UNIVERSITY OF OKLAHOMA
GRADUATE COLLEGE

REVERSE BACKPROJECTION METHOD FOR SIMULATION OF PASSIVE BISTATIC SAR DATA

A THESIS
SUBMITTED TO THE GRADUATE FACULTY
in partial fulfillment of the requirements for the
Degree of
MASTER OF SCIENCE

By

TAYLER BRUMBLE
Norman, Oklahoma
2017

REVERSE BACKPROJECTION METHOD FOR SIMULATION OF PASSIVE BISTATIC SAR DATA

A THESIS APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

Dr. Nathan A. Goodman, Chair

Dr. Tian-You Yu

Dr. Caleb J. Fulton

For Amy. Thank you for everything.

Acknowledgements

I would like to thank members of Dr. Goodman's research group for their advice and assistance throughout the completion of this work. Dr. J.R. Lievsay for teaching me about GPU programming and other MATLAB functions that increased the speed of my program. David Lucking for teaching me about how to use a UNIX environment and helping me debug errors related to OU's supercomputing server. I would also like to thank Dr. Faruk Uysal for his assistance in developing the reverse backprojection algorithm, and Amy Weigant for her tips on software organization. Finally, this work could not have been completed without the guidance of my adviser, Dr. Nathan Goodman, who I thank for his countless hours of assistance and all his patience.

Table of Contents

List of Tables

List of Figures

Abstract

Synthetic Aperture Radar (SAR) is a valuable tool for acquiring information about landscapes through forming two-dimensional images remotely. SAR has useful applications for defense, intelligence, humanitarian, and urban planning efforts, among others [1]. Because of increased competition for suitable transmission frequencies in the electromagnetic spectrum from wireless communication corporations, passive bistatic radar has been extensively studied over the last few decades as a means of circumventing that obstacle. However, it does not come without its own introduced challenges - because telecommunication waveforms are continually changing, that variability introduces a large computational burden. In order to model data generation in a simulation environment, it is necessary to develop processing methods that will perform in a timely manner. In this work we present the Reverse Backprojection Algorithm, derived from the adaptable, commonly-used, yet brute force backprojection imaging algorithm, as a novel approach to generate simulated passive bistatic SAR data. To demonstrate the algorithm's effectiveness, we conduct a variety of simulations employing 4G Long Term Evolution (LTE) waveforms. Because LTE waveforms are not designed for radar usage, self-ambiguities within the waveform autocorrelation function degrade image quality. Simulations performed on point scatterers and distributed scenes illustrate these introduced artifacts, and meaningful comparisons are provided to determine which configurations more suited for SAR operation and identify additional processing

needs.

Chapter 1

Introduction

1.1 Motivation

The motivation for this work is to develop a novel data simulation method to model synthetic aperture radar (SAR) data formed from a passive bistatic radar (PBR) employing Long Term Evolution (LTE) cellular communication waveforms as the chosen illuminators of opportunity. Because SAR data sets are large, SAR image formation is computationally complex. Therefore, algorithm efficiency and speed are of great importance to SAR algorithm engineers. In order to test and develop SAR techniques, SAR engineers rely on high fidelity, efficient simulations. In the context of passive bistatic SAR (PBSAR), where the waveform is different for each pulse collection and unknown to the algorithm designer, efficiency is an even greater priority because of higher computational demand. Because collecting experimental data is expensive and logistically complicated, developing a faster, reliable means of modeling radar data is necessary to further explore bistatic radar processing methods. For the past few decades, SAR algorithm engineers have developed and refined image formation algorithms that are fast, flexible, and efficient, to suit the needs of intensive data collection. In this work, we explore the commonly used and flexible backprojection imaging algorithm - investigating whether we can exploit its flexibility “in reverse” to generate pulse-compressed radar data from a simulated image rather than forming an image from collected raw data in order to create a simulation tool that models SAR systems. Previous work completed in [2] examined passive bistatic radar geometry and the bistatic and multistatic ambiguity functions using LTE as a illuminator of opportunity. In Chapter 1 we introduce the concepts of PBR as well as examine different options for illuminators of opportunity as reviewed in literature. In Chapter 2 we describe fundamentals of SAR and the geometry of PB-SAR systems. In Chapter 3 we discuss the need for novel data simulation methods as well; we follow that discussion with a description of the conventional backprojection algorithm and how we can exploit its features to simulate pulse-compressed data from an image scene. We present results and analyses from our simulations in Chapter 4, and give concluding remarks in Chapter 5.

1.2 Introduction to Passive Bistatic Radar

In this section, we will introduce fundamental concepts of passive bistatic radar, as well as discuss options for illuminators of opportunity. We summarize previous work exploring these options in passive bistatic radar literature, and discuss advantages and disadvantages of each modulation scheme. Because we ultimately decided to use LTE waveforms in our simulations, this section also includes a description of the properties of LTE waveforms, their composition, and their components.

A bistatic radar is a system with separated transmitting and receiving hardware. The transmitter and receiver can be mounted on the same platform, or be located kilometers away. Passive bistatic radar systems exploit transmitted waveforms from external transmitting sources such as communication towers instead of producing

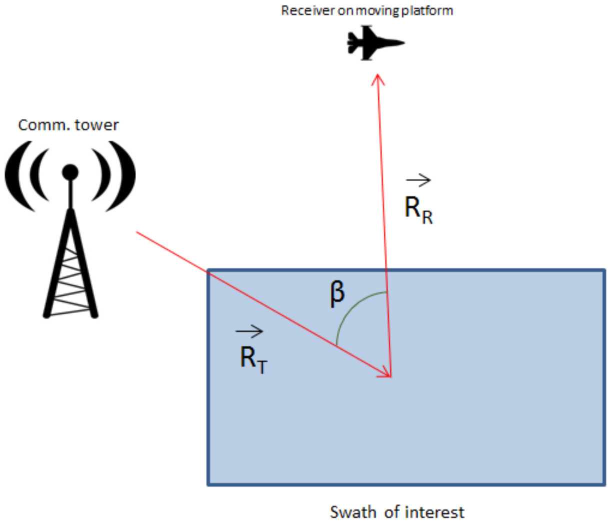


Figure 1.1: Basic bistatic radar setup

their own. Using a communication waveform in place of a specially-designed radar waveform as a transmitted signal introduces many challenges, while the system as a whole can have key advantages. Passive bistatic radars do not require frequency allocation in an already cluttered electromagnetic spectrum, weigh less, are more mobile and agile, and are cheaper to construct due to the lack of transmitting hardware. PBRs may offer a significantly larger radar cross section (RCS) than monostatic radars, may offer greater information content in SAR data regarding feature extraction and classification, and may operate more covertly because they do not transmit their own waveforms [3], [4], [5]. Figure 1.1 illustrates a basic PBR setup with an airborne receiver and a stationary telecommunications tower.

The added complexity from using a communication waveform requires compensation on the back end. The radar engineer must consider properties such as effective radiated power, system geometry, operating frequency, antenna patterns, and bandwidth to optimize system performance [3], [6]. Because radar systems require low interference in order to operate in transmit and receive modes, an increasingly congested electromagnetic spectrum has spurred competition between radar engineers and telecommunication companies [6]. Both radar and telecommunication systems operate favorably at frequencies less than 5 GHz; because corporations are willing and able to pay upwards of £12 billion GBP (about 16 billion USD) for desirable spectrum bandwidth, radar engineers have compensated by adopting spectrum sharing techniques, including PBR [7]. Studies have been conducted to evaluate the performance of a variety of waveforms such as FM radio, DVB-T, DAB, LTE, and WiMAX for passive bistatic radar [6], [8]–[12]. Griffiths and Baker determined in [13] that digital waveforms are much more favorable for PBR because they are much more consistent over time and do not vary greatly on signal content like analog signal sources; their spectra have noise-like qualities which are advantageous for range compression and Doppler estimation [9]. Additionally, analog sources have bandwidths on the order of kHz (in contrast to digital sources, which often have bandwidth on the order of MHz), which does not allow for sufficiently fine range resolution [14]. Because of these limitations facing analog transmissions, this work explores usage of various digital transmissions, including DVB and LTE, as illuminators of opportunity for passive bistatic radar. Additionally, research from Evers et. al suggest that deterministic features of digital waveforms can be used to acquire *a priori* knowledge of the signal and eliminate the need to collect a direct-path surveillance signal [8], [15].

Digital Video Broadcast-Terrestrial (DVB-T) is a communications standard for television data used in many regions of the world that employs orthogonal division multiplexing [9]. DVB-T modulation follows a standard outlined in [16] for a signal structure that includes data frames, a pilot signal, and guard intervals. DVB-T can operate in one of two modes, 2k or 8k-mode, with 2048 or 8192 subcarriers, respec-

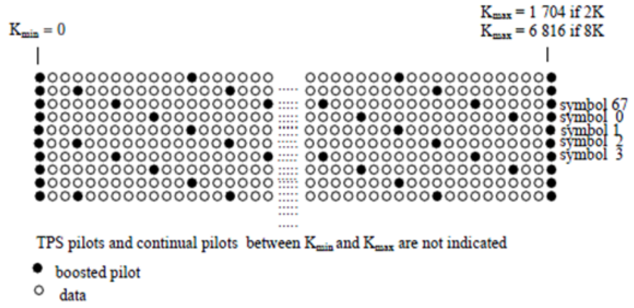


Figure 1.2: DVB-T Frame Structure [16]

tively. Figure 1.2 shows a DVB-T signal's frame structure and distribution of pilot signals. 2k-mode offers a bandwidth of 6 MHz and 8k-mode offers a bandwidth of 8 MHz within the ultra high frequency band (UHF) [8] [16]. DVB-T symbols are grouped into frames consisting of 68 OFDM symbols [9]. Because the contained MPEG-2 video data passes through bit-randomization and inner and outer-coding stages before being projected onto the signal constellation, the data on these carriers appears random and noise-like [9], [17]. However, signal characteristics of DVB-T cause peaks to appear in its cross-correlation, resulting in ambiguities in the range-Doppler domain [9]. Because these ambiguities could negatively impact target detection and classification when DVB-T is used as an illuminator of opportunity, Palmer et al. in [9] propose to use mismatched matched filtering techniques to eliminate the ambiguity in the DVB-T signal's ambiguity function.

