ABSTRACT

Ambiguity Function Magnitude Inversion and
Applications of Morphological Dilation in POCS

Albert R. Yu, Ph.D.

Mentor: Robert J. Marks II, Ph.D.

This dissertation examines morphological dilation for applications in Projection onto Convex Sets (POCS) as well as the inversion of ambiguity function magnitude. In general, POCS solvers are Least-Squares (LS) algorithms which minimize the L^2-norm of a proposed solution. However, there are situations where other error metrics can be advantageous. One such metric is the weighted minimized-maximum error, or minimax which minimizes the L^∞-norm. Multiple methods for evaluating the weighted, minimax error are investigated, and this dissertation will introduce a modified alternating projections algorithm utilizing morphological dilation on convex sets to solve for the minimax. This is shown to have notable improvements over standard POCS solvers for selective signal synthesis applications, including Fresnel diffraction synthesis, Computed Tomography (CT) and associative memory image reconstruction. When multiple, conflicting objective functions are present, minimax solvers can be demonstrated to be an unbiased solver among multiple conflicting constraints, avoiding the Least-Squares tendency to shift a solution towards the centroid.

In addition, the ambiguity function magnitude inversion is shown to be possible and a regularized method for quickly inverting a given function to a valid family of source signals is detailed. The ambiguity function is a fundamental aspect of radar signal processing that is frequently described as non-invertible from its magnitude as the transform is not one-to-one. In the past, an inversion to constant phase shift is possible with the full magnitude and phase of the function, but the phase information is frequently stripped as extraneous for analysis. Unfortunately this practice prevents a clear inversion. However, this paper demonstrates that inversion to a valid spawning signal is possible, and outlines a regularized method for achieving the desired magnitude response. This will give radar designers direct control over crafting ambiguity functions with mission-critical characteristics.

Ambiguity Function Magnitude Inversion and Applications of Morphological Dilation in POCS

by

Albert Reynold Yu, B.S.B.E., M.S.E.C.E.

A Dissertation

Department of Electrical and Computer Engineering

Kwang Y. Lee, Ph.D., Chairperson

Submitted to the Graduate Faculty of
Baylor University in Partial Fulfillment of the
Requirements for the Degree
of

Doctor of Philosophy

Approved by the Dissertation Committee

Robert J. Marks II, Ph.D., Chairperson

Keith E. Schubert, Ph.D.

Scott Koziol, Ph.D.

Brian Garner, Ph.D.

Accepted by the Graduate School

December 2018

J. Larry Lyon, Ph.D., Dean

Page bearing signatures is kept on file in the Graduate School.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank all the people who supported me for the quest to this dissertation. First I want to express my deepest appreciation to my adviser Dr. Robert J. Marks II for his unwavering guidance, friendship and encouragement. I would not be pursuing a Ph.D. without your inspiration. I also want to thank my family, for their patience, love and support. This has been a tremendous journey and I am eternally grateful.

Further thanks to the members of my committee, Drs. Keith Schubert, Scott Koziol, Eric Blair, and Brian Garner, and the Computational Intelligence Laboratory, Iwan Sandjaja, David Nemati and Eric Holloway, and Dr. Charles Baylis's help on the ambiguity function project. This work was partially funded by the National Science Foundation (Grant No. ECCS-1343316).

DEDICATION

To my family, friends and colleagues who helped guide me through this journey

CHAPTER ONE

Introduction

Motivation

This dissertation investigates novel inversion techniques and applies them in a variety of engineering applications. There are two prongs to this research: investigating a modified Projection onto Convex Set (POCS)-based algorithm utilizing morphological dilation used for minimizing the maximum or minimax error, and developing a regularized method for ambiguity function magnitude inversion for applications in radar signal processing. Dilated POCS is compared to standard POCS implementations which achieve a minimized L^2-norm or Least-Squares result.

The advent of fuzzy logic led to a new notion of fuzzy POCS as an alternative to Least-Squares minimization. In classic POCS, constraints are either met or failed with explicit values of true or false. Fuzzy logic attempts to quantify vagueness with a partial truth, and has led Zadeh to introduce fuzzy set theory [1]. A set or constraint can be repeatedly enlarged or fuzzified until previously non-intersecting sets all touch. The minimum enlargement of each set required in order for the greedy limit cycle to disappear and all sets to touch is the ideal solution. The key issue is how to fuzzify a set. One method is to fuzzify via morphological dilation. In morphological image processing, dilation is a technique of expanding shapes by the form of a dilation kernel. For example, if a certain image structure is known to contain holes, dilation can be used to expand the structure to fill in all holes before being eroded back to the original size. The size and shape of the kernel must be considered, as kernels that are too large risk subsuming all fine details. Mathematically, this is frequently accomplished via Minkowski addition [2] between the constraint and chosen kernel.

The applications investigated include Computed Tomographic (CT) image reconstruction, Fresnel diffraction synthesis, associative memory, and ambiguity function magnitude inversion. In CT imaging, an x-ray scanner irradiates a target object from multiple viewing angles and measures the time delay of received signals. These measurements are collated to form the target's sinogram. The task of image reconstruction is to invert the sinogram into an image of the original target. This problem is typically framed as a matrix inversion, but is impractical do perform directly due to the large size of images. As a result, POCS techniques such as Simultaneous Algebraic Reconstruction Technique (SART) [3] and its variants are used instead, as these algorithms can be highly parallelizable. This leads to a Least-Squares reconstruction that minimizes the sum of square errors to the given sinogram. However, this may not always be the best metric for reconstruction. In this inversion, each sinogram pixel is ray traced through the original image in order to identify a path matrix. The Least-Squares solution reduces the collective squared error among all paths in reconstruction, which can lead to loss of details when only a few select angles image certain details. A minimax approach can in certain scenarios address this issue and lead to an improved result.

Fresnel diffraction synthesis is another application where a typical projection-based approach may lead to a Least-Square result that is inferior to the minimax result. In diffraction synthesis, the goal is to design an aperture for transmission of a signal that will achieve the desired image intensities downrange from the source. This is accomplished by propagating the aperture to each downrange plane and specifying a desired target pupil, then projecting back via the Method of Angular Spectrum [2]. However, the tendency of Least-Square methods to bias the solution towards similar constraints leads to underrepresented, conflicting image constraints to be deemphasized. Minimax methods can address this issue by ensuring equal representation among the given pupil constraints.

The problem of associative memory is one of template matching with incomplete information. POCS can be used in this area to construct an image from a given snippet and a library of known images. The given signal segments can vary in size and similarity, leading to a conflict with the library. Minimax POCS can reveal an alternative solution to the common Least-Squares approach and emphasize different features in the synthesis process.

The ambiguity function (AF) used in radar signal processing is frequently described as non-invertible when given only its magnitude as the transform is not one-to-one. Inversion to a constant phase shift is possible given both the magnitude and phase, but in practice the AF is expressed in terms of its magnitude only, removing valuable phase information and impeding recovery. Thus, radar signal designers often use iterative methods to formulate signals with desirable properties by inspection. Using a modified Gerchberg-Saxton algorithm, we examine several transformations affecting phase recovery and develop a regularized inversion algorithm of the AF magnitude to valid spawning signals.

Outline

Chapter Two is a Literature Review over convex sets and signals, projection techniques, morphological dilation and the minimax. It will describe the relationship between dilated sets and the minimized-maximum weighted error, and continue into various algorithms for calculating the minimax solution. Chapter Three will investigate the application of dilated POCS to Computed Tomography (CT) image reconstruction and Fresnel diffraction synthesis, and how the minimax algorithm can improve results under certain noise effects. Chapter Four will investigate dilated projections to the topic of associative memory. Chapter Five will examine the inversion of ambiguity function magnitude inversion, and Chapter Six will conclude the dissertation and summarize the issues with dilated POCS for minimax optimization.

CHAPTER TWO

Background

Convex Sets

Geometric Convex Sets

A set X is convex if and only if for all vectors $\vec{x}_0, \vec{x}_1 \in X$, we have

$$\vec{x} = \lambda\vec{x}_0 + (1-\lambda)\vec{x}_1 \in X \qquad (2.1)$$

for $0 \leq \lambda \leq 1$. All points along the line formed between $\vec{x}_0$ and $\vec{x}_1$ is also within X. A point outside the set would violate convexity. Some geometric examples are shown in Figure 2.1. Convex sets ensure that the weighted averages of solutions in the set will stay within the set. If a set isn't convex, then taking a weighted average of solutions may lead to a point that is no longer valid. At its boundaries, a convex set that includes its limit points is called a *closed* convex set.

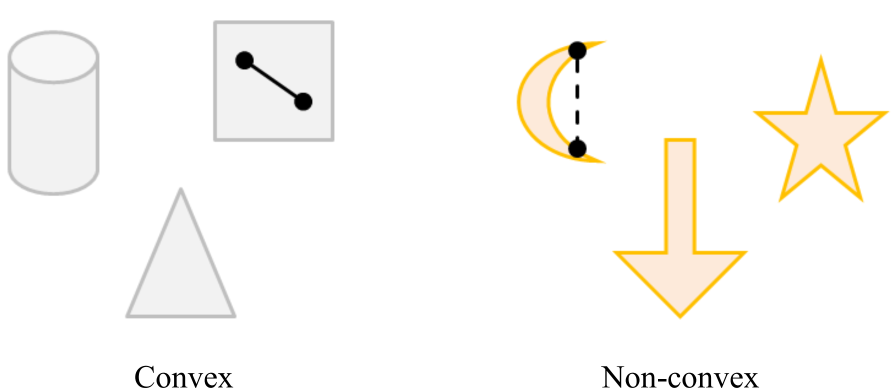


Figure 2.1. This figure shows geometric examples of convex and non-convex sets. For a convex set, any weighted average of two points will produce a point that is still inside the set. This is not the case for non-convex sets, where averaging points may produce an invalid solution.

Convex Projection

Projecting onto a convex set S involves finding a unique point on S that is closest to a given position. Starting at point $\vec{x}$, let the vector $\vec{s}$ point from $\vec{x}$ to the set C. The projection of $\vec{x}$ onto C is the vector $\vec{s}$ that minimizes its distance. That distance is usually the Euclidean distance or L^2-norm and is unique. This is depicted in Figure 2.2.

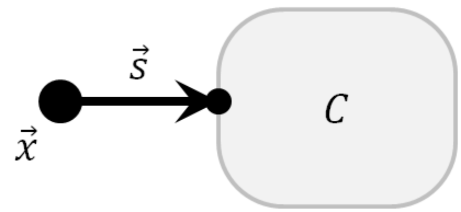


Figure 2.2. A projection of a point $\vec{x}$ onto a convex set C. The projection yields a unique vector $\vec{s}$ that minimizes the distance from $\vec{x}$ to set C.

If $\vec{x}$ is already within C, then that minimum distance is zero, $\vec{s} = 0$ and the point does not move and the projection does nothing. Projection results in a fixed point that depends on initialization. The projection operation is idempotent: any further projections will not change the outcome. This is described by

$$P_C(\vec{x}) = P_C\big(P_C(\vec{x})\big). \tag{2.2}$$

Minkowski Addition

Convex sets can be added together through Minkowski addition. The definition of Minkowski addition is given as follows: for the sets X and Y,

$$Z = X \oplus Y = \{\vec{z} \quad | \quad \vec{z} = \vec{x} + \vec{y} \quad \forall \quad \vec{x} \in X, \vec{y} \in Y\}. \tag{2.3}$$

When both X and Y are convex then this procedure results in a new set Z that is also convex. This can be shown from (2.1) and (2.3), where the given sets X and Y are convex. Then for any two points $\vec{z}_0$ and $\vec{z}_1$ in Z such that $\vec{z}_0 = \vec{x}_0 + \vec{y}_0$ and $\vec{z}_1 = \vec{x}_1 + \vec{y}_1$ where $\vec{x}_0, \vec{x}_1 \in X$ and $\vec{y}_0, \vec{y}_1 \in Y$, then for $0 \leq \lambda \leq 1$ the sum

$$\vec{z} = \lambda\vec{z}_0 + (1-\lambda)\vec{z}_1 = \lambda(\vec{x}_0 + \vec{y}_0) + (1-\lambda)(\vec{x}_1 + \vec{y}_1) \in X \oplus Y \tag{2.4}$$

Where $\vec{z} \in Z$ satisfies (2.1). This operation is also commutative. Minkowski addition is frequently used in morphological image processing where it is known as *dilation*; so named for enlarging a given binary mask by the desired dilation kernel. This technique is frequently paired with *erosion*, or the Minkowski difference, in a procedure known as *closing*, which is an idempotent operation used to fill in gaps in a binary mask.

Convex Signals

While a geometric interpretation of convexity gives an intuitive grasp of convex sets and projections, this concept can be extended to include many different types of signals. The following convex constraints: bounded energy, fixed area, identical middles, and duration-limited signals, are described in detail in [2] and will be used as the projection operators for the various applications described later in Chapters Three, Four and Five.

Bounded Energy

The set of signals with bounded energy includes any signal $x(t)$ such that

$$S = \{\, x(t) \quad | \quad \|x(t)\| \leq r \,\}$$

for some constant r such that the square of the L^2-norm is less than r. This set forms a hypersphere of radius r and is called a Ball. A projection of any signal $y(t)$ onto S is given by

$$P_S\big(y(t)\big) = \begin{cases} y(t) & , \quad \|y(t)\| \le r \\ \dfrac{r y(t)}{\|y(t)\|} & , \quad \|y(t)\| > r. \end{cases}$$

Thus, any value of $y(t)$ exceeding r in magnitude has its magnitude lowered to r and may be referred to as projecting onto the norm ball. This projection will be used in Chapter Three in the application of diffraction synthesis in order to ensure that the energy through the aperture is bounded to a specified value.

Constant or Fixed Area

The set of signals with constant area includes any signal $x(t)$ such that, over a fixed interval $[a, b]$

$$S = \left\{ x(t) \,\middle|\, \int_a^b x(t)dt = c \right\}$$

for some constant area c. A projection of any signal $y(t)$ onto S is given by

$$P_S\big(y(t)\big) = \begin{cases} y(t) + \dfrac{1}{b-a}\left(c - \displaystyle\int_a^b y(t)dt\right) & , \quad a \le t \le b \\ y(t) & , \quad \text{else.} \end{cases}$$

Thus, the values of $y(t)$ within $[a, b]$ are uniformly raised or lowered by a constant amount such that the new area is equal to c. In Chapter Three, the diffraction synthesis pupil will use this projection as a convex alternative to a non-convex Gerchberg-Saxton methodology [4]. The purpose of this projection in this application is to add intensity and prevent a degeneracy condition in the reconstructed signal and ensure the pupil acquires the desired shape.

Identical Middles

The set of signals with identical middles includes any signal $x(t)$ such that, over a fixed interval $[a, b]$

$$S = \{\, x(t) \mid x(t) = c(t) \ \forall\, t \in [a, b] \,\}$$

for some fixed $c(t)$. Thus, $c(t)$ forms the identical middle for all $t \in [a, b]$, and $x(t)$ is allowed to be anything else outside this interval. A projection of any signal $y(t)$ onto S is given by

$$P_S\big(y(t)\big) = \begin{cases} c(t) & , \quad a \le t \le b \\ y(t) & , \quad \text{else.} \end{cases}$$

Thus, the values of $y(t)$ within $[a, b]$ are set to the identical middle $c(t)$. The signal can take on any value outside this middle. This projection will be used in Chapter Four for associative memory to impose the multiple conflicting signal segments onto the synthesized image.

Duration-Limited

The set of signals that have finite support includes any signal $x(t)$ such that, over a fixed interval $[a, b]$

$$S = \{\, x(t) \mid x(t) = 0 \ \forall\, t \notin [a, b] \,\}.$$

A projection of any signal $y(t)$ onto S is given by

$$P_S\big(y(t)\big) = \begin{cases} y(t) & , \quad a \le t \le b \\ 0 & , \quad \text{else.} \end{cases}$$

Any nonzero value of $y(t)$ for a given t outside $[a, b]$ is set to zero. Chapter Five will implement an adaptive window based on this concept for ambiguity function magnitude inversion.

Projection onto Convex Sets

Projection onto Convex Sets (POCS) is a popular, longstanding class of algorithms for Least-Squares or L^2-norm minimization. The concept was introduced by John von Neumann in 1933 as the Method of Alternating Projections (MAP) [5], and has since been rediscovered many times by [6] and [7]. A randomly initialized point will reach a fixed point solution along the intersection of convex sets via repeated, sequential projections upon each set. However, this method assumes that such an intersection exists. If no intersection exists, then alternating projections will result in a greedy limit cycle that is dependent on initialization and has no unique solution. POCS and its variants have been applied to a variety of fields including signal recovery [8], artificial neural networks [9], [10], [11], medical imaging [11], [12], and time-frequency analysis [13], [14]. These were popularized by Kaczmarz [15], Gordon, Bender and Herman [16], Youla and Webb [12], Sezan and Stark [11] for applications in medical imaging and restoration. In many of these situations, one is given a set of convex constraint parameters to be met. A solution that satisfies all constraints corresponds to common point along the intersection of all sets. Alternating projections is a simple yet effective method at locating such common points.

Alternating Projections

This process will approach a fixed point for intersecting sets, which is highly dependent on initialization, as shown in Figure 2.3. However, this is not the case if the sets do not intersect. In the case of non-intersecting sets, alternating projections will reach a greedy limit cycle, as depicted in Figure 2.4.

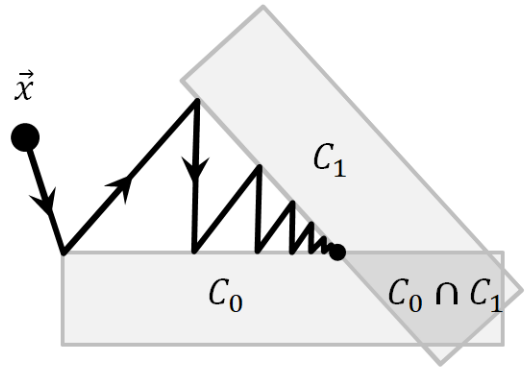


Figure 2.3. An illustration of the convergence achieved via alternating projections. Projecting onto two, intersecting sets from a random initialization, POCS will converge onto a fixed point on the intersection of the sets that highly depends on initialization.

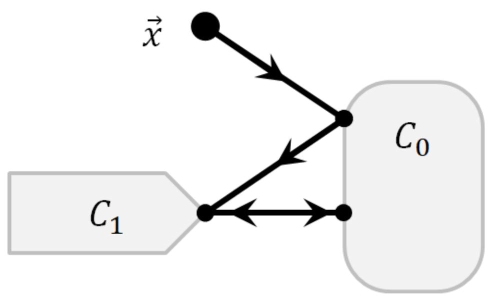


Figure 2.4. Alternating projections of $\vec{x}$ onto C_0 and C_1 does not lead to a common point, as both sets do not intersect. Instead, a limit cycle results that corresponds to the minimum distance between the two sets.

Unfortunately when there are three or more non-intersecting sets, alternating projections converges to a greedy limit cycle that has no clear usefulness that is highly dependent on the order of projections, as shown in Figure 2.5. In this example, alternating from $C_0 \rightarrow C_1 \rightarrow C_2$ produces a clearly different limit cycle than the reverse process of $C_2 \rightarrow C_1 \rightarrow C_0$. While the two-set example in Figure 2.4 represented the closest distance between C_0 and C_1, here there is no such useful property.

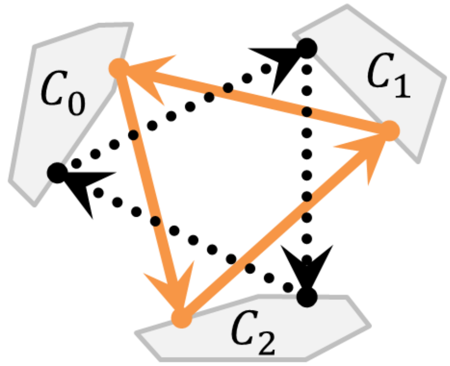


Figure 2.5. When there are three or more non-intersecting sets, alternating projections can converge to different limit cycles depending on the order of projections. Unlike the two set case, where the limit cycle represents the closest distance between the sets, the limit cycle for three or more sets has no known useful properties.

One way to attempt to resolve this issue is to introduce a relaxation term λ such that $0 < \lambda \leq 1$. This term is multiplied to the projected step, thereby truncating each iteration's step size. Relaxation, seen in Figure 2.6, can achieve significantly improved results in the rate or quality of convergence.

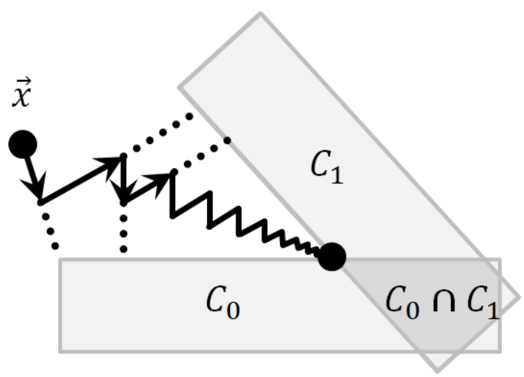


Figure 2.6. A relaxed alternating projection results where each projected step is scaled down by 0.5. This results in smaller step sizes, but in certain situations can improve convergence by ensuring the projections follow the "inside track" toward the fixed point.

It can also reduce the size of the final limit cycle, as seen in Figure 2.7. However, unless an optimal relaxation size is used, a limit cycle may still exist for nonintersecting sets.

One way to deal with this is to have the relaxation term λ get smaller with iteration, but this must be chosen carefully to prevent the step size from becoming too small before the algorithm has had a chance to converge. Overall, a unique, least-squares solution that minimizes the sum of squared distances to each set is preferred, which leads to the implementation of simultaneous projections.

