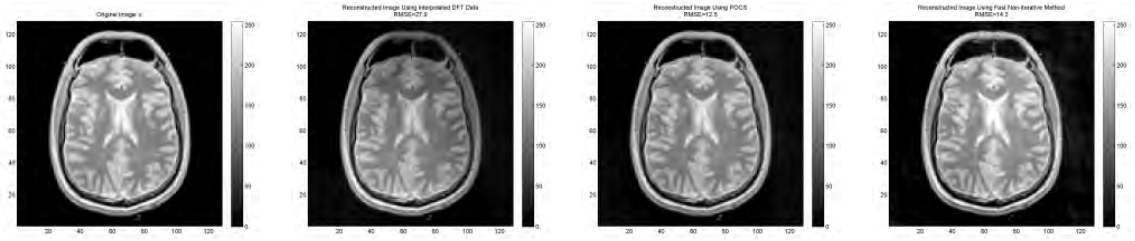
Figure 4.20: RMSE vs. approximate operations of reconstruction methods compared against inverse filtered method using noisy data.

4.5.2.11 Quick Reconstruction of Other Images

Other images can be reconstructed quickly using the same precomputed filter $h(i_1, i_2)$ and $H(k_1, k_2)$ if the 2-D DTFT samples are also collected on the same frequency configuration. Again a just-determined filter is used and the 2-D DTFT samples are nearest neighbor interpolated to a 2-D DFT grid but no explicit noise is added.

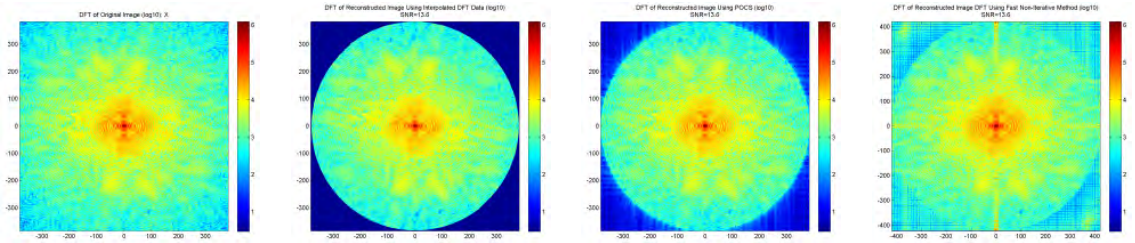
MRI Head

Data samples are simulated from an MRI reconstructed image of the head ("MRI head") as shown in Fig. 4.22(a). The effective signal-to-noise ratio is 13.6 dB. The POCS image with an RMSE of 12.5 requires 2.4 times as many operations as the "fast non-iterative" image with an RMSE of 14.3. Subtle artifacts are seen in the "fast non-iterative" image which can also be seen equivalently in its 2-D DFT in the vertical and horizontal high frequency locations.



(a) Original image (b) Interpolated-DFT image, RMSE=27.9 (c) POCS solution image, RMSE=12.5 (d) Inverse filtered image, RMSE=14.3

Figure 4.21: "MRI head" reconstructed images in log scale compared against inverse filtered method.



(a) Original 2-D DFT (b) Interpolated DFT (c) POCS solution 2-D DFT (d) Inverse filtered 2-D DFT

Figure 4.22: 2-D DFTs of "MRI head" reconstructed images in log scale compared against inverse filtered method.

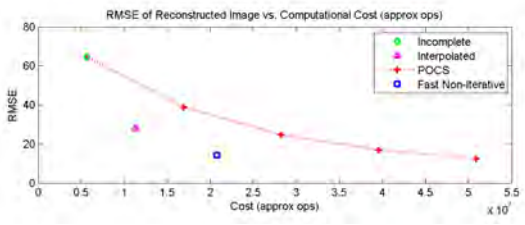
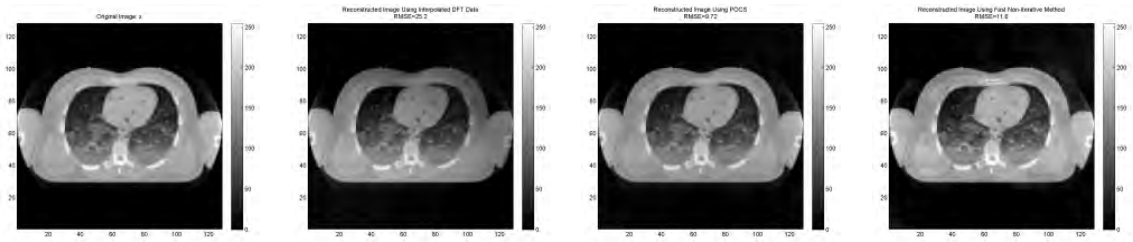


Figure 4.23: RMSE vs. approximate operations of reconstruction methods compared against inverse filtered method for "MRI head".

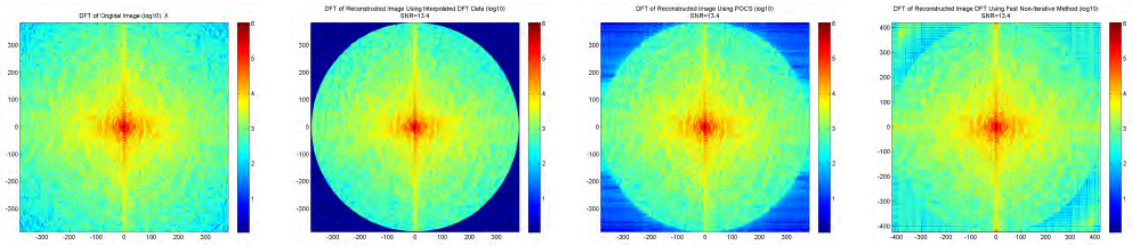
CT Thorax

Data samples are simulated from a CT reconstructed image of the thorax ("CT thorax") as shown in Fig. 4.25(a). The effective signal-to-noise ratio is 13.4 dB. The POCS image with an RMSE of 9.72 requires 2.4 times as many operations as the "fast non-iterative" image with an RMSE of 11.6. Horizontal streaking artifacts are seen in the "fast non-iterative" image which can also be seen equivalently in its 2-D DFT in the vertical high frequency locations.



(a) Original image (b) Interpolated-DFT image, RMSE=25.2 (c) POCS solution image, RMSE=9.72 (d) Inverse filtered image, RMSE=11.6

Figure 4.24: "CT thorax" reconstructed images in log scale compared against inverse filtered method.



(a) Original 2-D DFT (b) Interpolated DFT (c) POCS solution 2-D DFT (d) Inverse filtered 2-D DFT

Figure 4.25: 2-D DFTs of "CT thorax" reconstructed images in log scale compared against inverse filtered method.

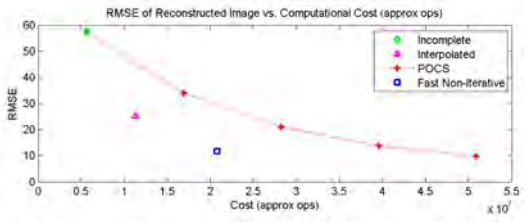
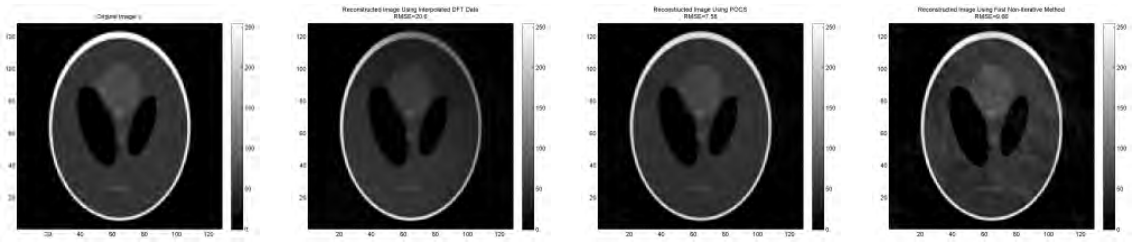


Figure 4.26: RMSE vs. approximate operations of reconstruction methods compared against inverse filtered method for "CT thorax".

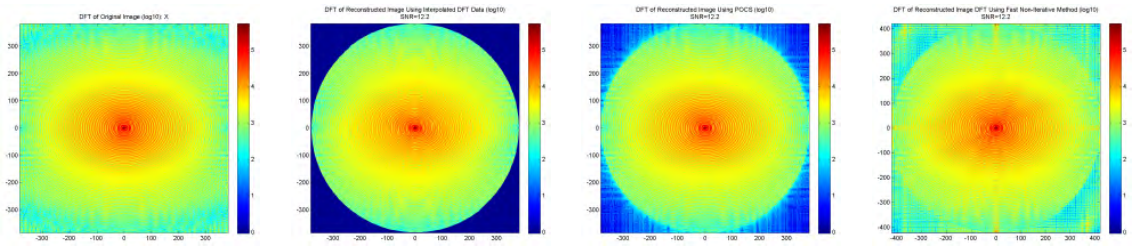
Shepp-Logan Phantom

Data samples are collected from a Shepp-Logan phantom image of a simulated head ("Phantom") as shown in Fig. 4.28(a). The effective signal-to-noise ratio is 12.2 dB. The POCS image with an RMSE of 7.58 requires 2.4 times as many operations as the "fast non-iterative" image with an RMSE of 9.66. Wrinkling artifacts are seen in the "fast non-iterative" image which can also be seen equivalently in its 2-D DFT in the vertical and horizontal high frequency locations.



(a) Original image (b) Interpolated-DFT image, RMSE=20.6 (c) POCS solution image, RMSE=7.58 (d) Inverse filtered image, RMSE=9.66

Figure 4.27: "Phantom" reconstructed images in log scale compared against inverse filtered method.



(a) Original 2-D DFT (b) Interpolated DFT (c) POCS solution 2-D DFT (d) Inverse filtered 2-D DFT

Figure 4.28: 2-D DFTs of "Phantom" reconstructed images in log scale compared against inverse filtered method.

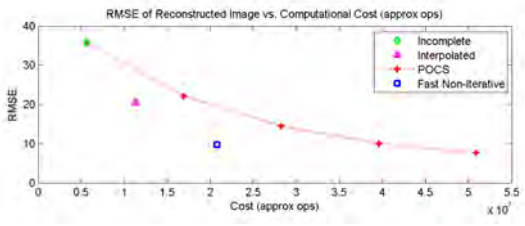


Figure 4.29: RMSE vs. approximate operations of reconstruction methods compared against inverse filtered method for "Phantom".

4.5.3 Finite-Support Regularization Filter Simulation Results

4.5.3.1 Settings

Results using a filter generated using finite-support regularization are presented. The "CT head" image $x(i_1, i_2)$ has size $M \times M = 128 \times 128$. The DFT samples are measured along a square-inscribed spiral pattern fitted onto a $N \times N = 768 \times 768$ grid (where the ratio N/M is 6.00). The frequency values are sampled at discrete grid locations, loosely following an analytical spiral, and thus are exact DFT values. The conjugate symmetric set of samples $X_{data}(k_1, k_2)$ is used in the reconstruction, assuming the solution is real. No noise is added to the DFT samples initially.

The filter $h(i_1, i_2)$ is $P \times P = (N - M + 1) \times (N - M + 1) = 641 \times 641$ constructed in one case using projection onto convex sets (POCS) and in another case using a finite-support regularized (FSR) preconditioned conjugate gradient (PCG) method.

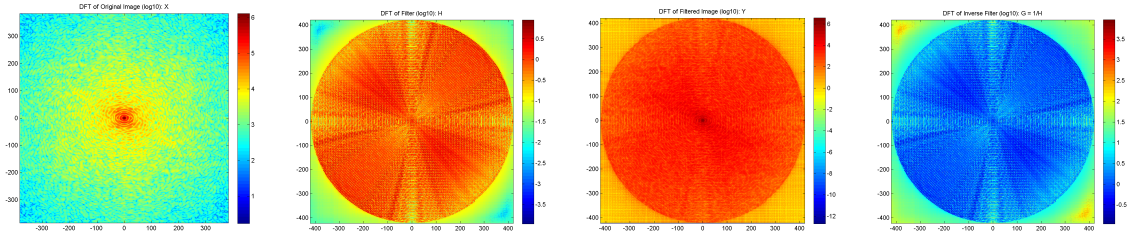
In summary, the goal is to deconvolve the solution x from the filtered image $y = h * *x$, where the 2-D DFT of the filtered image is the product of 2-D DFT of the filter and the known 2-D DFT data samples as such $Y = H \cdot X_{data}$.

4.5.3.2 Just-Determined Filter Comparison

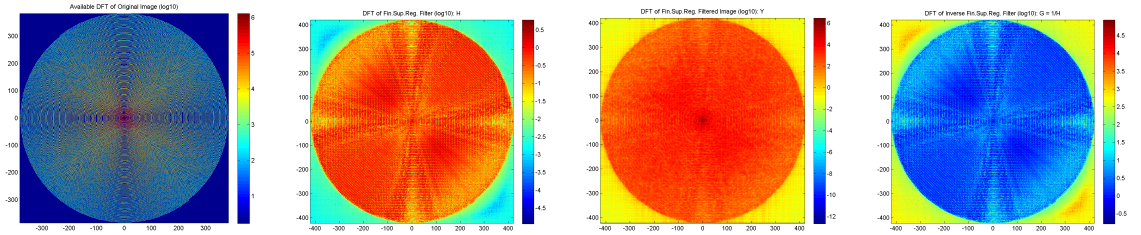
The number of unknown values of $h(i_1, i_2)$ is $P^2 = (N-M+1)^2 = 410,881$ and the number of zero 2-D DFT values $H(k_1, k_2)$ is set to also be 410,881. The spiral pattern has 60 revolutions and up to 16384 angular samples per revolution (essentially a contiguous path). The DFT samples with the highest frequency locations are trimmed so that $h(i_1, i_2)$ can be solved as a just-determined system. Note that this makes the overall reconstruction problem highly overdetermined by a factor of 10.9 with 178,943 known 2-D DFT samples X_{data} in Fig. 4.30(e) to solve for only 16,384 image pixels in x.

Both the POCS filter in Fig. 4.30(b) and finite-support regularized filter in Fig.

4.30(f) are each generated with 250 iterations with two $N \times N$ 2-D DFT's per iteration. The root mean squared error (RMSE) of the 2-D DFT locations of the filter that should be zero for the POCS filter is 4.77×10^{-2} and for the FSR filter is 3.01×10^{-3}, about a 16-fold decrease. This improvement can be seen in the lower and more uniform values of the region outside of the spiral in the 2-D DFT of the FSR filter in Fig. 4.30(f) as compared to Fig. 4.30(b). This is more evident in the 2-D DFT of the filtered images, Y, in Fig. 4.30(c) and Fig. 4.30(g) and in the inverse filters, $G = 1/H$, in Fig. 4.30(d) and Fig. 4.30(h).