DTV is the standard for digital broadcast television used in North America, as defined by the Advanced Television Standard Committee (ATSC) [18]. DTV is modulated using 8-vestigial sideband (VSB), which is not OFDM-based like digital television standards elsewhere [9]. Perhaps the most obvious advantage of choosing DTV as a passive coherent location waveform is its wide coverage throughout the United States of America [19], [20]. 8-VSB is a modulation scheme resulting from an 8-level amplitude-shift-keyed (ASK) signal that has been filtered to a single sideband [21]. The signal bandwidth is limited to 6 MHz by passing the upper sideband through a root-raised-cosine filter [19]. Like DVB-T, the signal generation process produces data that resemble a pseudo-random sequence; a constant value of 1.25 is added to the odd-valued symbols to generate the carrier, but that characteristic can be removed by using mismatched filtering [19]. One disadvantage of employing DTV is that the received signal must both be decoded and remodulated to regenerate the direct-path signal and suppress direct path interference [19]. A radar decoder must be comparable to a DTV television receiver for DTV to be a feasible waveform selection; because DTV signals are also prone to multipath distortion, it is challenging to create a filter based on just a training sequence and the decoder must implement an adaptive equalization filter [19], [22]. Although DVB-T and DTV are feasible waveform options for passive bistatic radar, we ultimately decided to extensively study LTE waveforms, which are described in the following section.

1.3 LTE Waveforms

LTE cellular communication networks employ Orthogonal Frequency Division Multiple Access (OFDMA), and are structured in a standard defined in [23]. LTE cellular communication towers employ frequency-division duplexing (FDD), which allocates smaller bands of the the tower's available frequency spectrum to each user in proportion to their needs. Figure 1.3 presents an example of OFDMA with FDD; six color-coded users communicate with the tower simultaneously and are each given a different number of subcarriers over time to meet their needs. The most basic element of an LTE transmission is a data symbol with duration T_u. The symbol is composed of a superposition of either 2048 or 4096 mutually orthogonal, quadrature phase shift keying (QPSK), 16-quadrature amplitude modulation (16-QAM), or 64 QAM modulated subcarriers [8]. Some of the subcarriers are carry data, others as part of the DC subcarrier. Each of these channels made up of subcarriers are separated by 15 kHz guard intervals [24]. Figure 1.4 gives representation of an LTE waveform's frequency domain signal structure, showing the distribution of subcarriers with respect to frequency. In order to improve communication in a cluttered environment, the beginning of a data symbol contains a copy of the end portion of a data symbol, referred to as a cyclic prefix [25]. Different fractions of the symbol length are used - an LTE symbol can have a *normal* or *extended* cyclic prefix [23], which determine the ratio of the symbols used. Each LTE data symbol is concatenated in the time domain to form 10 ms radio frames that are made up of .5 ms slots and 1 ms subframes; the time duration of each of these elements remains fixed, while the total number of symbols varies depending on the type of cyclic prefix used, as evidenced in Figure 1.5 (which is similar to a Figure given by Anritsu in [24]) [15]. While the entire radio frame could potentially be used as a radar pulse, it is in our best interest to use only a portion of the entire waveform in order to maximize bandwidth, optimize resolution and effects of second-trip echoes, and decrease the computational expense of data simulation and imaging.

We can simulate LTE waveforms by utilizing built-in programs within the MATLAB ®LTE System Toolbox [26]; for this work, we will focus primarily on LTE downlink (DL) transmissions. LTE DL transmissions can occupy bandwidths of 1.4, 3, 5, 10, 15, and 20 MHz, corresponding to 6, 15, 25, 50, 75, and 100 resource blocks per channel [27]. Figure 1.7, taken from [2], show two sample generated LTE waveforms using the MATLAB toolbox; Figure 1.7(a) shows the time and frequency domain structure of a single LTE channel with 50 resource blocks, and Figure 1.7(b) shows a waveform with 6 concatenated channels containing 50 resource

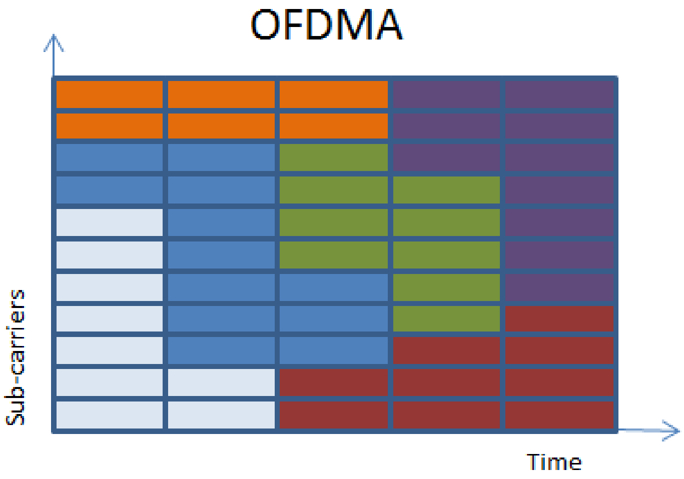


Figure 1.3: Each color represents data allotted to a different user in OFDMA modulation

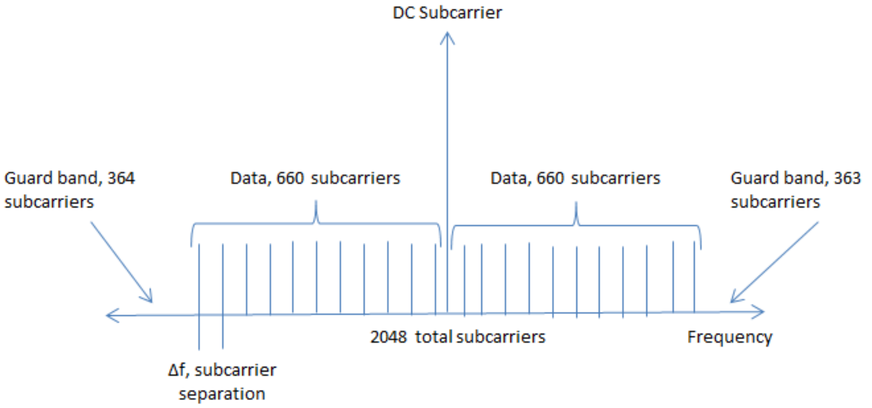


Figure 1.4: LTE waveform frequency domain structure

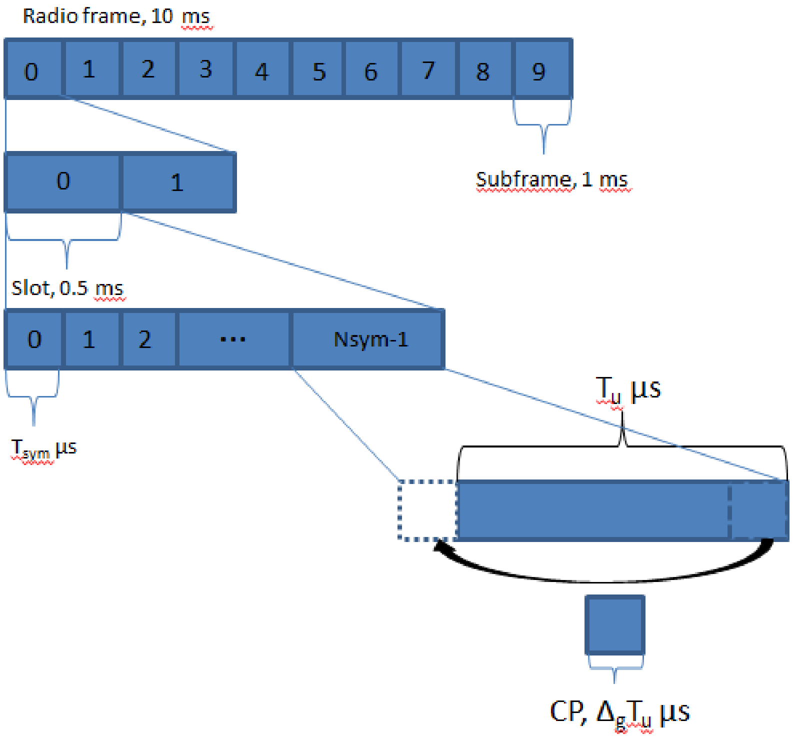


Figure 1.5: LTE frame breakdown

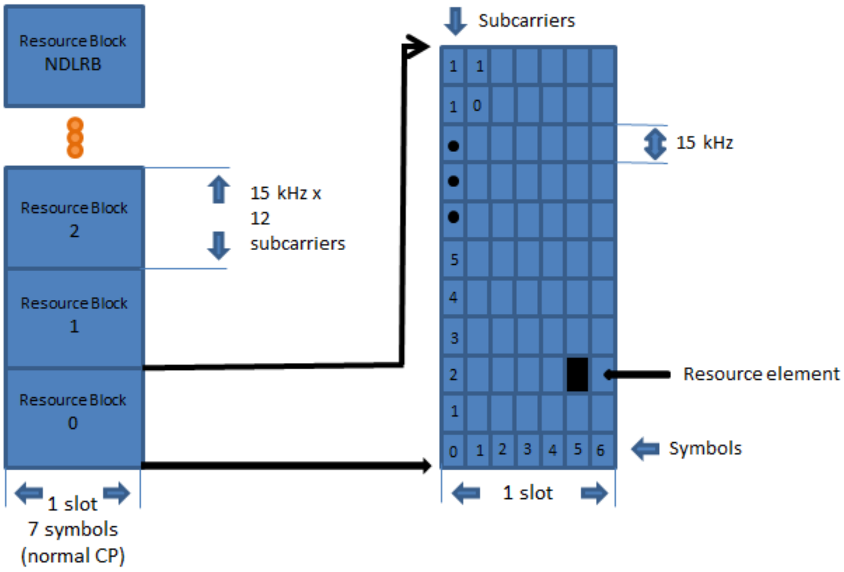
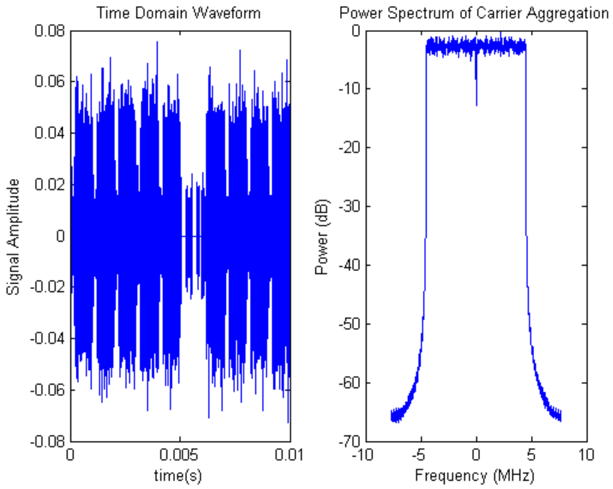
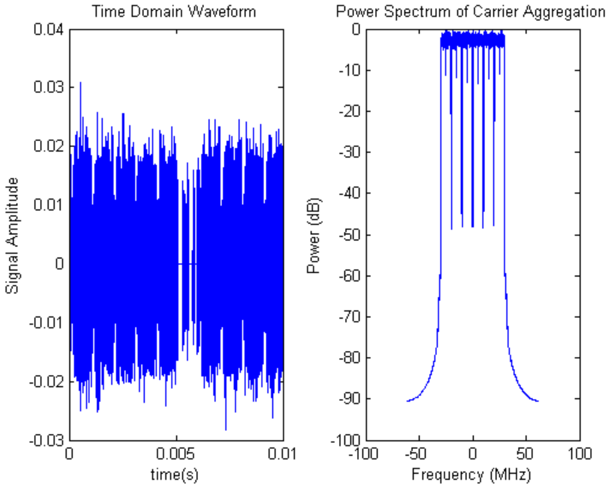