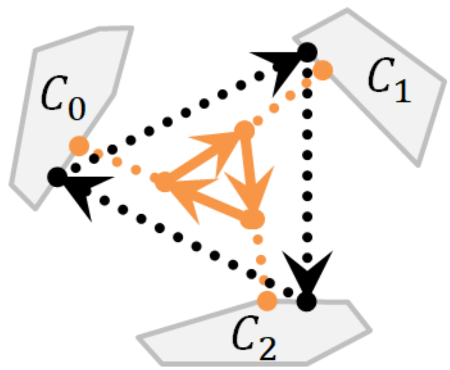


Figure 2.7. Here, a $\lambda = 0.5$ relaxation results in a significantly smaller limit cycle despite utilizing the same projection sequence. In some implementations, λ is reduced with iteration, resulting in the step size approaching zero. This can lead to the limit cycle approaching a least-squares solution between the given convex sets.

Simultaneous Projections

To counteract the multiple possible limit cycles reach with alternating projections, a unique, Least-Squares solution is preferred. Consider the following implementation of simultaneous projections [2],

$$P(\vec{x}) = \sum_{i \in C} w_i P_i(\vec{x}) \tag{2.5}$$

where

$$\sum_{i \in C} w_i = 1.$$

When w_n are all equal value, (2.5) becomes the equation for a centroid, or arithmetic mean. The total error of a point $\vec{x}$ is given by

$$D(\vec{x}) = \sum_{i \in C} w_i \|\vec{x} - P_i(\vec{x})\|^2 \tag{2.6}$$

Thus the goal is to minimize the total error, yielding a Least-Squares solution. Solving via simultaneous projections yields a Least-Squares centroid and is thus dominated by the center-of-mass of the constraint sets. Consequently, this prioritizes solutions near a higher density of constraint sets, diminishing the influence of constraints that are relative outliers. This is frequently resolved by varying the weights in order to explore all convex combinations, relying on the domain expertise of the operator in assigning more importance to one constraint over another. Figure 2.8 depicts a simultaneous projection onto multiple convex sets. The update step can be significantly lower than that of alternating projections as any update step is averaged over all other projection steps. As a result, hybrid methods are often employed instead to speed up convergence.

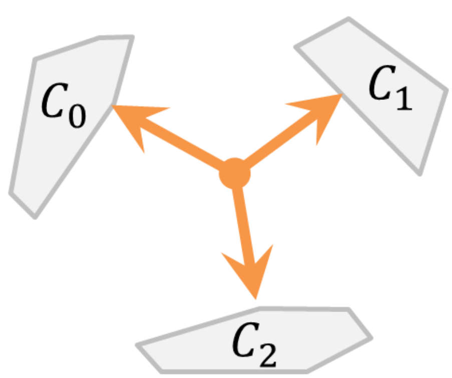
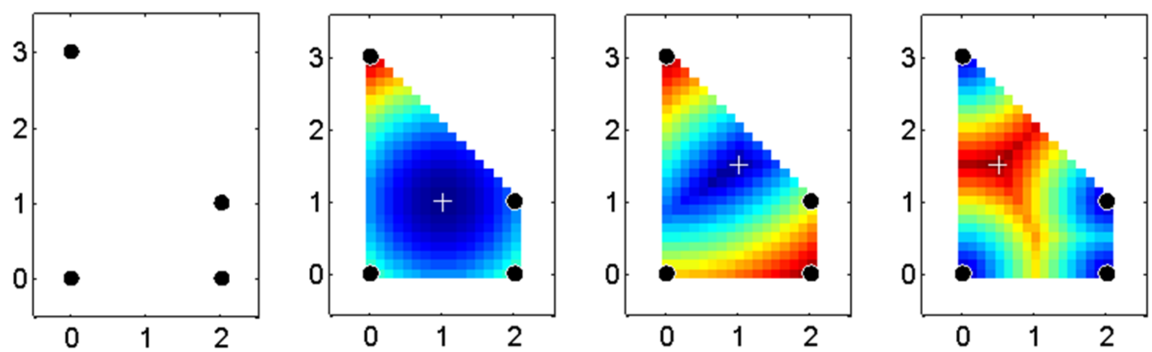


Figure 2.8. In simultaneous or averaged projections, a given point is simultaneously projected onto multiple constraint sets, and the sum of projections is used to update the position of the solution.

Minimax

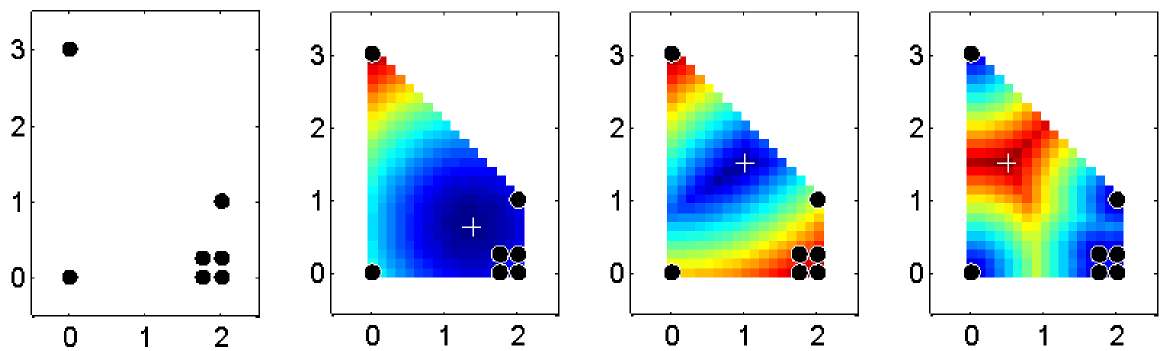
The error in (2.6) is expressed in terms of the L^2-norm squared, yielding a Least-Squares solution. Other norms may be used. When the L^1-norm is used the minimized function becomes the sum of absolute values, also known as Least-Absolute Deviations (LAD). When the L^∞-norm is used, the resulting optimization is known as the minimax and is used to minimize the maximum error. A geometric example using points is shown in Figure 2.9, demonstrating the differences between the Least-Squares solution, minimax, and maximin.



(a) Given Points (b) Least-Squares (c) Minimax (d) Maximin

Figure 2.9. Four convex sets are given as points in (a). The Least-Squares solution (b) is centroid of the four points, equivalent to the arithmetic mean. Minimax (c) locates the center of the Smallest Enclosing Ball (SEB) that minimizes the maximum distance to any point. Maximin (d) is not equivalent to the minimax, and locates the center of Voronoi partitions.

The maximin is also shown to demonstrate how it differs from the minimax. Figure 2.10 demonstrates how adding more constraints in the form of three more points will shift the centroid and thus Least-Squares solution. The minimax and maximin are unaffected by these additional constraints as they are within the maximum errors. Thus, the minimax solution is sensitive only to outlying constraints that push the boundaries of the space occupied by convex combinations of the constraints and will be robust against minor variations of already satisfied restrictions. In contrast, Least-Squares solutions will always shift with a changing center-of-mass, which may not be desired.



(a) Given Points (b) Least-Squares (c) Minimax (d) Maximin

Figure 2.10. Multiple convex sets are given as points in (a). The Least-Squares (b), minimax (c) and maximin (d) solutions are shown. The addition of three points onto Figure 2.9 results in a shift in the centroid and thus Least-Squares solution, while the minimax and maximin are unaffected.

For a geometric minimax consisting of m constraints in n dimensions, the minimax is defined by its boundary points which can be anywhere from one to the minimum of m and $n + 1$. Certain algorithms, such as Welzl's minidisk, recursively search through smaller subsets of boundary points in order to locate the SEB.

Smallest Enclosing Ball Solvers

The minimax is used in a variety of fields. One of the original applications of the minimax was in the 1-center problem [17], or smallest enclosing circle. This is depicted in Figure 2.11. The goal is to locate on a map an optimal position that would minimize the maximum distance to a given list of targets. The 2-center or higher variants involves placing two or more optimal centers such that the maximum distance to any center is minimized, and so on. At higher dimensions, the smallest enclosing circle becomes the Smallest Enclosing Ball, or SEB. Let P be the set of m points in n-dimensional space that needs to be enclosed with the smallest possible ball. The following algorithms have been developed to solve for the Smallest Enclosing Ball.

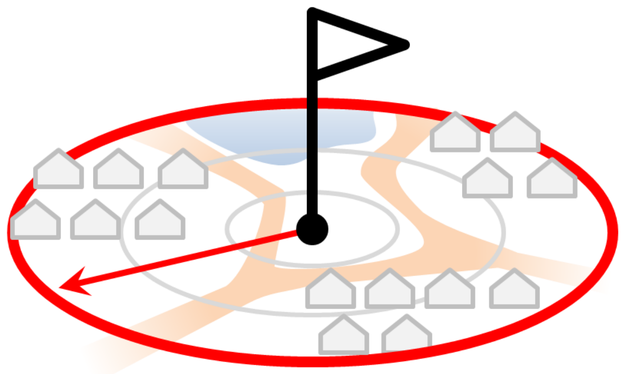


Figure 2.11. The 1-center problem is an application of minimax towards the smallest enclosing circle. The goal of this problem is to find the location of a shopping center that minimizes the maximum distance to any local residence.

Ritter's Bounding Sphere

Ritter's Bounding Sphere [17] is a simple approximation for the SEB at low dimensions. It is an iterative algorithm that takes an initial guess $\vec{x}_0$ performs the following actions:

1. Find $\vec{u} \in P$ that is maximum distance from $\vec{x}_k$.
2. Find $\vec{v} \in P$ that is maximum distance from $\vec{u}$.
3. Set $\vec{x}_{k+1} = \frac{1}{2}(\vec{u} + \vec{v})$.
4. Set $k = k + 1$ and repeat from 1.

Distances computations in n-dimensions need to be performed m times for each point in the set P, giving the algorithm a time complexity of $O(mn)$. Thus, this algorithm is a comparatively efficient method for approximating the minimax. The algorithm identifies the farthest pair of points in P. However, there is no guarantee that these two points define the boundary of the smallest enclosing ball. As a result, the algorithm performs poorly at higher dimensions. Nevertheless, the algorithm is popular for 2D and 3D applications for fast approximation due to its simplicity.

Welzl's Algorithm

Welzl's algorithm [18] is based on Seidel's "prune and search" algorithm [19], introduced a recursive algorithm for finding the SEB known as minidisk. It begins by randomly drawing $n + 1$ points in P to form the boundary set R, from which the circumcenter is computed. Subsets of R recursively tested to find the minimum number of points needed to define the circumcenter from these points. From here, the remaining points in P are tested against this boundary and the smallest enclosing ball is grown until all points are satisfied. Up to $n + 1$ points define the boundary, and thus $(n + 1)!$ recursive $n + 1$ computations are needed to explore each combination. This is performed over m points, giving the method a time complexity of $O(m(n + 1)(n + 1)!)$. Unfortunately even modest increases to n will make the Welzl algorithm untenable, so this method is kept to lower dimensions.

Badoiu-Clarkson

The minimum enclosing ball can be found via the Badoiu and Clarkson algorithm [20], which is a relatively simple procedure involving three primary steps. A simplified version is provided by Martinetz, Mamlouk and Mota [21] as follows.

1. Initialize with a guess of the balls center $\vec{x}_0$ at step $k = 0$, using any point in the set P.

2. For each step k, find $\vec{u}_k$ to be the farthest point in P from the current center $\vec{x}_k$

$$\underset{\vec{u} \in P}{\operatorname{argmax}} \|\vec{u}_k - \vec{x}_k\|.$$

3. Update the center location via

$$\vec{x}_{k+1} = \vec{x}_k + \frac{1}{k+1}(\vec{u}_k - \vec{x}_k).$$

Note that while $\vec{x}_0$ is canceled in the first iterative step, it is still needed in choosing the first, farthest point $\vec{u}_k$. The remaining terms can be shown to sum to

$$\vec{x}_{k+1} = \frac{1}{k+1}(\vec{u}_0 + \vec{u}_1 + \cdots + \vec{u}_k).$$

Each additional iteration tugs on the point of the center guess in the direction of the current, farthest point. Thus, $\vec{x}_k$ becomes some convex combination of all points $\vec{p}$ in the set P, or

$$\vec{x}_k = \lambda_0 \vec{p}_0 + \lambda_1 \vec{p}_1 + \cdots + \lambda_{m-1} \vec{p}_{m-1}$$

where $\lambda_0 + \lambda_1 + \cdots + \lambda_{m-1} = 1$. Each point in the set P nudges the center in accordance to how many times it is the farthest away point, achieving an equilibrium as $k \to \infty$. The accuracy of this algorithm depends on the iteration i to increase the resolution on the convex combination coefficients λ. The run time of the algorithm is inversely proportional to the error. The algorithm has a time complexity of $O\left(\frac{mn}{\epsilon}\right)$ to lower the relative error of the final estimated bounding radius to below ϵ.

Fischer's SEB

Fischer's SEB is a two-step, shrink and move process that gradually encloses a ball around all points in the set [22]. A random point $\vec{x}_0$ in P is chosen as an initial starting center, and the ball radius r is computed from the farthest away point in P. This results in an over-dilated ball. Once initialized, the shrink and move process begins. For shrinking, the radius r is lowered until one or more points touch the surface of the ball. The points touching the boundary are kept for the move step. To move the center $\vec{x}$, Fischer takes advantage of a property of the minimax. Given all points constituting the boundary, the minimax solution must lie on a line normal to the circumcenter or minimax of that boundary. Any point on said line will be equally distant to each point on the boundary, and thus a line search is needed to place the new center. Figure 2.12 depicts the process of finding the next search direction. The new ball is then shrunk, and a new line search is performed on the line normal to the new circumcenter. However, since the circumcenter is difficult to compute, the affine projection onto the boundary is used instead.

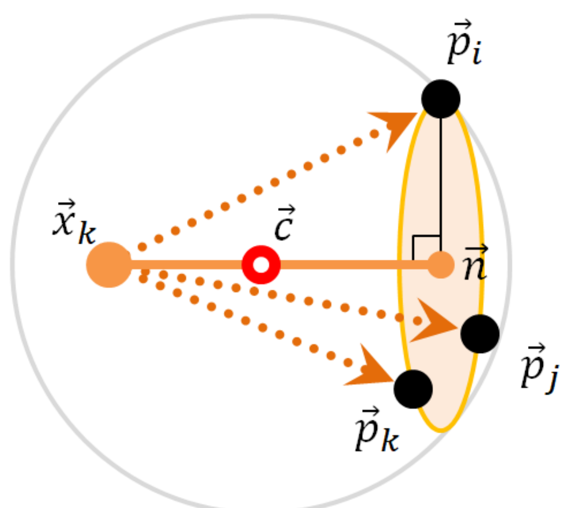


Figure 2.12. An illustration of Fischer's move step: for a given point $\vec{x}_k$, the farthest away points $\vec{p}_i$, $\vec{p}_j$, and $\vec{p}_k$ form the boundary of the current smallest enclosing ball. The point $\vec{n}$ is a projection of $\vec{x}_k$ onto the affine space defined by $\vec{p}_i$, $\vec{p}_j$, and $\vec{p}_k$. A line search between $\vec{x}_k$ and $\vec{n}$ is performed to compute the next center for SEB.

Linear Programming

Linear Programming (LP) [23], [24], [25], is a special case of convex optimization where the objective function and convex constraints are linear. Linear Programs take the form

$$\begin{array}{rcrcl} \underset{\vec{x}}{\text{minimize}} & : & \vec{c}^{\,T}\vec{x} & & \\ \text{subject to} & : & A\vec{x} & \leq & \vec{b}, \\ & & \vec{x} & \geq & \vec{0}. \end{array}$$

Here, $A \in \mathbb{R}^{m\times n}$ while $\vec{c}, \vec{x} \in \mathbb{R}^n$ and $\vec{b} \in \mathbb{R}^m$. This form is called the *primal* problem. The objective function $\vec{c}^{\,T}\vec{x}$ indicates a direction of increasing cost over the space defined by $\vec{x} \geq \vec{0}$. The gradient of an objective function is depicted in Figure 2.13, while Figure 2.14 depicts the effects of bounding constraints on the feasible search space.

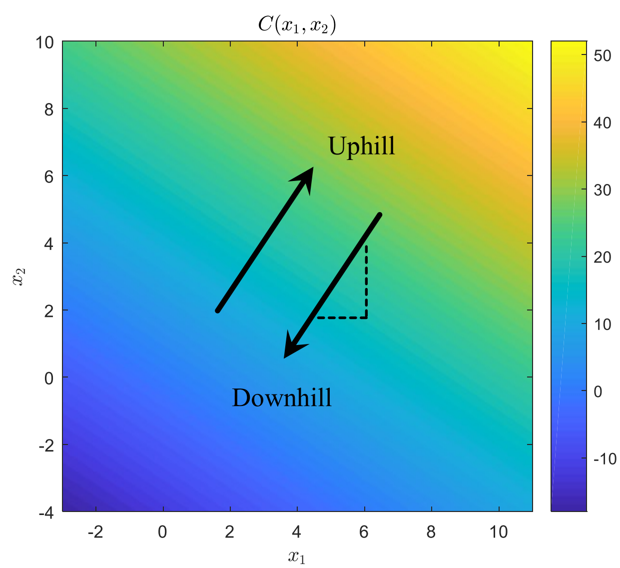


Figure 2.13. A 2D depiction of the objective function, indicating an up and downhill direction for maximization or minimization.

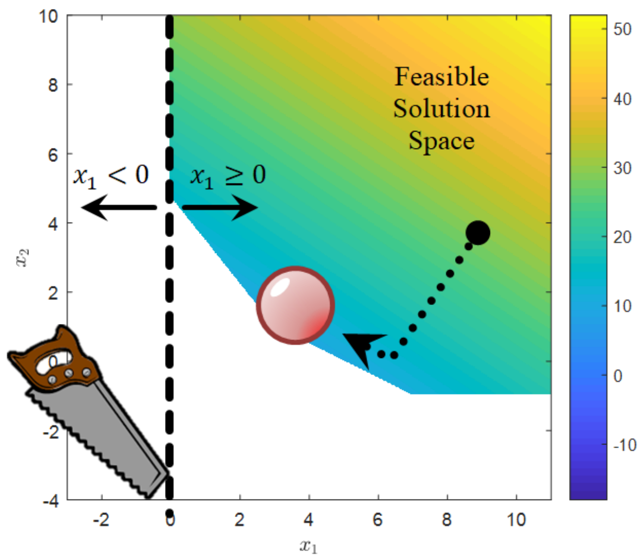


Figure 2.14. Boundary conditions indicate feasible half-spaces, the intersection of which forms the feasible solution space. For Linear Programs a feasible minimum will be found along a boundary of the feasible space. Conceptually, a marble rolling downhill will reach a minimum along the boundary.

However, this minimization can also be expressed as maximization.

$$\begin{aligned} \underset{\vec{y}}{\text{maximize}} \quad &: \quad \vec{b}^{\,T}\vec{y} \\ \text{subject to} \quad &: \quad A^T\vec{y} + \vec{s} = \vec{c}, \\ & \qquad\qquad \vec{s} \geq \vec{0}. \end{aligned}$$

This second form is called the *dual* problem, and the newly introduced term $\vec{s}$ is called the *dual slack*. These two forms are Lagrangian duals and can be related by the inequality

$$\vec{c}^{\,T}\vec{x} \geq \vec{b}^{\,T}\vec{y}.$$

This is known as the *duality gap*, and is often used as a condition for optimization in Interior Point (IP) methods. The duality gap becomes equality at the optimal solution, since

$$
\begin{aligned}
\vec{c}^{\,T}\vec{x} - \vec{b}^{\,T}\vec{y} &= (A^T\vec{y} + \vec{s})^T\vec{x} - (A\vec{x})^T\vec{y} \\
&= \vec{s}^{\,T}\vec{x} + (A^T\vec{y})^T\vec{x} - (A\vec{x})^T\vec{y} \\
&= \vec{s}^{\,T}\vec{x} \\
&\geq \vec{0}.
\end{aligned}
$$

Equality occurs at the optimal $\vec{x}$ and $\vec{y}$. This gives rise to the Karush-Kuhn-Tucker (KKT) conditions [23], [25], [24] for optimality, which are:

1. Dual feasibility $\quad A\vec{x} = \vec{b}$
2. Primal feasibility $\quad A^T\vec{y} + \vec{s} = \vec{c}$
3. Complementary slackness $\quad x_i s_i = 0 \quad , \quad i = 1, \dots, n$
4. Domain $\quad \vec{x}, \vec{s} \geq \vec{0}.$

While POCS may be used to solve certain Linear Programs, alternate methods are preferred in practice. Two popular types of Linear Programming solvers are the simplex method and Interior-Point (IP) methods. Simplex methods [26] maintain the first three KKT conditions and iteratively solve for the fourth, while Interior Point methods maintain the feasibility and domain and iteratively solves for the complementary slackness condition.

Simplex Methods

Dantzig's simplex method [26] revolves around setting each constraint defining a half-space. At higher dimensions, this can lead to slow convergence as the algorithm skirts the surface of the solution space. The Simplex algorithm is diagramed in Figures 2.15 to 2.18. The key aspects of the Simplex method are

- *The Origin*: The origin is a convenient starting point, and the cumulative displacement of the starting position indicates the final minimized objective function value.
- *Slack variables*: A slack variable indicates a constraints displacement from the origin. The larger the negative value the farther removed from the origin, whereas if it is positive the origin is already included and satisfied.
- *Entry Points*: Regarding slack variables, the more negative the slack variable, the farther removed from the origin. Ideally, all variables include or just touch the origin. Thus, picking the farthest away constraint is a good entry point into the iterative process.
- *Pivot Points*: The pivot point checks an entry points sensitivity to the objective function (by comparing ratios), indicating the direction of pivot. The element with the smallest ratio is chosen as the pivot point to encourage the minimum.