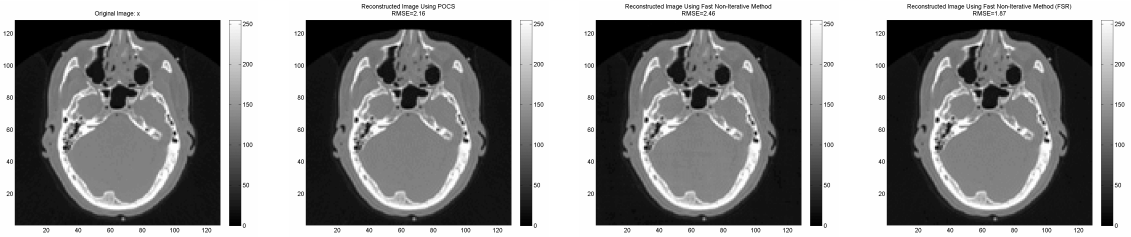
(a) Log DFT of true image (b) Log DFT of POCS filter (c) Log DFT of POCS filtered true image (d) Log DFT of Inverse POCS Filter



(e) Known Log DFT of true image (f) Log DFT of FSR filter (g) Log DFT of FSR filtered true image (h) Log DFT of inverse FSR filter

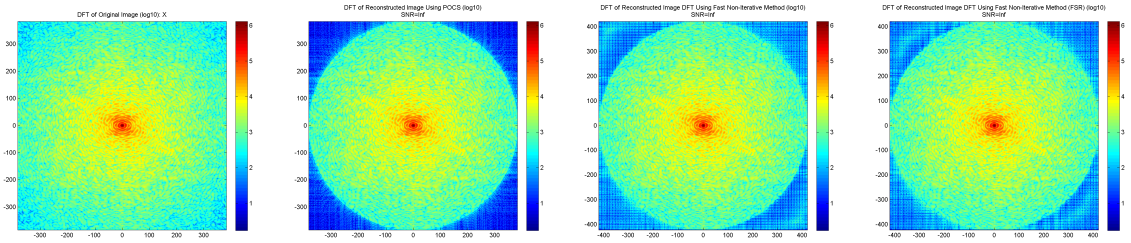
Figure 4.30: Frequency data and **just-determined** POCS versus FSR filter comparison.

As seen from Fig. 4.31-4.32, the POCS and both fast non-iterative using inverse filter methods produce reconstructed images with low error at around 2, compared with the RMSEs of the incomplete-DFT and cubic interpolation methods in Fig. 4.33 and in Table 4.3. However the fast non-iterative methods require about 5 times less operations than the POCS reconstruction method.



(a) True image (b) POCS solution image, RMSE=2.16 (c) Inverse POCS-filtered image, RMSE=2.46 (d) Inverse FSR-filtered image, RMSE=1.87

Figure 4.31: Reconstructed images using **just-determined** POCS versus FSR filters.



(a) True image DFT (b) POCS solution DFT (c) Inverse POCS-filtered DFT (d) Inverse FSR-filtered DFT

Figure 4.32: Log DFT of reconstructed images using **just-determined** POCS versus FSR filters.

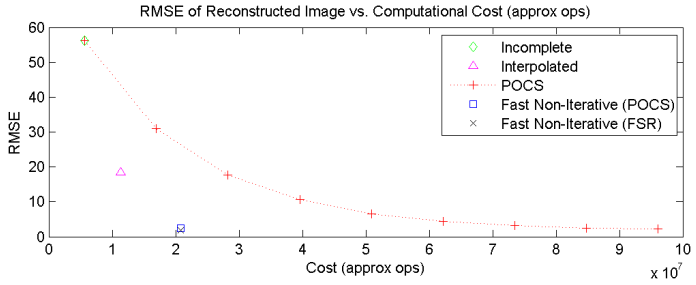


Figure 4.33: RMSE vs. approximate operations of reconstruction methods using **just-determined** POCS versus FSR-generated filters.

Table 4.3: Fast Non-Iterative Reconstruction Error for Just-Determined Filters

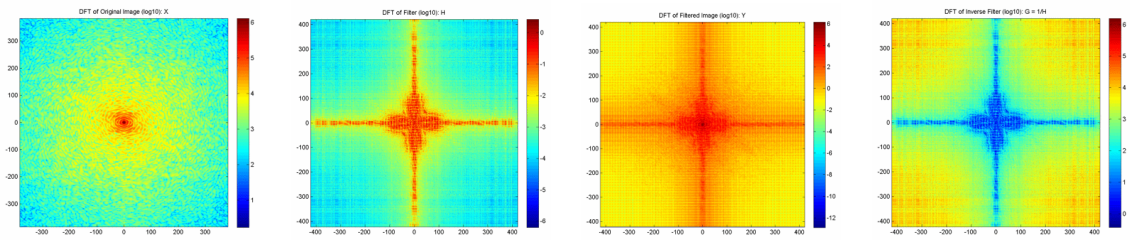
Reconstruction Method	RMSE	Iterations	Operations
Incomplete DFT	56.1	1	5.7×10^6
Cubic Interpolation	18.5	1	1.1×10^7
POCS RMSE-Matched	2.16	8	9.6×10^7
Fast Non-Iterative (POCS)	2.46	1	2.1×10^7
Fast Non-Iterative (FSR)	1.87	1	2.1×10^7

4.5.3.3 Over-Determined Filter Comparison

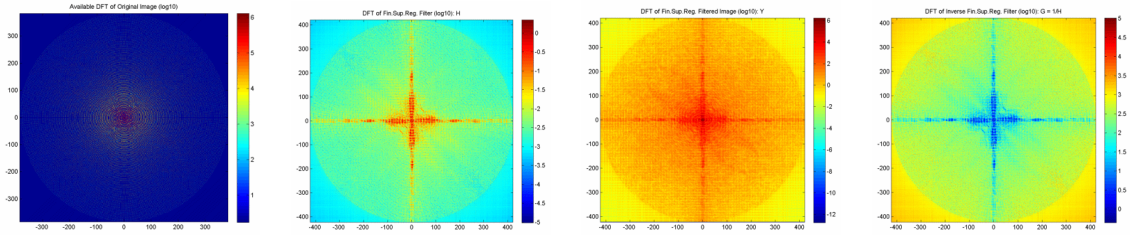
To reduce the overdetermined factor of the known 2-D DFT samples X_{data} over the number of solution pixels in x, the number of zero 2-D DFT values $H(k_1, k_2)$ is set to be 549,277 instead of 410,881. The number of unknown values of $h(i_1, i_2)$ still remains at $P^2 = (N - M + 1)^2 = 410,881$. The filter h is then overdetermined by a factor of 1.34 which will lead to approximate solutions. The number of known 2-D DFT samples is now 40,547 compared to 16,384 image pixels, making the reconstruction problem overdetermined by a reduced factor of 2.47 instead of 10.9. The less dense spiral pattern now has 44 revolutions and up to 512 angular samples per revolution shown in in Fig. 4.34(e).

Both the POCS filter in Fig. 4.34(b) and FSR filter in Fig. 4.34(f) again are each generated with 250 iterations with two $N \times N$ 2-D DFT's per iteration. The RMSE of the 2-D DFT locations of the filter that should be zero for the POCS filter is 3.31×10^{-3} and for the FSR filter is 2.76×10^{-3}. The RMSEs are close however from observing the 2-D DFTs of the filters in Fig. 4.34(b) and Fig. 4.34(f), the FSR filter has larger values that coincide with the spiral mask and the non-spiral values are more uniform than those of the POCS filter. This is again more evident in the 2-D DFT of the filtered images, Y, in Fig. 4.34(c) and Fig. 4.34(g) and in the inverse filters, $G = 1/H$, in Fig. 4.34(d) and Fig. 4.34(h).

As seen from Fig. 4.35-4.36, the inverse POCS-filtered image has the worst RMSE



(a) Log DFT of true image (b) Log DFT of POCS filter (c) Log DFT of POCS filtered true image (d) Log DFT of inverse POCS filter



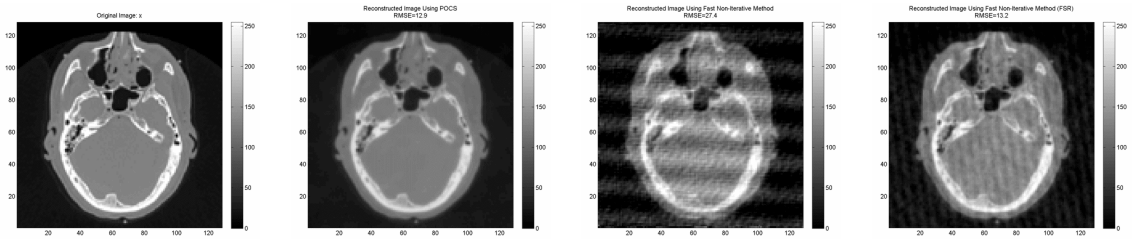
(e) Known log DFT of true image (f) Log DFT of FSR filter (g) Log DFT of FSR filtered true image (h) Log DFT of inverse FSR filter

Figure 4.34: Frequency data and **over-determined** POCS versus FSR filters.

Table 4.4: Fast Non-Iterative Reconstruction Error for Over-Determined Filters

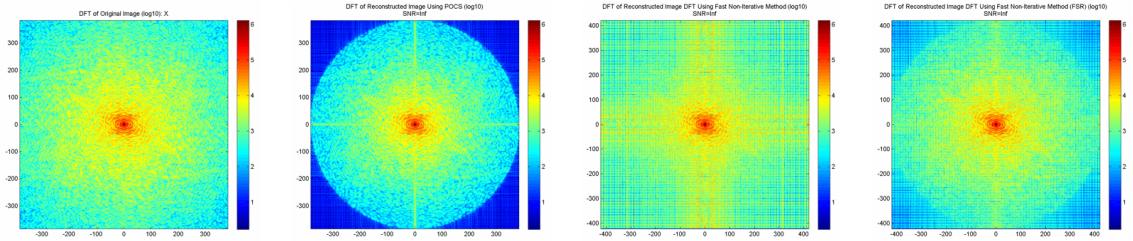
Reconstruction Method	RMSE	Iterations	Operations
Incomplete DFT	65.4	1	5.7×10^6
Cubic Interpolation	27.1	1	1.1×10^7
POCS RMSE-Matched	12.9	5	6.2×10^7
Fast Non-Iterative (POCS)	27.4	1	2.1×10^7
Fast Non-Iterative (FSR)	13.2	1	2.1×10^7

at 27.4 of the three images. The POCS solution method with an RMSE of 12.9 has lost some intensity and the inverse FSR-filtered image with an RMSE of 13.2 has some rippling artifacts. Both fast non-iterative could benefit from some regularization. However the aforementioned methods performed better than the incomplete-DFT and cubic interpolation methods shown in Fig. 4.37 and in Table 4.4. The fast non-iterative methods require about 3 times less operations than the POCS reconstruction method.



(a) True image (b) POCS solution image, RMSE=12.9 (c) Inverse POCS-filtered image, RMSE=27.4 (d) Inverse FSR-filtered image, RMSE=13.2

Figure 4.35: Reconstructed images using **over-determined** POCS versus FSR filters.



(a) True image DFT (b) POCS solution DFT (c) Inverse POCS-filtered DFT (d) Inverse FSR-filtered DFT

Figure 4.36: Log DFT of reconstructed images using **over-determined** POCS versus FSR filters.

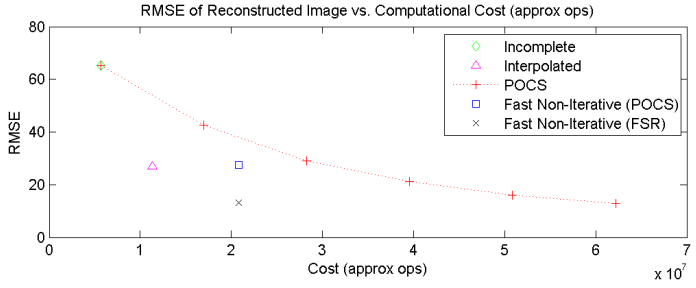
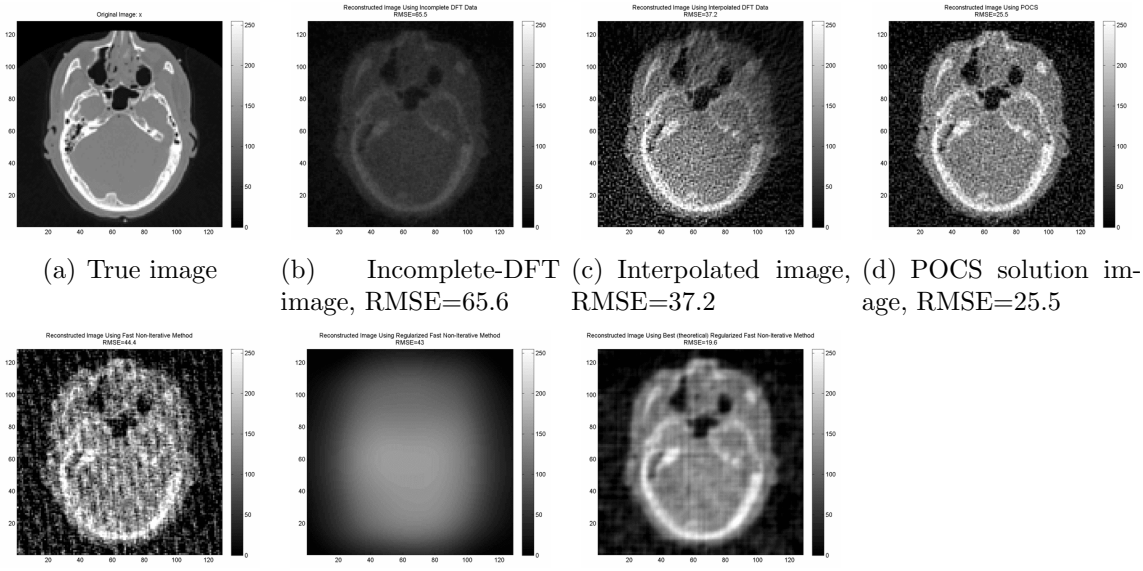


Figure 4.37: RMSE vs. approximate operations of reconstruction methods using **over-determined** POCS versus FSR filters.

4.5.3.4 Noisy Data

The settings from the overdetermined filter case from Section 4.5.3.3 is maintained except for the addition of $\sigma = 20N = 1.5\times10^4$ white Gaussian noise to the known 2-D DFT values resulting in a SNR of 6.28 dB.

A full comparison of all reconstruction methods is seen from Fig. 4.38-4.39. With the addition of noise to the measurement data, it can be seen that even the inverse FSR-filtered method in Fig. 4.38(e) regularization is needed. A second derivative regularization term that penalizes differences is employed. Automatically choosing the regularization parameter $\lambda = 7.5\times10^1$ leads to an over-regularized solution using the inverse FSR-filter in Fig. 4.38(f). The ideal best regularized inverse FSR-filter solution with $\lambda = 3.2\times10^{-2}$ is in Fig. 4.38(g) with the lowest RMSE as shown in Fig. 4.40.



(a) True image

(b) Incomplete-DFT image, RMSE=65.6

(c) Interpolated image, RMSE=37.2

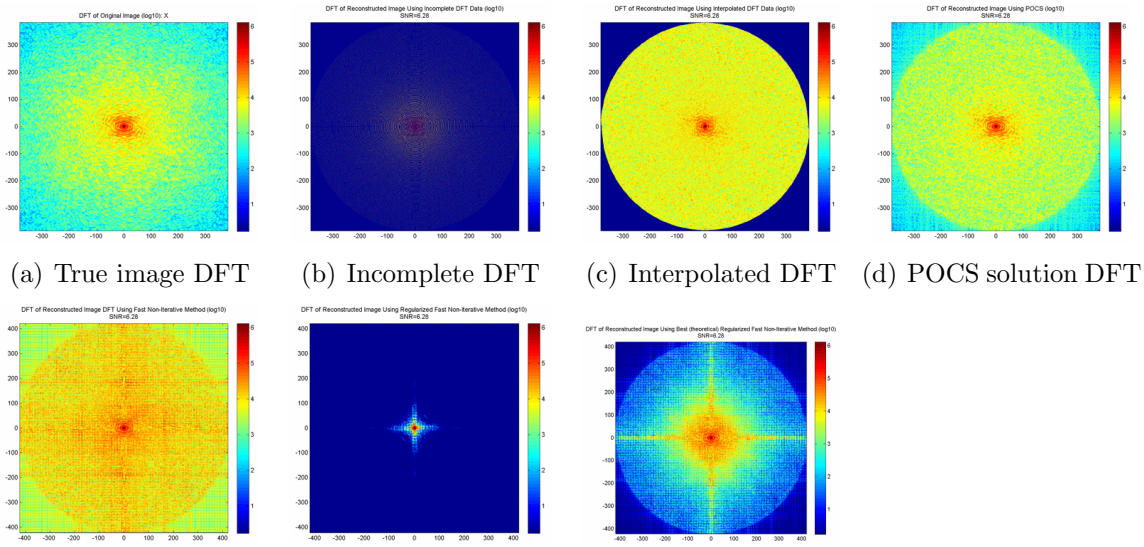
(d) POCS solution image, RMSE=25.5

(e) Inverse FSR-filtered image, RMSE=44.4

(f) Regularized inverse FSR-filtered image ($\lambda = 7.5\times10^1$), RMSE=43

(g) Best regularized inverse FSR-filtered image ($\lambda = 3.2\times10^{-2}$), RMSE=19.6

Figure 4.38: Noisy reconstructed images using an **over-determined** FSR filter.