Figure 1.6: LTE resource block composition

blocks each. Figure 1.6, which is similar to a Figure found in [24], illustrates the composition of each resource block within an LTE transmission. Because of the OFDM modulation, the LTE tower can broadcast many channels at once; the resulting waveform has a bandwidth of the concatenated channels. One reason LTE waveforms are of particular interest to us is their ubiquitous presence in populated areas, which is advantageous for the potential of real-time data collection in future work. According to [8], each LTE tower covers a land area of between 5 and 100 km, though smaller coverage regions of 5 km are the most common. The full range of LTE operation carrier frequencies is 450 MHz to 2.6 GHz [28]. The FCC has licensed two bands within the evolved universal terrestrial radio access (E-UTRA) band 5, Class A and Class B to the Oklahoma City metropolitan statistical area (MSA) [24] [29]; in that band, transmissions can have bandwidths of 1.4, 3, 5, and 10 MHz [24]. We will simulate carrier frequencies consistent to these values in all simulations described in the following chapters.



(a) A single-channel LTE Waveform with 50 resource blocks



(b) A six channel LTE Waveform with 50 resource blocks each.

Figure 1.7: Complex-baseband representation of sample LTE waveforms

LTE symbols are also formed with deterministic features: cell specific reference elements (CSREs), primary synchronization signals (PSSs), and secondary synchronization symbols (SSSs), that can be used to mitigate the need for a direct path signal, as is the case with DVB-T. Unfortunately, these deterministic features can cause difficulties with cross-ambiguities [15]. The effects from these deterministic elements are evident in the results featured in later chapters. LTE waveforms are made up of 504 physical (PHY) level cell-identities (CIDs) so that cellular devices communicating with LTE transmitters can differentiate information from different transmitters; each CID is made up of a unique primary and secondary synchronization signal (PSS and SSS) pair [23]. The PSS is given as a 62-length Zadoff-Chu sequence, defined in [23], depending on the CID number, $u \in \{25, 29, 34\}$ [8], [23]. These values are assigned to 31 subcarriers centered around the DC subcarrier on the last symbol in the first (0) and eleventh (10) slots [23]. The remaining subcarriers in the symbol are modulated normally (using QAM) [8]. Similarly to the PSS, the SSS is created by interleaving two 31-length sequences and mapped onto 62 subcarriers around the DC subcarrier in slots 0 and 10 [8], [23]. In order for a mobile device to further discern the source of received information, cell-specific resource elements (CSREs) are applied to the transmitted signal [8]. A pseudorandom Gold sequence of length 31 is used to provide complex modulation coefficients for each CSRE according to a QPSK scheme [8], [23]. The CSREs are contained within symbols 0 and 4 within a slot for signals employing a normal CP and symbols 0 and 3 for an extended CP [8]. The position of the CSREs is determined by the CID number; they are placed on every six subcarriers. Consecutive symbols containing CSREs are offset in frequency (on different subcarriers). More information regarding CSRE positioning is beyond the scope of this work, but can be found in the LTE Standard [23]. In this work we will primarily examine LTE waveforms as an illuminator of opportunity because of the availability of LTE transmitters, the noise-like spectrum of digital waveforms, the relatively high bandwidth of LTE transmissions, and to explore how frame composition (number of resource blocks and bandwidth) affects imaging. We present results from a simulation tool that is capable of using any waveform as a radar pulse, and we will make comparisons between using bistatic waveforms and conventional linear frequency modulated (LFM) waveforms in Chapter 4. The flexibility of our simulation tool will greatly benefit SAR algorithm engineers by generating pulse-compressed data that can be used to explore SAR signal processing and image formation methods. Now that we are familiar with passive bistatic radar and LTE waveforms from Chapter 1, we will introduce concepts in SAR and go into detail regarding the geometry of passive bistatic radar systems in Chapter 2.

Chapter 2

SAR Theory and Geometry

2.1 Intro to Synthetic Aperture Radar

In this section, we introduce the concepts of synthetic aperture radar (SAR) and provide equations relevant to our applications of these concepts. The techniques in this work exploit synthetic aperture radar (SAR) imaging, which is a method of obtaining fine-resolution, two-dimensional radar images. SAR is an effective imaging technique for remote sensing because the utilized electromagnetic frequencies allow for imaging that is unaffected by time of day and ambient weather conditions [30]. In order to obtain good quality images, it is important for the image resolution to be constant throughout the entire scene [31]. The scene dimensions are defined as range (or down-range, the direction perpendicular to the platform's trajectory), and cross-range (the along-track dimension). Because of the nature of SAR geometry, the system exhibits different image resolutions along the two dimensions. The basis for SAR is the "synthetic aperture viewpoint", which is employed because sufficient cross-range resolution could not be attained otherwise in "real beam" imaging operation. In real-beam imaging, the cross-range resolution of any given pixel in the imaged scene can be approximated by the width of the beam at that point, $R_0\theta_{az}$, where R_0 is the down-range distance to the given pixel and θ_{az} is the 3-dB azimuthal beamwidth of the radar antenna. By that metric, two scatterers would need to be the width of the beam apart in order to be resolved. Because airborne SAR typically operates at ranges on the order of kilometers, real beam operation yields resolution much too coarse to produce meaningful SAR images. In order to alleviate this problem, the "synthetic aperture viewpoint" utilizes a single antenna array and the platform motion to construct a "synthetic aperture" as it transmits waveforms and collects data. Each position where transmissions occur act as an element of the array, and each transmission's returns are combined so that the results mimic those of a very large array, allowing for sufficiently fine resolution for imaging. In order to introduce fundamental principles of SAR, we will first discuss these principles for monostatic radar, a configuration with a co-located transmitter and receiver, and then expand our analysis to passive bistatic SAR. For monostatic SAR, the resolution in down-range is

$$\Delta R = \frac{c}{2BW}, \tag{2.1}$$

and the resolution in the cross-range dimension is

$$\Delta CR = \frac{\lambda R_0}{2vT_a}, \tag{2.2}$$

where λ is the wavelength of the transmitted waveform, v is the velocity of the target with respect to the radar, and T_a is the time it takes for the SAR platform to travel across the entire aperture. Because of the platform motion, there is a lower bound on cross range resolution for monostatic, stripmap SAR:

$$\Delta CR = \frac{D_{az}}{2}, \tag{2.3}$$

which is significant because (2.3) is independent of range. Effects of waveform frequency and platform motion influence the simulation and will be evident in the results of the later sections.

Pulse-Doppler radars measure reflectivity by transmitting pulses and recording the complex-valued voltage of the echo over time [31]. In conventional monostatic radar, the receiver does not collect samples while the antenna is transmitting to avoid interference from from the transmitted signal. Because for our case, the transmitter and receiver are separate - we will model our system as collecting samples from both a nearby transmitter and the ground plane beneath it. The receiver samples the returned voltage at the Nyquist rate, which is the inverse of the received signal's bandwidth. The samples will be demodulated and processed as complex baseband signals. The sampled data will follow a typical pulse-Doppler radar data structure - the radar will sort the samples from each pulse as a row in a data cube, as illustrated in Figure 2.1. Each sample in each row corresponds to its own individual range gate. We will refer to the axis made up of these range gates as the "fast-time" axis, since each sample represents an echo from a different time delay; the axis has units of seconds with spacing equal to the sampling rate and represents a discrete set of observed echo delays. We will call the other axis the "slow-time" axis, and each row along that axis will be made up of the echoes from a single pulse, for a total of N rows, which represent the N transmitted pulse and their received echoes. We can calculate the time delay for each sample as

$$\tau = \frac{2R}{c}, \tag{2.4}$$

where R is the range to whatever produces the echo. Because each row corresponds to a different pulse and a different platform position, we can gather information

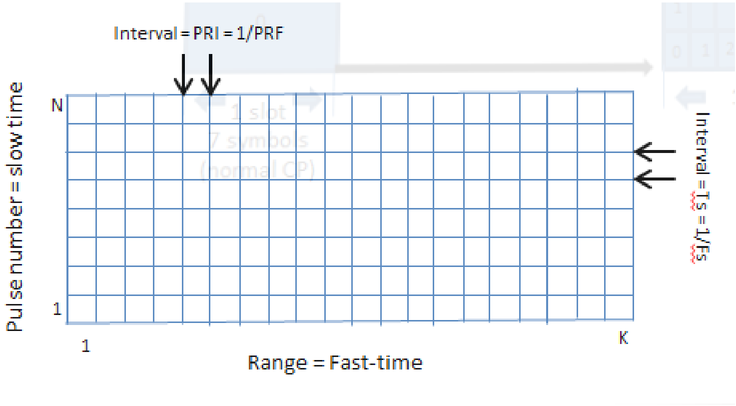


Figure 2.1: Pulse-Doppler data arrangement

about platform motion by examining pulse-to-pulse changes. We will use the phase information from the sampled data in both dimensions to form images, which will be discussed more in Chapter 3. The way the Pulse-Doppler data is arranged so that it can be used in our imaging algorithm is given in Figure 2.1; each row consists of the fast-time samples for that pulse, n, and the N pulses are stacked on top of each other.

We now expand our discussion to passive bistatic SAR. Because the range R_B in a passive bistatic radar is made up of two components - the range between transmitter and scatterer, R_T, and the range between scatterer and receiver, R_R. We express the time delay, τ, as a function of the two-way bistatic range:

$$\tau = \frac{R_B}{c}. \tag{2.5}$$

Down-range and cross-range resolution are also altered by the geometry inherent in passive bistatic SAR (PBSAR) systems. The range resolution is dependent on the bistatic angle, β, the angle between the two range components:

$$\Delta R = \frac{c}{2BW cos(\beta/2)}, \tag{2.6}$$

where the minimum possible resolution (when $cos(\beta/2) = 0$) is equal to the monostatic range resolution in (2.1), indicating that there is no angle between R_R and R_T. This section discussed the fundamentals of SAR; now that we are familiar with synthetic aperture radar, we will move on to specifics of passive bistatic SAR geometry.