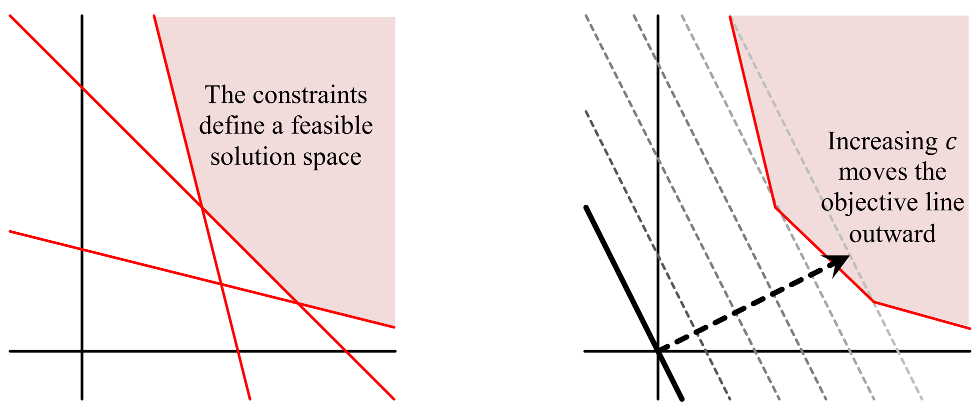


Figure 2.15. (Left) the given constraints form a fence around a feasible solution space. (Right) the objective function is shown as the black solid line. Increasing the value ***c*** shifts the objective function out. We want to find the point right when the objective function just touches the feasible space. But this is easier said than done.

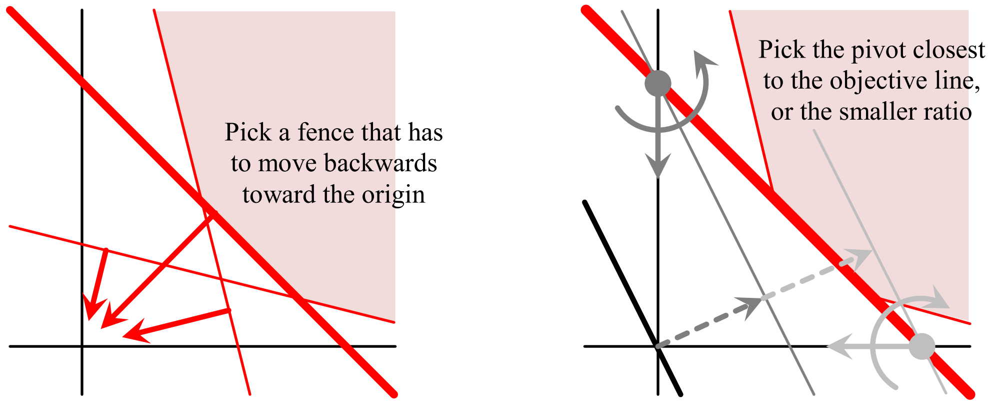


Figure 2.16. (Left) select a fence that is far away from the origin. Here, the one in bold is selected as the entry point. (Right) next pick a pivot point that is closest to the origin. This can be found via ratio of slopes.

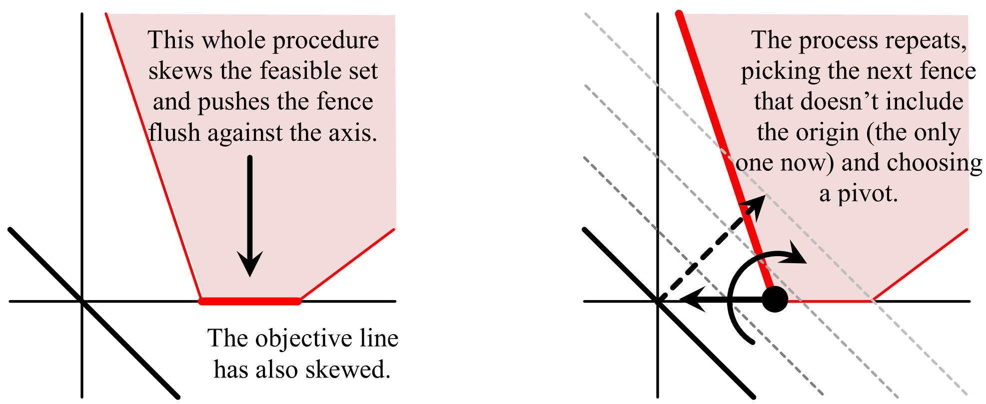


Figure 2.17. (Left) the result of pivoting is the feasible set flush against the axis. (Right) there are still fences that don't include the origin, so the process repeats.

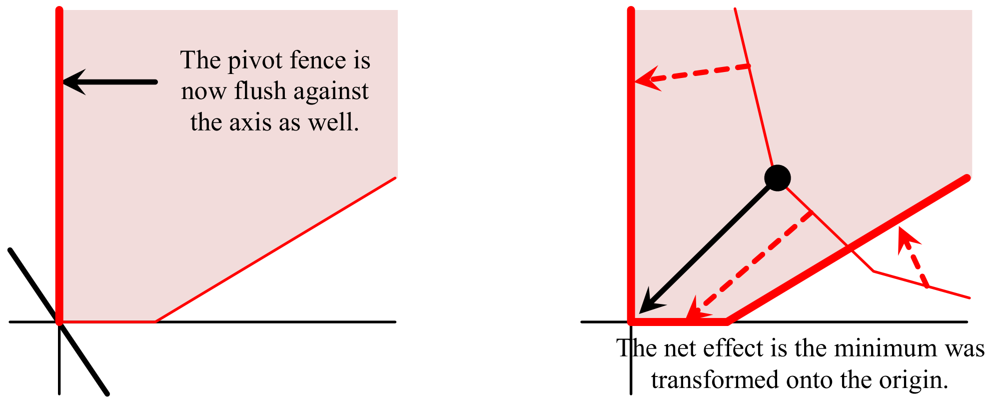


Figure 2.18. (Left) the result of pivoting the second fence. (Right) the overall change in position of the fence tells us where the minimum was in the original plot.

Interior-Point Methods

Interior-point methods converge on the optimal solution from within the convex hull. This can lead to slower convergence for low dimensional problems, but can be significantly faster at higher dimensions. Many Interior-Point methods have been developed to compete with simplex methods.

Dikin's Affine Scaling Method

Dikin's algorithm [27] has been rediscovered many times and is also known as the affine scaling method. Consider an update step of $\vec{f}_k$.

$$\begin{aligned} \vec{f}_{k+1} &= \vec{f}_k + a\Delta\vec{f}_k \\ G\vec{f}_{k+1} &= G\vec{f}_k + aG\Delta\vec{f}_k \\ &= \vec{g} \end{aligned} \tag{2.7}$$

where a is an arbitrary step size modifier greater than zero. At steady-state,

$$aG\Delta\vec{f}_k = \vec{0}.$$

Which indicates that $\Delta\vec{f}_k$ is in the nullspace of G. A projection P of a vector onto the nullspace of G is given by

$$P = I - G^T(GG^T)^{-1}G. \tag{2.7}$$

Identifying the direction of steepest descent

$$\Delta\vec{f}_k = -P\vec{c}$$

because

$$\begin{aligned}\vec{c}^{\,T}\vec{f}_{k+1} &= \vec{c}^{\,T}\vec{f}_k + a\vec{c}^{\,T}\Delta\vec{f}_k \\ &= \vec{c}^{\,T}\vec{f}_k - a\vec{c}^{\,T}P\vec{c} \\ \vec{c}^{\,T}\vec{f}_{k+1} &< \vec{c}^{\,T}\vec{f}_k\end{aligned}$$

since $a > 0$ and $\vec{c}^{\,T}P\vec{c} > 0$. In addition, a must be chosen such that $\vec{f}_k - aP\vec{c} \geq \vec{0}$ since $\vec{f} \geq \vec{0}$.

Primal-Dual Newton-Step Methods

Primal-Dual Newton-Step barrier methods [23], [28], [29] maintain the KKT conditions. A logarithmic barrier function is applied to the primal

$$\begin{aligned}\underset{\vec{x}}{\text{minimize}} &: \quad \vec{c}^{\,T}\vec{x} - \tau\sum_j \log(x_j) \\ \text{subject to} &: \quad A\vec{x} = \vec{b}, \\ &\quad\ \ \vec{x} \geq \vec{0}.\end{aligned}$$

Similarly, the dual becomes

$$\begin{aligned}\underset{\vec{y}}{\text{maximize}} &: \quad \vec{b}^{\,T}\vec{y} + \tau\sum_j \log(s_j) \\ \text{subject to} &: \quad A^T\vec{y} + \vec{s} = \vec{c}, \\ &\quad\ \ \vec{s} \geq \vec{0}.\end{aligned}$$

Taking the KKT conditions yields

$$
\begin{aligned}
A^T\vec{y} + \vec{s} - \vec{c} &= \vec{0} \\
A\vec{x} - \vec{b} &= \vec{0} \\
\vec{x} \circ \vec{s} &= \tau\vec{1}.
\end{aligned}
$$

To compute a Newton step, the gradient must be formed form the above conditions. This can be found via the Jacobian,

$$
J = \begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{bmatrix}
$$

which yields the Newton step

$$
\begin{bmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta\vec{x} \\ \Delta\vec{y} \\ \Delta\vec{s} \end{bmatrix} = - \begin{bmatrix} A^T\vec{y} + \vec{s} - \vec{c} \\ A\vec{x} - \vec{b} \\ \vec{x} \circ \vec{s} - \tau\vec{1} \end{bmatrix}
$$

Solving for $\Delta\vec{x}$, $\Delta\vec{y}$ and $\Delta\vec{s}$ gives the update

$$
\begin{bmatrix} \vec{x}_{k+1} \\ \vec{y}_{k+1} \\ \vec{s}_{k+1} \end{bmatrix} = \begin{bmatrix} \vec{x}_k \\ \vec{y}_k \\ \vec{s}_k \end{bmatrix} + a_k \begin{bmatrix} \Delta\vec{x}_k \\ \Delta\vec{y}_k \\ \Delta\vec{s}_k \end{bmatrix}
$$

where a_k is an step-size modifier greater than zero such that the conditions on $\vec{x}, \vec{s} > 0$ are maintained. Each iteration, τ is shrunk by a chosen $0 < \delta < 1$ such that

$$
\tau_{k+1} = \delta\tau_k
$$

in order to tighten the 3rd KKT condition with each iteration.

Linear Programming Minimax

The problem of minimized maximum can be extended to matrix inversion and thus Linear Programming. The goal is to resolve $\vec{b} = A\vec{x}$, where $\vec{x} \in \mathbb{R}^n$, $\vec{b} \in \mathbb{R}^m$, and $A \in \mathbb{R}^{m \times n}$. Direct inversion becomes computationally impractical for large values of m and n, requiring alternative methods of evaluation. Many problems, including tomographic image reconstruction, require these techniques. Minimax error involving matrices can be reformulated as a Linear Programming problem and efficiently solved. The goal is to find $\vec{x}$ in order to minimize the norm of the residual

$$\text{minimize} \quad \left\| \vec{a}_i^T \vec{x} - b_i \right\|_p$$

When $p = 2$, this yields the standard, Least-Squares approach, minimizing the sum of squares of the residual. Setting $p = 1$ gives the Least Absolute Deviation approach, minimizing the sum of absolute values. For minimax $p = \infty$ such that

$$\text{minimize} \quad \max_i \left| w_i (\vec{a}_i^T \vec{x} - b_i) \right|$$

This can be reformulated as

$$\begin{aligned} &\text{minimize} && \delta \\ &\text{subject to} && \delta \geq w_i(\vec{a}_i^T \vec{x} - b_i) \geq -\delta, \\ & && \vec{x} \geq \vec{0} \end{aligned}$$

and this is resolved by substituting $\vec{x}' = \begin{bmatrix} \vec{x} \\ \delta \end{bmatrix}$ and minimizing the last element in vector $\vec{x}'$.

Let $\vec{u}$ be the inverse of vector of weights resulting from dividing both sides of the constraints by w_i, or where

$$u_i w_i = 1.$$

Here the values of u_i indicate how a given constraint scales with the minimized maximum error, and is inversely proportional to w_i. Thus, the more important a constraint, the slower it dilates. When $w_i = 0$, the constraint is trivial and $u_i \to \infty$ and does not need to be considered. This can also be extended to situations where constraints must be strictly followed. If a certain constraint i exists that must have no error in it, then the rate of dilation $u_i = 0$ and a projection onto this constraint is directly onto the set. This can be useful for situations where no compromise can be made.

$$\begin{aligned} &\text{minimize} && \delta \\ &\text{subject to} && \delta u_i \geq \vec{a}_i^T \vec{x} - b_i \geq -\delta u_i, \\ & && \vec{x} \geq \vec{0}. \end{aligned}$$

The constraints can be rewritten in standard form as

$$\begin{bmatrix} A & -\vec{u} \\ -A & -\vec{u} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \delta \end{bmatrix} \leq \begin{bmatrix} \vec{b} \\ -\vec{b} \end{bmatrix} \tag{2.7}$$

and the objective function to minimize can be written as

$$\begin{bmatrix} \vec{0}_n \\ 1 \end{bmatrix}^T \begin{bmatrix} \vec{x} \\ \delta \end{bmatrix}.$$

where $\vec{0}_n$ is a column vectors of zeros of length n. The inequality in (2.7) is often rewritten as an equality condition by augmenting $\vec{x}$ with slack variables. Define $\vec{r} \geq 0$ to be a length m column vector. Then (2.7) can be rewritten as

$$\begin{aligned} &\text{minimize} && \begin{bmatrix} \vec{0}_n \\ \vec{0}_m \\ 1 \end{bmatrix}^T \begin{bmatrix} \vec{x} \\ \vec{r} \\ \delta \end{bmatrix} \\ &\text{subject to} && \begin{bmatrix} A & -I_m & -\vec{u} \\ -A & -I_m & -\vec{u} \end{bmatrix} \begin{bmatrix} \vec{x} \\ \vec{r} \\ \delta \end{bmatrix} = \begin{bmatrix} \vec{b} \\ -\vec{b} \end{bmatrix}, \\ & && \begin{bmatrix} \vec{x} \\ \vec{r} \\ \delta \end{bmatrix} \geq \vec{0}. \end{aligned} \tag{2.8}$$

where I_m is an identity matrix of size $m \times m$. The Linear Program can be written as

$$\begin{aligned} &\text{minimize} && \vec{c}^{\,T} \vec{f} \\ &\text{subject to} && G\vec{f} = \vec{g}, \\ & && \vec{f} \geq \vec{0} \end{aligned} \tag{2.9}$$

where

$$\vec{c} = \begin{bmatrix} \vec{0}_n \\ \vec{0}_m \\ 1 \end{bmatrix}, \quad G = \begin{bmatrix} A & -I_m & -\vec{u} \\ -A & -I_m & -\vec{u} \end{bmatrix}, \quad \vec{f} = \begin{bmatrix} \vec{x} \\ \vec{r} \\ \delta \end{bmatrix}, \quad \vec{g} = \begin{bmatrix} \vec{b} \\ -\vec{b} \end{bmatrix}.$$

Thus the given components are G, $\vec{g}$ and $\vec{c}$ form a Linear Program where the goal is to find $\vec{f}$.

CHAPTER THREE

Morphological Dilation for Applications in POCS

This chapter is an unpublished manuscript by: A. R. Yu, K. E. Schubert and R. J. Marks II, "Morphological Dilation for Applications in POCS."

Introduction

This manuscript investigates modifications to POCS in order to minimize the maximum error through morphological dilation. The work is inspired and mentored by Dr. Robert J. Marks II who posed the various applications of dilated POCS and guided the research process. Dr. Keith E. Schubert also provided valuable insight into the various implementations of Computed Tomographic (CT) reconstruction techniques used in medical imaging. This work presents the theory of the minimax solution through morphological dilation and the potential advantages of this technique over current Least-Squares algorithms. These advantages are demonstrated through applications of morphological dilation to CT image reconstruction and Fresnel diffraction synthesis.

Projection onto Convex Sets is a popular method for Least-Squares or L^2-norm minimization. It is based on the Method of Alternating Projections (MAP) introduced by John von Neumann in 1933 [5]. Initially, alternating projections requires a nonempty intersection of convex constraints in order to arrive at a fixed point solution; otherwise a greedy limit cycle is reached [2]. A diverse assortment of POCS techniques were developed to address this issue, including the addition of relaxation terms and averaged or simultaneous projections in order to resolve the limit cycle and achieve a Least-Squares solution when a unique solution does not exist. These POCS algorithms are applied to a variety of fields including signal recovery [8], artificial neural networks [9], [10], [30], medical imaging [11], [31], [12], and time-frequency analysis [13], [14].

The goal in many restoration and synthesis problems is to discover any image or signal that expresses a predefined set of desired qualities. These characteristics are frequently presented as a set of convex constraints and can take many forms, including signals with bounded energy, time-duration limited, band-limited signals, or signals with fixed area [2]. A solution that satisfies all the constraints occurs within the intersection of these convex sets. However, when these constraints do not intersect then no common solution exists and alternating projections will fail to converge. A Least-Squares approach is typically implemented to resolve this issue with the goal to minimize the weighted Total Variation,

$$\min_{\vec{x} \in S} \sum_{i \in C} w_i \|\vec{x} - P_i(\vec{x})\|^2$$

where w_i are weights on the set of convex constraints C such that

$$\sum_{i \in C} w_i = 1$$

and P_i is the projection operator for the ith convex constraint and S is the search space for $\vec{x}$. For POCS the weighted L^2-norm is used and corresponds to a centroid. However, there are situational benefits to other norms. In Least-Absolute Deviations (LAD), the L^1-norm is used and is known to be a robust alternative that is less susceptible to outliers and optimal for certain error types [32]. When the L^∞-norm is used, the minimization becomes

$$\min_{\vec{x} \in S} \max_{i \in C} \left(w_i \|\vec{x} - P_i(\vec{x})\| \right)$$

and the resulting optimization is called the weighted minimax.

The minimax problem has been thoroughly investigated extensively for many disparate applications and has many specialized solvers. A classic example of minimax explores the 1-center problem [33] in order to locate an optimal construction placement for a shopping center that minimizes the maximum distance to all local residences on a map. An analogous problem in computational geometry is known as the Smallest Enclosing Ball (SEB) problem and expands the 1-center problem to higher dimensional balls with applications in computer graphics and robotics [34]. Minimax can also be applied to matrix inversion, where it is typically framed as a Linear Program (LP). In statistical decision theory the minimax is explored to minimize the maximum cost or worst-case scenario. Each application has developed various *ad hoc* solvers for computing the minimax. For SEB, popular solvers include Welzl's miniball [18], Gartner, Fischer and Zurcher's algorithms [22], [35], and the Badoiu-Clarckson algorithm [20]. Particle Swarm Optimization has also been used for the 1-center problem [36]. Both Interior-Point and simplex methods can be used for minimax matrix inversion.

This paper explores a generalized POCS-based solver that utilizes morphological dilation in order to achieve the weighted minimax solution. The impetus for utilizing POCS is in accommodating the variety of potential convex signal processing constraints that may not have a direct, consistent correspondence as performing a projection can involve a different operator for each constraint. For example, the set of band-limited signals is projected upon via low-pass filtering at the required cut-off frequency, while the set of bounded signals is projected upon via truncation. In addition, morphological dilation with specialized convex kernels can situationally improve POCS reconstructions.

We present a POCS-based method for computing the minimax and demonstrate its practical impact through applications in Fresnel diffraction synthesis and Computed Tomography (CT) image reconstruction and contrast the minimax to current POCS techniques and their corresponding Least-Squares results.

Least-Squares versus Minimax

Alternating projections onto convex sets is best visualized geometrically as in Figure 3.1. Given two intersecting sets, any randomly initialized point will converge to a fixed point within the intersection via repeated, alternating projections. When sets do not intersect then no common solution exists and the alternating projections approach a greedy limit cycle as seen in Figure 3.2. Limit cycles depend on the initialization and order of projections. To resolve this situation and arrive at a distinct solution, a Least-Squares algorithm is typically implemented to find the centroid within these constraints.

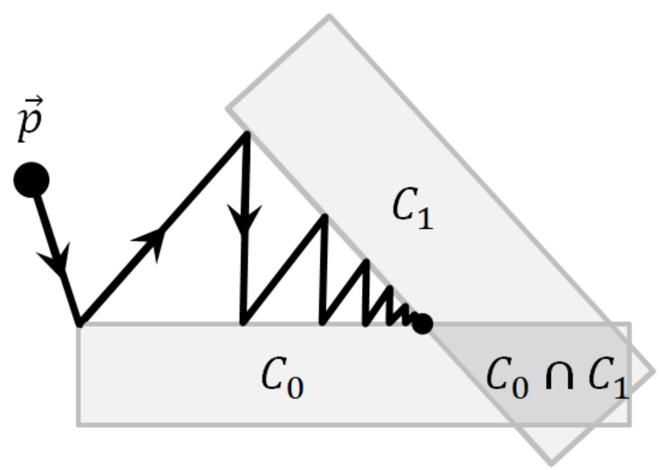


Figure 3.1. An illustration of the convergence of alternating projections. POCS will converge onto a point on the intersection of convex constraint sets from any random initialization.

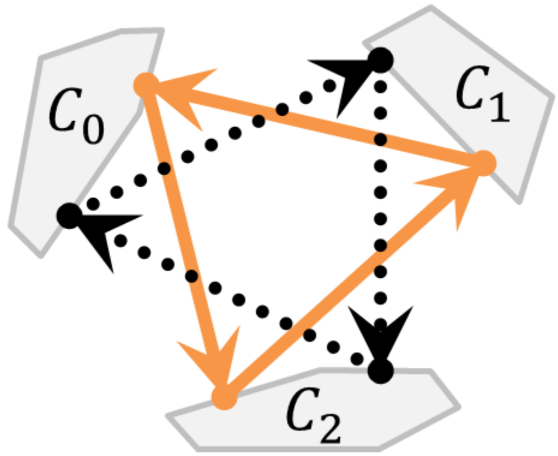


Figure 3.2. An illustration of various greedy limit cycles in alternating projections for non-intersecting convex sets. Limit cycles may not be unique. Here, projecting $C_0 \to C_1 \to C_2$ yields a different limit cycle to $C_2 \to C_1 \to C_0$. A Least-Squares approach is often used to resolve this issue and produce a unique result.