(a) True image DFT (b) Incomplete DFT (c) Interpolated DFT (d) POCS solution DFT

(e) Inverse FSR-filtered DFT (f) Regularized inverse FSR-filtered DFT ($\lambda = 7.5\times10^{1}$) (g) Best regularized inverse FSR-filtered DFT ($\lambda = 3.2\times10^{-2}$)

Figure 4.39: Log DFT of noisy reconstructed images using an **over-determined** FSR filter.

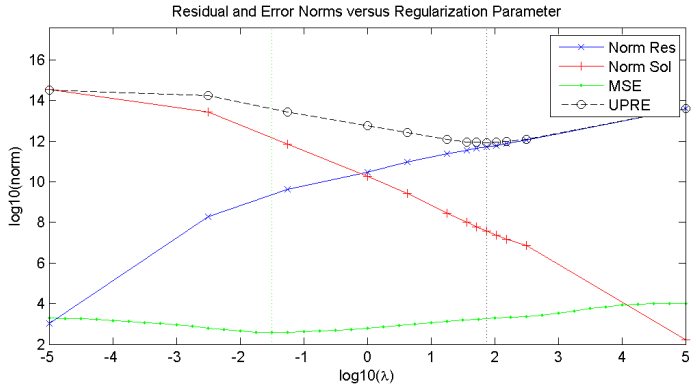


Figure 4.40: RMSE vs. regularization parameter λ of noisy reconstruction methods using an **over-determined** FSR filter.

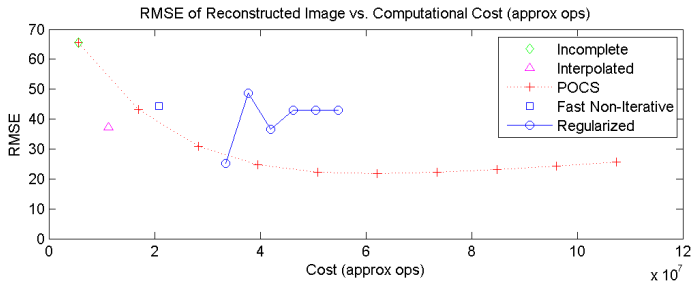


Figure 4.41: RMSE vs. approximate operations of noisy reconstruction methods using an **over-determined** FSR filter.

Table 4.5: Noisy Fast Non-Iterative Reconstruction Error for an Over-Determined FSR Filter

Reconstruction Method	RMSE	Iterations	Operations
Incomplete DFT	65.6	1	5.7×10^{6}
Cubic Interpolation	37.2	1	1.1×10^{7}
POCS RMSE-Matched	25.5	9	1.1×10^{8}
Fast Non-Iterative (FSR)	44.4	1	2.1×10^{7}
Reg. Fast Non-Iterative (FSR) ($\lambda = 7.5\times10^{1}$)	43.0	5	5.5×10^{7}
Best Reg. Fast Non-Iterative (FSR) ($\lambda = 3.2\times10^{-2}$)	19.6	1	2.9×10^{7}

4.5.4 Run Time Comparison

Run-time of the method developed in section is compared to that of a POCS iteration. The "CT head" image $x(i_1, i_2)$ has size $M \times M = 128 \times 128$. The Discrete Fourier Transform samples are measured along a square-inscribed spiral pattern fitted onto a $N \times N = 768 \times 768$ grid (where the ratio N/M is 6.00). The filter $h(i_1, i_2)$ is $P \times P = (N - M + 1) \times (N - M + 1) = 641 \times 641$ constructed using a finite-support regularized preconditioned conjugate gradient method. In summary, the goal is to deconvolve the solution x from the filtered image $y = h * *x$, where the 2-D DFT of the filtered image is the product of 2-D DFT of the filter and the known 2-D DFT data samples as such $Y = H \cdot X_{data}$ all of size $L \times L$ which is larger than N. The reconstruction was run on a 3.40 GHz Pentium 4 processor with 3GB of RAM.

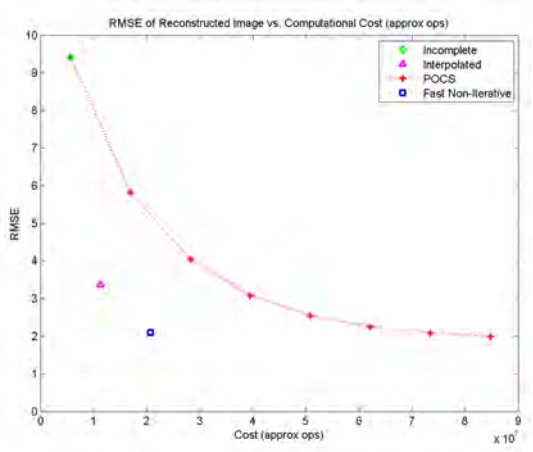
Fig. 4.42 shows the error of the reconstruction versus both the approximate operations and the more accurate computation time in seconds. From Table 4.6, we can see that the 1st iteration of POCS is composed of basically 3 $N \times N$ 2-D FFTs and takes 411ms. On the other hand, the "fast non-iterative" method requires 1 $N \times N$ 2-D FFT, 1 $L \times L$ element-wise division, and 2 $L \times L$ 2-D FFTs, and takes 688ms for when $L = 844$ and 484ms for when $L = 900$.

Solving the deconvolution problem of size $L \times L = 844 \times 844$ where $L = 1.1 \cdot N$ is just a scaled value of N takes the fast non-iterative method 688ms equivalent to a little more than 2 iterations of the POCS method. Detailed help documentation on Matlab's "fftn()" function states: "The execution time for FFT depends on the length of the transform. It is fastest for powers of two. It is almost as fast for lengths that have only small prime factors. It is typically several times slower for lengths that are prime or which have large prime factors." The prime factors of 844 are $[2, 2, 211]$ and hence the large 211-length FFT operation is slow. In comparison $N = 768$ has small prime factors of $[2, 2, 2, 2, 2, 2, 2, 2, 3]$.

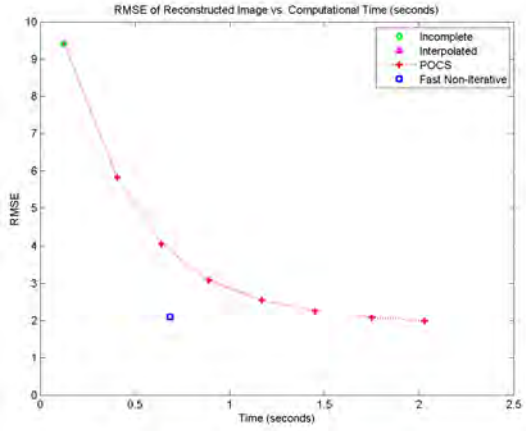
Therefore if we select L different from N but still factoring into small prime factors

we can speed up the $L \times L$ 2-D FFT. One arbitrary choice is $L = \lceil \sqrt{N} \rceil^2 = 900$ which has small prime factors $[2, 2, 3, 3, 5, 5]$. Now we can see in Fig. 4.42(d) that the "fast non-iterative" method is within 1 and 2 iterations of POCS, with a careful selection of size of the $L \times L$ 2-D FFT.

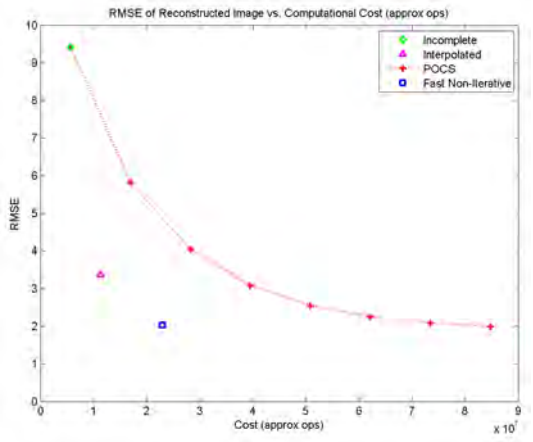
A side note is that the Delaunay triangulation based interpolation actually took much longer than approximated at about 30s.



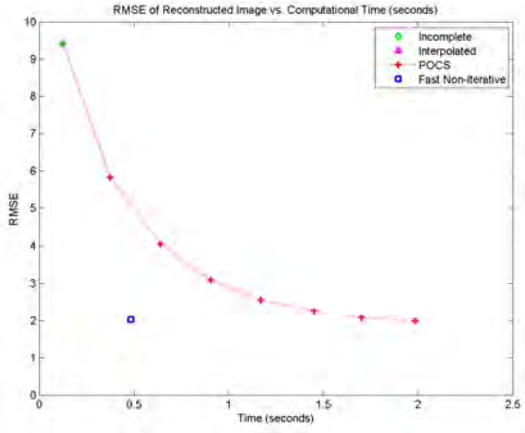
(a) Error vs approx. ops. ($L = 1.1 \cdot N = 844$)



(b) Error vs run time (seconds) ($L = 1.1 \cdot N = 844$)



(c) Error vs approx. ops. ($L = \lceil \sqrt{N} \rceil^2 = 900$)



(d) Error vs run time (seconds) ($L = \lceil \sqrt{N} \rceil^2 = 900$)

Figure 4.42: Error vs approximate operations and run times.

Table 4.6: Fast Non-Iterative Simulation Run Time Comparison

Reconstruction Method	Operation	Support RMSE	Number of Iterations (n)	Approximate Operations	Runtime (seconds)
Incomplete DFT		9.41	1	5.7×10^6	**0.125**
Cubic Interpolation		3.36	1	1.1×10^7	**29.7**
POCS (RMSE-Matched)		1.99	7	9.6×10^7	**2.04**
POCS (1 Iteration)		5.82	1	9.6×10^7	**0.411**
	$x = \text{ifft2}(X_{\text{data}})$				0.12
	$x = \text{real}(x \cdot h_{\text{mask}})$				0.01
	$X = \text{fft2}(x)$				0.11
	$X(\text{indData}) = X_{\text{data}}(\text{indData})$				0.03
	$x_{\text{rec}} = \text{ifft2}(X)$				0.12
Fast Non-Iterative (L=844)		2.09	1	2.1×10^7	**0.688**
Fast Non-Iterative (L=900)		2.03	1	2.1×10^7	**0.484**
	$Y = H \cdot X_{\text{data}}$				0.02
	$y = \text{ifft2}(Y)$				0.11
	$Y_L = \text{fft2}(y)$				0.10
	$G_L = 1/H_L$				0.09
	$X_L = Y_L \cdot G_L$				0.02
	$x_{\text{rec}} = \text{ifft2}(X_L)$				0.13

4.5.5 Actual CT Results (M=256)

4.5.5.1 Settings

Results using actual CT data, provided by Adam M. Alessio of the University of Washington and described in Section 3.6.3.1, are presented. A filtered back-projection reconstructed image of size 889×889 of a physical phantom using the whole parallel beam converted sinogram is used for reference.

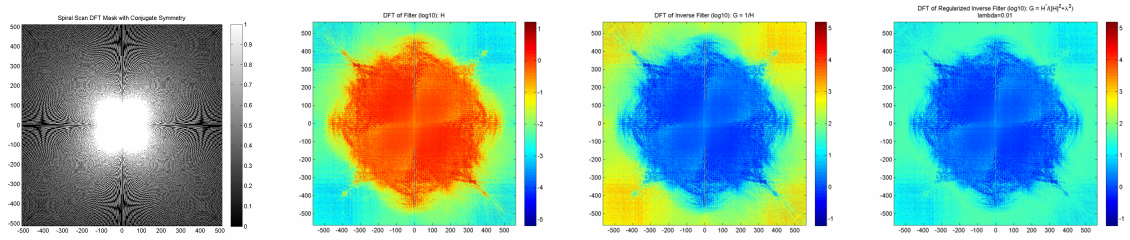
The size of the image $x(i_1, i_2)$ to reconstruct is $M\times M = 256\times256 = 65,536$. The 1-D DTFT samples of the rebinned parallel projections must fill a $N\times N = 1024\times1024$ 2-D DFT grid (where the ratio N/M is 4.00). Then the support of the filter $h(i_1, i_2)$ is $P\times P = (N-M+1)\times(N-M+1) = 769\times769 = 591,361$. Therefore the number of zeros in the $N\times N = 1024\times1024$ 2-D DFT of the filter $H(k_1, k_2)$ should also be P^2 to be just-determined. Then the size of the measured data or known 2-D DFT samples of $X_{data}(k_1, k_2)$ must be $N^2 - P^2 = 1024^2 - 769^2 = 457,215$, the number of samples that are not zeroed out by the filter.

Then 369 angularly uniformly spaced 1-D DTFTs of the projection slices, extended to the corners, sufficiently nearest neighbor fills 453,709 locations of the DFT grid shown in Fig. 4.43(a). This makes the filter slightly over-determined by a factor of

1.01. The filter $h(i_1, i_2)$, shown in Fig. 4.43(b), is constructed using the finite-support regularized preconditioned conjugate gradient method.

Because the DTFT of the projections are circularly band-limited in their measurements, the unfilled locations are set to zero. The conjugate symmetric set of samples $X_{data}(k_1, k_2)$ is used in the reconstruction, assuming the solution is real. The $N \times N$ grid is 43% filled with known values. The problem is overdetermined by $453,709/M^2 = 6.92$.

In summary, the goal is to deconvolve the solution x from the filtered image $y = h**x$, where the 2-D DFT of the filtered image is the product of 2-D DFT of the filter and the known 2-D DFT data samples as such $Y = H \cdot X_{data}$. Deconvolution is performed by multiplication of Y by the inverse filter $G = 1/H$, shown in Fig. 4.43(c), or the regularized inverse filter $G_{reg} = H^*/(|H|^2 + \lambda^2)$, shown in Fig. 4.43(d), where $\lambda = 1 \times 10^-2$.