2.2 Geometry

In order to develop an algorithm that models pulse-compressed data from SAR, we must first familiarize ourselves with the effects of the geometry of passive bistatic configurations on collected PBSAR data. Figure 2.2 shows a basic depiction of stripmap operation mode (though some of our processing methods in described in Chapter 3 will use conventions employed in spotlight mode). We will model the platform motion using the typical "move-stop-move" assumption; because the platform's velocity is much slower than the speed of light, each pulse is assumed to be transmitted and received at a fixed location. The surface to be imaged is referred to as the "ground plane." The ground plane, or "scattering grid," $P(x, y)$, is modeled in a two-dimensional x-and-y coordinate system, broken into discrete, square grid regions called "patches," $p(x, y)$. We will consider an airborne radar platform traveling in the positive x-direction at a constant velocity.

The platform's direction of motion will be modeled as the x-axis of the coordinate grid. Instead of considering only the platform/receiver location with respect to the ground plane as in monostatic SAR, for bistatic SAR we must examine the effects of transmitter-receiver pair locations on the resulting images. Because of the separation between the transmitter and receiver, the isorange contours are elliptical and the power contours are Cassini ovals as opposed to the circular contours

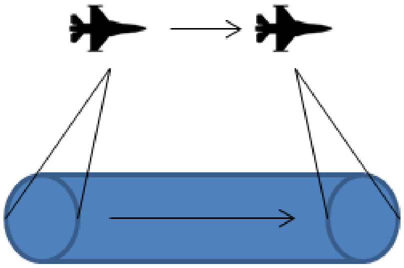


Figure 2.2: Stripmap SAR

in monostatic radar [32]. The elliptical nature of the isorange contours complicates bistatic radar performance analyses. In the monostatic case, the ambiguity function depends on the range and Doppler shift with respect to a single point, whereas the bistatic ambiguity function depends on the two-part bistatic range and a Doppler shift related to the bistatic angle [31], [33].

Figure 2.3 illustrates the bistatic SAR geometry modeled in this work. The variables $\vec{R_T}$ and $\vec{R_R}$ represent the vectors between the transmitter and each patch, and the platform (receiver) and each patch, respectively. We will refer to the scalar quantities R_T and R_R as the length of these vectors. The distance between the transmitter and receiver, is referred to as the bistatic baseline. The platform position is $RX = (R_x, R_y, 0)$, the transmitter position is $TX = (T_x, T_y, 0)$, and each patch has a position $(x_p, y_p, 0)$. Because the platform is in motion, the receiver position $(R_x, R_y, 0)$ varies along the x-axis according to the pulse number, n. The transmitter is assumed to be a stationary communications tower, so its position will not vary over time. The platform collects data from a region of interest of an arbitrarily-selected area composed of square patches, $p(x, y)$, with length and width determined by the radar's range resolution, ΔR. The receiver samples the incoming waveform at a rate of BW samples/sec, where BW is the bandwidth of the LTE waveform. Configurations can also include multiple transmitters or trans-

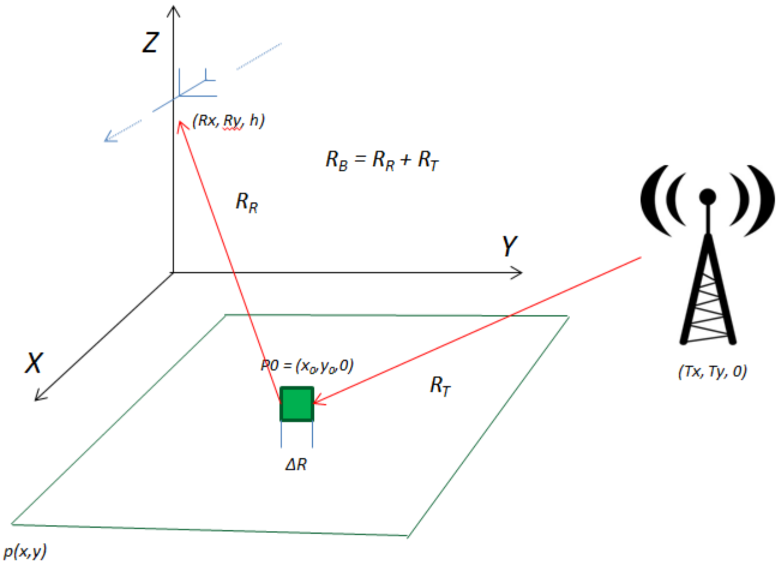


Figure 2.3: Bistatic SAR geometry

mitters; those configurations are called “multistatic.” We investigated properties of multistatic radar for target detection in [2], but further analysis is beyond the scope of this work.

In monostatic radar, an antenna orthogonal to the direction of motion transmits pulses, but in bistatic radar, the receiving antenna senses reflections off the region of interest from a nearby communications tower. Consider a scatterer, P_0, located at (x_0, y_0), where x_0 and y_0 are the center of the ground plane. The bistatic range to P_0 is the sum of the transmitter and receiver ranges, R_R and R_T, which can be found by applying the distance formula:

$$R_{B0}(n) = R_{R0}(n) + R_{T0} \tag{2.7}$$

where

$$R_{T0} = \sqrt{(T_x - x_0)^2 + (T_y - y_0)^2}, \tag{2.8}$$

and

$$R_{R0}(n) = \sqrt{(R_x(n) - x_0)^2 + (R_y - y_0)^2}. \tag{2.9}$$

We can express the quantity $(R_y - y_0)$ as a constant, R_p, such that

$$R_{R0}(n) = \sqrt{(R_x(n) - x_0)^2 + R_p^2}, \tag{2.10}$$

and

$$R_{B0}(n) = R_{T0} + R_p\sqrt{1 + \frac{(R_x - x_0)^2}{R_p^2}}. \tag{2.11}$$

The range $R_{B0}(n)$ varies hyperbolically as the radar moves along its track. The pulse length t_p is determined by the scene dimensions, since we only need however many samples it takes to map out the entire ground plane. We can extrapolate (2.7) to find the bistatic range for each point in the ground plane:

$$R_B(n) = R_R(n) + R_T. \tag{2.12}$$

From the range matrix in (2.12), we can find the time delay to every patch:

$$\tau(n) = R_B(n)/c, \tag{2.13}$$

where c is the speed of light, and find the minimum and maximum time delays, τ_{min} and τ_{max} in $\tau(n)$.

For analysis of bistatic radar, it can also be beneficial to be familiar with properties of ellipses. Contours of equidistant bistatic range lie on ellipses with the receiver and transmitter locations as their foci. For a bistatic radar, the semi-major axis of the ellipse is:

$$a = \frac{R_T + R_R}{2}. \tag{2.14}$$

If $R_R = R_T$, as is the case with a monostatic radar, the resulting isorange contour is a circle of radius a [3]. The semi-minor axis of a bistatic isorange contour is [3],

$$b = \sqrt{a^2 - \frac{L^2}{4}}. \tag{2.15}$$

From there, we can derive the generalized equation for an isorange contour [32]:

$$\frac{x^2}{a^2} + \frac{y^2}{a^2 - L^2} = 4. \tag{2.16}$$

The eccentricity, e, of an ellipse is

$$e = \frac{L}{2a}, \tag{2.17}$$

and an eccentricity $e = 0$ corresponds to a circular contour evident in a monostatic radar [3]. Further details regarding isorange contours are discussed in Chapter 4.

As (2.9) suggests, only the x-position with the respect to the radar changes during data collection. Because the platform is not transmitting pulses, it operates only in receiver mode in intervals dependent on the difference between the minimum and maximum bistatic ranges. Because only the x-coordinate is varying, we can use an approximation derived from the Taylor series for $\sqrt{1+x} = 1 + \frac{x}{2} - \frac{x^2}{8} \dots$ for (2.11) to find the range at each successive position, as long as y is much greater than 0. Therefore, R_B is given by [31]

$$R_B = R_T + \left[1 + \frac{(R_R(n) - x)^2}{2R_p^2}\right]. \tag{2.18}$$

From (2.18), we can gather that the collected slow-time data will have quadratic phase modulation because of the platform data [31]. The modulation will "spread" the bandwidth in the Doppler dimension and will thus enable fine cross-range resolution.

If the platform moves across a significant distance relative to the size of the illuminated scene, the range corresponding to the point scatterer P will change. If this change in range is greater than the range bin size, range migration will occur and could cause the focusing in the resulting image to be degraded. The difference in range to P over the synthetic aperture is :

$$\Delta R_m = \frac{D_{SAR}}{R_p} R_x(n) = \frac{v}{T_a} R_p R_x(n) \tag{2.19}$$

where D_{SAR} is the length of the synthetic aperture and T_a is the total time it takes to trek across the entire aperture [31].

So far we have discussed in this chapter the effects of bistatic and SAR geometry on the received data, but we have not yet discussed the effects of the platform velocity. In a case where both the platform and targets of interest are stationary, the radar will not observe a Doppler shift, as is the case with conventional monostatic radar. However, the effects of platform and/or target velocity must be considered in SAR processing, because the time-varying phase history is used to form images. However, for many remote sensing applications and our simulations, it is assumed that the ground-plane is a stationary surface, therefore the Doppler shift will only depend on the motion of the platform with respect to the ground plane. Because the platform is moving with respect to each patch, each patch will observe a different Doppler shift, depending on angles and the direction of motion. If the platform is moving with some velocity, v, the frequency of the received signal will be shifted due to the Doppler effect, which is dependent on the radar waveform's wavelength.

According to [33], one must consider higher-order terms when analyzing the effects of target or platform velocity if either has varying acceleration. However, because we are assuming that the platform is flying at a constant velocity, those terms are negligible in the context of our analysis. For bistatic radar, usually the observed Doppler shift of each patch depends on the bistatic angle, β, which can be found by using the law of cosines:

$$\beta = cos^{-1}\left(\frac{R_R^2 + R_T^2 - L^2}{2R_R R_T}\right). \tag{2.20}$$

Because of the bistatic geometry, the observed Doppler shift is the component in the direction of the bistatic bisector, $\frac{\beta}{2}$, the line of sight (LOS) visible to the receiver. In previous work we derived the component in the direction of $\frac{\beta}{2}$ [2]. However, because the transmitter is stationary and the platform is moving in our simulations, the observed Doppler shift, f_D, only depends on the angle ϕ, the angle between v and each patch,

$$f_D = \pm\frac{2v}{\lambda_c} cos\big(\phi\big), \tag{2.21}$$

where λ_c is the wavelength of the carrier frequency. We define patches the platform is moving *toward* as having a positive Doppler shift, and patches the platform is moving *away from* as having negative Doppler shifts. Now that we have discussed fundamentals of SAR and passive bistatic radar geometry, we will advance our discussion to existing data simulation methods and the need for new approaches given challenges in passive bistatic radar signal processing.