Least-Squares algorithms minimize the L^2-norm, or the sum of squares. Consider the simultaneous projection onto all convex sets [2], [11],

$$P(\vec{x}) = \sum_{i \in C} w_i P_i(\vec{x}).$$

When w_i are all equal, this corresponds to the equation for a centroid, or arithmetic mean, and the solution corresponds to the Least-Squares minimization

$$D_\infty = \sum_{i \in C} w_i \|\vec{x}_\infty - P_i(\vec{x}_\infty)\|^2.$$

Consider the simple 2D example of non-intersecting convex sets consisting of three distinct points (black dots) shown in Figure 3.3. The bottom left is the origin and the other two dots are on the axes one unit away. For an arbitrary point $\vec{x} = (x_0, x_1)$, the weighted sum of distances is

$$D = \sum_{i \in C} w_i \|\vec{x}_\infty - P_i(\vec{x}_\infty)\|^2 = x_0^2 + x_1^2 + w_0(1 - 2x_0) + w_1(1 - 2x_1).$$

Then

$$\frac{\partial D}{\partial x_0} = 2x_0 - 2w_0$$

and

$$\frac{\partial D}{\partial x_1} = 2x_1 - 2w_1.$$

Setting these to zero gives

$$\vec{x}_\infty = (w_0, w_1).$$

For equal weights,

$$(w_0, w_1, w_2) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$$

we reach the steady-state solution shown in Figure 3.3. Solving via Least-Squares yields a centroid dominated by the center-of-mass of the constraints. Consequently, this prioritizes solutions near a higher density of constraint sets, diminishing the influence of constraints that are relative outliers. This is frequently resolved by varying the weights, relying on the domain expertise of the operator in assigning more importance to one target over another.

An alternative approach is to minimize the L^∞-norm. This process results in a solution with the minimized maximum error among all the given constraints and is thus independent of the centroid of the solution set. This distinction can be seen through the example in Figure 3.4. Note that while the Least-Squares approach corresponds to a centroid, the minimax approach is not the equidistant solution (or circumcenter) as non-max errors falling within the convex hull of constraint sets are effectively ignored. As the smallest envelope, the minimax solution is even more sensitive to outlying error that Least-Squares.

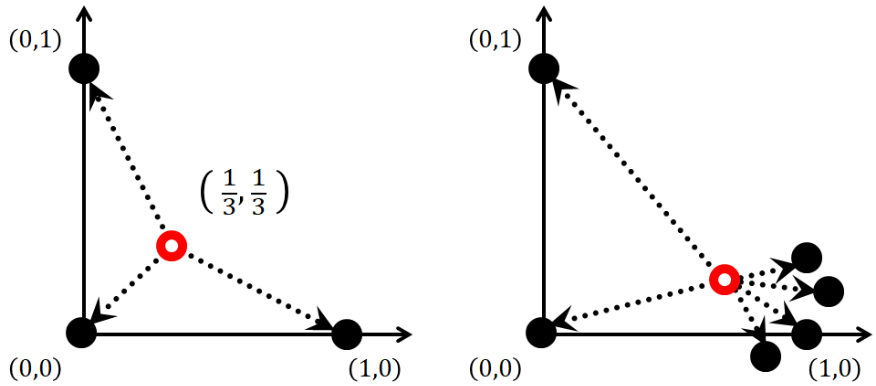


Figure 3.3. In this geometric example, the Least-Squares solution approaches the centroid, an arithmetic mean (left). Adding more constraints (right) will shift the solution towards the densest region and may not be ideal for the situation at hand.

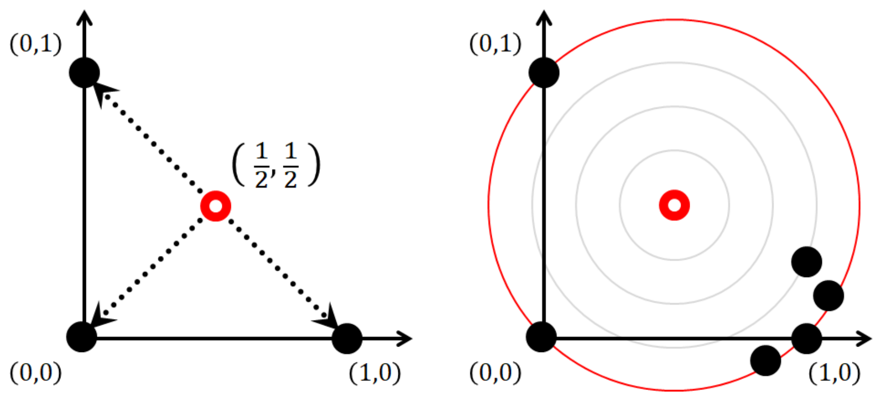


Figure 3.4. The minimax solution (left) will locate a position that has a minimized maximum weighted projected distance to any constraint. Additional constraints within this max distance will not shift the solution. When all constraints are equally weighted, this geometric example is the Smallest Enclosing Ball problem (right).

Solving for Weighted Minimax via POCS

Many small-scale, low-dimensional algorithms exist to solve or approximate the SEB. Ritter's Bounding Sphere [17] is a simple and popular low-dimensional algorithm that can quickly come to an approximation of SEB by identifying the midpoint of the two points farthest away from each other, but this is rarely exact and is not suited for higher dimensions. Welzl's Miniball [18] takes a randomized approach to identify the minimum points to form the smallest convex hull, but this also has issues with large-scale implementation. Fischer's SEB [22] uses a two-step, shrink and move approach that can achieve exact solutions to SEB and performs well for large problems. The Badoiu-Clarkson [20] achieves asymptotic error bounds and is simple to implement. In each case, the minimax is the minimized maximum L^2 error corresponding to a smallest ball. A generalized projection algorithm allows changes to the kernel to explore beyond the SEB paradigm.

We use morphological dilation with specific kernels to alternating projections to determine the minimax solution. Dilation in morphological image processing involves Minkowski addition [2]. A set X is dilated with a kernel Y when

$$X \oplus Y = \{ \vec{z} \mid \vec{z} = \vec{x} + \vec{y} \ \forall \ \vec{x} \in X, \vec{y} \in Y \}.$$

This operation is commutative and conserves convexity. A geometric example is shown in Figure 3.5. Morphological dilation with a hypersphere has the effect of minimizing the maximum distance to any given constraint set. This dilation hypersphere is a ball, defined by

$$B(\vec{c}, r) = \{ \vec{x} \in S \mid \|\vec{x} - \vec{c}\|_2 \leq r \}$$

where $\vec{c}$ is the ball's center and r is its radius. However, the dilation kernel does not have to be restricted to a ball. Any convex kernel convolved with a convex constraint will still yield a convex set. As such, a different kernel may be situationally advantageous. Projection onto non-ball kernels can be performed by dilating the constraint sets directly. The smallest enclosing ball in Figure 3.4 is equivalent to dilating each constraint set as in Figure 3.6. Dilating the constraints allows for a variety of kernels to apply. Both ball and box dilation kernels are demonstrated in the following example applications. Dilating the convex constraints can be accomplished in two ways.

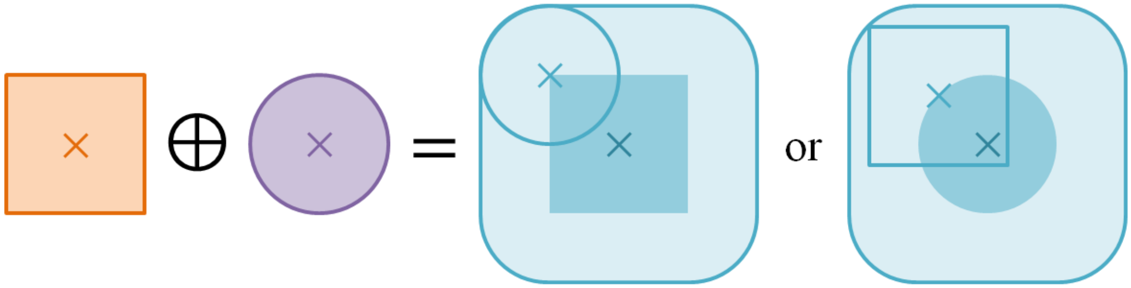


Figure 3.5. A geometric example of dilation and its commutative property, with the center of each kernel at the ×.

1. Project with a truncated step as consistent with dilation with a ball kernel. This is a SEB approach for which there exist many efficient algorithms.
2. Project onto convex constraints that have been dilated with a desired kernel. This approach allows for varying types of dilation kernels to be applied.

POCS can be modified to apply to both methodologies. Note that constraints that are unevenly weighted have corresponding rates of dilation, as shown in Figure 3.7. Many points will have solutions easily satisfied, corresponding to an over-dilation, as in point B in Figure 3.8, effectively removing these points from further consideration. In alternating projections, the ith projected step $\vec{s}_i$ is given by

$$\vec{s}_i = w_i(P_i(\vec{x}) - \vec{x}).$$

For dilated projections, this projected step is truncated by the dilated size of the kernel. For SEB situations this truncation corresponds to the radius of the hypersphere r. The ideal radius r can be found through binary search. For a given $\vec{x}$, the updated, dilated projected step $\vec{s}_{i,r}$ is given by

$$\vec{s}_{i,r} = \begin{cases} \left(1 - \dfrac{w_i r}{\|\vec{s}_i\|}\right)\vec{s}_i & , \quad \|\vec{s}_i\| \geq r \\ 0 & , \quad \text{otherwise.} \end{cases}$$

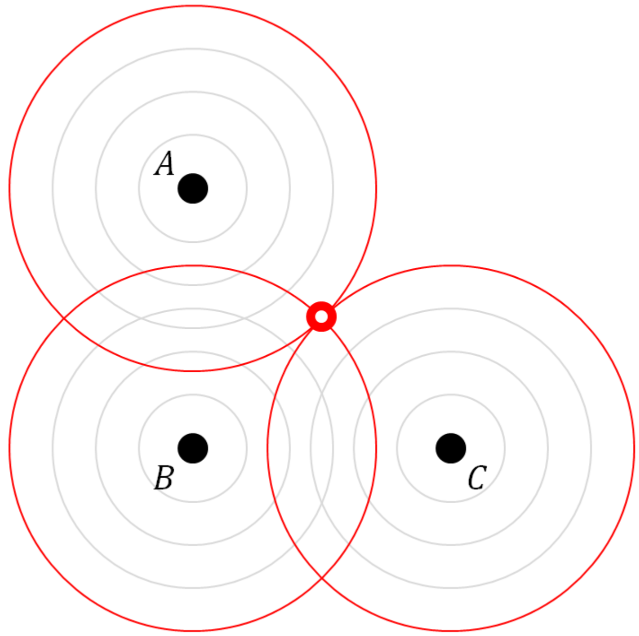


Figure 3.6. In this example, three convex sets (black dots) are dilated to progressively larger disks until all sets initially touch. This dilated solution (red dot) corresponds to solving the SEB problem.

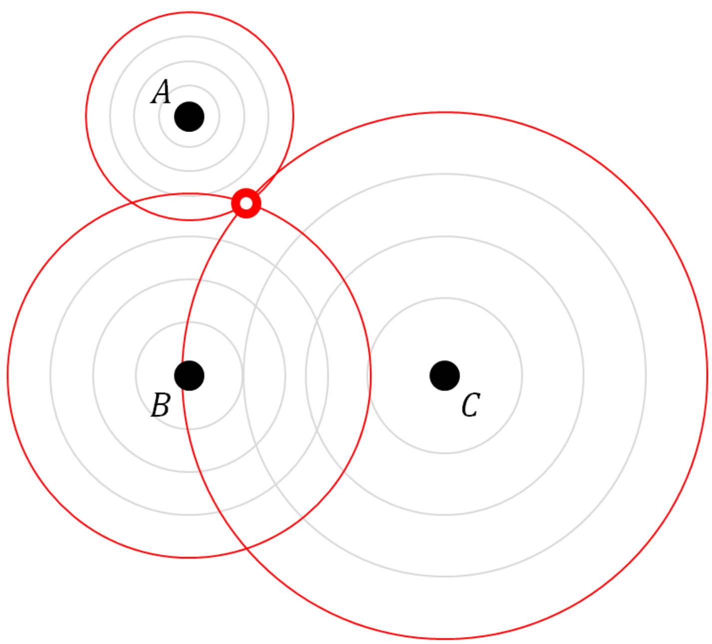


Figure 3.7. Changing the relative rates of dilation changes the weight or relative importance of each set. Smaller dilation corresponds to stricter constraints.

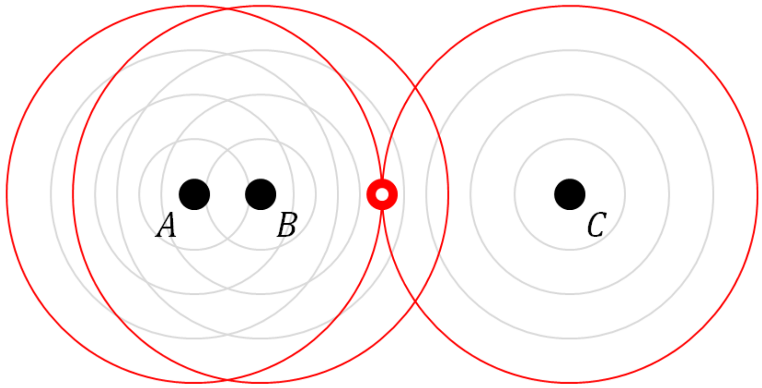


Figure 3.8. A degenerate case where the dilated solution (red dot) is not the circumcenter. The point B is within the SEB and becomes a satisfied constraint. In this situation only A and C determinet he solution's location.

This process is depicted for a ball kernel geometrically in Figure 3.9. If another kernel is used, then the truncation value r will depend on the constraint set and orientation and thus encouraging dilation of the constraint sets instead.

The simplest method for identifying r is through binary search. Alternating projections is performed to convergence, tracking the length of the limit cycle. If the dilation is too small, all the constraints do not intersect and the limit cycle will be nonzero as in Figure 3.2, necessitating an increase to r. If the limit cycle converges to near zero, the dilated sets overlap and r is decreased. Alternatively to binary search, the remaining length of the limit cycle can be used to approximate the necessary increases to r, acting as a "drawstring" in order to identify the minimax solution. However we found that binary search outperformed these methods at larger scales.

The truncation term shares similarities to the relaxation parameter in standard POCS to improve convergence. However, the dilated solution is a profoundly different result as no action is taken if the current solution already satisfies the given dilated objective. This overcomes the Least-Squares inclination towards the center-of-mass.

Furthermore, the size of the kernel applied to each constraint set is an inverse reflection on the weight or relative importance of that set. In general, the larger the kernel applied, the larger the resulting convolved constraint set, and thus the less strict the requirement to satisfy that particular constraint as depicted in Figure 3.7.

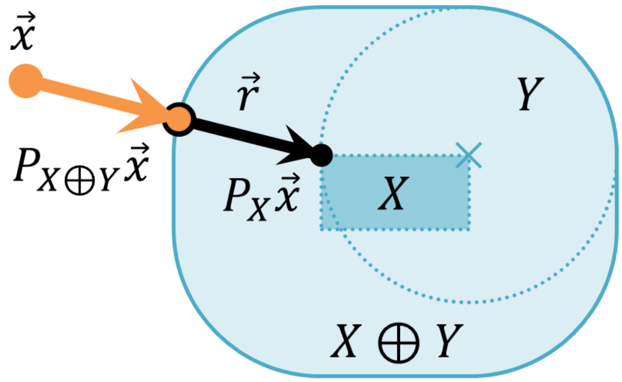


Figure 3.9. The unweighted rectangular constraint X is dilated by a disk Y to give the dilated constraint $X \oplus Y$. In the situation where Y is a ball, a projection onto the dilated set $P_{X\oplus Y}(\vec{x})$ is the projection onto $P_X(\vec{x})$ minus $\vec{r}$ from $\vec{x}$. Since Y is a ball $B(\vec{0}, r)$, then $\|\vec{r}\| = r$ or the radius of Y.

Diffraction Synthesis

Background

Diffraction synthesis can be solved through an alternating projections approach that produces a Least-Squares solution. The goal of diffraction synthesis is to design an aperture that will produce the desired images at given target planes, as shown in Figure 3.10. Each target plane represents a constraint set, and each image plane can be calculated from the method of angular spectrum. This allows for modeling both the forward and reverse propagation of the aperture plane. We model a Helium-Neon laser at 6328Å, over a 1cm × 1cm aperture.

In practice, diffraction synthesis is typically resolved using a Gerchberg-Saxton algorithm [4] or its variants. Gerchberg-Saxton is a phase recovery algorithm used to recover the phase information of the magnitude-only image. Here, the algorithm iterates between the source and target planes, imposing the desired image intensities while keeping the phase. While Gerchberg-Saxton algorithms will typically produce good results, the imposition of fixed magnitudes in not convex. The resulting images will be complex valued. To maintain the viability of POCS, we impose an approximate convex constraint of fixed real area, leading to degraded synthesis performance. The convex set S corresponding to the set of signals with constant real area is expressed by

$$S = \left\{ \vec{x} \;\middle|\; \sum_{i \in C} \Re(x_i) = \rho \right\}$$

where $\vec{x}$ is a complex vectorized image, C is the interval in question and ρ is the fixed area. A projection onto this set is given by

$$\left(P_S(\vec{x})\right)_i = \begin{cases} x_i & , \quad i \notin C \\ x_i - \dfrac{1}{N}(\rho_x - \rho) & , \quad i \in C \end{cases}$$

where

$$\rho_x = \sum_{i \in C} \Re(x_i)$$

and

$$N = \sum_{i \in C} 1.$$

The convex constraint sets implemented include bounded energy for the aperture and fixed real area for the pupils. For the aperture, signal intensity is projected down to the norm ball. This constraint is not strict, but a soft bound intended to counteract pupil planes adding energy into the signal. For each pupil, the shapes enclosed must have fixed real area. The sum of the real part of signals within the pupil must reach a fixed sum. This serves to prevent the degeneracy where all signals become zero. Projecting onto fixed real area involves adjusting the real part of each pixel value to achieve the desired sum simultaneously. Projecting onto the norm ball involves capping the magnitude of each pixel.

Method of Angular Spectrum

Signal propagation can be modeled through multiple methods. A general, computationally expensive method is through the Rayleigh-Sommerfeld diffraction, given by

$$E(x,y,z) = \iint_{u,v} \frac{ze^{-jkr}}{2\pi r^2}\left(jk + \frac{1}{r}\right) E(u,v,0)du\,dv$$

where $r = \sqrt{(x-u)^2 + (y-v)^2 + z^2}$ and the wave number $k = 2\pi/\lambda$, where λ is the wavelength of the aperture signal. A key issue here is the computation of r in the integral, which is prohibitively expensive for all but the simplest of examples. This leads to the Fresnel approximation,

$$E(x,y,z) = \frac{e^{-jkz}jk}{2\pi z} \iint_{u,v} e^{-\frac{jk}{2z}((x-u)^2+(y-v)^2)} E(u,v,0)du\,dv$$

The approximation is made for $z \gg x, y$ where z approaches r. The key benefit of this formulation is allowing the inversion to occur through the 2D Fourier Transform of the source plane $E(u,v,0)$ and a propagating function. In addition, the signal can be propagated backwards through z through an inverse propagating function. Thus the method of Angular Spectrum can be summarized as

1. Sample over the complex values of a cross-sectional area of the aperture plane in a grid pattern.
2. Take the 2D Fourier transform, decomposing the signal into its angular components.
3. Apply the propagating function to distance z and take the inverse Fourier Transform.

This method can be repeated for forward and backward propagations.

Synthesis Results

A comparison of Least-Squares and dilated projections are shown in Figures 3.11 and 3.12. In this example, the bat image is desired within 0.025m to 0.075m from the aperture, while a man image should appear in the distance at 1.0m. Standard POCS will result in the weighted average of these constraints, producing the Least-Squares solution seen in Figure 3.11 where the centroid is near the bat pupils. The result is higher projected distance to the man image. Heavily weighting the far image would be required to lower its error. In contrast, the dilated solution shown in Figure 3.12 treats each constraint equally and approaches a solution with the minimized maximum error. Details in the bat images are sacrificed for a clearer picture of the man. This demonstrates a potential weakness to Least-Squares methods, as satisfying one bat pupil in the near-field region will likely satisfy the others, thereby making the other, similar constraints redundant. The centroid nature of Least-Squares depicted in Figure 3.3 results in these constraints being overemphasized, pulling emphasis away from the farther-field man image. The minimax solution weights all the constraints equally without operator foreknowledge in order to find a solution balanced between constraint sets and enhances details in the underrepresented final man image. Since the bat image has three times the representation of the man image, the Least-Squares solution has a third the maximum Mean-Squared Error (MSE) in reconstruction of 0.011 to 0.033. With the minimax approach, the maximum MSE is suppressed to 0.014. Distortion caused by projecting onto convex pupils of fixed area instead of non-convex fixed magnitude leads to non-uniform images of varying intensities. An alternative convex constraint is presented in Figure 3.13.

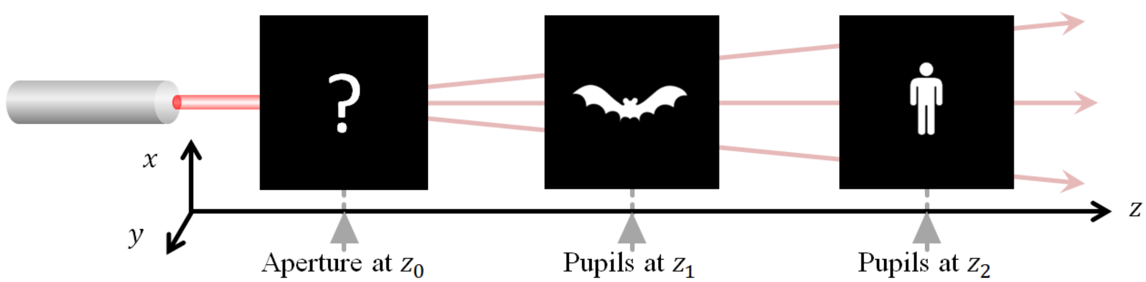


Figure 3.10. The diffraction synthesis problem involves designing an aperture at z_0 that will give the desired diffraction pattern or image at fixed propagation distances in the near to far-field. The x and y-coordinates refer to the aperture or image plane and z represents the propagated distance. The pupil at z_1 should appear like a bat and like a man at z_2.