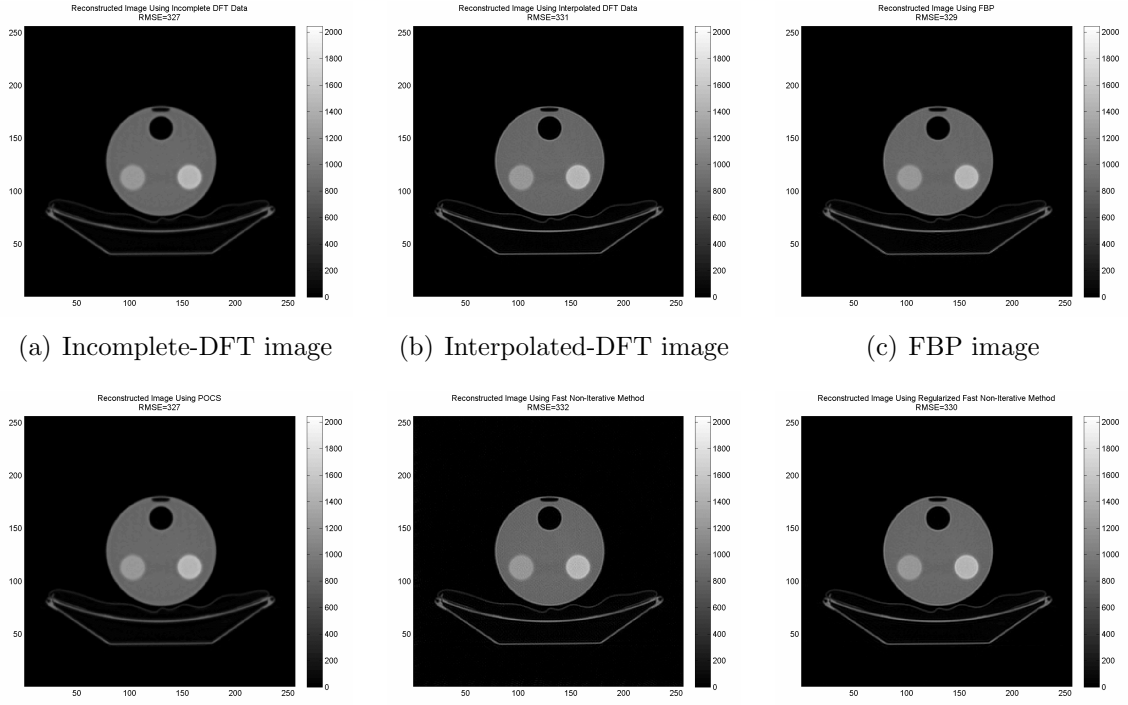
(a) Mask of available 2-D DFT samples (b) Log DFT of filter, H (c) Log DFT of inverse filter, G (d) Log DFT of regularized inverse filter, G_{reg}

Figure 4.43: Actual CT data frequency mask and filters for M=256 and N=1024.

4.5.5.2 Fast Non-Iterative Reconstruction

The filtered image $y(i_1, i_2)$ is transformed into the frequency domain with an $L \times L = 1128 \times 1128$ 2-D FFT where $L = 1.1 \cdot N$, deconvolved with the inverse filter $G(k_1, k_2)$ or the regularized inverse filter $G_{reg}(k_1, k_2)$, and finally an inverse 2-D FFT transforms it back to a $L \times L$ zeropadded solution where the $M \times M$ reconstructed image $x_{rec}(i_1, i_2)$ is extracted. The fast non-iterative methods are compared

with an incomplete-DFT inverse 2-D DFT method, a linear interpolation method, non-windowed filtered back-projection method, and the projection onto convex sets method (for 3 iterations) as shown in Fig. 4.44. The runtime comparisons shown in Table 4.7 were measured on a workstation with a 3.40GHz Pentium4 CPU with 3GB of RAM.



(a) Incomplete-DFT image (b) Interpolated-DFT image (c) FBP image

(d) POCS image (e) Inverse filtered image (f) Regularized inverse filtered image

Figure 4.44: Actual CT reconstructed images in tight support for M=256 and N=1024.

Fig. 4.45 shows the in finer detail the reconstruction artifacts of each method. The incomplete-DFT reconstruction in Fig. 4.45(a) is blurry due to lacking higher frequency information in its DFT as seen in Fig. 4.46(a), yet it runs the fastest at 0.234 seconds shown in Table 4.7. The interpolated-DFT reconstruction in Fig. 4.45(b) has sharp edges yet some mild ringing is present around the edges most likely due to the sharp circular windowing in its DFT shown in Fig. 4.46(b), and it is slow taking 51 seconds. The non-windowed FBP reconstruction in Fig. 4.45(c) also has

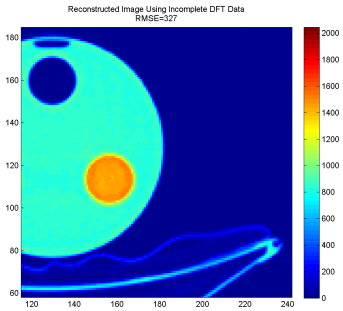
Table 4.7: Fast Non-Iterative Actual Data Runtime Comparison with M=256 and N=1024.

Reconstruction Method	Number of Iterations (n)	Approximate Operations	Runtime (sec)
Incomplete DFT	1	1.1×10^7	0.234
Cubic Interpolation	1	2.1×10^7	51.0
FBP	1	4.4×10^9	72.5
POCS	3	5.2×10^7	1.36
Fast Non-Iterative	1	3.9×10^7	1.16
Reg. Fast Non-Iterative	1	9.3×10^7	1.19

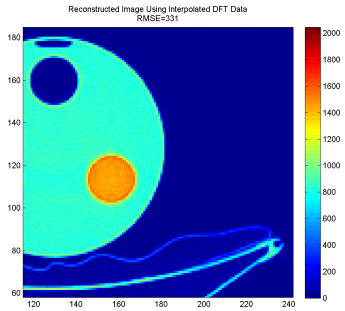
sharp edges and no ringing as the whole DFT is reconstructed shown in Fig. 4.46(c), yet it is the slowest taking 72.5 seconds.

The 3-iteration POCS reconstruction in Fig. 4.45(d) is blurry most likely to the early termination yet even though its DFT can be seen to be circularly band-limited like the interpolated DFT, no ringing exists due to a smoother windowing effect of the early termination. The 3 iterations of POCS is fast taking 1.36 seconds yet a sharper reconstruction would require many more iterations. The fast non-iterative reconstruction in Fig. 4.45(e) has sharp edges but speckled noise is visible due to amplification of noise in the higher frequencies as seen in Fig. 4.46(e). The regularized fast non-iterative reconstruction in Fig. 4.45(f) has sharp edges and suppressed noise as seen in higher frequencies in Fig. 4.46(f), yet the regularization parameter must be carefully selected as was done in this case. Both fast non-iterative methods were fast taking 1.16 and 1.19 seconds respectively.

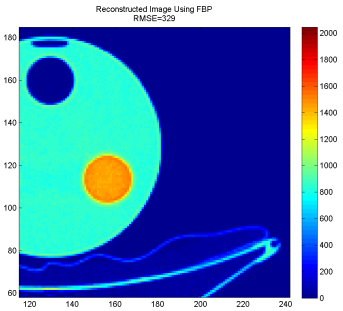
In conclusion, the regularized fast non-iterative method that used a filter (that took 6min to precompute offline) before and after deconvolution achieved the same quality of the FBP but with much less computation time. The advantage is once the frequency configuration is fixed then the filter can be reused for any new data.



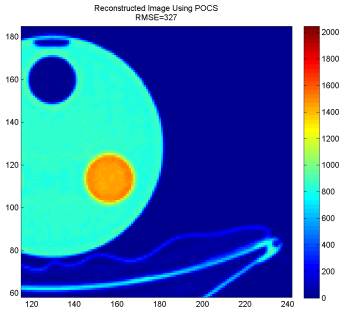
(a) Incomplete-DFT image



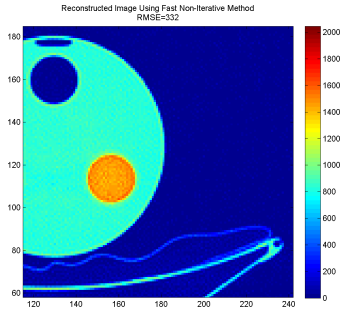
(b) Interpolated-DFT image



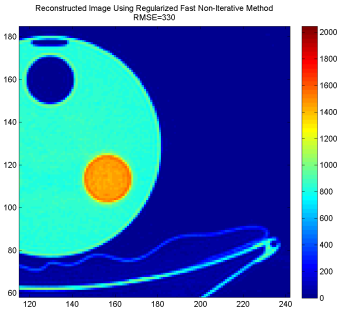
(c) FBP image



(d) POCS image

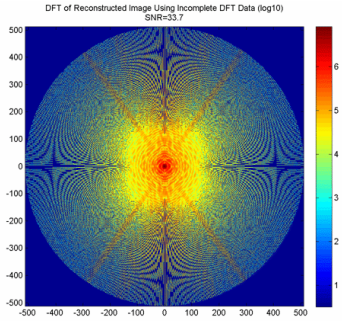


(e) Inverse filtered image

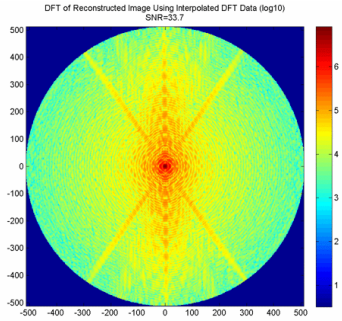


(f) Regularized inverse filtered image

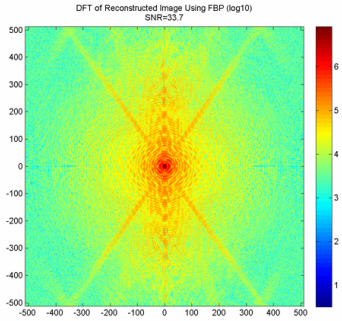
Figure 4.45: Zoomed and colormapped actual CT data reconstructed images for M=256 and N=1024.



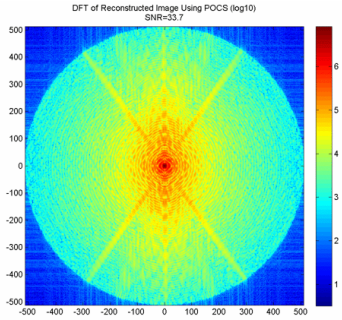
(a) Incomplete DFT



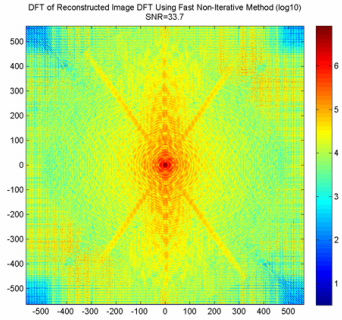
(b) Interpolated DFT



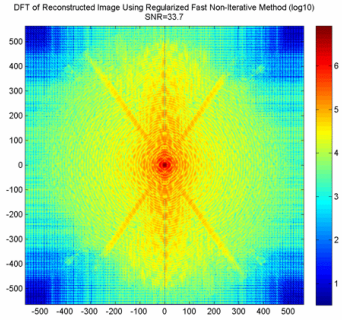
(c) DFT of FBP



(d) DFT of POCS



(e) DFT of inverse filtered



(f) DFT of regularized inverse filtered

Figure 4.46: 2-D DFTs of actual CT data reconstructed images in log scale for M=256 and N=1024.

4.5.6 Large Actual CT Results (M=512)

4.5.6.1 Settings

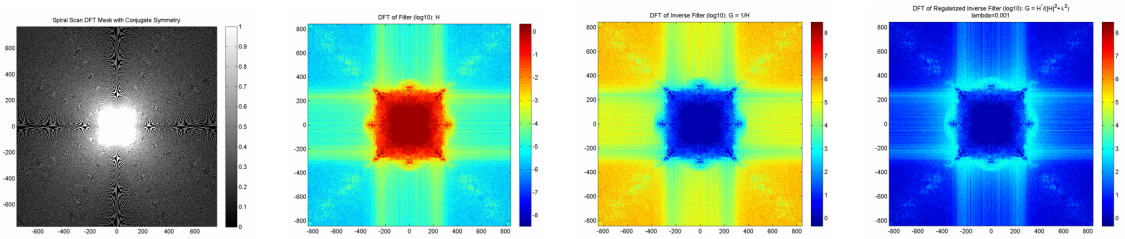
The larger size of the image $x(i_1, i_2)$ to reconstruct is $M \times M = 512 \times 512 = 262,144$. The 1-D Discrete-Time Fourier Transform samples of the rebinned parallel projections must fill a $N \times N = 1536 \times 1536$ 2-D DFT grid (where the ratio N/M is 3.00). Then the support of the filter $h(i_1, i_2)$ is $P \times P = (N - M + 1) \times (N - M + 1) = 1025 \times 1025 = 1,050,625$. Therefore the number of zeros in the $N \times N = 1536 \times 1536$ 2-D Discrete Fourier Transform of the filter $H(k_1, k_2)$ should also be P^2 to be just-determined. Then the size of the measured data or known 2-D DFT samples of $X_{data}(k_1, k_2)$ must be $N^2 - P^2 = 1536^2 - 1025^2 = 749,527$, the number of samples that are not zeroed out by the filter.

Then 385 angularly uniformly spaced 1-D DTFTs of the projection slices, extended to the corners, sufficiently nearest neighbor fills 749,527 locations of the DFT grid shown in Fig. 4.47(a). This makes the filter over-determined by a factor of 1.53. The filter $h(i_1, i_2)$, shown in Fig. 4.47(b), is constructed using the finite-support regularized preconditioned conjugate gradient method.

Again because the DTFT of the projections are circularly band-limited in their measurements, the unfilled locations are set to zero. The conjugate symmetric set of samples $X_{data}(k_1, k_2)$ is used in the reconstruction, assuming the solution is real. The $N \times N$ grid is 32% filled with known values. The problem is overdetermined by $749,527/M^2 = 2.86$.

In summary, the goal again is to deconvolve the solution x from the filtered image $y = h**x$, where the 2-D DFT of the filtered image is the product of 2-D DFT of the filter and the known 2-D DFT data samples as such $Y = H \cdot X_{data}$. Deconvolution is performed by multiplication of Y by the inverse filter $G = 1/H$, shown in Fig. 4.47(c), or the regularized inverse filter $G_{reg} = H^*/(|H|^2 + \lambda^2)$, shown in Fig. 4.47(d), where

$\lambda = 1 \times 10^{-3}$.



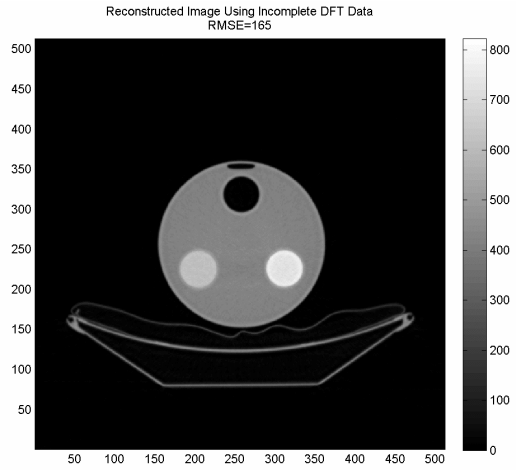
(a) Mask of available 2-D DFT samples (b) Log DFT of filter, H (c) Log DFT of inverse filter, G (d) Log DFT of regularized inverse filter, G_{reg}

Figure 4.47: Actual CT data frequency mask and filters with M=512 and N=1536.

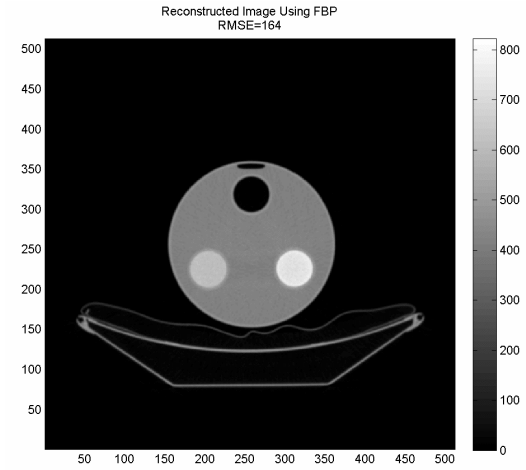
4.5.6.2 Fast Non-Iterative Reconstruction

The filtered image $y(i_1, i_2)$ is transformed into the frequency domain with an $L \times L = 1690 \times 1690$ 2-D FFT where $L = 1.1 \cdot N$, deconvolved with the inverse filter $G(k_1, k_2)$ or the regularized inverse filter $G_{reg}(k_1, k_2)$, and finally an inverse 2-D FFT transforms it back to a $L \times L$ zeropadded solution where the $M \times M$ reconstructed image $x_{rec}(i_1, i_2)$ is extracted. The fast non-iterative methods are compared with an incomplete-DFT inverse 2-D DFT method, a linear interpolation method, non-windowed filtered back-projection (FBP) method, and the projection onto convex sets (POCS) method (for 3 iterations) as shown in Fig. 4.48. The runtime comparisons shown in Table 4.8 were measured on a workstation with a 3.40GHz Pentium4 CPU with 3GB of RAM.