Chapter 3

Reverse Backprojection Algorithm

In this chapter, we present our received signal model and the matched filter algorithm for data collection. We discuss the impracticality of existing data simulation methods for usage in passive bistatic radar, and hypothesize that we can adapt existing efficient image formation algorithms, including the backprojection algorithm, in order to simulate data. In Section 3.2, we examine the backprojection algorithm, from its origins in medical imaging to more efficient contemporary adaptations. In Section 3.3, we present the reverse backprojection method and its implementation based on the principles given in Section 3.2 that simulates data by employing the backprojection method *in reverse*. Lastly, Section 3.4 gives the complexity of our novel approach and compares its computational expense to more conventional data simulation methods.

3.1 Need for More Efficient Simulation Methods

In this section, we discuss our models for the received signal and range swath and the matched filter algorithm for pulse-compressed data simulation. In order to gather the SAR phase history to reconstruct an image, we model both a direct-path signal, $s(t)$ and a signal that has been reflected off the ground plane, $s(t-\tau)$, where τ is the observed delay. After simulating an LTE waveform, we model the collected direct-path signal for each pulse, $s_n(t)$, as a short interval of the desired length in samples, l_p, given in (3.2). In conventional radar operation where the same pulse is repeatedly transmitted, it is not necessary to collect a matched filtered waveform for every pulse; however, because of the constantly-changing nature of LTE DL transmissions, this step must be repeated for each platform position, n, for all N pulses. This process is repeated in intervals specified by the system's chosen pulse repetition interval (PRI), T, which is a constant value much longer than t_p. The entire length of the data collection, the coherent processing interval (CPI), is therefore NT.

We then model the signals reflected from the ground plane that have been collected by the receiver, $s_n(t-\tau)$. The received signal, $s(n, t-\tau(x,y))$, is a copy of the direct-path signal for the n^{th} pulse that has been scaled by the target's RCS, $\sigma(x,y)$, has been delayed by the bistatic propagation delay, $\tau(x,y)$,

$$s(n, t-\tau(x,y)) = \sigma(x,y)s_n(t-\tau(x,y))exp\left(\frac{-jkR_B(x,y;n)}{c}\right), \tag{3.1}$$

where ΔR_B is the difference in range between each range bin and that of the scene center, P_0. The length of $s(n, t-\tau)$ that is recorded is determined from the number of samples needed to cover the entire range swath.

Because the platform is physically separated from the ground plane, there will be idle time between pulse transmission and the non-zero minimum propagation delay, τ_{min}. Therefore the number of samples, l_p, we need to map out the entire scene size is

$$l_p = (\tau_{max} - \tau_{min})F_s, \tag{3.2}$$

where F_s is the sampling frequency. For our purposes, the waveform is sampled at a rate of $F_s = BW$ samples per second. The length l_p is therefore the time-bandwidth product.

The received signal data are stored in the format discussed in Chapter 2 and illustrated in Figure 2.1. The received signals from each of the n received pulses are stored on subsequent rows in a pulse-Doppler data matrix. Because the receiving antenna is illuminating a swath of many patches simultaneously in SAR, the received signal is a integral of the echoes from the entire swath:

$$s(n, t - \tau(x, y)) = \int_X \int_Y \sigma(x, y) s_n(t - \tau(x, y)) e^{(\frac{-jkR_B(x,y;n)}{c})} dY dX. \tag{3.3}$$

Because calculating a continuous integral like (3.3) for the response of an entire scene is impossible in a programming environment, the area must be discretized to allow for a finite number of computations. Therefore, the simulated SAR radar scene is modeled as a grid of "patches" - square segments, $p(x, y)$. The scene should be fragmented in the down-range dimension into patches with width no wider than $w = \frac{1}{BW}$, so that the minimum image resolution for the bandwidth BW can be obtained. There are no simple equations for cross range resolution for passive bistatic SAR. Equation 2.2 is dependent on the monostatic propagation path in the down-range dimension, R_0, which of course increases further from the platform. However, in PBR, this R_0 has two components dependent on the transmitter location with respect to the range swath. Because no simple equations are available, engineers must evaluate the cross range resolution resulting from various SAR aperture lengths and transmitter-receiver pair geometries and set the scattering patch width in the cross range dimension to be smaller than the observed resolution value.

Trying to calculate the returned scattering over each individual patch is still a highly computationally expensive result, since each patch has its own unique delay, τ, that occurs on a continuous set of numbers:

$$s(n, t - \tau(x, y)) = \sum_{X} \sum_{Y} \sigma(x, y) s_n(t - \tau(x, y)) e^{\left(\frac{-jkR_B(x,y;n)}{c}\right)}. \tag{3.4}$$

Equation 3.4 would involve a series of linear interpolations of $s_n(t - \tau(x, y))$ for every patch, for every pulse. From this received signal, $s_n(t - \tau(x, y))$, perhaps the most intuitive way to form an image would be to use the matched filter algorithm, which optimizes the signal-to-noise ratio (SNR) [21]. We define m as the axis of range values corresponding to each index of the matched filter. The matched filter response of a scatterer located at range m is given by

$$\Psi(m) = \frac{1}{NK} \sum_{n=1}^{N} \sum_{l=1}^{K} s_n(t - \tau(x, y)) exp\left(\frac{-jkR_B(x, y; n)}{c}\right), \tag{3.5}$$

where $s_n(t - \tau(x, y))$ is the received signal from (3.1). Equation 3.5 would need to be computed for each pixel within the imaged scene, for every pulse. This calculation would be too computationally expensive for practical applications, since its computational complexity is $O(N^4)$, where N refers to the number of pixels in the scattering grid [30]. Additionally, mismatches due to atmospheric turbulence, flight path trajectory, and coherence of the microwave source will result in phase errors and negatively affect image resolution [34]. Therefore we must use approximations in order to reduce the number of computations in simulating the pulse-compressed data output. In Section 3.3, we present the reverse backprojection method as a solution to the challenges presented in this section; the approach given in Section 3.3 is dependent on the theory behind the backprojection imaging algorithm, which is described in detail in the following section.

3.2 Theory of Backprojection

In order to examine the methods applied in the subsequent sections, it is necessary to discuss the theory of image formation for SAR imaging. Image formation techniques for SAR can be classified in two broad groups: Fourier-based methods and convolution-backprojection [35]. FFT-based methods require a much lighter computational load, but require all pulses in the coherent processing interval to be collected before forming an image [35]. Backprojection is a commonly used image formation algorithm for SAR based on the matched filter algorithm, that is computationally expensive but flexible [30]. Backprojection is advantageous for SAR imaging because newly acquired phase information can be updated into the image during data collection - which is propitious for parallel processing on a Graphical Processing Unit (GPU). Additionally, backprojection can be applicable for all trajectories - both for spotlight-mode and stripmap SAR [35].

The basis of the backprojection algorithm for SAR is based on principles of computer-aided tomography (CAT) utilized for medical images described in [36], [37]. Computer-aided tomography is an imaging technique that produces a 2D cross-sectional image of a 3D object by processing projectional views of the object taken at many look angles [37]. Tomographic reconstruction is based on the projection-slice theorem [37]. According to the projection slice theorem, the re-

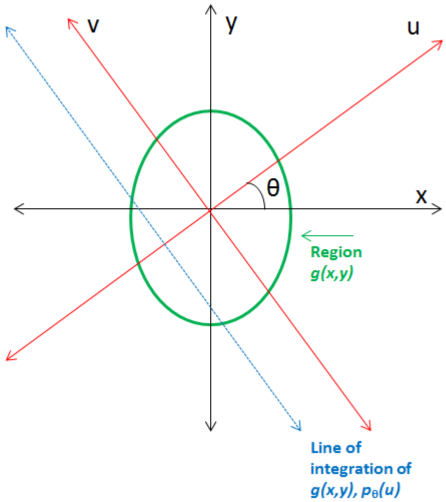


Figure 3.1: Illustration of the Projection Slice Theorem

ceived waveform obtained at each look angle approximates a slice of the 1D Fourier transform of a projection of the ground patch at that look angle [37]. If we define $g(x, y)$ as a signal to be reconstructed using tomographic methods and $p_\theta(u)$ as the projection of g at angle θ, then $p_\theta(u)$ evaluated at each u is a line integral in the direction of v:

$$p_\theta(u) = \int_{-\infty}^{\infty} g(ucos(\theta) - vsin(\theta), usin(\theta) + vcos(\theta))dv, \tag{3.6}$$

Figure 3.1, similar to a figure given by Munson et al. in [37], illustrates one of the line integrals in (3.6). In the frequency domain, the Fourier transform of $p_\theta(u)$, $P_\theta(U)$, is a slice of the 2D Fourier transform of $G(X, Y)$ at angle θ and is expressed in [37] as

$$P_\theta(U) = G(Ucos(\theta), Usin(\theta)). \tag{3.7}$$

Tomographic systems commonly reconstruct $g(x, y)$ by employing the convolution-backprojection method [36]. An important difference between SAR and CAT is that

SAR data are inherently narrowband because they lie in ground plane segments whose extents are established by the frequency content of the transmitted signal [37]. Because the transmitted signal $s(t)$ has finite bandwidth, and the received signal $g(x, y)$ is the convolution of an echo of the transmitted signal convolved with the reflectivity of the ground patches, $g(x, y)$ therefore also has finite bandwidth [38]. As the radar moves through the synthetic aperture, backprojection methods *project* each imaged scatterer onto the radar line of sight. The radar's LOS is usually defined as the displacement vector between the platform's antenna phase center (APC) and the scene center; for the purposes of this work the LOS will be defined as a vector representing the projection of each range bin, m, onto the image scene. Each scatterer will be represented in the LOS vector in range bin corresponding to its distance to the antenna phase center. Because each pulse can only resolve scatterers in range, range bins will usually contain reflections from multiple scatterers [39]. As the radar moves along its track and collects more pulses, the scatterers will be projected onto different range bins, which garners more information about the image scene. This information is evident in the Doppler shift observed by the received signal. As the radar collects more pulses, the phase history of the received pulses changes; changes in phase history over the course of the CPI allows the algorithm to project the scatterers onto the correct pixels within each range bin. Tomographic-based reconstruction *projects* the Fourier transform of each radar LOS slice *back* onto a user-specified 2D grid of image pixels, $I(x, y)$, hence the origin of the word *backprojection* [39]. Because the location of each range bin does not align with to an exact location on $I(x, y)$, interpolation methods are necessary for successful imaging; interpolation is performed to the nearest neighbor, which requires upsampling but does not introduce a substantial computational burden [35]. This operation is different for bistatic SAR because the range to each pixel depends on the two-part

R_B and the isorange contours are elliptical; however, we did not experience substantial difficulty in imaging since the transmitter is stationary. In order to perform the intermediate steps of upsampling and interpolating, the carrier phase rotation must be removed from the data [35]. We accomplish this step by applying a phase correction as described in [30].