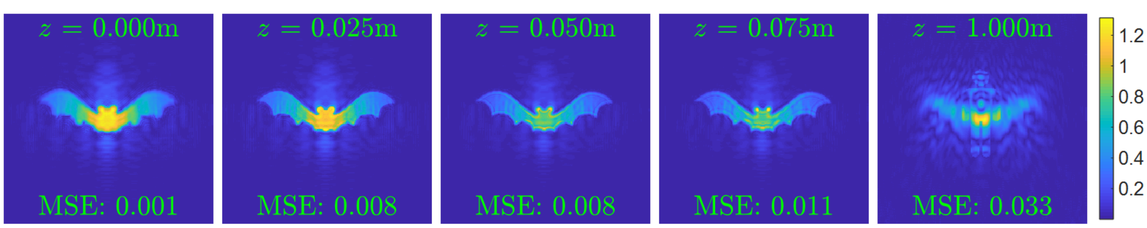


Figure 3.11. Normalized images of the simultaneous POCS approach results in the Least-Squares solution to the diffraction synthesis problem. A bat image is required for the near-field and a man in the far field. Due to repeated, strict reinforcement of the bat image in the near-field, the resulting solution deemphasizes the far-field pupil.

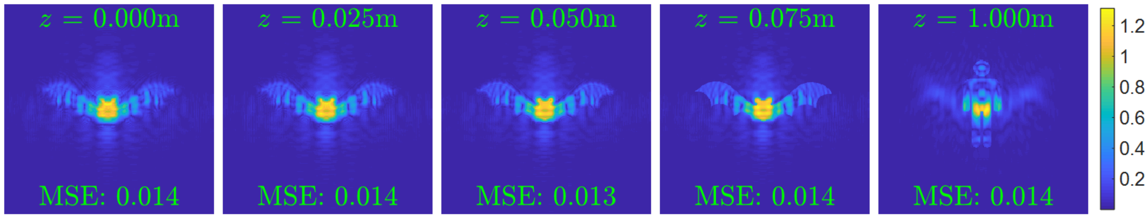


Figure 3.12. Normalized images of the dilated approach minimizes the maximum error all constraints, improving the far-field image of a man at the expense of the closer bat images.

In a communications situation it may be advantageous to broadcast a signal around an obstacle. Here, the image of a watertower is introduced at 0.3m as an obstacle to avoid. The resulting synthesized signals under Least-Squares and minimax POCS are shown in Figures 3.14 and 3.15. The propagated signal must avoid an obstacle at 0.3m, in the form of a watertower silhouette. In this case, the additional constraint projects all values within the watertower image to zero, ensuring no energy passes through it. In the dilated case, this corresponds to allowing no dilation of the obstacle constraint: the solution must lie within that convex set. Similar to the previous example, the bear image has twice the representation of the hand image. Thus, the Least-Squares solution favors the bear with an MSE of 0.004 to the hand with an MSE of 0.009. For minimax, the max MSE is suppressed to 0.005.

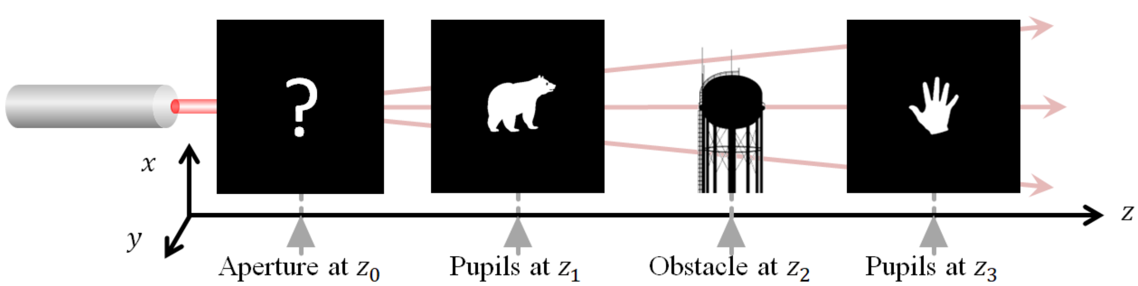


Figure 3.13. This problem involves an obstacle placed in the path of a desired target pupil. The synthesized aperture should avoid the obstacle and form the image behind it.

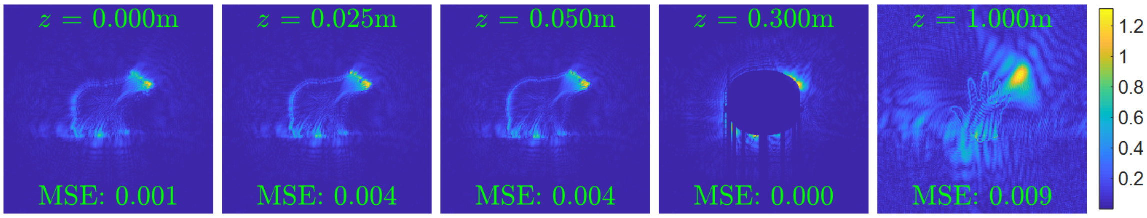


Figure 3.14. The normalized Least-Squares approach synthesizes an aperture to avoid the watertower at 0.3m. The hand image is sacrificed for improved performance on the two bear images.

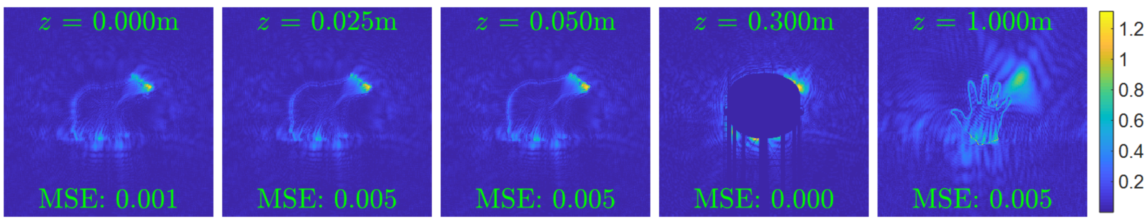


Figure 3.15. The normalized dilated approach avoids the watertower and improves the far-field hand pattern at the expense of the near-field bear images. The errors are distributed evenly between pupils as the algorithm is not given a reason to weigh one constraint more than another.

Computed Tomography

Background

We apply dilated kernels to the issue of medical imaging via Computed Tomography (CT) reconstruction. Here, a target object is imaged at various angles resulting in projection slices corresponding to the time-delay of the scan passing through the imaged body from specific viewpoints. These projections from various angles form the object's profile or sinogram as shown in Figure 3.16, where the target object used is a 100×100 image of the modified Shepp-Logan phantom. A beam path matrix is used to model this interaction. Thus, the problem of medical image reconstruction from a sinogram is synonymous with matrix inversion. However, the problem becomes large-scale with increasing pixel resolution, necessitating efficient, parallelizable inversion techniques. POCS-based solutions such as Algebraic Reconstruction Technique (ART) [16], [2], Simultaneous Iterative Reconstruction Technique (SIRT) and Simultaneous Algebraic Reconstruction Technique (SART) [3] are based on alternating and averaged projections. Alternatively, Filtered Back-Projection (FBP) can also yield good results by back-projecting the sinogram under a ramp filter. Due to issues with noise and streak artifacts, FBP is generally replaced with iterative methods in practice. Nevertheless, FBP is a fast, non-iterative process that can be used to initialize a POCS inversion.

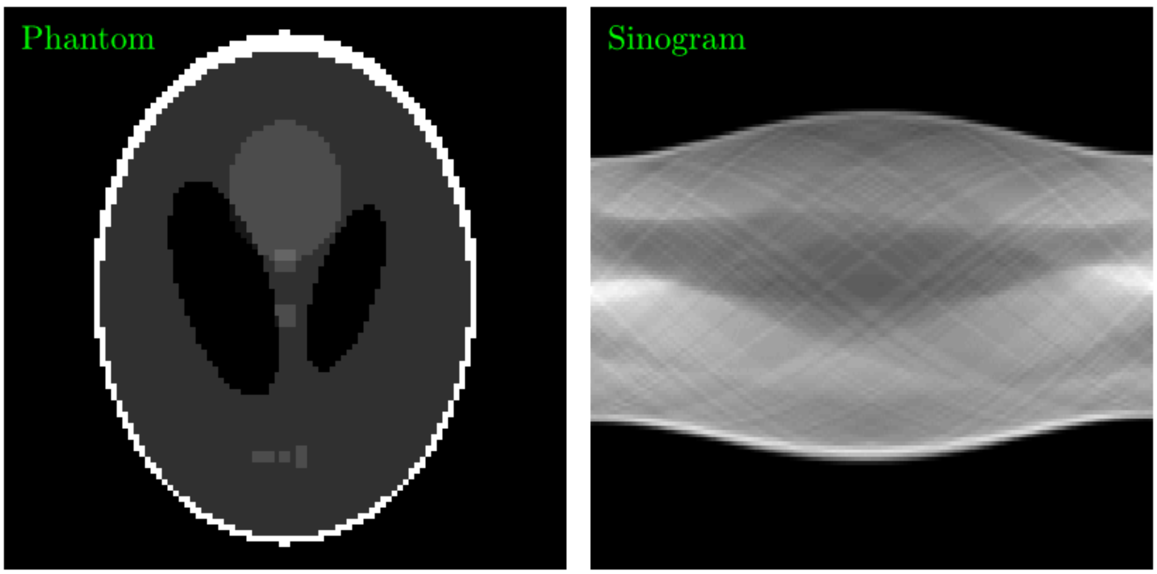


Figure 3.16. A 100×100 modified Shepp-Logan phantom and its corresponding sinogram, sampled at 180 one-degree angles. The Shepp-Logan phantom is a popular test image for image construction representing the features of the human brain and skull. Here, the phantom used is a normalized greyscale image within [0,1].

For an $m \times n$ path matrix, n corresponds directly with the number of pixels in the image, while m is the product of the length of projection bins and the number of angles tested. For the simple 100×100 phantom in Figure 3.16 using 180 one-degree slices into 1-pixel wide bins results in a sparse matrix with dimensions 26100×10000. In the following examples SART is used to form the Least-Squares solution while alternating projections onto dilated sets is used for the minimax solution. Image recovery error is measured with respect to the given sinogram and the reconstructed image's sinogram, not the reconstructed image itself.

Box-Dilated Sinogram

This CT example demonstrates how box-dilation is situationally advantageous to ball kernels. Rather than truncate the projected step, the sinogram constraint may be dilated instead. The vectorized sinogram can be dilated horizontally for lateral movement of the imaged object, or vertically for noise. An exclusively vertical dilation of the sinogram is the classic minimax error problem. A slice of this noisy sinogram for the Shepp-Logan phantom reconstruction is seen in Figure 3.17. This box-dilation can be reshaped to emphasize noise or lateral movement by removing slack in the other dimension.

CT Reconstruction Results

The results of the dilated algorithm are compared with a Least-Squares technique and FBP. The recovered images and sinograms are displayed in Figures 3.18 and 3.19. Figure 3.18 demonstrates the slight advantage of dilated kernels for zero-mean Gaussian noise. The Least-Square algorithm minimizes the MSE of the reconstructed image's sinogram to the observed sinogram, reducing the FBP MSE from 1.6175 to 0.8465, but this does not translate to coherent recovered images. Projecting onto the dilated sets yielded advantageous results, despite a significantly higher MSE of 1.5338. Figure 3.19 introduces lateral shifts in addition to zero-mean uniform noise to the sinograms, resulting in the application of box kernels for dilation. For the Shepp-Logan phantom, the Least-Squares approach improves the FBP MSE from 3.2632 to 2.4062. The dilated approach raises the MSE to 3.0848 but result produced a grainy image with sharp edges and suppressed streak artifacts.

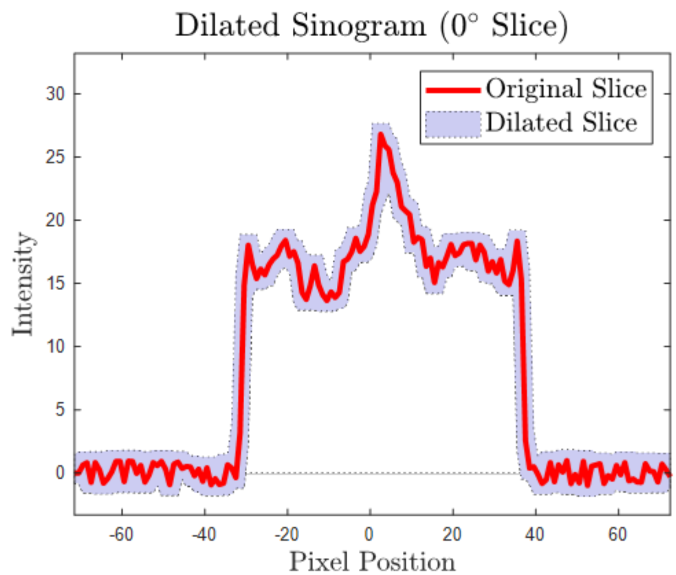
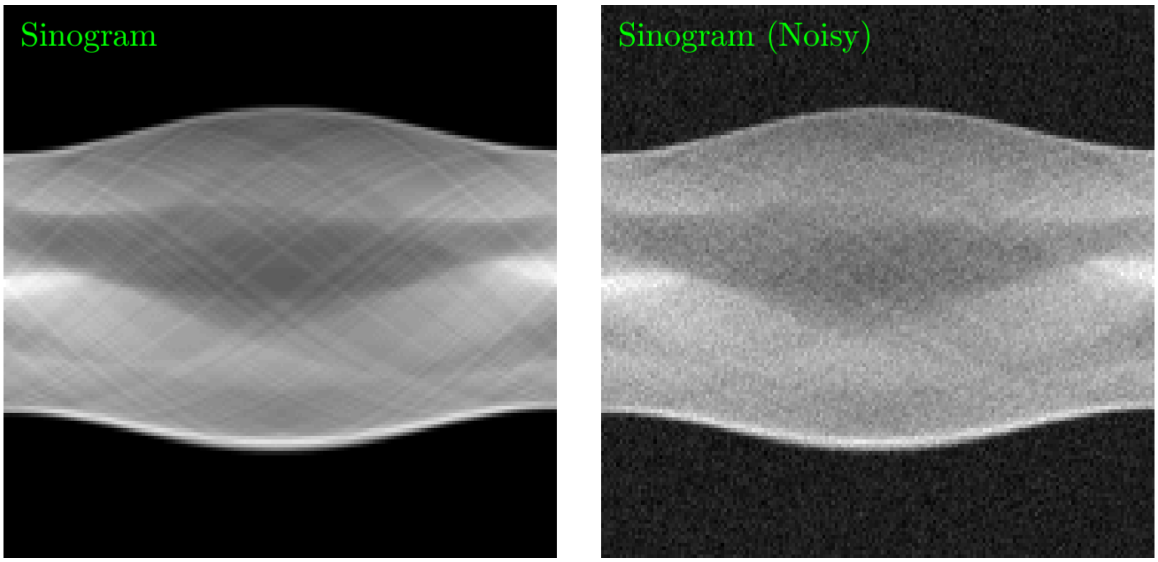


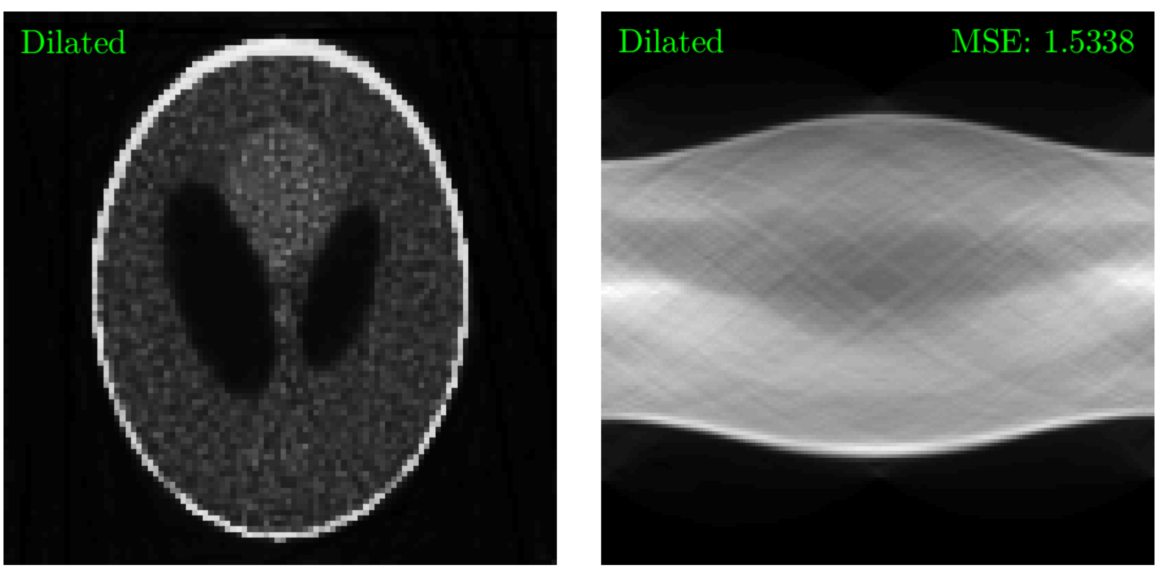
Figure 3.17. The 0° slice of the sinogram, dilated with a box corresponding to lateral vibration and signal noise. The noise dilation changes with each iteration as the solver ascertains the noise level while maximum lateral movement is constrained. Rather than project onto the given (red) sinogram itself, the current iteration of the reconstructed image is projected to within the boundaries of the dilated slice instead.

Conclusion

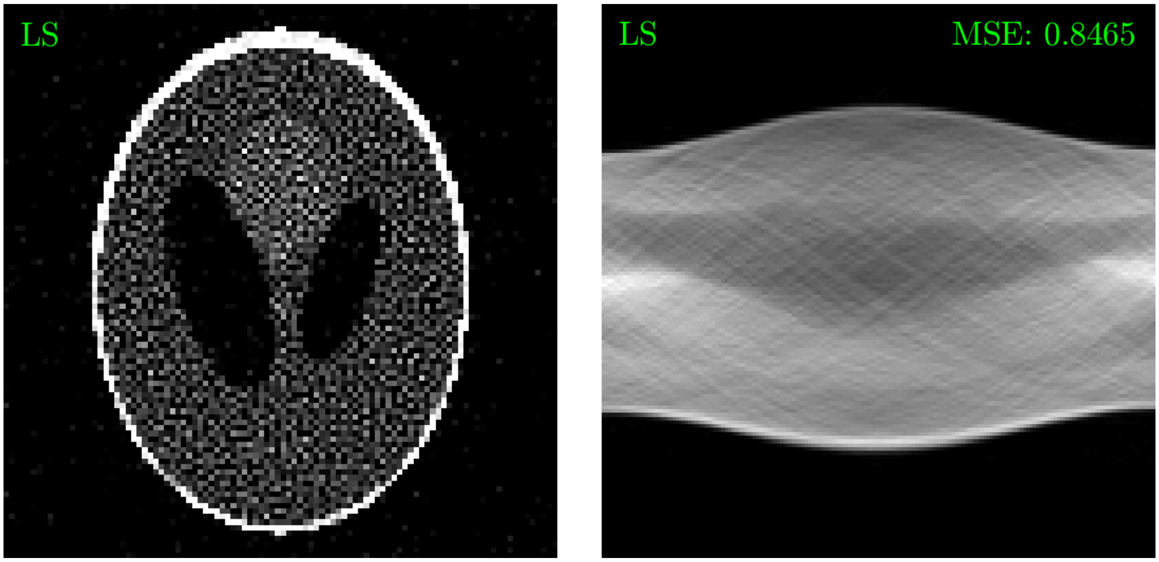
Dilated projections can enhance the outcome of POCS in situations where multiple, non-intersecting convex constraint sets are dictated. Instead of converging to a Least-Square error, dilated projections can achieve minimax weighted error solutions that can have applications in diffraction synthesis and tomographic image reconstruction. Dilation subsumes convex constraints that are easily satisfied, removing their grander impact on the solution and avoiding a centroid bias. Subsumed constraints can be skipped, improving the speed of the algorithm with iteration. Further improvement in algorithmic performance is needed to identify the ideal kernel beyond binary search. *Ad hoc* minimax solvers will outperform the dilated projections for specific applications, but dilated POCS has an advantage in versatility among diverse convex constraints.