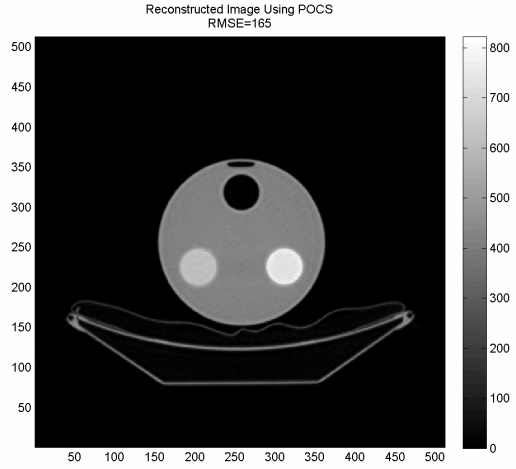
In conclusion, the regularized fast non-iterative method that used a filter (that took 13min to precompute offline) before and after deconvolution achieved satisfactory quality within that of the FBP but with much less computation time. The advantage is once the frequency configuration is fixed then the filter can be reused for any new data. The POCS method performed well because the first iteration is the same as the incomplete-DFT for which the overdetermined system produced a good reconstruction.



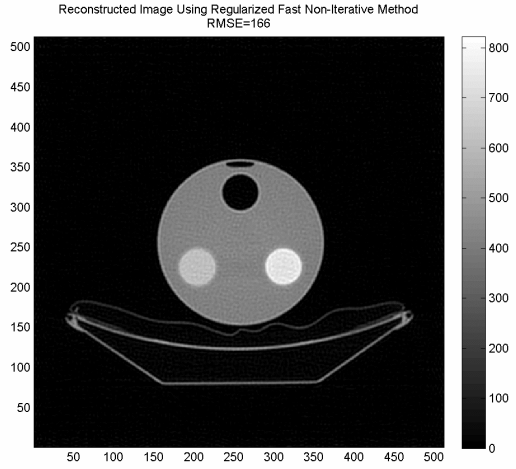
(a) Incomplete-DFT image



(b) FBP image

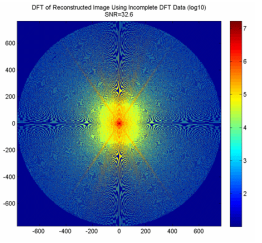


(c) POCS image

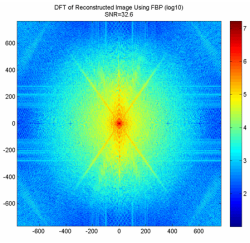


(d) Regularized inverse filtered image

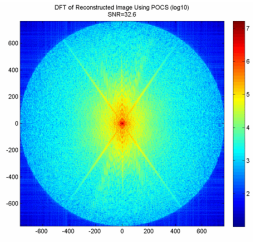
Figure 4.48: Actual CT data reconstructed images in tight support with M=512 and N=1536.



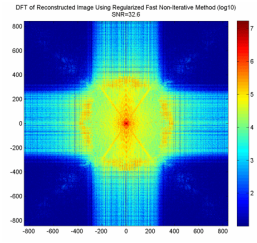
(a) Incomplete DFT



(b) DFT of FBP



(c) DFT of POCS



(d) DFT of regularized inverse filtered

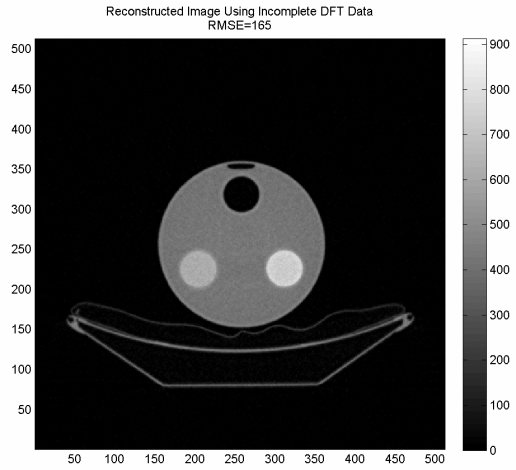
Figure 4.49: 2-D DFTs of actual CT data reconstructed images in log scale with M=512 and N=1536.

Table 4.8: Fast Non-Iterative Actual Data Runtime Comparison with M=512 and N=1536.

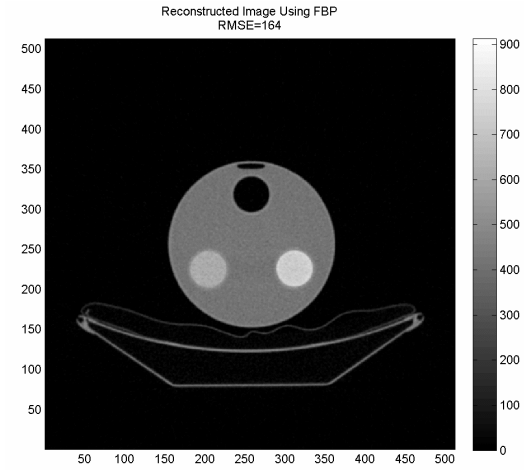
Reconstruction Method	Number of Iterations (n)	Approximate Operations	Runtime (sec)
Incomplete DFT	1	2.5×10^{7}	0.547
FBP	1	9.1×10^{10}	176
POCS	3	1.3×10^{8}	2.98
Reg. Fast Non-Iterative	1	2.2×10^{8}	2.25

4.5.6.3 Low-Noise Fast Non-Iterative Reconstruction

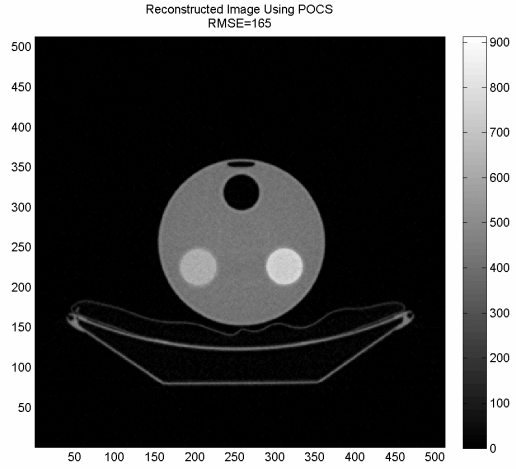
Real additive white Gaussian noise with a standard deviation of $1\times10^{3}\cdot N = 1.5\times10^{6}$ is added to the projection data leading to an SNR of 13.9 (dB). The regularization parameter for the fast non-iterative method is increased to 5×10^{-3}. Figures 4.50-4.51 show the reconstructions using low-noise data.



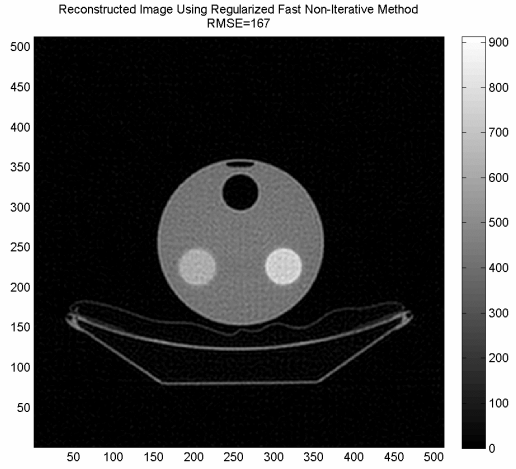
(a) Incomplete-DFT image



(b) FBP image

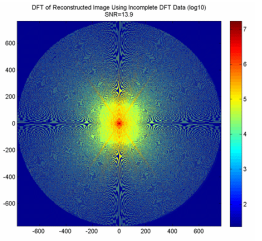


(c) POCS image

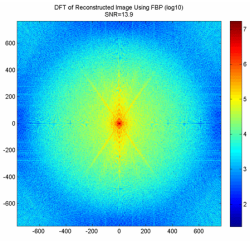


(d) Regularized inverse filtered image

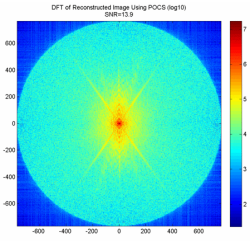
Figure 4.50: Low-noise actual CT data reconstructed images in tight support with M=512 and N=1536.



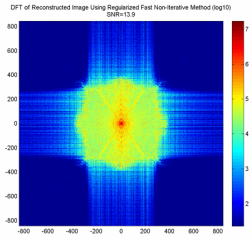
(a) Incomplete DFT



(b) DFT of FBP



(c) DFT of POCS

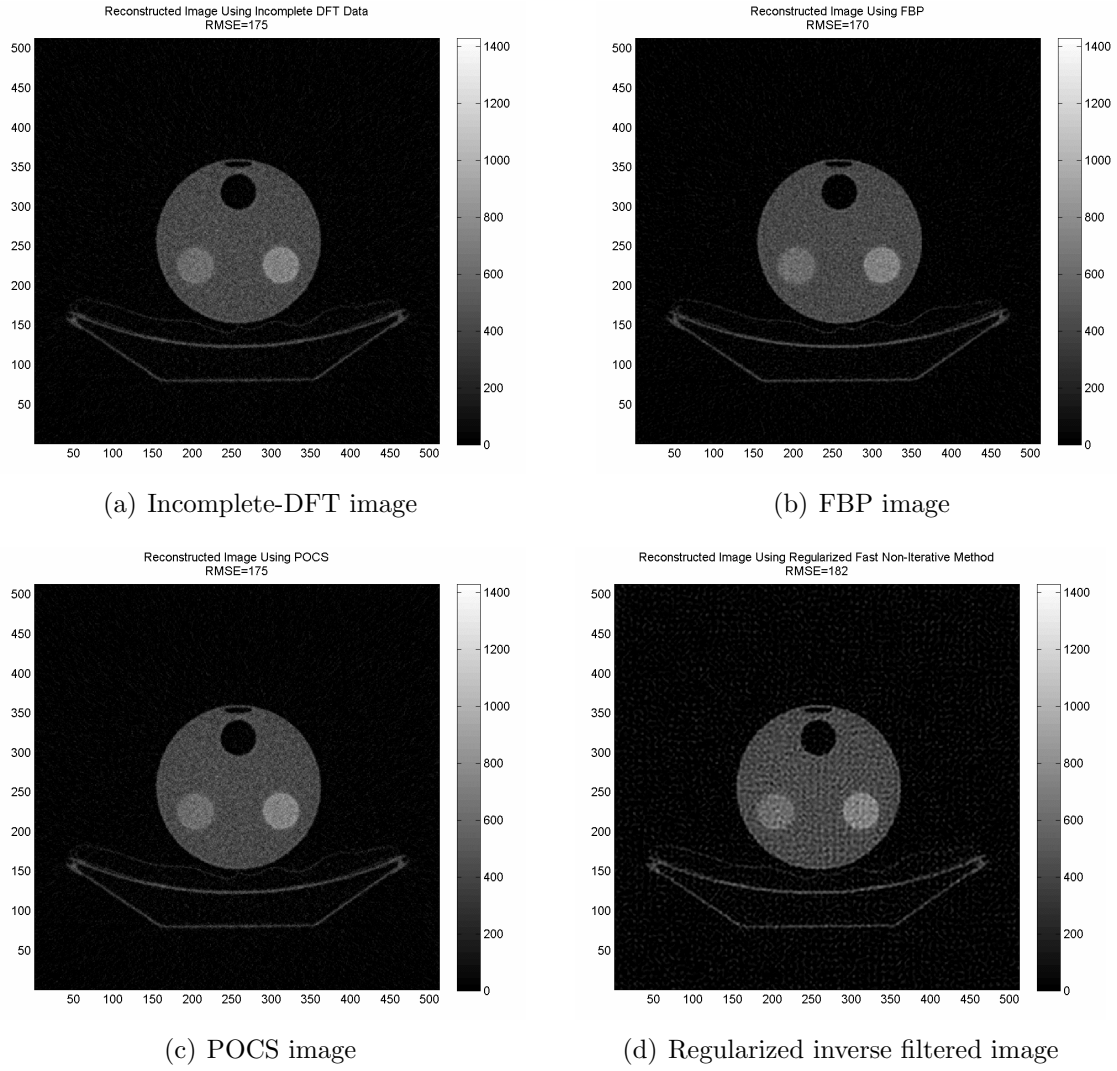


(d) DFT of regularized inverse filtered

Figure 4.51: 2-D DFTs of low-noise actual CT data reconstructed images in log scale with M=512 and N=1536.

4.5.6.4 High-Noise Fast Non-Iterative Reconstruction

Real additive white Gaussian noise with a standard deviation of $5\times10^3 \cdot N = 7.7\times10^6$ is added to the projection data leading to an SNR of 1.58 (dB). The regularization parameter for the fast non-iterative method is increased to 1×10^{-2}. Figures 4.52-4.53 show the reconstructions using high-noise data.



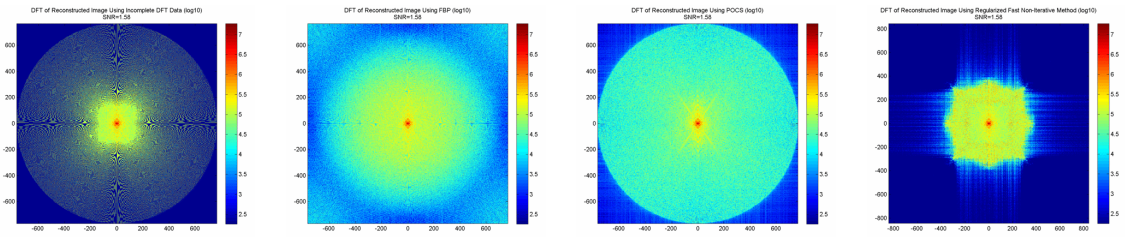
(a) Incomplete-DFT image

(b) FBP image

(c) POCS image

(d) Regularized inverse filtered image

Figure 4.52: High-noise actual CT data reconstructed images in tight support with M=512 and N=1536.



(a) Incomplete DFT (b) DFT of FBP (c) DFT of POCS (d) DFT of regularized inverse filtered

Figure 4.53: 2-D DFTs of high-noise actual CT data reconstructed images in log scale with M=512 and N=1536,

4.5.7 Larger Actual CT Results (M=888)

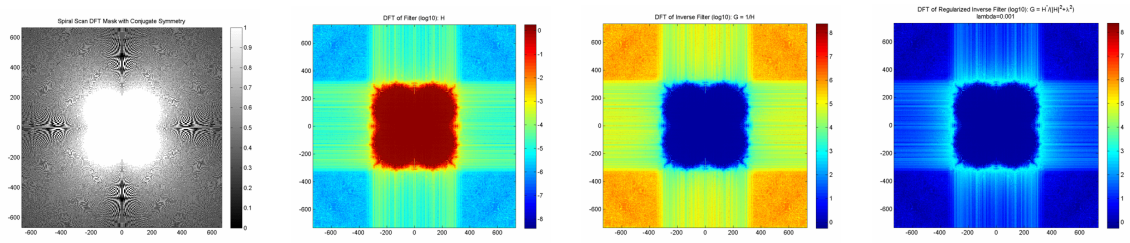
4.5.7.1 Settings

The size of the image to reconstruct is increased to $M \times M = 888 \times 888 = 788,544$ to match the resolution of the projection data. The 1-D Discrete-Time Fourier Transform samples of the rebinned parallel projections must fill a $N \times N = 1332 \times 1332$ 2-D Discrete Fourier Transform grid (where the ratio N/M is 1.50). Then the support of the filter $h(i_1, i_2)$ is $P \times P = (N - M + 1) \times (N - M + 1) = 445 \times 445 = 198,025$, smaller than the previous. Therefore the number of zeros in the $N \times N = 1536 \times 1536$ 2-D Discrete Fourier Transform of the filter $H(k_1, k_2)$ should also be P^2 to be just-determined. Then the size of the measured data or known 2-D DFT samples of $X_{data}(k_1, k_2)$ must be $N^2 - P^2 = 1332^2 - 445^2 = 1,034,271$, the number of samples that are not zeroed out by the filter.