Over the last few decades, algorithm engineers have modified the backprojection algorithm to be faster and less computationally complex, as traditional backprojection has $O(N^3)$ complexity [40]. Yegulalp in [35] describes methods for "fast-backprojection" that attempt to circumvent shortcomings in computational complexity and time expense. Yegulalp proposes implementing backprojection on a pulse-by-pulse basis rather than a pixel-by-pixel basis. The data from each pulse is processed as a subaperture, and the final image is formed by coherently adding the images from each subaperture [35]. The pulse-by pulse subaperture methods reduce computational complexity from $O(N^3)$ to $O(N\sqrt{N})$ for a scene size of $N \times N$.

In order to implement backprojection in software, we form the image grid, which has a spacing determined by the system's range resolution. We will refer to the x and y-axis of this grid as x_i and y_i, to specify that the spacing of the imaging grid is not necessarily the same as that of the scattering grid $p(x, y)$. For each platform position, we calculate the bistatic range, R_B, to each point in the grid, $I(x_i, y_i)$. Note that the range from the transmitter to each pixel is independent of platform motion and does not change during the coherent processing interval. We then determine, for each pulse, which pixels lie within the set R_B minimum and maximum bistatic ranges observed in the collected data, $R_B = \{RB_{min}, \ldots, RB_{max}\}$:

$$\Xi(x_i, y_i, n) = \{R_B(x_i, y_i, n) > R_{Bmin} \cap R_B(x_i, y_i, n) < R_{Bmax}\}. \tag{3.8}$$

Variable Ξ represents a subset of the grid $p(x_i, y_i)$ with range from the radar between the minimum and maximum bistatic range observed over the CPI. We oversample the matched filter response given in (3.5) by a factor of 10; oversampling by a factor of 10 is regarded as a rule of thumb in order to perform the necessary nearest-neighbor interpolation [30]. We apply a carrier phase correction so that the scatterer in the center of the scene has zero phase

$$\phi(x_i, y_i, n) = exp(jk\Delta R_B(x_0, y_0, n)) \tag{3.9}$$

where $\Delta R(x_0, y_0, n)$ is the difference in the bistatic range between the platform location on the nth pulse and pixels, $p(x_i, y_i)$ [30]. In (3.9), ΔR_B represents the total bistatic range. This ΔR_B is usually expressed as a one-way range in literature, which requires an additional factor of 2. We interpolate the oversampled matched filter data onto the pixels imaged during each pulse:

$$\Upsilon(x_i, y_i, n) = s_{int}(\Psi(m_u), \Xi(x_i, y_i, n)), \tag{3.10}$$

where the subscript u indicates that the matched filter Ψ has been upsampled and $s_{int}(.)$ represents the interpolation operation. The interpolation operation projects the matched filter data defined on a uniformly-spaced upsampled grid, m_u, onto the pixels in the image grid specified by $\Xi(x_i, y_i, n)$. Because the received signal is bandlimited, a sinc kernel is ideal for this interpolation [30]. The interpolation is performed by linear interpolation on the zero-padded inverse Fourier transform. Because the interpolation is performed on a pulse-by-pulse basis, we combine the projected range data for each pulse, $\Upsilon(x_i, y_i, n)$, to produce an image, Ω, defined on the grid (x_i, y_i) with the appropriate phase correction to the pixels in Ξ:

$$\Omega(x_i, y_i) = \sum_{n=1}^{N} \Upsilon(x_i, y_i, n)\phi(\Xi(x_i, y_i, n)). \tag{3.11}$$

The result is a matrix with size $P_i \times P_i = P_i^2$ where the axes are the x and y coordinates (in meters) of the resulting SAR image that represent the true dimensions of the illuminated swath.

The backprojection algorithm is a "brute force" method of imaging that is not known for being computationally efficient. However, improvements since the algorithm was first proposed in 1992 in [34] have made backprojection far easier to implement. Figure 3.2 gives the relatively straightforward structure of the backprojection algorithm; there is a single loop, which goes through the N subapertures corresponding to each pulse, adding the contribution of each to form an image. The number of operations in producing the SAR image from the matched filter data is dependent on the size of the image, P_i^2, the number of pulses N, and the length of the upsampled matched filtered data, $l_{m_u} = 10l_{\rho_{xx}}$ where $l_{\rho_{xx}}$ is the length of the autocorrelation function of the pulse ($l_{\rho_{xx}} = 2l_p - 1$). The total number of operations for each pulse, n is: $9I^2 + l_{m_u}$. Because, $P_i^2 = 500^2 = 250,000$ for our simulations, which is an order of magnitude larger than l_{m_u}, the computational expense will be dominated by the desired size of the image, P_i^2.

This concludes our discussion of the backprojection imaging algorithm. The following section describes the reverse backprojection method for data simulation that is based on applying principles of backprojection imaging *in reverse*. Instead of forming an image from pulse-compressed data, the reverse backprojection method operates in the opposite direction to simulate pulse-compressed data from a SAR image.

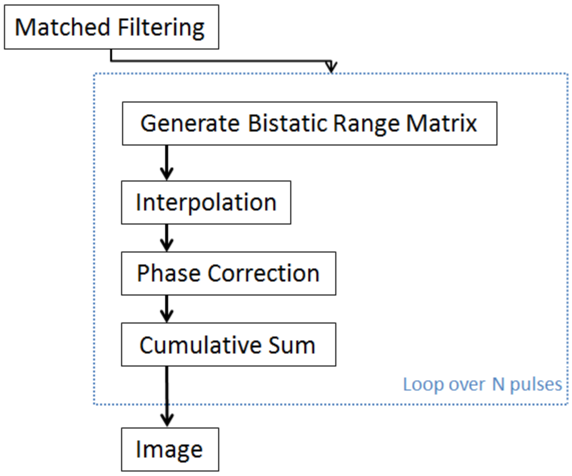


Figure 3.2: Backprojection Algorithm

3.3 Reverse Backprojection Algorithm Description

In this section we present the novel reverse backprojection method for pulse-compressed data simulation, which applies principles of the backprojection imaging algorithm described in the previous section. In order to describe a data collection simulation, it is necessary to first explain how pulse-Doppler radars can be used to acquire information about targets they illuminate. We describe in Section 2.1 how the target's range can be determined by the time delay of the received waveform, τ, and in Section 2.2 how information regarding velocity (with respect to the platform, in this work) can be inferred from a frequency shift of the received waveform, f_D. We can isolate the point target response of a SAR scene in range by cross-correlating the received signal with its matched filter, which in our case is a flipped, conjugated copy of the sampled direct-path signal. As we explored the ambiguity function in bistatic radar system in [2], target detection involves calculating the cross-correlation function between the direct path signal, $s_n(t)$, and a set of received pulses, $s_n(t-\tau)$:

$$\rho_{xx}(n,t) = \int_{-\infty}^{\infty} s_n(t) s_n^*(t-\tau)dt \tag{3.12}$$

as described in [41]. The autocorrelation function has length $l_{\rho_{xx}}$, which is equal to one less than twice the length of $s_n(t-\tau)$ (the signal $s_n(t)$ that has been sampled by an analog-to-digital converter), assuming that the direct-path signal and the received pulses have equal lengths. Because the imaged region of interest is assumed to be some distance away from the radar, the radar's ADC will wait until the bistatic time delay of the closest patches before taking samples. We will call the minimum amount of samples required to cover the range swath, l_0 and the maximum, l_p; the number of necessary samples to cover the entire range swath is therefore $l_p - l_0$. Because each sampled LTE pulse is different, we also divide by the energy in the nth pulse, $E[n]$, to normalize the pulses. From this matched filter data we can form an image as described in the previous section. Figure 3.3 depicts an example matched filter output for an LTE pulse and Figure 3.4 shows the autocorrelation function, $\rho_{xx}(\tau_m)$ of 16 randomly generated LTE pulses with the same number of resource blocks. The matched filter output, $\xi(m)$ has sidelobes approximately 12 dB down from the main peak, which corresponds to the point at which the waveform overlaps its matched filter completely during convolution.

The method described in Chapter 3.2 would require simulated autocorrelation functions for every point in the imaged scene on a continuous set of delays. We can exploit the conventional backprojection algorithm used for imaging by performing backprojection *in reverse* to generate simulated SAR data on a set of quantized delays. We calculate the contribution of each patch to the sampled data by scaling pre-calculated autocorrelation functions by the clutter RCS values, $\sigma(x,y)$, that are shifted by a finite number of quantization levels, Q, within a resolution cell. Be-