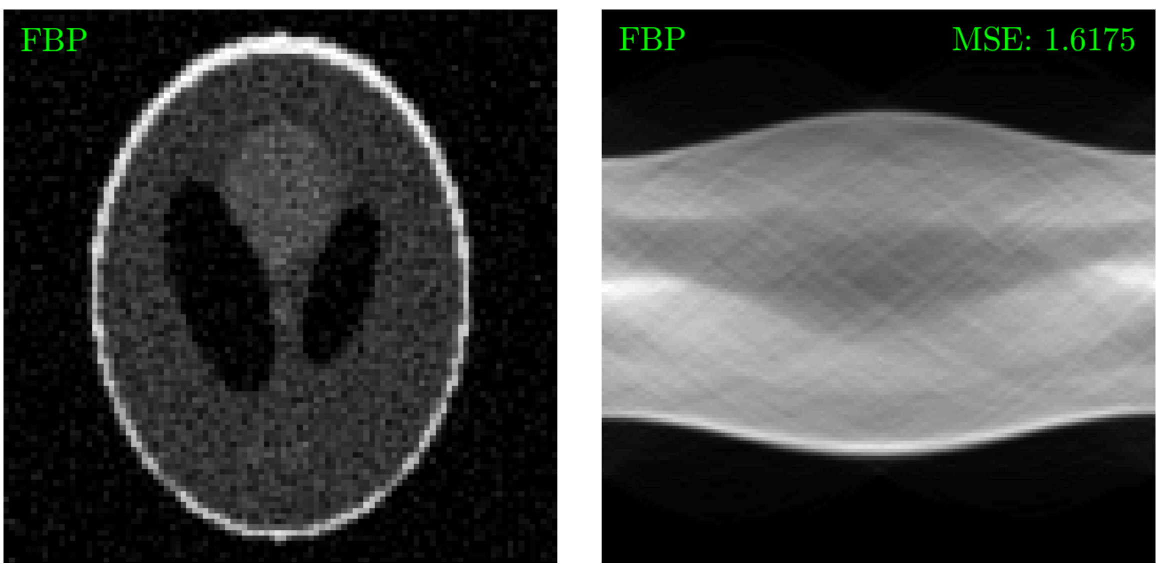
(a) Original (Unknown) and Noisy Sinogram (Given)



(b) Dilated Recovery Image and Sinogram

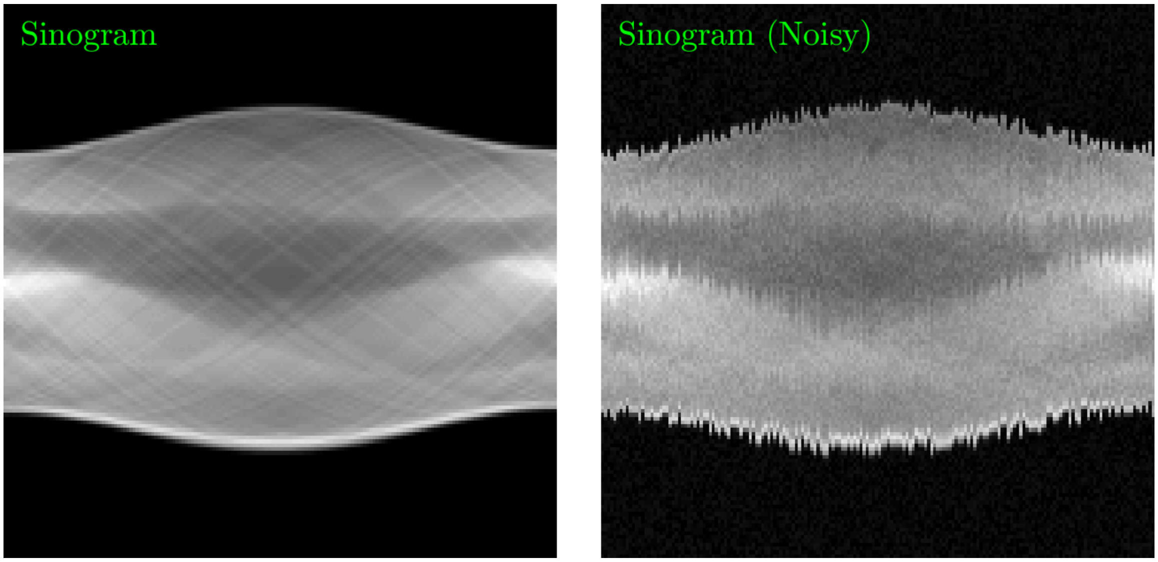


(c) Least-Squares Recovery Image and Sinogram

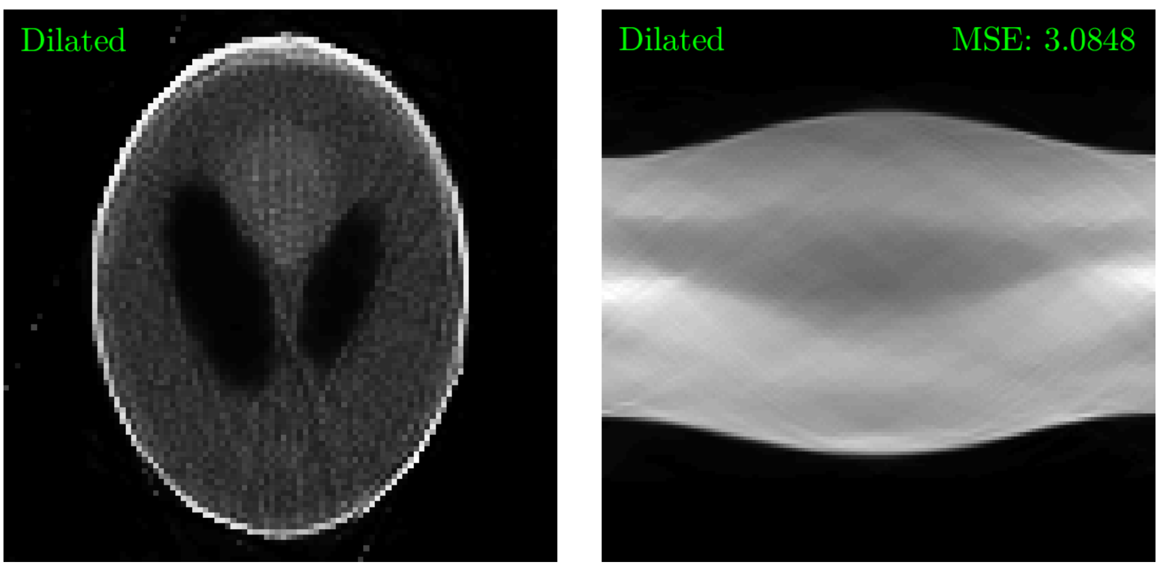


(d) Filtered Back-Projection Image and Sinogram

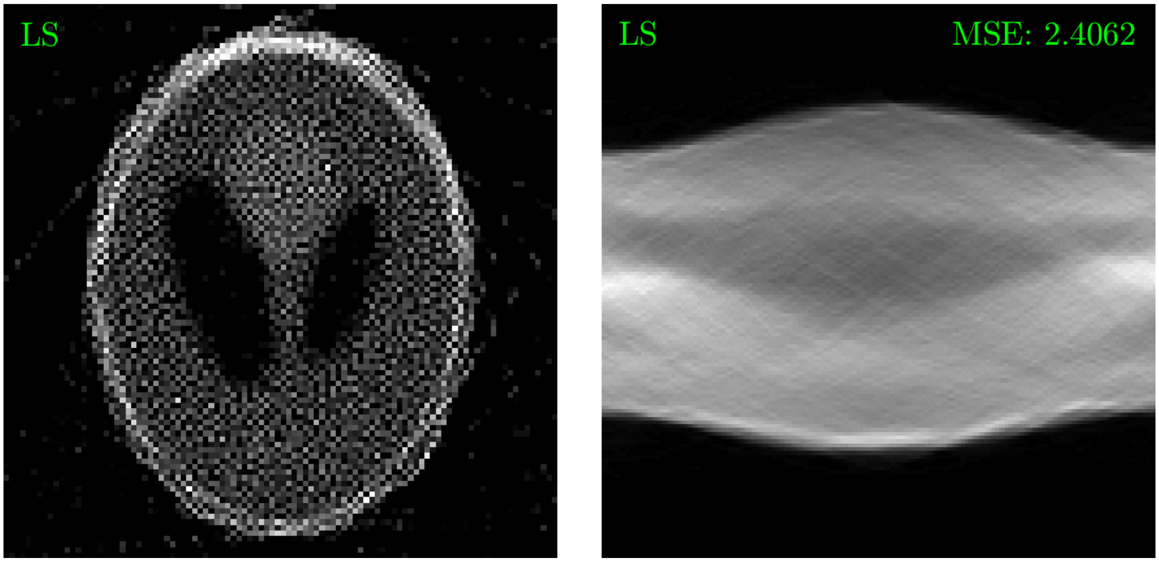
Figure 3.18. The Shepp-Logan phantom sinogram (a) is corrupted by Gaussian random noise in with $\sigma = 1.0$ and is used to reconstruct an image using (b) dilation, (c) Least-Squares algorithms, and (d) filtered back-projection. The Least-Squares sinogram is the closest L^2-norm to the given noisy sinogram, but the dilated result has the cleanest edges and features.



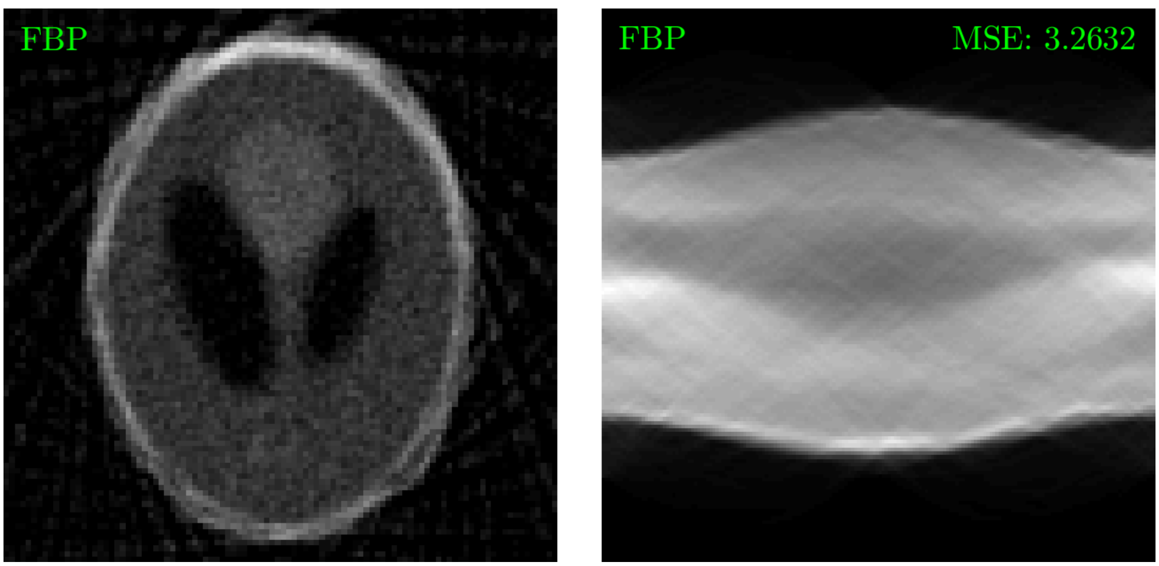
(a) Original (Unknown) and Noisy Sinogram (Given)



(b) Dilated Recovery Image and Sinogram



(c) Least-Squares Recovery Image and Sinogram



(d) Filtered Back-Projection Image and Sinogram

Figure 3.19. The Shepp-Logan phantom sinogram (a) is corrupted by uniform random noise in [-1, 1] and uniform random lateral motion in [-2, 2] pixels. Reconstructions are made using (b) dilation, (c) Least-Squares algorithms, and (d) filtered back-projection. The Least-Squares sinogram has the lowest MSE, but the dilated result's image has the cleanest features. Streak artifacts from the target's motion are seen in the Least-Squares and FBP reconstruction.

CHAPTER FOUR

Associative Memory

Introduction

POCS may be applied to associative memories [2]. The goal of associative memory is to determine if and where a given signal segment matches a signal in a library of templates. For example, a partial or cropped snapshot of a person may be compared to a prepared library of suspects in order to identify the target. Given a library of vectors $\vec{a}_0, \vec{a}_1, \ldots$ and a vector segment $\vec{b}$ associated with the library, the goal is to identify the library vector that is closest to matching $\vec{b}$.

$$\begin{array}{c} (\vec{b}) \\ \downarrow \\ - \\ 1 \\ 2 \\ 1 \\ - \\ 3 \\ 2 \\ - \end{array} \qquad \left(\begin{array}{cccc} \vec{a}_0 & \vec{a}_1 & \vec{a}_2 & \vec{a}_3 \\ & & \downarrow & \\ 3 & 3 & 1 & 5 \\ 1 & 2 & 1 & 1 \\ 2 & 4 & 2 & 3 \\ 3 & 1 & 1 & 2 \\ 1 & 1 & 4 & 4 \\ 4 & 1 & 2 & 1 \\ 1 & 3 & 2 & 5 \\ 5 & 2 & 5 & 4 \end{array} \right).$$

In this example, the dashes represent the unknown segments of $\vec{b}$, and $\vec{a}_2$ is the closest match. The error metric is often implemented through a Least-Squares approach in order to minimize the squared difference between the given signal segment $\vec{b}$ and each vector in the library. This can be expressed as

$$D(\vec{a}_j) = \sum_{i \in C} \left| \vec{b}_i - (\vec{a}_j)_i \right|^2 \qquad (4.1)$$

where C is the known indices of $\vec{b}$. The optimization becomes

$$\underset{j}{\operatorname{argmin}} D(\vec{a}_j). \tag{4.2}$$

This is a Least-Squares optimization that can be determined through POCS. However, dilated POCS can also be performed in order to arrive at a minimax solution.

POCS Implementation

POCS can be used to perform associative memory in order to arrive at a Least-Squares solution. One way of implementing POCS is to project onto the affine space defined by the library set. For a library A composed of n images,

$$A = [\vec{a}_0 \quad \vec{a}_1 \quad \cdots \quad \vec{a}_{n-1}]$$

where $\vec{a}_i$ is a column vector formed by vectorized images in the library. The update on a reconstruction $\vec{x}_k$ is given by

$$\vec{x}_{k+1} \quad = \quad \vec{x}_k + a\Delta\vec{x}_k. \tag{4.3}$$

At steady state, the update step equals zero. For the library set, the update contribution is a projection onto the library set A, or

$$P \quad = \quad A(A^T A)^{-1} A^T. \tag{4.4}$$

Since A is constructed via a library of vectorized images, the size of A can make the computation of P prohibitively expensive. One possibility to resolve this is to invert via QR decomposition. Alternatively, (4.4) can be split into

$$P \quad = \quad AA^+ \tag{4.5}$$

where A^+ is the Moore-Penrose pseudoinverse of A, and can also be computed via POCS algorithms. The projection onto an image segment $\vec{b}$ can be performed via projection onto identical middles, as described previously in the Identical Middles section of Convex Signals in Chapter Two.

Image Library

An image library of 64 scientists, mathematicians and engineers is used. Each portrait is a 300×240 pixels in size, and vectorized to form a 72000×64 library matrix A. At these dimensions, the projection in (4.5) will be performed via POCS.

Figure 4.1. A base library of 64 portraits of various scientists, mathematicians and engineers serving as images for associative memory.

The Least-Squares difference between each image is shown in Figure 4.2. This matrix indicates which portraits are most similar and dissimilar. For example, the images of Marks, Garner, and Schubert are fairly similar as opposed to Liebniz, Moivre and Pascal. Table 4.1 depicts the top ten similar and dissimilar portraits as characterized by the equation in (4.1). The key dissimilarity is the periwigs worn.

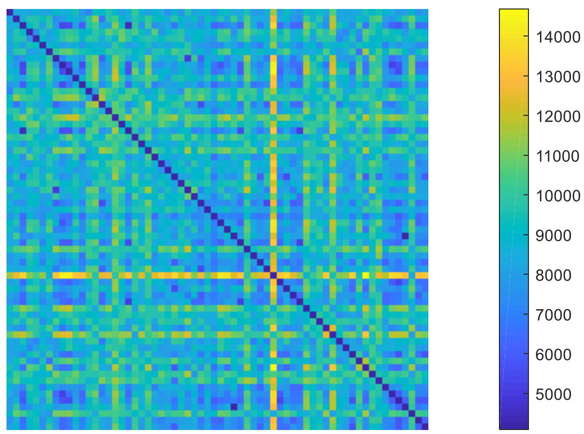


Figure 4.2. A 64 × 64 matrix indicating the Least-Squares distance between each image. The minimums indicate portraits that are near or similar to each other, while larger values indicate a greater deviation between images.

Table 4.1. Top 10 similar and different portraits.

Source	Top 10 Similar	Top 10 Dissimilar
Marks	Garner, Planck, Pythagoras, von Neumann, Laplace, Rutherford, Pareto, Markov, Venn, Kalman	Liebniz, Pascal, Euclid, L'Hopital, Moivre, Poisson, Taylor, Bessel, Radon, Ampere
Schubert	Koziol, Rutherford, Venn, Markov, Kalman, Garner, Allman, von Neumann, Lebesgue, Maxwell	Liebniz, Pascal, Moivre, L'Hopital, Fermat, Euclid, Newton, Radon, Euler, Taylor
Koziol	Schubert, Kalman, Rutherford, Markov, Venn, von Neumann, Ampere, Garner, Lebesgue, Allman	Liebniz, Pascal, Moivre, Fermat, Euler, Euclid, L'Hopital, Newton, Al Khwarizmi, Radon
Blair	Godel, Hilbert, Einstein, Riemann, Maxwell, Lebesgue, Rutherford, Kepler, Cantor, Markov	Liebniz, Pascal, L'Hopital, Fourier, Radon, Moivre, Descartes, Euclid, Taylor, Newton
Garner	Marks, Laplace, Pythagoras, von Neumann, Ramanujan, Rutherford, Allman, Kalman, Schubert, Markov	Liebniz, Pascal, Euclid, Moivre, L'Hopital, Poisson, Fermat, Bessel, Taylor, Napier

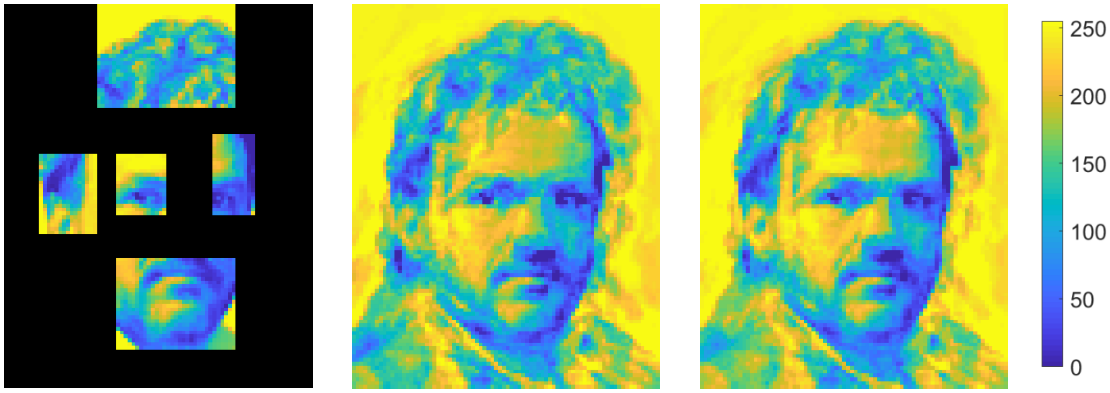
Image Synthesis

Both Least-Squares and minimax POCS algorithms will perform associative memory. The differences between these two methodologies appear when conflicting seeding image segments are used. Figure 4.3 shows the reconstruction that occurs when five different seeds are used: Mark's hair, and Liebniz's periwig, Garner's right eye, Taylor's left eye, and Koziol's mouth. The synthesized images are very similar with subtle differences in the eyes and expressions. In particular, Least-Squares produced an image with eye placements close to the seeded images, while minimax introduces a new eye from the library that is a closer match to the other.

(a) Conflicting Seeds (b) Minimax (c) Least-Squares

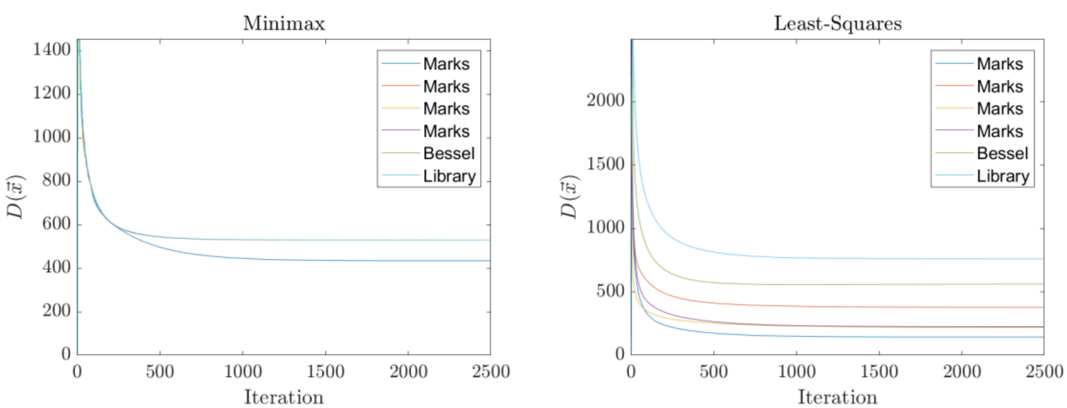
Figure 4.3. (a) Five conflicting sections consisting of Marks, Liebniz, Garner, Taylor, and Koziol. (b) A minimax and (c) Least-Squares synthesis.

The effect of the centroid on the Least-Squares solution is also apparent in the synthesized images. Figure 4.4 is composed of one section of Bessel's hair, and four of Mark's face. The minimax solution produces a gradual transition from face to hair, while the Least-Squares solution is biased towards the face, resulting in clear transition between constraint sets. A plot of errors with iteration is shown in Figure 4.5.



(a) Conflicting Seeds (b) Minimax (c) Least-Squares

Figure 4.4. (a) Five conflicting sections, one of Bessel and four of Marks. (b) A minimax and (c) Least-Squares synthesis.



(a) Minimax (b) Least-Squares

Figure 4.5. A plot of errors with iteration for (a) minimax and (b) Least-Squares. Both algorithms converge near 1000 iterations.

Summary

Minimax POCS can be applied as an alternative to Least-Squares methods for synthesizing images from a template library using conflicting seeds. Here, the convex sets chosen were the affine space defined by the library and the identical middles defined by the seed images and reiterating the advantages of a general projection algorithm for including the variety of potential convex constraints. The minimax solution exhibits deference to the centroid, which can be useful when addressing images libraries with varying levels of similarity. The resulting synthesized images tend to exhibit a greater degree of blending, leading to a more cohesive reconstruction.

CHAPTER FIVE

Ambiguity Function Magnitude Inversion

This chapter is an unpublished manuscript by: A. R. Yu, C. Baylis, and R. J. Marks II, "Regularized Phase Recovery for Ambiguity Function Magnitude Inversion."

Introduction

This manuscript investigates a regularized methodology for inverting the ambiguity function magnitude into valid spawning signals. The work is mentored by Dr. Robert J. Marks II who posed the use of Gerchberg-Saxton methods for phase recovery. Dr. Charles Baylis and Dr. Marks developed the ambiguity function inversion with full phase information, and Albert Yu developed and implemented the magnitude-only inversion.

The ambiguity function (AF) [37], [38], [39], [40], [41], [42], [43], [44] results when a matched filter is used to simultaneously determine the range and Doppler of a transmitted radar signal and is vital in identifying a target's position and velocity. Ideally, a chosen radar signal will yield an AF with a magnitude that is strictly an unambiguous Dirac point mass, but there are no signals that yield such an AF. Thus a tradeoff exists between precisely determining a target's range and Doppler. Furthermore, no unique inverse exists for mapping an AF back to an originating signal when only the AF magnitude is known, as the transform is not one-to-one. Acquiring the desired AF properties is frequently described as requiring the prior knowledge and expertise of the radar designer [45] who must choose a potential signal and check the AF response for the necessary characteristics. This has led to the area of AF synthesis [46], [47] where a designer iteratively adjusts a source signal to give a desired AF response by inspection. However, these methods do not necessarily result in inversions, but signals that yield an AF within the given constraints.