Then 667 angularly uniformly spaced 1-D DTFTs of the projection slices, extended to the corners, sufficiently nearest neighbor fills 739,953 locations of the DFT grid shown in Fig. 4.54(a). This makes the filter over-determined by a factor of 3.74. The filter $h(i_1, i_2)$, shown in Fig. 4.54(b), is constructed using the finite-support regularized preconditioned conjugate gradient method.

Because the DTFT of the projections are circularly band-limited in their measurements, the unfilled locations are set to zero. The conjugate symmetric set of

samples $X_{data}(k_1, k_2)$ is used in the reconstruction, assuming the solution is real. The $N \times N$ grid is 58% filled with known values. The problem is overdetermined by $1,034,271/M^2 = 1.31$. The regularization parameter is set as $\lambda = 1 \times 10^{-3}$.



(a) Mask of available 2-D DFT samples (b) Log DFT of filter, H (c) Log DFT of inverse Filter, G (d) Log DFT of regularized inverse filter, G_{reg}

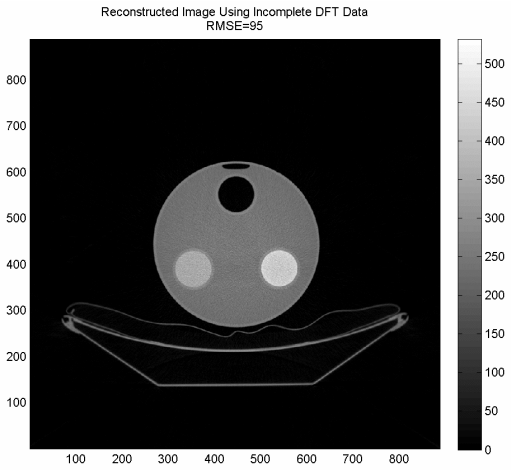
Figure 4.54: Actual CT data frequency mask and filters with M=888 and N=1332.

4.5.7.2 Fast Non-Iterative Reconstruction

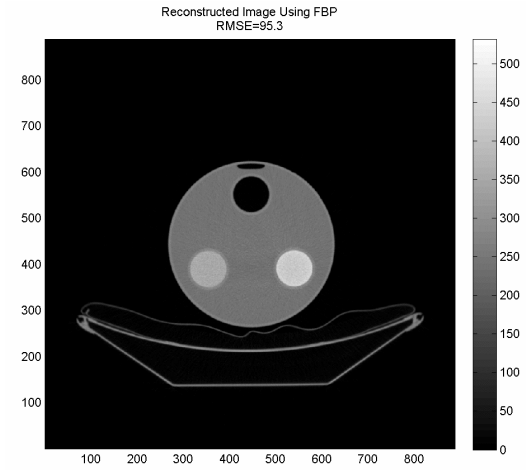
Figures 4.55-4.56 show the larger reconstructions yet with relatively smaller 2-D DFT grid as specified by the smaller grid size N.

Table 4.9: Fast Non-Iterative Actual Data Runtime Comparison with M=888 and N=1332.

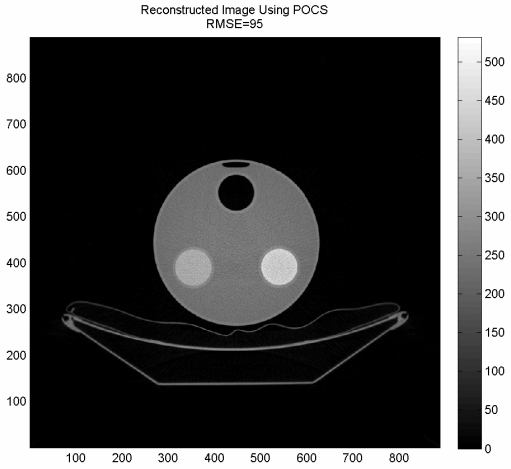
Reconstruction Method	Number of Iterations (n)	Approximate Operations	Runtime (sec)
Incomplete DFT	1	1.8×10^7	0.781
FBP	1	9.6×10^9	233
POCS	3	9.2×10^7	3.56
Reg. Fast Non-Iterative	1	1.6×10^8	4.73



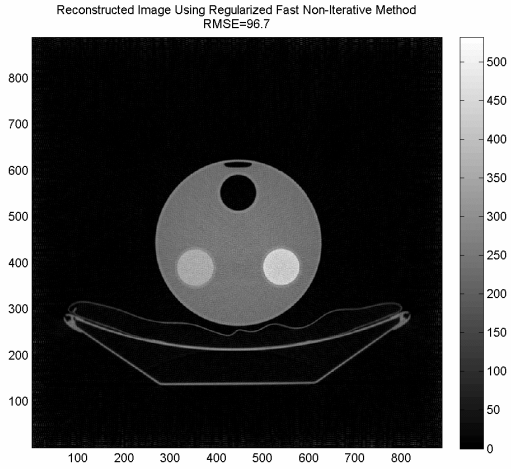
(a) Incomplete-DFT image



(b) FBP image

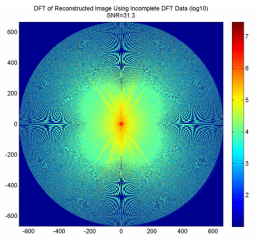


(c) POCS image

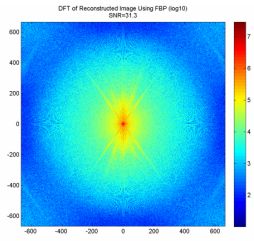


(d) Regularized inverse filtered image

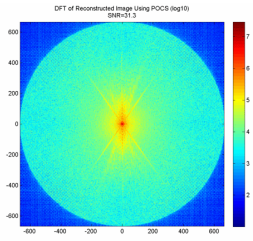
Figure 4.55: Actual CT data reconstructed images in tight support with M=888 and N=1332.



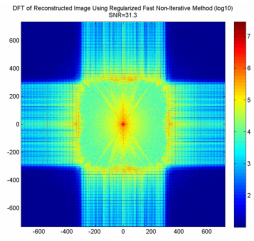
(a) Incomplete DFT



(b) DFT of FBP



(c) DFT of POCS



(d) DFT of regularized inverse filtered

Figure 4.56: 2-D DFTs of actual CT data reconstructed images in log scale with M=888 and N=1332.

CHAPTER V

Divide-and-Conquer Approach and Results

This section presents a divide-and-conquer method that divides the frequency domain into subbands thus splitting a large image reconstruction problem into smaller ones.

We present a non-iterative algorithm for the reconstruction of an $(M \times M)$ image from a sufficient number of 2-D DTFT samples. The algorithm uses Gabor logons or equivalently Gabor filters, localized in space and frequency, to partition the problem into a set of smaller problems (divide-and-conquer), each of which is solved and then combined into the final reconstruction. The algorithm also can be used to obtain a low-resolution but unaliased reconstruction, and to regularize the problem by discarding sub-problems that are themselves ill-conditioned.

5.1 Introduction

5.1.1 Problem Statement

Once again the goal is to reconstruct an $(M \times M)$ discrete image $x(i_1, i_2)$ from some of the values of its 2-D Discrete-Time Fourier Transform in (2.1). We make no assumption about whether the image is complex-valued or real-valued. We formulate the problem for square image support; modification to non-square support is trivial. We assume that the frequency locations are already known.

5.1.2 New Approach

The approach used in this paper summarizes as follows (more details are provided in the next section):

1. Project the image onto an over-complete set of Gabor logons, with some overlap between subbands;
2. Each projection yields a self-contained finite-support image reconstruction subproblem from irregular 2-D DTFT samples, which are now in a subband of the original problem;
3. Each of these subproblems is solved separately, using any procedure. Any badly conditioned problem can be regularized or even discarded;
4. The 2-D DFT of the solution to each subproblem is computed, and the 2-D DFT of the Gabor logon is divided out where the latter is not close to zero;
5. The computed 2-D DFTs for each subband are combined to give the 2-D DFT of the original image, which is then computed with an inverse 2-D DFT.

Advantages of the new approach are as follows:

- A large problem is replaced with many smaller and similar problems (divide-and-conquer);
- Each subproblem can be regularized independently, depending on its conditioning. Poorly conditioned or underdetermined subproblems (not enough frequency samples in that subband) can be discarded altogether, regularizing the overall problem;
- An *unaliased* low-resolution image can be reconstructed using the lowest-frequency subband. This may be sufficient for recognition in some applications.

5.2 Divide Step: Gabor Logons

A Gabor logon is essentially a modulated Gaussian

$$\phi(t, t_o, \omega_0) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(t-t_o)^2/(2\sigma^2)} e^{j\omega_o t} \tag{5.1}$$

Its exponential-squared drop-off in both time t and frequency ω means that the Gabor logon has virtually compact support in both time and frequency centered in time at $t = t_o$ and frequency at $\omega = \omega_o$. A Gaussian is used since it is most heavily concentrated in time and frequency; a parameter σ^2 trades off concentration in time and frequency.

Gabor logons are common choices for window function in the Short-Time Fourier Transform (STFT). However we are *not* performing a time-frequency decomposition of the image reconstruction from irregular Fourier samples problem, since no explicit time-frequency representation interpretation is necessary.

Here we make the following changes to the Gabor logon basis function:

- We use discrete-time n instead of continuous-time;
- We use DTFT instead of Fourier transform to define its spectrum;
- We *truncate* in time n so that it has finite support;
- Despite these changes the spectrum has essentially compact support.

Empirically we have observed that the basis function

$$\phi(n, k) = \begin{cases} 0.9^{n^2/k} & for |n| \leq 6\sqrt{k} \\ 0 & for |n| > 6\sqrt{k} \end{cases} \tag{5.2}$$

works well as a $(1/2)^k$-band filter (e.g., $k = 0 \rightarrow$ half-band filter). Note that since its drop-off is exponential-squared in time, the duration of basis function $\phi(n, k)$ does *not* increase by a factor of k, but by $\sqrt{k}$. This means that the procedure of this paper

is more efficient for large k. Also note that $\phi(n,k)$ has duration $12\sqrt{k}+1$.

5.3 Conquer Step: Subband Subproblems

The 2-D DTFT of the projection having compact $(N \times N)$ support (here ** denotes 2-D convolution)

$$y(i_1,i_2) = x(i_1,i_2) * *\phi(i_1,k)\phi(i_2,k) \tag{5.3}$$

is essentially zero except for frequencies

$$|\omega_1|, |\omega_2| < \phi(1/2)^k \tag{5.4}$$

Downsampling $y(i_1,i_2)$ by k leads to the following *subband subproblem*:

- Reconstruct $N \times N$ subimage $y(ki_1, ki_2)$, where
- $N = (M + 12\sqrt{k})/2^k$ from
- Frequency samples $X(\omega_1,\omega_2)\Phi(\omega_1)\Phi(\omega_2)$;
- all in the original subband $|\omega_1|, |\omega_2| < \pi(1/2)^k$;
- now expanded to the full band $|\omega_1|, |\omega_2| < \pi$.

For example, let the original image be (128×128). Using $k = 3$, we decompose the original problem into $(2^k)^2 = (2^3)^2 = 64$ subproblems, each requiring reconstruction of an $(N \times N)$ subimage where

$$N = (128 + 12\sqrt{8})/2^3 = 20.2 \rightarrow N = 21 \tag{5.5}$$

from its 2-D DTFT frequency samples

$$X(8\omega_{1i}, 8\omega_{2i}), |\omega_1|, |\omega_2| < \pi/8 \tag{5.6}$$

in a subband of the original problem. This requires the solution of 64 linear systems with $21^2 = 441$ unknowns, instead of one with $128^2 = 16384$ unknowns.

5.4 Complete Procedure

1. Define $(2^k)^2$ subproblems as above, using *modulation* either to shift the subproblem to the origin, or to shift the basis function to the frequency subband;
2. Solve each subproblem separately;
3. Do this *in parallel* for each of the $(2^k)^2$ subproblems;
4. Compute an $(N \times N)$ 2-D DFT of the solution;
5. Divide this point-by-point by the 2-D DFT of the *downsampled* $\phi(n, k)$
6. Combine these $(2^k)^2$ 2-D DFTs (one for each of $(2^k)^2$ subbands) into the 2-D DFT of the overall image;
7. Use *overlapping* subbands so frequencies at which $\phi(n, k)$ is small can be discarded.

5.5 Results

5.5.1 Simulation Results

5.5.1.1 Settings

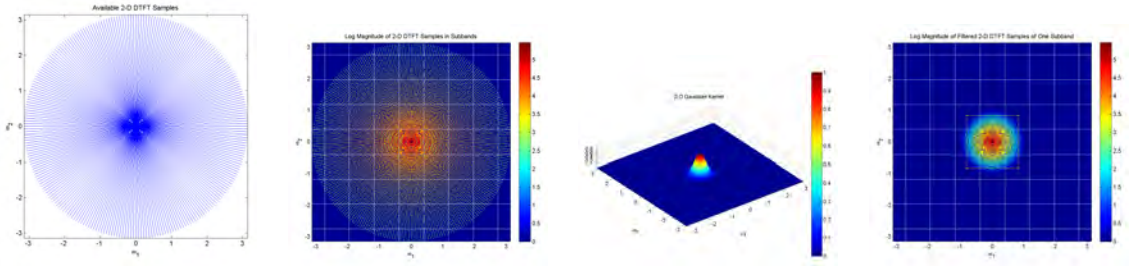
The "Shepp-Logan" head phantom image of size $M \times M = 128 \times 128$ is used first. The DTFT samples are measured on a CT pattern at 128 angles and 128 radial samples per angle as shown in Fig. 5.1(a). The problem is divided into $8 \times 8 = 2^K \times 2^K$ subbands where $K = 3$ as shown in Fig. 5.1(b). Therefore each subproblem to solve has size 16×16 however we employ overlapping of subbands by a factor of $\alpha_{overlap} = 2$, which increases the size of each subproblem to 32×32. The all-white Gaussian noise added to the DTFT samples has standard deviation of $\sigma_N = 5 \times 10^2$ resulting a SNR

of 25.7 dB.