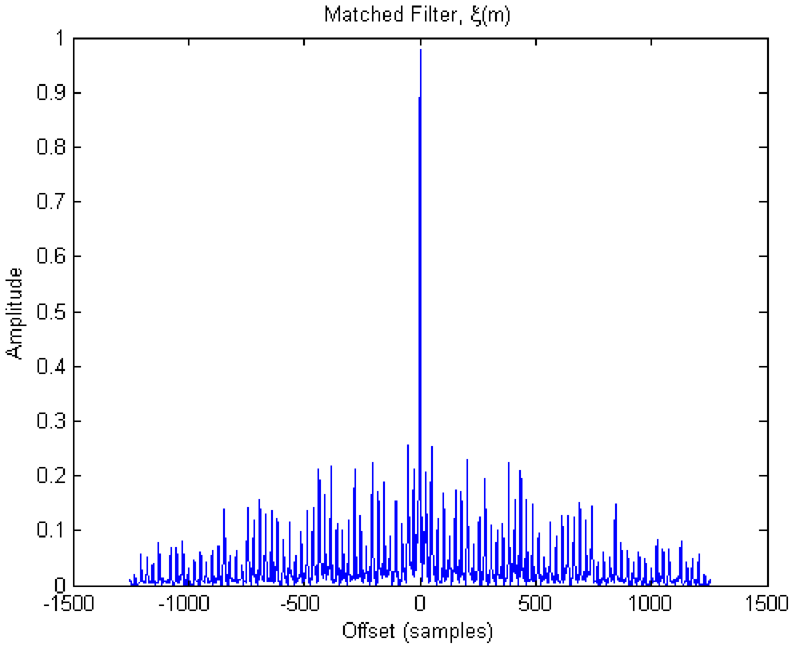


Figure 3.3: Matched filter output for an LTE pulse

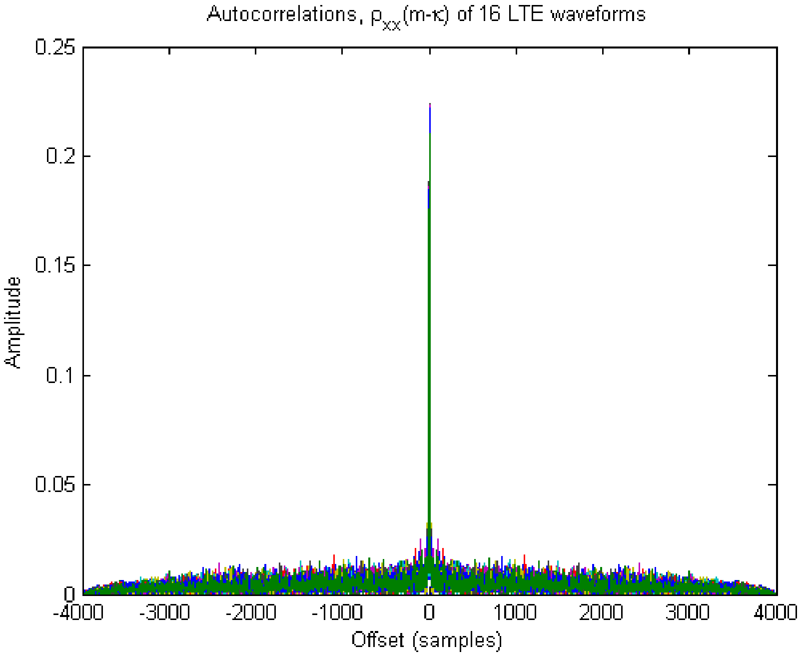


Figure 3.4: Autocorrelation functions of 16 LTE pulses

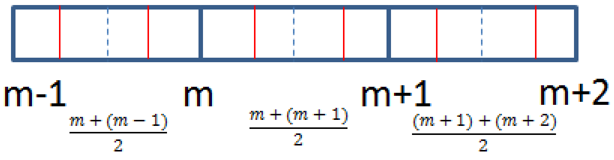


Figure 3.5: Upsampled range bin axis. Red lines correspond to new effective range bins added by the shifted autocorrelation matrix

cause the center of each patch most likely does not align with the center of a range bin, by creating an upsampled LOS vector with discrete delay values, τ_m, we can accurately simulate the matched filtered data by performing a series of discretized interpolations. Figure 3.5, consisting of three consecutive range bins on axis, m, illustrates the upsampled range axis that allows us to simulate a scatterer's delay to a fraction of a range bin; for three quantization levels, the two shifted waveforms correspond to delays one-quarter of the way between adjacent range bin centers. More effects of the chosen number of quantization levels are discussed in Section 4.5. Instead of calculating these contributions in a pixel-by-pixel manner, we follow a similar approach described in [35] that calculates the contribution of all the pixels during each *pulse collection* as a subaperture. Because each range bin in the radar's LOS vector corresponds to an elliptical contour, $\zeta(x, y; m)$, of pixels in the scattering grid $p(x, y)$, it is most effective to calculate the contributions of each of the range bin's contours at once, since each sample of the matched filtered return represents an isorange contour in the image scene.

In order to create the autocorrelation matrix, $\rho_{xx}(q, \tau_m; n)$, we make Q copies of each row of $\rho_{xx}(\tau_m)$ and apply an array of shifts, κ,

$$\kappa = \frac{[\lfloor \frac{-Q}{2} \rfloor \ldots 0 \ldots \lfloor \frac{Q}{2} \rfloor]}{\lceil \frac{Q}{2} \rceil} \tag{3.13}$$

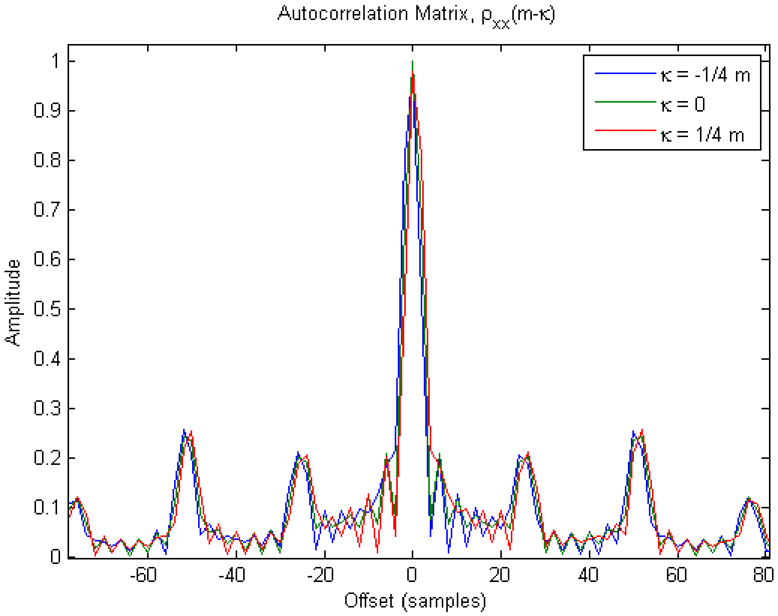


Figure 3.6: Shifted waveforms, $\rho_{xx}(m-\kappa)$

so that the end result for each platform position, n, is a matrix with an odd number of rows, $\rho_{xx}(q,\tau_m)$, where the center row experiences zero shift. The peaks of the Q rows of $\rho_{xx}(q,\tau_m)$ are, therefore, offset by increments of $\kappa = \frac{1}{Q+1}$, as κ approaches $\pm\frac{1}{2}$. Because each sample in the received radar data corresponds to a range resolution cell, the shifts chosen are within ± half a sample of the autocorrelation peak, $\rho_{xx}(0)$, the center of the function. We create this matrix of shifted functions that we can scale shifted copies of the function to better simulate the data from an upsampled axis of range bin delays; an example with 3 quantization levels is given in Figure 3.6. To compute the contributions to each range bin, we loop through each value of m, in set $M = \{l_0, l_0+1, \ldots, l_p\}$ and first, calculate the range corresponding to the center of each range bin, $\xi(m)$ in M:

$$\xi(m;n) = c(m-1)F_s - R_{B0}(n), \tag{3.14}$$

where $R_{B0}(n)$ is the bistatic range to the center pixel from platform position n; this value is subtracted so that the center pixel will always occupy the range bin corresponding to a range of 0. We define $R_{B\Delta}$ as the distance between each scatter's bistatic range, $R_B(x, y; n)$ (for the current pulse n) and the bistatic range to P_0, $R_{B0}(n)$ and subtract that value from $\xi(m)$ in order to compensate for the range to the phase center

$$\Delta m(x, y) = \xi(m) - R_{B\Delta}(x, y; n), \tag{3.15}$$

where $\Delta m(x, y)$ is the fraction of a sample between the center of each range bin in M and the bistatic range of each patch. Samples with an absolute value of 1/2 or less are within half a sample of the range bin center and are in that range bin's contour, $\zeta(x, y; m)$; therefore the RCS values of these patches contribute to the range response of that sample:

$$\zeta(x, y; m) = \begin{cases} 1 & |\Delta m| < \frac{c}{2BW} \\ 0 & |\Delta m| \geq \frac{c}{2BW} \end{cases} \tag{3.16}$$

The variable $\zeta(x, y; m)$ is a Boolean matrix representing the values in $p(x, y)$ corresponding to each particular range bin, m. Those values are quantized to determine how much the autocorrelation must be shifted to model the range response from each pixel belonging to the contour, $\zeta(x, y; m)$:

$$\zeta(x, y; m_q) = \left| \frac{\Delta m(\zeta(m)) Q \beta}{c} - \frac{Q}{2} \right|. \tag{3.17}$$

where $\zeta(x, y; m_q)$ ranges from $q = \{1, 2, \ldots, Q\}$. Each value of $\zeta(x, y; m_q)$ will correspond to a row of $\rho_{xx}(q, \tau_m; n)$; these values determine which shifted autocorrelation function has a peak closest to the true delay for each scatterer in $\zeta(x, y; m)$. We choose this range so that each $\zeta(x, y; m_q)$ can correspond to a matrix row. In order to properly scale the appropriate $\rho_{xx}(q, \tau_m; n)$ of those pixels in $\zeta(x, y; m)$, we must extract the RCS constants from those pixels. We must also add the carrier phase rotation associated with the exact $R_B(x, y; n)$ of each pixel; because these values are all pre-calculated and stored, this operation is not computationally expensive and

$$J = p(\zeta(x, y; m))e^{-jkR_B(\zeta(x,y;m))}, \tag{3.18}$$

where k is the wavenumber, $\frac{2\pi f_c}{c}$. In order to add the right amounts of each shifted row of $\rho_{xx}(q, \tau_m; n)$ to the updating sum of the SAR data, D, we must loop through each q in Q and sum the values of J whose indices correspond to each Δm_q:

$$\sigma_q(q) = \sum_{q=1}^{Q} J(\zeta(x, y; m_q) == q). \tag{3.19}$$

This operation yields the total RCS constant, $\sigma_q(q)$ for each q, and determines how much we scale each row in $\rho_{xx}(q, \tau_m; n)$ to represent the contribution from each scatterer in $\zeta(x, y; m)$ to the SAR data. When we calculate the total contribution of each $\zeta(x, y; m)$, we will therefore only have to compute a Q number of products if $\sigma_q(q)$ are calculated previously. Because there is an additional time offset corresponding to where a particular pulse lies within the radar's CPI, we extract a separate interval of $l_p - l_0$ samples from the Q rows of $\rho_{xx}(q, \tau_m; n)$ so that the peak of the chosen $\rho_{xx}(q, \tau_m)$ will be in the m^{th} position with a slight offset. We determine the starting and ending indices, $i_o(m)$ and $i_{end}(m)$ for each m:

$$i_0(m) = l_{center} - ((m - l_0) - 1), \tag{3.20}$$

where l_{center} is the midpoint of $l_p - l_0$, and

$$i_{end}(m) = i_0(m) + l_p - 1. \tag{3.21}$$

Finally, for each range bin we multiply the RCS constant $\sigma_q(q)$ associated with the relevant data points of each row of $\rho_{xx}(q, \tau_m; n)$:

$$D(n, \tau_m) = \sum_{m=l_0}^{l_p} \sum_{q=1}^{Q} \sigma_q(q) \rho_{xx}(q, i_0(m) : i_{end}(m)), \tag{3.22}$$

so that the final result is the simulated data from the entire scene for each of the n pulses. The data from each of the n pulses is stored in the way described in Chapter 2.1, in a format can be easily used for image formation. The following section will discuss the computational complexity of the reverse backprojection method and the improvement in efficiency compared to matched filtering.