Full inversion to a family of potential spawning signals is possible when both magnitude and phase of the AF are known [37], [39]. The phase information of the AF is critical for such inversions. We present a methodology for inverting the Woodward AF magnitude to a valid source signal using a modified version of the Gerchberg-Saxton algorithm for phase recovery [4], [2]. The magnitude-phase AF inversion algorithm outlined in [48] is refined with a regularized window to account for the simplified inversion and reduce cumulative phase effects of random initializations in reconstruction, removing the apparent high-frequency components and drastically improving convergence. This procedure will give a radar operator direct control over the specification of the desired AF.

The Woodward Ambiguity Function

The ambiguity function examines the correlation between a source template signal and Doppler shifted received signals. The Woodward ambiguity function [44] of a potentially complex baseband signal $s(t)$ is given by

$$\chi(\tau, u) = \int_{-\infty}^{\infty} s(t)s^*(t-\tau)e^{-j2\pi ut}dt \tag{5.1}$$

where τ is the propagation delay and u is the relative Doppler shift between the target and receiver. We adopt the shorthand notation of (5.1) by

$$s(t) \Rightarrow \chi(\tau, u) \tag{5.2}$$

Other definitions similar to (5.1) exist that result in the same AF magnitude [39].

Motivated by the notation in (5.2), the inverse of the AF is

$$\chi(\tau, u) \Leftarrow s(t). \tag{5.3}$$

We now show the inverse in (5.3) is unique to within a constant phase.

Full Inversion with Known Phase

Given both phase and magnitude, the ambiguity function can be inverted to a spawning signal. We generalize the inversion given by Eustice et al. [39]. The full ambiguity function can be expressed in terms of the source signal $s(t)$ via the inverse Fourier Transform of (5.1),

$$s(t)s^*(t-\tau) = \int_{-\infty}^{\infty} \chi(\tau, u) e^{j2\pi ut} du. \tag{5.4}$$

Choose a specific fixed time $t = \xi$ where the signal $s(t)$ is nonzero. This requires some foreknowledge on the part of the radar signal designer. Our inversion algorithm, presented later, will automatically account for this value. Setting $\tau = 0$ yields

$$|s(\xi)|^2 = s(\xi)s^*(\xi) = \int_{-\infty}^{\infty} \chi(0, u) e^{j2\pi u\xi} du. \tag{5.5}$$

The phase component of $s(\xi)$ is unknown, but the magnitude is given by

$$|s(\xi)| = \left(\int_{-\infty}^{\infty} \chi(0, u) e^{j2\pi u\xi} du \right)^{\frac{1}{2}}. \tag{5.6}$$

This reveals the magnitude of $s(t)$ at the specific time $t = \xi$. In polar form $s(\xi) = |s(\xi)| e^{j\angle s(\xi)}$ so (5.4) can be written as

$$s^*(\xi-\tau) = \frac{e^{-j\angle s(\xi)}}{|s(\xi)|}\int_{-\infty}^{\infty}\chi(\tau,u)e^{j2\pi ut}du \tag{5.7}$$

where $|s(\xi)|$ is the signel value scalar given in (5.6) and $\angle s(\xi)$ is the phase. This constant $\angle s(\xi)$ is not important since the corresponding ambiguity function is independent of its value. This can be seen from the definition of the ambiguity function in (5.1) where the $e^{j\angle s(\xi)}$ term in $s(t)$ is canceled by the $e^{-j\angle s(\xi)}$ term in $s^*(t-\tau)$. The constants $|s(\xi)|e^{j\phi}$ with ϕ, an arbitrary real phase, results in the same ambiguity function as when using the single complex number $s(\xi)$.

The inversion result in (5.7) shows the function $s^*(-\tau)$ centered at ξ. The value of ξ must be chosen such that $|s(\xi)| \neq 0$. Conjugating (5.7) with $t = \xi - \tau$ gives the final expression for $s(t)$

$$s(t) = \frac{e^{j\phi}}{|s(\xi)|}\int_{-\infty}^{\infty}\chi^*(\xi-t,u)e^{-j2\pi u\xi}du$$

or, using (5.6) and the notation in (5.3)

$$\chi(\tau,u) \Leftarrow s(t) = \frac{e^{j\phi}\int_{-\infty}^{\infty}\chi^*(\xi-t,u)e^{-j2\pi u\xi}du}{\left(\int_{-\infty}^{\infty}\chi(0,u)e^{j2\pi u\xi}du\right)^{\frac{1}{2}}}. \tag{5.8}$$

Transformations that Preserve Magnitude

The inversion in (5.8) requires both the phase and the magnitude of the ambiguity function. In practice, the phase is discarded as only the magnitude of the ambiguity function is needed to assess the trade-off between simultaneous measurement of range and Doppler. Inversion when only the magnitude is known is a more difficult problem, as many signals can yield the same AF magnitude. Using the notation in (5.2) we can show for example that:

- The magnitude of the AF if a shifted signal is the same as the AF of signal unshifted.

$$s(t - t_0) \Rightarrow \chi(\tau, u)e^{-j2\pi u t_0}. \tag{5.9}$$

- Likewise, any signal with a linear change in phase will yield the same ambiguity function magnitude.

$$s(t)e^{j(kt-\phi)} \Rightarrow \chi(\tau, u)e^{-jk\tau}. \tag{5.10}$$

  An advantage for real signals is that this linear change along $u = 0$ of the ambiguity function corresponds directly to $e^{-jk\tau}$, and can easily be approximated and removed.

- A signal's transposition gives

$$s(-t) \Rightarrow \chi(\tau, -u).$$

  If $s(t)$ is real, a signal transposition also gives the same ambiguity function magnitude, because

$$\chi(\tau, -u) = \chi^*(\tau, u). \tag{5.11}$$

An inversion of the AF magnitude is not unique. Therefore we will settle for any well-behaved function that produces the desired AF magnitude. Our proposed method for recovering a valid spawning signal to achieve a desired AF magnitude is based on a modified version of the Gerchberg-Saxton algorithm for phase recovery.

Gerchberg-Saxton Algorithm

The Gerchberg-Saxton Algorithm [8], [9], [14] is an iterative algorithm for image phase recovery originally used in optics for finding phase given the image intensities on two planes separated by the act of diffraction. The Fourier Transform and its inverse are used as the propagation function into the far-field plane. The Fourier transform of $r(t)$ is $R(u)$

$$r(t) \rightarrow R(u) = \int_{-\infty}^{\infty} r(t) e^{-j2\pi ut} dt \tag{5.12}$$

and the inverse Fourier Transform of $Q(u)$ is $q(t)$

$$Q(u) \rightarrow q(t) = \int_{-\infty}^{\infty} Q(u) e^{j2\pi ut} du. \tag{5.13}$$

The Gerchberg-Saxton Algorithm seeks to recover a signal $r(t) = |s(t)|e^{j\angle r(t)}$ and therefore $R(u) = |S(u)|e^{j\angle R(u)}$ given only the magnitudes $|s(t)|$ and $|S(u)|$. The algorithm is summarized in Figure 5.1. A signal is iteratively Fourier transformed and inverse transformed and, in each domain, the changing phase is kept and the known magnitudes $|s(t)|$ and $|S(u)|$ are imposed. Although the algorithm has not been proven to converge, implementations are often successful.

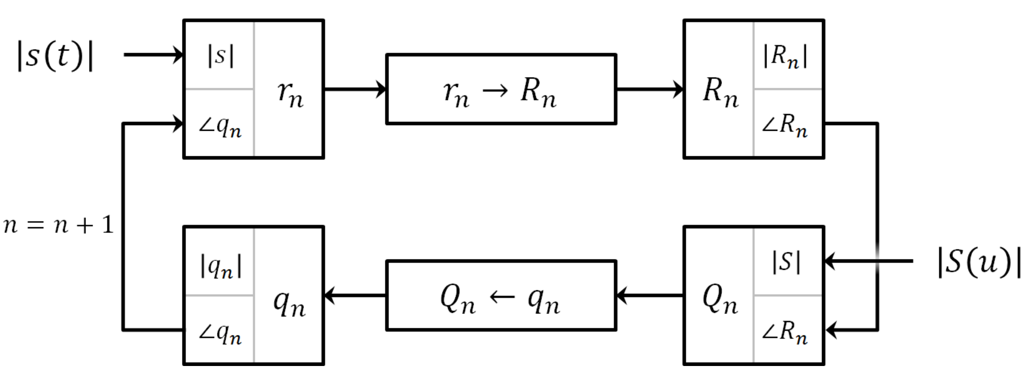


Figure 5.1. An illustration of the Gerchberg-Saxton algorithm. The algorithm initializes $r_0(t)$ with the given magnitude $|s(t)|$ and a random phase $\angle q_0$.

Magnitude-Only Inversion

Consider a target ambiguity function magnitude $\chi(\tau, u)$. The goal is to find any signal $r(t)$ such that

$$r(t) \Rightarrow \chi_r(\tau, u) = |\chi(\tau, u)| e^{j\angle\chi_r(\tau,u)}.$$

The AF magnitude of $r(t)$ will also match under any translation and linear phase shift, and may also include reflections in time if $r(t)$ is purely real. Thus, a given AF magnitude can correspond to a signal arbitrarily translated in time. However, by choosing a value forξ, the inversion is limited to the general proximity around it. The length of this proximity can be derived from the given AF magnitude itself by examining its non-zero elements in t, as demonstrated in Figure 5.2. As a result, the inversion only needs to examine the reconstruction about ξ within this bandwidth, as shown in Figure 5.3.

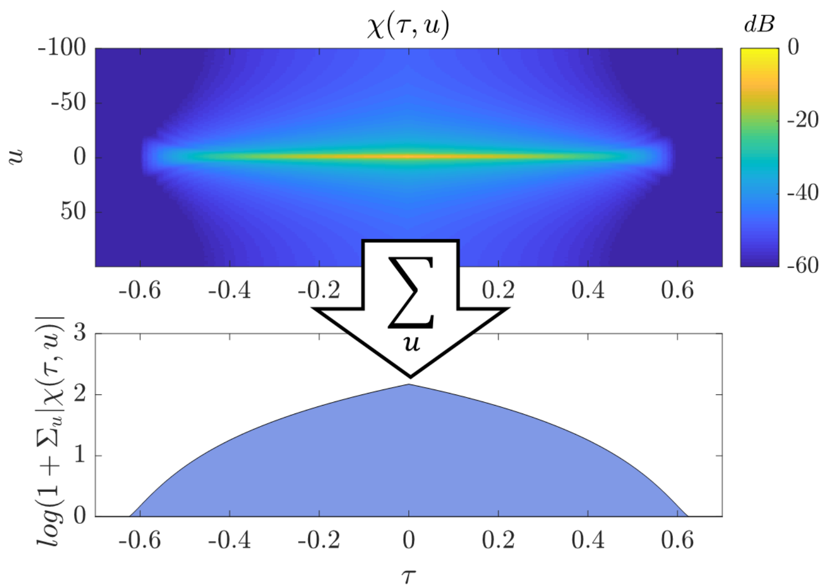


Figure 5.2. The given AF magnitude can be used to estimate the time length T samples of a spawning signal. In this example, a summation along u of the AF magnitude reveals a nonzero bandwidth window of $2T - 1$ samples. The original signal must be T samples in total length.

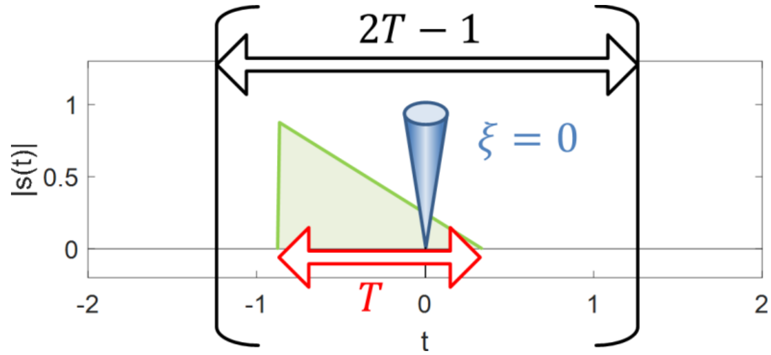


Figure 5.3. By fixing $\xi = 0$ in the inversion, a length T sample signal is pinned in translation and limited to the $2T - 1$ sample region centered about the origin. With the random initial phase, the reconstruction can occur anywhere within this window each iteration.

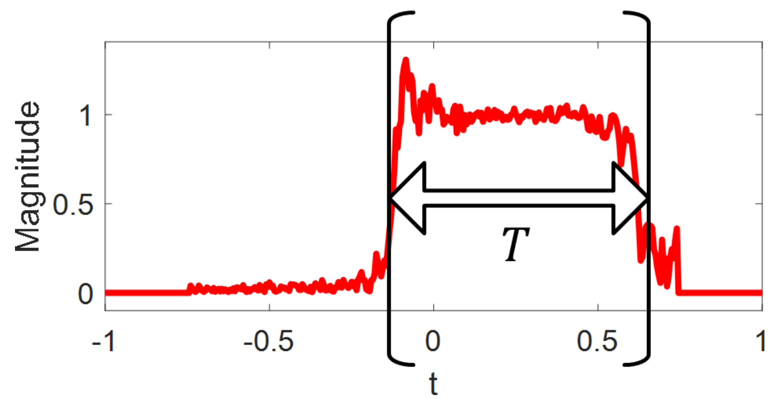


Figure 5.4. A magnitude plot of the reconstructed signal at an early iteration. The given AF is that of a rectangle pulse. The location of the recovered signal shifts with each iteration of the inversion.

Simplifying the Inversion

In lieu of (5.8) we have

$$r(t) = \frac{e^{j(kt-\phi)} \int_{-\infty}^{\infty} \chi^*(\xi - t, u) e^{-j2\pi u\xi} du}{\left(\int_{-\infty}^{\infty} \chi(0, u) e^{j2\pi u\xi} du\right)^{\frac{1}{2}}}. \tag{5.14}$$

where $r(\xi)$ is chosen to be nonzero. When $r(\xi)$ is close to zero, the inversion will become ill-conditioned. By setting $\xi = 0$ and letting the algorithm invert the signal around this point, we have the simplified inversion

$$r(t) = \frac{e^{j(kt-\phi)} \int_{-\infty}^{\infty} \chi^*(-t,u)du}{\left(\int_{-\infty}^{\infty} \chi(0,u)du\right)^{\frac{1}{2}}}. \tag{5.15}$$

As shown in Figure 5.3, fixing $\xi = 0$ limits the effect of translation on the reconstruction of a T sample source signal to $[-T, T]$.

Regularized Window

The reconstruction can be further refined by determining the length of the original signal T from the given AF magnitude and incorporating it into each of its iterations. This window does not have to be exact, and should allow some leeway in the recovery process. The estimation from Figure 5.2 should include an additional buffer for the AF that tapers in magnitude at the edges. The chief advantage of this is in addressing discontinuities in the spawning signal, such as ramp or rectangle functions.

The algorithm is presented in Figure 5.5. This regularized, adaptive window truncates $r(t)$ to be of length T by setting values outside the window to 0. Aligning this window is the main issue. Let

$$w_\delta\big(r(t)\big) = \begin{cases} r(t) & , \quad \delta \le t \le \delta + T \\ 0 & , \quad \text{else.} \end{cases}$$

then we evaluate

$$\underset{\delta}{\operatorname{argmin}} \ \left\| \chi(\tau, 0) - w_\delta\big(r(t)\big) \star w_\delta\big(r(t)\big) \right\|$$

where $\star$ is the cross-correlation.

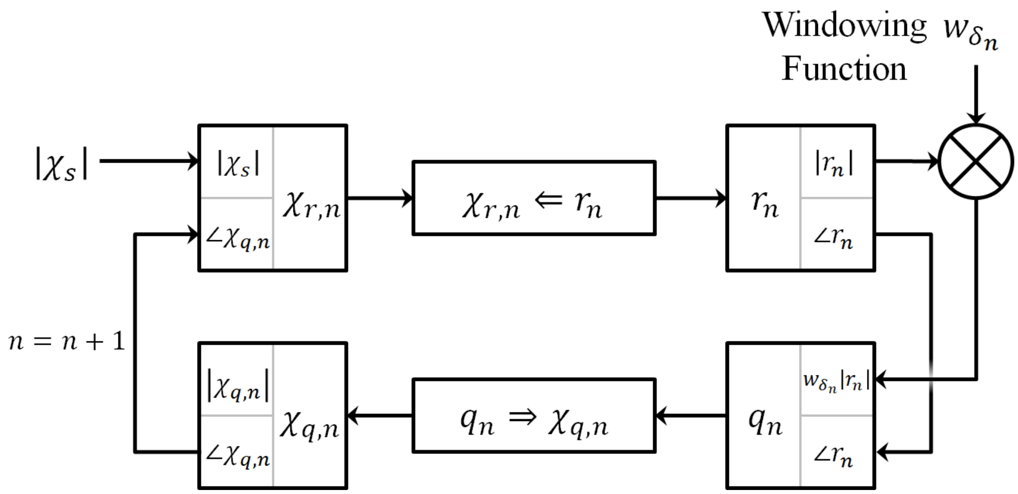


Figure 5.5. An illustration of the modified phase-recovery algorithm for AF magnitude inversion. Here, $\chi_{r,0}$ is initialized with random phase. Unlike the standard Gerchberg-Saxton algorithm, only one intensity $|\chi_s|$ is given.

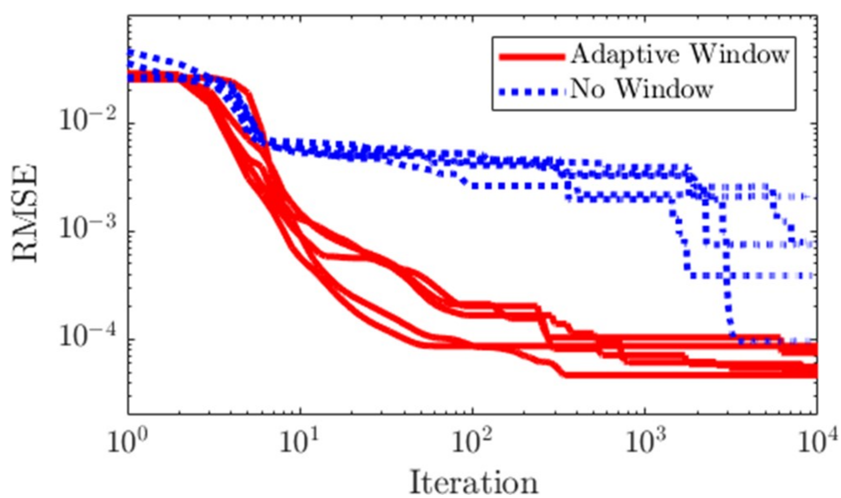


Figure 5.6. A log-log plot of the running RMS errors of multiple random initializations for both windowed (red solid line) and non-windowed (blue dotted lines) attempts at inverting the AF of a linear chirp pulse. Convergence is slow for discontinuous functions, but windowing can drastically improve performance.

Addressing Linear Phase

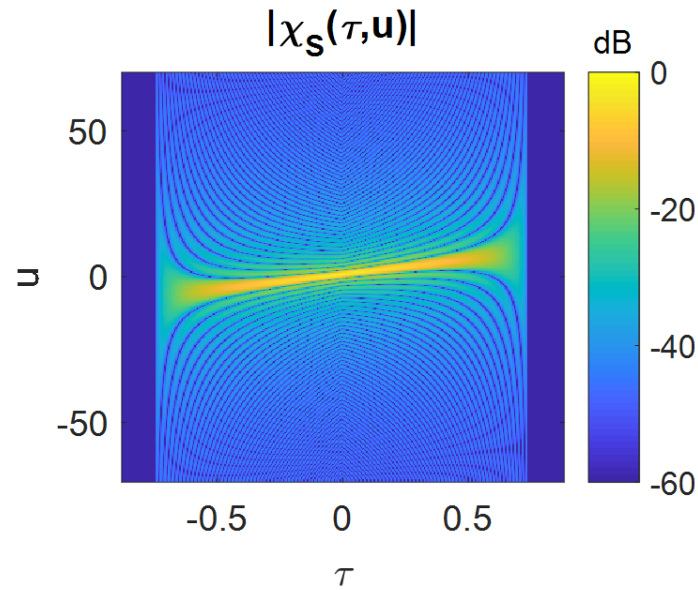
The linear phase component $e^{j(kt+\phi)}$ of this inversion will have considerable influence on the recovered signal, as any constant linear phase shift will yield the same

AF magnitude. An inversion from magnitude only using a randomly initialized phase will have some phase component associated with this linear-time rotation. Left unchecked, this value can grow dramatically with each iteration, leading to the appearance of highly oscillatory components in the resulting signal. While this effect is inconsequential to the resulting AF magnitude, the high oscillation of the restoration still needs to be avoided for clarity. If $r(t) \Rightarrow \chi_r(\tau, u)$ then $r(t)e^{j(kt+\phi)}$ will have the same AF magnitude. To prevent this from affecting the recovery with mounting iterations, this influence can be assessed and removed. A simple solution is to set the phase difference of two sequential points in $r(t)$ to 0. The inversion will unravel the phase of the rest of the signal about these components.