The modified Gabor logon is a modulated 2-D Gaussian kernel

$$\phi(n_1, n_2; k_1, k_2, \sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(n_1^2+n_2^2)}{2\sigma^2}} e^{\frac{j2\pi(k_1 n_1 + k_2 n_2)}{2^K}}, \; k_1, k_2 = 0, ..., 2^K - 1 \qquad (5.7)$$

as shown in Fig. 5.1(c) where we have chosen $\sigma = \frac{1}{\alpha_{overlap}} 2^K$. The modified Gabor logon is used to convolve the image and isolate a subband shown in Fig. 5.1(d)



(a) CT Configuration of available 2-D DTFT samples (b) Known 2-D DTFT samples with subband boundaries (c) Modified Gabor logon or 2-D Gaussian kernel (d) 2-D DTFT samples projected onto an overlapped subband

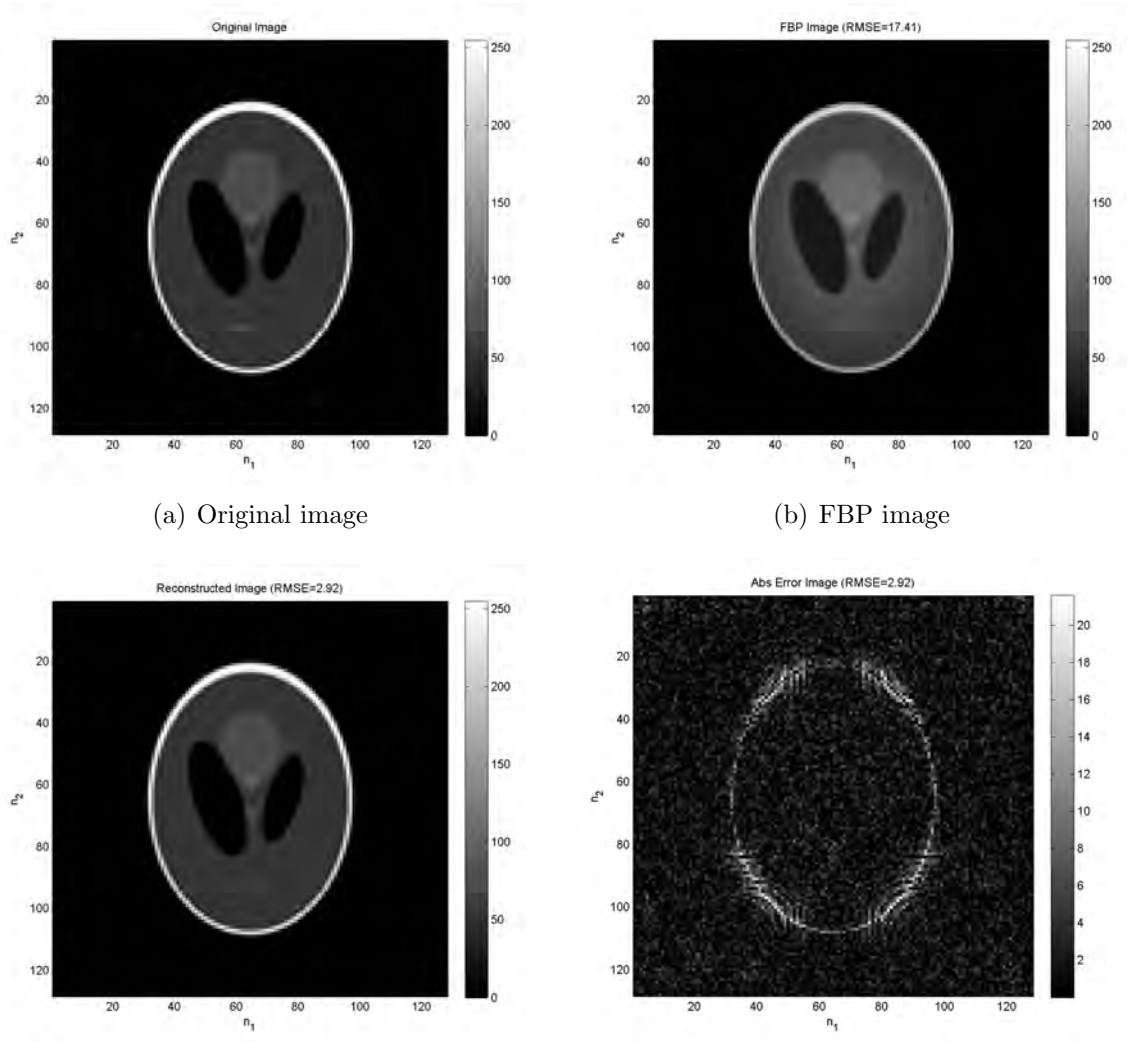
Figure 5.1: Subdivision strategy.

Given the possibly over-determined system matrix A and data vector b in normal equation form, each subbanded problem is solved in the least squares sense using the QR decomposition to attain the solution vector x using Tikhonov regularization with regularization parameter λ. The comparison method of filtered back-projection (FBP) reconstruction uses cubic interpolation and the ramp filter (Ram-Lak) with no other lowpass filtering.

5.5.1.2 Simulation Data Reconstruction with Uniform Regularization

We solve the reconstruction problem with a regularization parameter of $\lambda = 1 \times 10^1 = 10$ applied uniformly to all subproblems. The subband decomposition reconstructed image, in Fig. 5.2(c), has an RMSE of 2.92, as shown in Fig. 5.2(d), and took a total of 100 seconds, at 1.56 seconds per subproblem. The benchmark

FBP reconstruction, in Fig. 5.2(b), has an RMSE of 17.4 with an artifact of a brighter center and took 1.61 seconds. Their respective 2-D DFTs in log scale are shown in Fig. 5.3.



(a) Original image

(b) FBP image

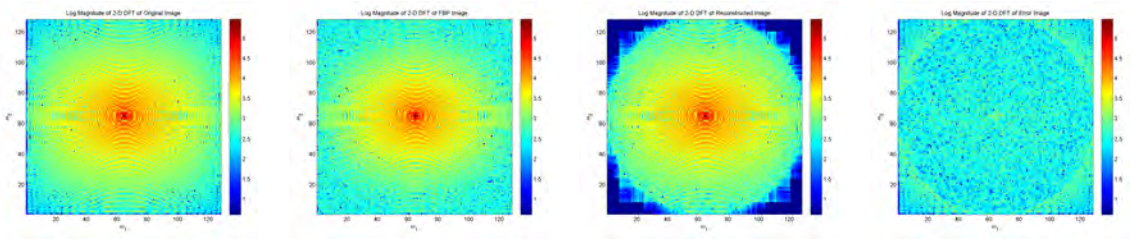
(c) Subband decomposition image

(d) Error of the subband decomp. image

Figure 5.2: Reconstructed images using uniform regularization of subproblems.

5.5.1.3 Noisy Simulation Data Reconstruction with Variable Regularization

We solve the reconstruction problem with a regularization parameter applied variably to the subproblems starting at the DC subproblem with $\lambda = 1 \times 10^1 = 10$ and

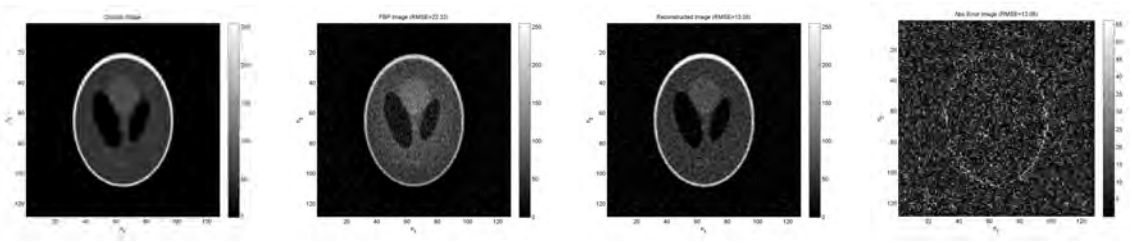


(a) DFT of original image (b) DFT of FBP image (c) DFT of subband decomposition image (d) DFT of error of the subband decomp. image

Figure 5.3: Log 2-D DFT of reconstructed images using uniform regularization of subproblems.

log-linearly increasing to $\lambda = 5 \times 10^1 = 50$ at the subproblem of the highest frequency. To emphasize the effectiveness of variable regularization the amplitude of the additive noise is increased to $\sigma_N = 5 \times 10^2$ and a resulting SNR of 5.74 dB.

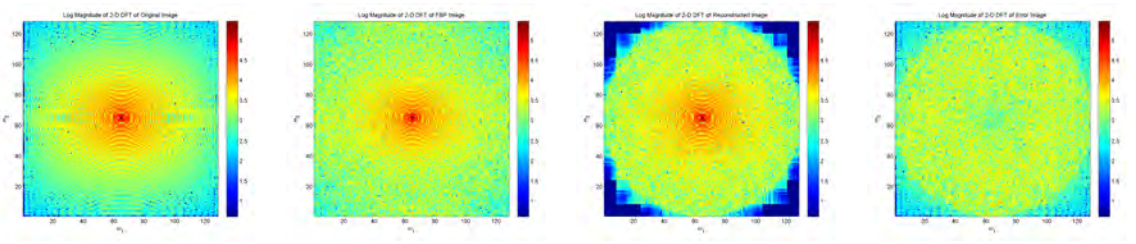
The subband decomposition reconstructed image using variable regularization and noisy data, in Fig. 5.4(c), has an RMSE of 13.08, as shown in Fig. 5.4(d), and took a total of 99 seconds, at 1.54 seconds per subproblem. The benchmark FBP reconstruction, in Fig. 5.4(b), has an RMSE of 22.3 and took 1.75 seconds. Their respective 2-D DFTs in log scale are shown in Fig. 5.5.



(a) Original image (b) FBP image (c) Subband decomposition image (d) Error of the subband decomp. image

Figure 5.4: Reconstructed images using variable regularization of subproblems.

Table 5.1 shows numerically the lower RMSE achieved using variable regularization over uniform regularization.



(a) DFT of original image (b) DFT of FBP image (c) DFT of subband decomposition image (d) DFT of error of the subband decomp. image

Figure 5.5: Log 2-D DFT of reconstructed images using variable regularization of subproblems.

Table 5.1: Divide-and-Conquer Reconstruction Errors using Variable versus Uniform Regularization

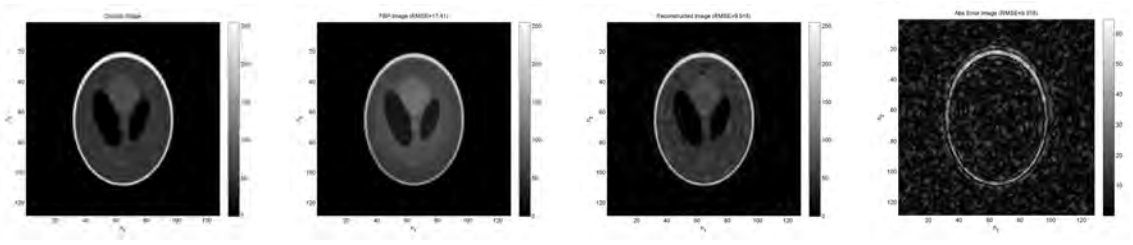
Type	λ_{low}	λ_{high}	RMSE
Uniform	5×10^{1}	5×10^{1}	16.31
Uniform	1×10^{1}	1×10^{1}	13.50
Uniform	2.5×10^{1}	2.5×10^{1}	13.22
Variable	1×10^{1}	5×10^{1}	13.08

5.5.1.4 Simulation Data Reconstruction with Enforced Just-Determined Subproblems

We solve the reconstruction problem with the previously used uniform regularization parameter and noise amplitude once again. However we illustrate how the computation of each subproblem can be sped up by reducing the data in each subproblem to be at most just-determined. In most cases, such as in CT, the subproblems near the origin (or DC) are highly over-determined. The following results show increased error with reduced computation time, yet the ratio of number of data samples to solution size can be adjusted to the desired amount.

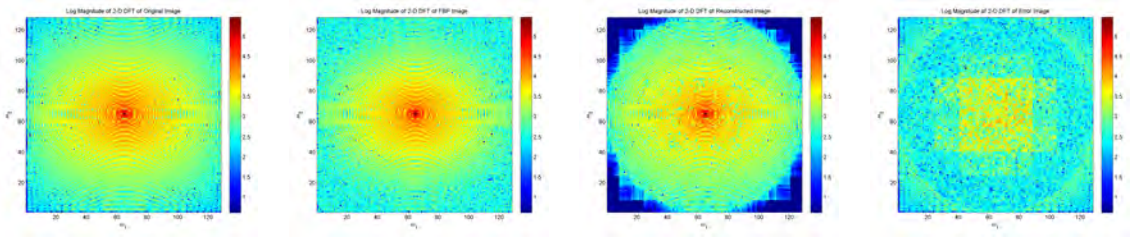
The subband decomposition reconstructed image using a cap on the data, in Fig. 5.6(c), has an RMSE of 9.92 (versus 2.92), as shown in Fig. 5.6(d), and took a total of 80 seconds (versus 100 seconds), at 1.25 seconds per subproblem (versus 1.54

seconds). The benchmark FBP reconstruction, in Fig. 5.6(b), has not changed with an RMSE of 17.4 and taking 1.61 seconds. Their respective 2-D DFTs in log scale are shown in Fig. 5.7.



(a) Original image (b) FBP image (c) Subband decomposition image (d) Error of the subband decomp. image

Figure 5.6: Reconstructed images using limited data in each subproblem.



(a) DFT of original image (b) DFT of FBP image (c) DFT of subband decomposition image (d) DFT of error of the subband decomp. image

Figure 5.7: Log 2-D DFT of reconstructed images using limited data in each subproblem.

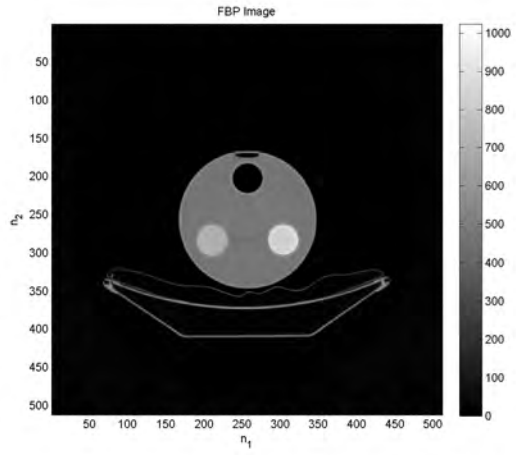
5.5.2 Actual CT Results

Results using actual CT data, provided by Adam M. Alessio of the University of Washington and described in Section 3.6.3.1, are presented. The actual CT data used has been downsampled by a factor of 2 so there are 205 angles and 445 bins. The reconstruction image size is $M \times M = 512 \times 512$. The problem is divided into $32 \times 32 = 2^K \times 2^K$ subbands where $K = 5$. Therefore each subproblem to solve has size 16×16 however we employ overlapping of subbands by a factor of $\alpha_{overlap} = 2$, which increases the size of each subproblem to 32×32. The all-white Gaussian noise

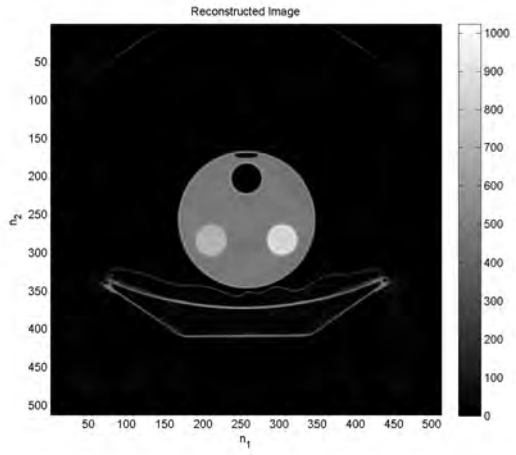
added to the DTFT samples has standard deviation of $\sigma_N = 5 \times 10^2$ resulting a SNR of 50.2 dB.

We solve the reconstruction problem with a regularization parameter of $\lambda = 1 \times 10^0 = 1$ applied uniformly to all subproblems. Each subband problem is solved using the preconditioned conjugate gradient method with a convergence tolerance of 1×10^{-4} and a maximum iteration of 10.

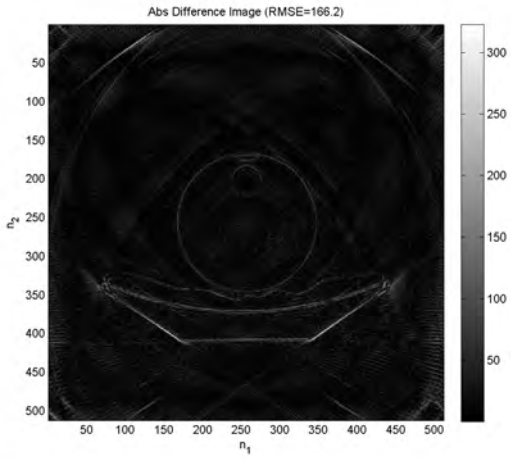
The subband decomposition reconstructed image is shown in Fig. 5.8(b) and took a total of 718 seconds (approximately 12 minutes), at 0.70 seconds per subproblem. The benchmark FBP reconstruction, in Fig. 5.8(a), took 10.0 seconds. Their difference image is shown in Fig. 5.8(c) while their respective 2-D DFTs in log scale are shown in Fig. 5.9.



(a) FBP image

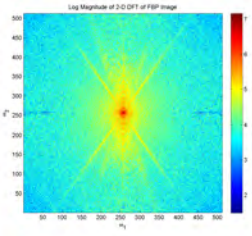


(b) Subband decomposition image

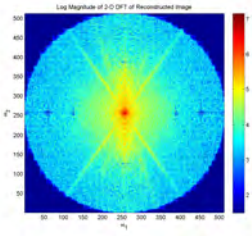


(c) Difference of the subband decomp. image

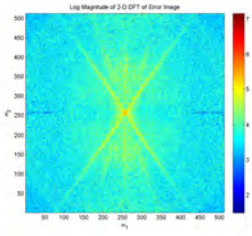
Figure 5.8: Divide-and-conquer reconstructed images using using actual CT data.



(a) DFT of FBP image



(b) DFT of subband decomposition image



(c) DFT of difference of the subband decomp. image

Figure 5.9: Log 2-D DFT of divide-and-conquer reconstructed images using actual CT data.

CHAPTER VI

Conclusions

6.1 Summary of Results

1. In conditioning approach in Chapter 3, using the Lagrange interpolation formula (2.16) for the just-determined case, I have shown a closed-form expression for the condition number (3.67) and for the upper bound of the condition number (3.70). The latter may be used in $O(M^2)$ time to measure the conditioning of a just-determined problem with high accuracy verified empirically by the correlation coefficient of 0.9. I have based the relationship between the condition number (3.67) and the variance sensitivity measure (3.75) by the insight that both measures increase when the distances between DTFT samples decrease. This relationship is also empirically verified by their correlation coefficient. For the over-determined case, the variance sensitivity measure performs well, verified empirically by the correlation coefficient of 0.8.

2. Also in the conditioning approach, I showed that if the Fourier samples are not constrained to intersections with radial lines, then perfect conditioning can be achieved which non-rectangular but regular frequency configurations are made clearer once unwrapped from 2D to 1D.

3. The non-iterative DFT-based image reconstruction is a new idea that presents

a non-iterative reconstruction algorithm that deconvolves a precomputed filter, zeroed out at the unknown 2-D DFT locations, from a filtered 2-D DFT set of data. This method is made fast by implementing the DFTs with the FFT algorithm. The construction of the filter requires it to satisfy some restrictions which can be approximated using POCS and hence the image will be an approximation. An improvement upon the filter creation is to use the finite-support regularization method to create better filters which converge faster in generating this filter than POCS.

4. The divide-and-conquer image reconstruction is also a new idea that reduces a large image reconstruction problem into smaller ones. The decomposition and the recombination of the subproblems worked nearly seamlessly. Improving the overall conditioning by applying the regularization of each subproblem separately proved to be successful in cases of large amounts of noise. Also limiting the data size per subproblem allowed a trade-off between computation speed and accuracy.

5. All three solution methods were performed on actual CT data to verify that our methods will find solutions to a real world application.

6.2 Evaluation of Results

6.2.1 Conditioning Approach

1. In the conditioning approach, in most cases the just-determined problem is poorly conditioned and hence we need to solve the over-determined problem. For the over-determined case, the Lagrange interpolation formula no longer applies as the closed form solution to the normal equation. Therefore only the upper-bound on the condition number is no longer valid. Therefore the variance sensitivity measure is left to estimate the conditioning.

2. Also I presented 3 methods for unwrapping the 2-D problem into 1-D. The Good-Thomas FFT approach had its limitations in that the Fourier samples must lie on a rectangular grid. The rotated-support Kronecker substitution relaxed the restriction of the Fourier sample locations to diagonal lines yet the spatial support was a diamond shape. The most convenient variant of this is the helical scan FFT which has a rectangular support and the Fourier samples must lie on just slightly diagonal lines.

3. When the situation allows for the selection of frequency sample locations we may use our variance measure to reduce the sensitivity. Simulated annealing is the global optimization algorithm which uses the variance measure to find the frequency configuration. When there are no constraints on the frequency locations such as non-tomographic applications, then perfect conditioning approaches maybe used to eliminate noise amplification in the problem.

6.2.2 Non-Iterative Approach

1. The non-iterative approach is recommended for when the frequency sample locations are already determined and fixed for multiple uses as we precompute this methods filter once. The speed advantage comes in this precomputation and the use of only 2-D FFTs. Though as inverse filtering and regularization is used in computing the quick solution, the solution in many cases can be biased.

2. The relative importance of the non-iterative approach is that this method can be applied to arbitrary frequency sampling configurations and not just to the CT configurations as shown in the results section. The specification of the data masking filter does not prescribe the locations of the data samples but only on the total numbers, which can be relaxed at the cost of the accuracy of the zero-masking ability of the filter.

3. The shape of the frequency response of the data masking filter in the non-iterative method can be interpreted in the following manner. The frequency response of the filter, H, zeros out DTFT values that are unknown and hence nearby values tend to be small whereas the non-zero filter frequency response values and their neighboring values are larger. When the data sampling configuration is dense near the center at DC and sparse at the high frequency locations such as in tomography, the data masking filter will be nonzero near DC and closer to zero further out. While the inverse data masking filter is then not exactly a cone filter as in the case of filtered backprojection, the inverse filter is a highpass filter. As in FBP, regularization of the inverse filter has similar effects as apodization.

4. The effects of filtering the incomplete DTFT data samples followed by deconvolution can be reinterpreted as an interpolation procedure yet with a large kernel as a function of the data masking filter, its accompanying filter specifications, and possible regularization. The reinterpretation can be seen more clearly from further evaluating the formulation (4.39) and masking use of the convolution-multiplication duality of the Fourier transform as such:

$$\begin{aligned} \hat{x}_M &= I_{crop} F_L^{-1} \cdot G_L \cdot F_L I_{pad} F_N^{-1} \cdot \tilde{H}_N \tilde{X}_N && (6.1) \\ &= I_{crop} F_L^{-1} F_L \cdot g_L \cdot \tilde{h}_L \cdot I_{pad} F_N^{-1} \tilde{X}_N && (6.2) \\ &= I_{crop} \cdot (g * \tilde{h})_L \cdot \tilde{x}_L && (6.3) \end{aligned}$$

where $\tilde{h}_L$ and g_L are circulant matrices that perform 2-D convolutions with the impulse responses of the forward and inverse data masking filters. The $\tilde{H}_N$ and its padded impulse response $\tilde{h}_L$ are noted that the zero value constraints are applied in the frequency domain and hence differ from the spatially $P \times P$

finite support h_L. Then $(g * \tilde{h})_L$ represents the "interpolation" kernel of the two filters spatially-convolved, which in the noisy model will include regularization, and $\tilde{x}_L$ is the padded incomplete-DFT reconstruction to be interpolated in the spatial domain.

5. In [12], similarities exist with our procedure with the iterative precomputation of the sampling density compensation weighting function is performed only using the sampling coordinate locations in k-space for MRI. However, even though the weighting function is multiplied to the data in the DFT domain, the complete procedure requires a convolution and a deconvolution in the DFT domain. Our method in contrast can be reinterpreted as having just one convolution to interpolate from one Cartesian grid to a larger one using 2-D DFTs with a pre-multiplication of the data masking filter and a post-multiplication of the regularized inverse of that filter.

6.2.3 Divide-And-Conquer Approach

1. The divide-and-conquer approach also is recommended for when the frequency sample locations are already determined and or when the image to reconstruct or dataset is quite large. This method allows the customization of solving each subproblem with regards to the amount of regularization and the overdetermining factor.

2. The divide-and-conquer method takes much longer than FBP *unless* each subband is solved in parallel, in this case is faster. If only a crude lowpass image is needed quickly, then only the lowest subband may be used.

3. The divide-and-conquer method is applicable to the general reconstruction problem from irregular samples and not just to tomographic projections as seen in the FBP results. Even for the tomographic projection data, each subband has

a different irregular sampling configuration to illustrate this point.

6.3 Suggestions for Future Research

1. The conditioning approach may benefit from a stronger theoretical link between the fast sensitivity measures and the condition number which explain the strong empirical evidence relating the two quantities.

2. Even though we estimate the condition number, there is a need to study the distribution of the actual condition number around the one estimated from the variance sensitivity measure. This can be accomplished by using Monte Carlo methods, where we simulate many examples and form a histogram. This in turn would be more useful than an upper bound even if one could be derived especially in the over-determined cases.

3. The estimated image of the non-iterative approach can be used as an initialization to the iterative methods such as the conjugate gradient algorithm and the POCS method, which both require a non-uniform FFT to implement. Yet the proposed initialization would reduce the overall time to convergence.

4. The divide-and-conquer image reconstruction can also be used as an initialization to the conjugate gradient method in the iterative algorithms.

5. As for the advantages of this divide-and-conquer method, the timings for each subproblem show that it is quite fast solved individually and can divide any large problem into smaller ones then can easily be solved either the iterative or the non-iterative approaches mentioned. Also a non-aliased low-resolution image is quickly computed which can then be updated with higher resolution subbands.

6. In the numerical results using computed tomography data, our methods directly apply in the parallel-beam projection selection in improving conditioning. For fan-beam data, interpolation is required to re-bin to parallel-beam projections. For the cases in variable angle projection selection, where a single parallel-beam projection is interpolation from multiple fan-beam projections, further considerations must be made to determine which fan-beam projections are required for a set of 2-D DTFT samples. The solution may require making hardware modifications such sampling data intermittently. Extending into 3 dimensions, cone-beam CT is another area to be explored.

7. As the numerical results focused on computed tomography, the reconstruction methods presented can also be applied to applications with complex-valued objects specifically in the areas of MRI and SAR.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] D. C. Munson, Jr., J. D. O'Brien, and W. K. Jenkins, "A tomographic formulation of spotlight-mode synthetic aperture radar," in *Proc. of IEEE*, vol. 71, Aug. 1983, pp. 917–925, no. 8.

[2] A. Macovski, *Medical Imaging Systems*. Prentice-Hall, 1983.

[3] A. E. Yagle, "An explicit closed-form solution to the limited-angle discrete tomography problem," *Int. J. Imaging Systems and Technology*, vol. 9, no. 2/3, pp. 174–180, 1998, special issue on Discrete Tomography.

[4] W. J. Rozwood, C. W. Therrien, and J. S. Lim, "Design of 2-D FIR filters by nonuniform frequency sampling," *IEEE Trans. Signal Processing*, vol. 39, no. 11, pp. 2508–2514, Nov. 1991.

[5] A. Zakhor and A. V. Oppenheim, "Reconstruction of two-dimensional signals from level crossings," in *Proc. of IEEE*, vol. 78, Jan. 1990, pp. 31–55, no. 1.

[6] A. Zakhor and G. Alvstad, "Two-dimensional polynomial interpolation from nonuniform samples," *IEEE Trans. Signal Processing*, vol. 40, no. 1, pp. 169–180, Jan. 1992.

[7] Y. Bresler and P. Feng, "Spectrum-blind minimum rate sampling and reconstruction of 2-D multiband signals," in *Proc. ICIP*, vol. 1, Sep. 1996, pp. 701–704.

[8] A. E. Yagle, "Closed-form reconstruction of images from irregular 2-D discrete Fourier samples using the Good-Thomas FFT," in *Proc. ICIP*, vol. 1, Sep. 2000, pp. 117–119.

[9] T. Strohmer, "Computationally attractive reconstruction of bandlimited images from irregular samples," *IEEE Trans. IP*, vol. 6, no. 4, pp. 540–550, Apr. 1997.

[10] A. Dutt and V. Rokhlin, "Fast fourier transforms for nonequispaced data," *SIAM J. Sci. Comp.*, vol. 14, no. 6, pp. 1368–1393, Nov. 1993.

[11] J. A. Fessler and B. P. Sutton, "Nonuniform fast Fourier transforms using min-max interpolation," *IEEE Trans. Signal Processing*, vol. 51, no. 2, pp. 560–574, Feb. 2003.

[12] J. G. Pipe and P. Menon, "Sampling density compensation in mri: Rationale and an iterative numerical solution," *Magnetic Resonance in Medicine*, vol. 41, no. 1, pp. 179 – 186, 1999.

[13] D. C. Youla and H. Webb, "Image restoration by the method of convex projections: Part 1 Theory," *IEEE Trans. Med. Imaging*, vol. 1, no. 2, pp. 81–94, Oct. 1982.

[14] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: Johns Hopkins University Press, 1996.

[15] B. C. Lee and A. E. Yagle, "A sensitivity measure for image reconstruction from irregular 2D DTFT samples," in *IEEE ICASSP*, vol. 4, May 2002, pp. 3245–3248.

[16] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*. NY: Addison-Wesley, 1974.

[17] J. R. Shewchuck, "An introduction to the conjugate gradient method without the agonizing pain," Carnegie Mellon University, Technical Report CMU-CS-94-125, Aug. 1994.

[18] T. Chan, "An optimal circulant preconditioner for Toeplitz systems," *SIAM J. Sci. Stat. Comput.*, vol. 9, pp. 766–771, 1988.

[19] N. K.Bose, "Asymptotic eigenvalue distribution of block-toeplitz matrices," *IEEE Trans. Inform. Theory*, vol. 44, no. 2, pp. 858–861, Mar. 1998.

[20] G. Beylkin, "The discrete radon transform," *IEEE Trans. SP*, vol. 35, no. 2, pp. 162–172, Feb. 1987.

[21] R. E. Blahut, *Fast Algorithms for Digital Signal Processing*. NY: Addison-Wesley, 1985.

[22] R. M. Mersereau and D. E. Dudgeon, "The representation of two-dimensional sequences as one-dimensional sequences," *IEEE Trans. ASSP*, vol. 22, no. 5, pp. 320–325, Oct. 1974.

[23] R. A. Horn and C. R. Johnson, *Matrix Analysis*. New York: Cambridge University Press, 1985.

[24] G. Szego, *Orthogonal Polynomials*, 4th ed. Providence, RI: Amer. Math. Soc., 1975, vol. 23.

[25] P. Vertesi, "Optimal lebesgue constant for lagrange interpolation," *SIAM J. Numer. Anal.*, vol. 27, no. 5, pp. 1322–1331, Oct. 1990.

[26] A. van der Sluis and H. A. van der Vorst, "The rate of convergence of conjugate gradients," *Numer. Math.*, vol. 48, pp. 543–560, 1986.

[27] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

[28] Y. Gao and S. J. Reeves, "Optimal k-space sampling in mrsi for images with a limited region of support," *IEEE Trans. Med. Imaging*, vol. 19, no. 12, pp. 1168–78, Dec. 2000.

[29] M. Woodward and F. Muir, "Hexagonal Sampling," in *Stanford Exploration Project*, vol. 38, 1983, pp. 183–194.

[30] M. Saranathan, V. Ramanan, R. G. Gulati, and R. Venkatesan, "Anthem: anatomically tailored hexagonal mri," *Magnetic Resonance Imaging*, vol. 25, no. 7, pp. 1039 – 1047, 2007.

[31] D. Callan, "When Is 'Rank' Additive?" *The College Mathematics Journal*, vol. 29, no. 2, pp. 145–147, Mar. 1998.

[32] C. R. Vogel, *Computational Methods for Inverse Problems (Frontiers in Applied Mathematics)*. New York: Soc. for Industrial and Applied Mathematics, 2002, vol. 10.