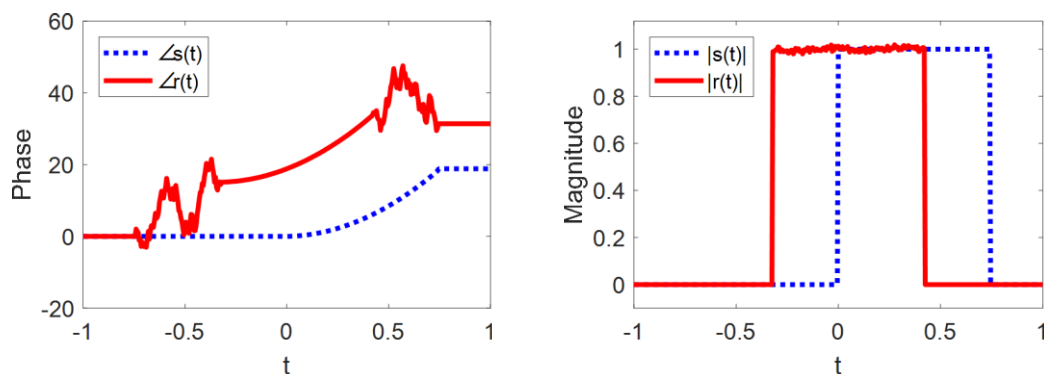
Signal Synthesis Examples

Recovering Magnitude with Regularized Windowing

The performance of the adaptive window is shown in Figure 5.6. For certain noncontinuous functions such as a rectangle or ramp function, the discontinuity of the signal's magnitude hinders convergence. By adaptively windowing the recovered function, there is a notable increase in the rate of convergence of the reconstruction. This is demonstrated in Figure 5.7. The floating window drastically improves convergence in these situations by enforcing strict limits inherent in the given AF magnitude on the length of the signal under inversion. Not all signals benefit from this windowing. Continuous functions such as the Gaussian pulse shown in Figure 5.8 experience no benefit from this procedure.



(a) Ambiguity Function of a Linear Chirp



(b) Magnitude Plot (c) Phase Plot

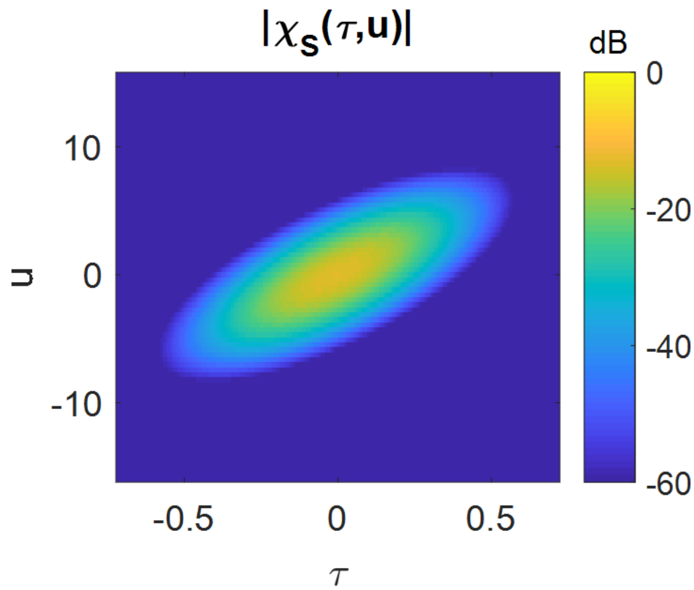
Figure 5.7. The AF magnitude (a) of a chirp function. As this signal is discontinuous, the inversion struggles in replicating the magnitude in (b) and convergence is slowed. The phase is accurately recovered in (c).

Recovering Phase

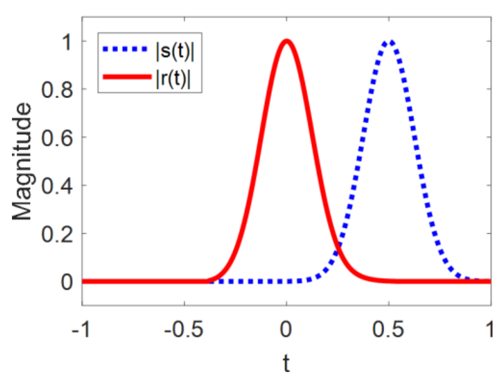
Figures 5.7, 5.8, and 5.9 demonstrate the algorithm's ability to recover nonlinear phase information in the spawning signal, such as for chirps. The random linear phase component obfuscates the true phase behavior of the signal but can be subsequently addressed. Figure 5.9 demonstrates this process with a spawning signal that has a rapidly deviating and irregular phase component. The uncorrected phase (c) appears as a high-frequency component until it is unwrapped in (d). From here, the first difference in the signal's unwrapped phase can be used to estimate the coefficients for the linear component $e^{j(kt+\phi)}$. This can then be subtracted from the result, yielding a much clearer phase plot (e). Since the linear component changes each iteration, the final removal of the linear phase component can be performed just once, post convergence.

Conclusion

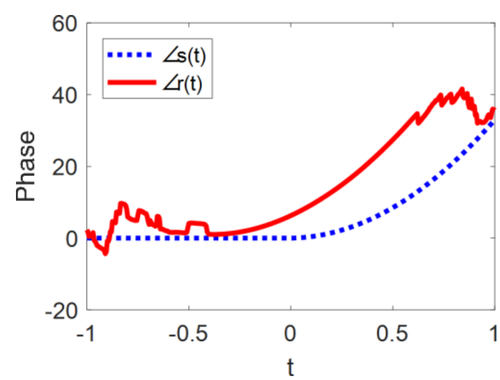
The lack of a unique inverse to the magnitude-only AF does not preclude inversion. Partial inversions can be made to discover valuable insight into the nature of a source signal for a given AF magnitude. Indeed, the modified Gerchberg-Saxton algorithm presented can successfully recover a valid spawning signal. Translations, linear phase rotations, and potentially even reflection are all transformations of the result to a family of congruent spawning signals. If further time resolution is required, the AF magnitude in this regard can be reshaped, at the expense of Doppler resolution, and vice versa. The inversion does not require a valid AF to be given, as the random initialization itself demonstrates. In this case, the iterated result will be a valid AF with many aspects of the desired AF incorporated into the signal. Overall, this inversion process may aid radar designers in molding their desired AF response by giving them a direct link to the family of source signal solutions.



(a) Ambiguity Function of a Gaussian Chirp

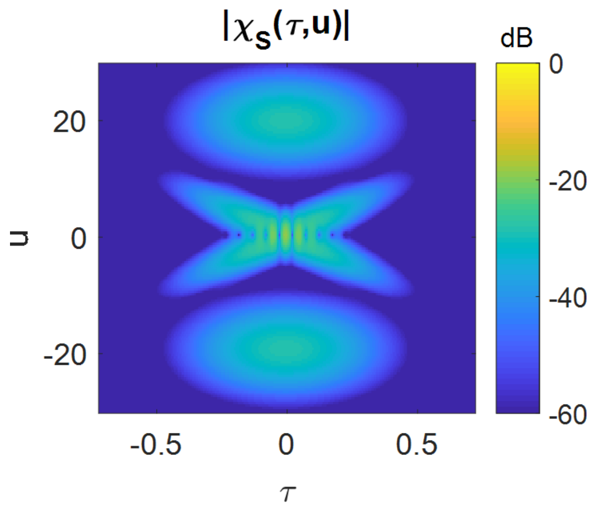


(b) Magnitude Plot

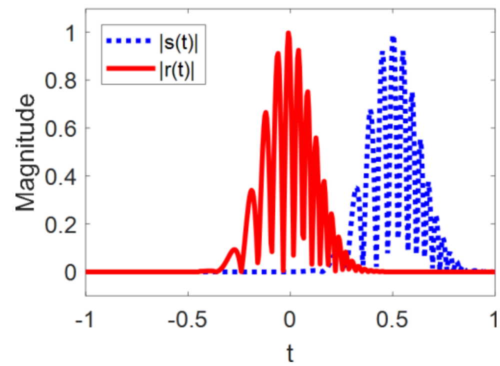


(c) Phase Plot

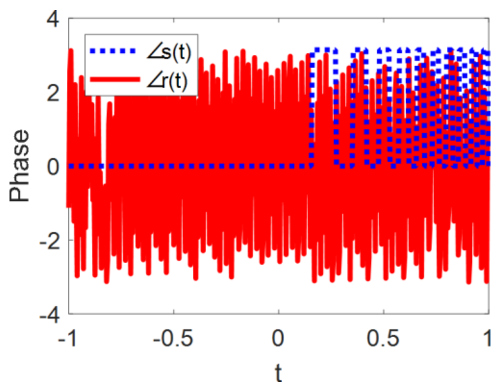
Figure 5.8. The given AF magnitude in (a) is inverted and compared to the original spawning signal $s(t)$. The magnitude plot (b) shows the reconstruction $r(t)$ centered about the fixed $\xi = 0$. In (c), the algorithm is able to recover the phase of the chirp where the signal has nonzero magnitude.



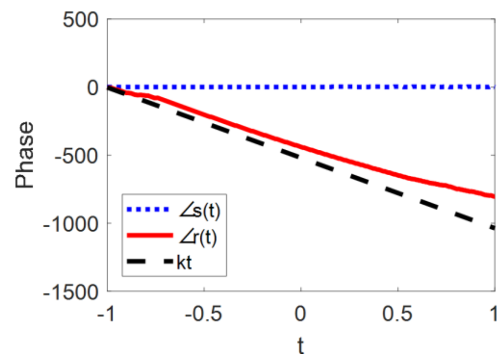
(a) AF of Real Part of a Gaussian Chirp



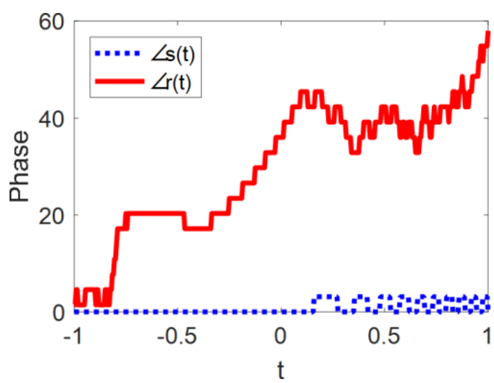
(b) Magnitude Plot



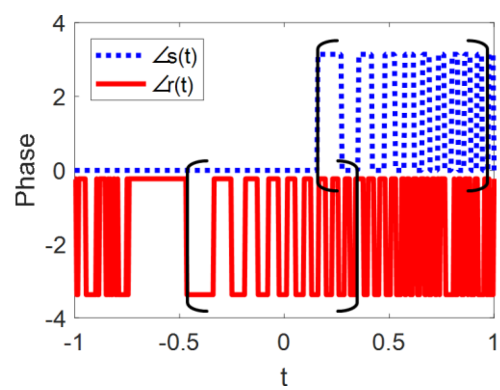
(c) Uncorrected Phase Plot



(d) Unwrapped Phase with Linear Phase Estimation



(e) Removed Linear Phase



(f) Corrected Phase Component

Figure 5.9. The given AF magnitude (a) is the real part of a Gaussian chirp $s(t)$. The inversion $r(t)$ shows the correct magnitude plot (b) but an obscure phase component (c). Unwrapping the phase reveals a large linear component (d), which when corrected (e) yields a matching final phase (f).

CHAPTER SIX

Conclusion

This dissertation explored a variety of inversion applications and modifications to standard POCS methods for computing the minimax. Morphological dilation is revealed to be a robust methodology for computing the minimax. The minimax results were contrasted with established Least-Squares algorithms to demonstrate the situational improvements possible from this methodology. A chief beneficial property of the minimax is its ability to avoid unintended preference for centroid solutions. Applications where multiple convex constraints are presented without regard for similarity between objectives can be surmounted by the minimax. By determining the algorithm though a modified POCS, the solver is generalized to include a wider variety of error metrics and convex constraints, expanding the scope and capability of minimax. This is shown to be useful in a variety of traditionally POCS applications, including Computed Tomographic image reconstruction, Fresnel signal synthesis and associative memory template matching. In addition, a regularized method for inverting an ambiguity function magnitude was developed and its ability to recover and process phase information is demonstrated. The minimax POCS may still be enhanced through improved methods beyond binary search for determining the correct dilation value. Higher dimensions still pose a significant roadblock for computing the minimax. These advancements need to be achieved to establish dilated POCS as a valid methodology for minimax.

BIBLIOGRAPHY

[1] L. A. Zadeh, "Fuzzy sets," *Information and Control,* vol. 8, no. 3, pp. 338-353, 1965.

[2] R. J. Marks II, Handbook of Fourier Analysis and Its Applications, New York, NY: Oxford University Press, 2009.

[3] A. H. Andersen and A. C. Kak, "Simultaneous algebraic reconstruction technique (SART): a superior implementation of the ART algorithm," *Ultrasonic Imaging,* vol. 6, no. 1, pp. 81-94, 1984.

[4] R. W. Gerchberg and W. O. Saxton, "A practical algorithm for the determination of the phase from image and diffraction plane pictures," *Optik,* vol. 35, no. 2, pp. 237-246, 1972.

[5] J. von Neumann, "On rings of operators. Reduction theory," *Ann. of Math,* vol. 50, no. 2, pp. 401-485, 1949.

[6] L. M. Bregman, "Finding the common point of convex sets by the method of successive projections," *Dokl. Akud. Nauk. SSSR,* vol. 162, no. 31, pp. 487-490, 1965.

[7] L. G. Gubin, B. T. Polyak and E. V. Raik, "The method of projections for finding the common point of convex sets," *USSR Computational Mathematics and Mathematical Physics,* vol. 7, no. 6, pp. 1-24, 1967.

[8] H. Peng and H. Stark, "Signal recovery with similarity constraints," *Journal of the Optical Society of America A,* vol. 6, no. 6, pp. 844-851, 1989.

[9] R. J. Marks II, "A class of continuous level associative memory neural nets," *Applied Optics,* vol. 26, no. 10, pp. 2005-2009, 1987.

[10] M. I. Sezan, H. Stark and S.-J. Yeh, "Projection method formulations of Hopfield-type associative memory neural networks," *Applied Optics,* vol. 29, no. 17, pp. 2616-2622, 1990.

[11] H. Stark and Y. Yang, Vector space projections: a numerical approach to signal and image processing, neural nets and optics, New York, NY: John Wiley & Sons, Inc., 1998.

[12] D. C. Youla and H. Webb, "Image restoration by the method of convex projections: part 1-theory," *IEEE Transactions on Medical Imaging,* vol. 1, no. 2, pp. 81-94, 1982.

[13] S. Oh, R. J. Marks II and L. E. Atlas, "Kernel synthesis for generalized time-frequency distributions using the method of alternating projections onto convex sets," *IEEE Transactions on Signal Processing,* vol. 42, no. 7, pp. 1653-1661, 1994.

[14] S. Oh, R. J. Marks II, L. E. Atlas and J. W. Pitton, "Kernel synthesis for generalized time-frequency distributions using the method of projections onto convex sets," in *Proc. SPIE 1348, Advanced Signal Processing Algorithms, Architectures and Implementations*, San Diego, CA, 1990.

[15] S. Kaczmarz, "Angenaherte Auflosung von Systemen linearer Gleichungen," *Bulletin International de l'Academie Polonaise des Sciences et des Lettres A,* vol. 35, pp. 335-357, 1937.

[16] R. Gordon, R. Bender and G. T. Herman, "Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography," *Journal of Theoretial Biology,* vol. 29, no. 3, pp. 471-481, 1970.

[17] J. Ritter, "An efficient bounding sphere," *Graphics Gems,* pp. 301-303, 1990.

[18] E. Welzl, "Smallest enclosing disks (balls and ellipsoids)," *New Results and New Trends in Computer Science, LNCS,* vol. 555, pp. 359-370, 1991.

[19] R. Seidel, "Linear programming and convex hulls made easy," in *Proc. 6th Annual ACM Symp. on Computational Geometry*, Berkeley, CA, 1990.

[20] M. Badoiu and K. L. Clarkson, "Smaller core-set for balls," in *Proc. 14th ACM-SIAM Symp. on Discrete Algorithms (SoDA)*, Baltimore, MA, 2003.

[21] T. Martinetz, A. M. Mamlouk and C. Mota, "Fast and easy computation of approximate smallest enclosing balls," in *2006 19th Brazilian Symp. on Computer Graphics and Image Processing*, Manaus, Brazil, 2006.

[22] K. Fischer, B. Gartner and M. Kutz, "Fast smallest-enclosing ball computation in high dimensions," in *Proc. 11th European Symp. on Algorithms (ESA)*, Budapest, Hungary, 2003.

[23] S. Boyd and L. Vandenberghe, Convex Optimization, Cambridge: Cambridge University Press, 2004.

[24] W. Karush, "Minima of Functions of Several Variables with Inequalities as Side Constraints," M.Sc. Dissertation, Dept. of Mathematics, University of Chicago, Chicago, Illinois, 1939.

[25] H. W. Kuhn and A. W. Tucker, "Nonlinear programming," in *Proc. of 2nd Berkeley Symp.*, Berkeley, CA, 1951.

[26] G. B. Dantzig, Linear Programming and Extensions, Princeton, NJ: Princeton University Press, 1963.

[27] R. J. Vanderbei and J. C. Lagarias, "I. I. Dikin's convergence result for the affine-scaling algorithm," *Mathematical developments arising from linear programming, Contemporary Mathematics,* vol. 114, pp. 109-119, 1990.

[28] N. Karmarkar, "A New Polynomial-time Algorithm for Linear Programming," *Combinatorica,* vol. 4, no. 4, pp. 373-395, 1984.

[29] S. Mehrotra, "On the implementation of a primal-dual interior point method," *SIAM J. Optimization,* vol. 2, no. 4, pp. 575-601, 1992.

[30] S.-J. Yeh and H. Stark, "Learning in neural nets using projection methods," *Optical Computing and Processing,* vol. 1, no. 1, pp. 47-59, 1991.

[31] S.-J. Yeh and H. Stark, "Iterative and one-step reconstruction from nonuniform samples by convex projections," *Optical Society of America A,* vol. 7, no. 3, pp. 491-499, 1990.

[32] P. Bloomfield and W. Steiger, Least Absolute Deviations: Theory, Applications and Algorithms, Birkhauser, 1983.

[33] R. L. Francis, "Letters to the editor - some aspects of a minimax location problem," *Operations Research,* vol. 15, no. 6, pp. 1163-1169, 1967.

[34] T. Larsson and L. Kallberg, "Fast and robust approximation of smallest enclosing balls in arbitrary dimensions," in *SGP'13 Proceedings of the Eleventh Eurographics Symp. on Geometry Processing*, Genova, Italy, 2013.

[35] S. Zurcher and B. Gartner, "Smallest Enclosing Ball for a Point Set with Strictly Convex Level Sets," Masters thesis, ETH Zurich, 2007.

[36] S. Mak and J. V. Roshan, "Minimax and minimax projection designs," *Journal of Computational and Graphical Statistics,* vol. 27, no. 1, pp. 166-178, 2018.

[37] D. Eustice, C. Baylis and R. J. Marks II, "Woodward's ambiguity function: from foundations to applications," in *2015 Texas Symp. on Wireless and Microwave Circuits and Systems (WMCS)*, Waco, TX, 2015.

[38] D. Eustice, C. Baylis, L. Cohen and R. J. Marks II, "Effects of power amplifier nonlinearities on the radar ambiguity function," in *2015 IEEE Radar Conf. (RadarCon)*, Arlington, VA, 2015.

[39] C. Baylis, L. Cohen, D. Eustice and R. J. Marks II, "Myths concerning Woodward's ambiguity function: analysis and resolution," *IEEE Transactions on Aerospace and Electronic Systems,* vol. 52, no. 6, pp. 2886-2895, 2016.

[40] M. Fellows, C. Baylis, L. Cohen and R. J. Marks II, "Calculation of the radar ambiguity function from time domain measurement data for real-time, amplifier-in-the-Loop waveform optimization," in *82nd ARFTG Microwave Measurement Conf.*, Columbus, Ohio, 2013.

[41] M. Fellows, J. Barlow, J. Barkate, M. Mosley, M. Flachsbart, C. Baylis, L. Cohen and R. J. Marks II, "Measurement-based radar waveform optimization using the ambiguity function and spectral mask," in *Texas Symp. on Wireless and Microwave Circuits and Systems (WMCS)*, Waco, TX, 2014.

[42] R. J. Marks II, J. F. Walkup and T. F. Krile, "Ambiguity function display: an improved coherent processor," *Applied Optics,* vol. 16, no. 3, pp. 746-750, 1977.

[43] R. J. Marks II and M. W. Hall, "Ambiguity function display using a single 1-D input," *Applied Optics,* vol. 18, no. 15, pp. 2539-2540, 1979.

[44] P. M. Woodward, Probability and Information Theory, with Applications to Radar, Oxford, U.K.: Pergamon, 1953.

[45] N. Levanon and E. Mozeson, Radar Signals, Hoboken, NJ: Wiley-IEEE Press, 2004.

[46] D. Eustice, C. Baylis, L. Cohen and R. J. Marks II, "Waveform synthesis via alternating projections with ambiguity function, peak-to-average power ratio, and spectrum requirements," in *2016 IEEE Radar and Wireless Symp. (RWS)*, Austin, TX, 2016.

[47] S. Sussman, "Least-square synthesis of radar ambiguity functions," *IRE Transactions on Information Theory,* vol. 8, no. 3, pp. 246-254, 1962.

[48] A. R. Yu, C. Baylis and R. J. Marks II, "Ambiguity function magnitude inversion via a modified Gerchberg-Saxton algorithm," in *IEEE WMCS Symp.*, Waco, TX, 2017.

[49] K. F. Cheung and R. J. Marks II, "Ill-posed sampling theorems," *IEEE Transactions on Circuits and Systems,* vol. 32, no. 5, pp. 481-484, 1985.

[50] D. R. Fuhrmann, P. J. Browning and M. Rangaswamy, "Signaling strategies for the hybrid MIMO phased-array radar," *IEEE Journal of Selected Topics in Signal Processing,* vol. 4, no. 1, pp. 66-78, 2010.

[51] R. J. Marks II and D. K. Smith, "Gerchberg-type linear deconvolution and extrapolation algorithms," in *Proc. SPIE 373 Transformations in Optical Signal Processing*, Seattle, WA, 1984.

[52] R. J. Marks II, "Posedness of a band-limited image-extension problem in tomography," *Optics Letters,* vol. 7, no. 8, pp. 376-377, 1982.

[53] M. I. Sezan and H. Stark, "Image restoration by the method of convex projections: part 2-applications and numerical results," *IEEE Transactions on Medical Imaging,* vol. 1, no. 2, pp. 95-101, 1982.