3.4 Reverse Backprojection Method Complexity

Now that we have proposed a novel approach to pulse-compressed data simulation, we must discuss its computational improvement compared to pre-existing simulation methods. Despite being a solution to a computationally expensive problem, the reverse backprojection algorithm is not computationally "cheap." The algorithm contains three nested for loops: a loop to account for all pulses, a loop for all range bins, and a loop for all quantization numbers; Figure 3.7 shows the flow and structure of the loops within the reverse backprojection algorithm. However, because the number of quantization levels, Q, is much smaller than the number of range bins K, increasing the number of quantization levels does not add a substantial amount of computational expense. The number of operations also greatly depends on the size,

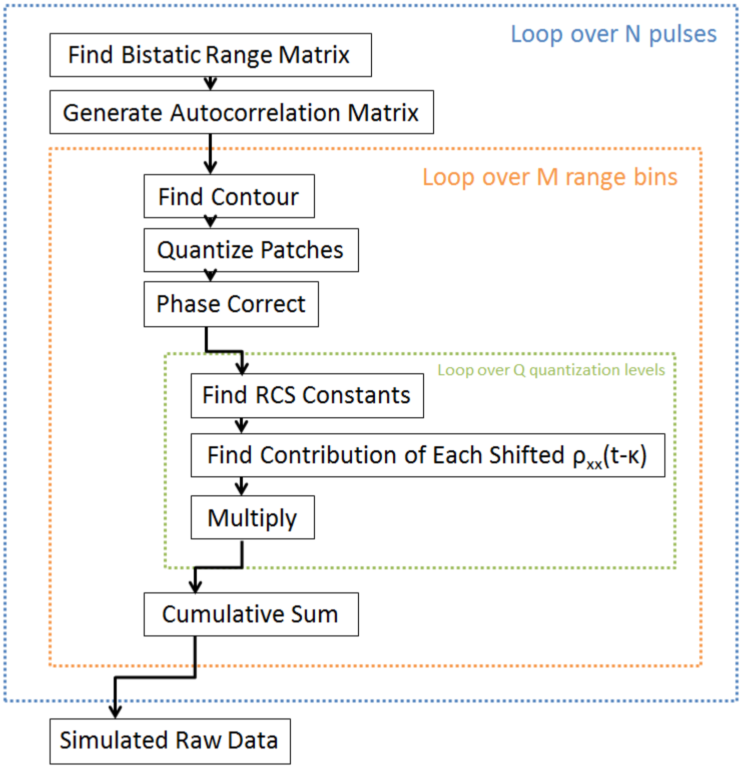


Figure 3.7: Reverse Backprojection Algorithm

$P \times P$, of the grid, $p(x, y)$; even though we are not performing pixel-by-pixel operations, we have to perform many searches over the entire grid. Therefore, increasing the size of the grid significantly increases processing time. Some of the operations are also dependent on the length of $\rho_{xx}(q, \tau_m; n)$, $l_{\rho_{xx}}$, which is dependent on the pulse length; using a longer waveform (such as an entire LTE radio frame) will significantly increase processing time. Therefore, we use a much smaller fraction of an entire radio frame as our transmitted signal. Within the outermost loop (over all N pulses), we calculate the appropriate bistatic range matrix, $R_B(x, y; n)$, the bistatic range to each pixel, R_{B0}, and create the autocorrelation matrix $\rho_{xx}(q, \tau_m; n)$. Within the first nested loop (over all M range bins), we determine which pixels belong to that range bin, quantize them, add the phase correction, and perform the operation in (3.22) once the appropriate $\sigma_q(q)$ have been determined in the innermost loop. The innermost loop (over Q quantization levels) finds the total $\sigma_q(q)$ for each Q. For this loop, computation speed is greatly improved by employing the MATLAB built-in functions **arrayfun** and **bsxfun**, which sort the $\sigma_q(q)$ in cell arrays that can quickly be summed and concatenated into a single matrix. Because Q is assumed to be considerably smaller than M and N, using **arrayfun** and **bsxfun** on this loop does demand an unendurable amount of RAM. Because each patch of $p(x, y)$ belongs to only one range bin contour, $\zeta(x, y; m)$ (which contain a variable number of patches, $p_{\zeta(x,y;m)}$), the total number of operations performed to each patch in each contour is therefore $Mp_{\zeta(x,y;m)} = M(P \times P) = MP^2$. For most of our simulations in Chapter 4, the chosen grid size is 2000x2000 and the associated autocorrelation length is 1,255. Because $2,000^2 = 4,000,000$ is much larger than 1,255, it is only necessary to consider the operations dependent on grid size for complexity characterization. For each pulse, the number of calculations is therefore approximately: $P^2(3MQ + 10M)$. This concludes our discussion of the reverse backprojection method for data simulation; Chapter 4 presents results using the methods described in this chapter.

Chapter 4

Results

In this chapter we present our results employing the novel reverse backprojection method for pulse-compressed SAR data simulation described in Chapter 3 for various passive bistatic radar configurations. In Section 4.1, we discuss simulation parameters and the data generated from the point scatterer response (PSR). We describe observed ambiguities in the resulting data and images originating from deterministic features contained within LTE waveform pulses in Section 4.2, and validate the effectiveness of our simulation method in accounting for nuances in generated data resulting from various waveform characteristics in Sections 4.2 and 4.3. In Section 4.4, we expand our analysis to distributed SAR scenes more closely resembling those in real-time SAR data collection. Finally, in Section 4.5, we discuss the effects of changing the number of quantization levels (as described in Section 3.3) on the simulated pulse-compressed data.

4.1 Point Scatterer Response

The image formed from the data collected from the SAR system in response to a single point scatterer in the center of the scene is known as the point scatterer response (PSR). The PSR will serve as an effective visualization tool to examine

PRI	1 ms
Platform velocity	200 $\frac{m}{s}$
LTE NDLRB	[25 25 25 25 25 25]
Bandwidth	29 MHz
Carrier Frequency	728 MHz
Platform positions	1024
Pulse length	628 samples
Quantization levels	3
Scene size levels	2.59x2.59 km

Table 4.1: Canonical Simulation Properties

how differences in waveform properties, geometry, and the number of quantization levels effect the resulting SAR image produced by our simulation. To do this, it is important to describe our canonical case; that is, the simulation properties that will be assumed to be unchanged for all simulations.

Table 4.1 lists the characteristics that will be assumed (unless otherwise noted) for each of the simulations presented in this chapter. The PRI and platform velocity are chosen so that we can visualize the effects of range migration in the simulated pulse-compressed data. The chosen LTE waveform with 6 channels containing 25 resource blocks each is chosen because waveforms with higher bandwidth give better resolution; the spectrum for the standard waveform is given in Figure 4.1. The given carrier frequency was selected because that is the center frequency of one of the allocated bands assigned by the FCC to Norman, Oklahoma for cellular communications. The scene size was chosen on the order of kilometers to model an airborne SAR system, and the pulse length is calculated from (2.1). The number of quantization levels is chosen to be 3, to examine the performance of the simulation using the lowest number of quantization levels (and computation time).

In order to illustrate the the point scatterer response, we performed simulations with a single point scatterer, P_0, located in the center of our imaged scene, as seen

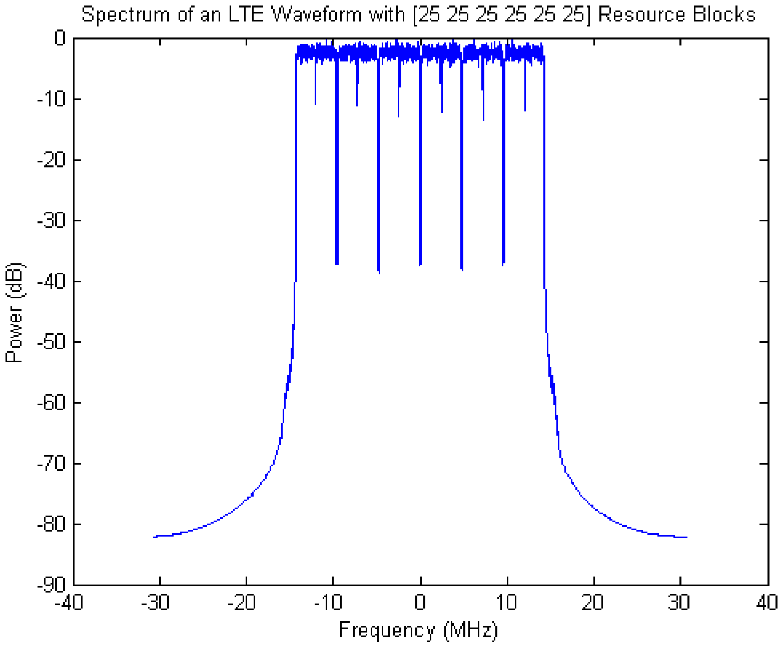


Figure 4.1: Spectrum of LTE waveform when NDLRB = [25 25 25 25 25 25]

in Figure 4.2 with an RCS amplitude of 5 and a random phase. Figure 4.3 shows the simulated pulse-compressed data output. Because the platform collects data over 1,024 positions, the response from the pixel P_0 migrates through range bins as the platform approaches the scatter's x-coordinate can be seen by close inspection of Figure 4.3. Because of periodic cross-ambiguities in the LTE pulsed waveform, some of the energy is distributed in range bins not corresponding to the correct range. Figure 4.4 is the zoomed-in image output when the data in Figure 4.3 is used in the backprojection imaging algorithm; because the data is collected over 1,024 pulses using a pulse corresponding to a fine resolution, the data appears very focused in both range and Doppler. We upsampled the image in Figure 4.4 by a factor of 10 so that fine details in the image could be more easily interpreted. Range ambiguities appear along the y-axis and Doppler ambiguities appear alongside the x-axis. These Doppler ambiguities result from aliasing in azimuth due to sidelobes

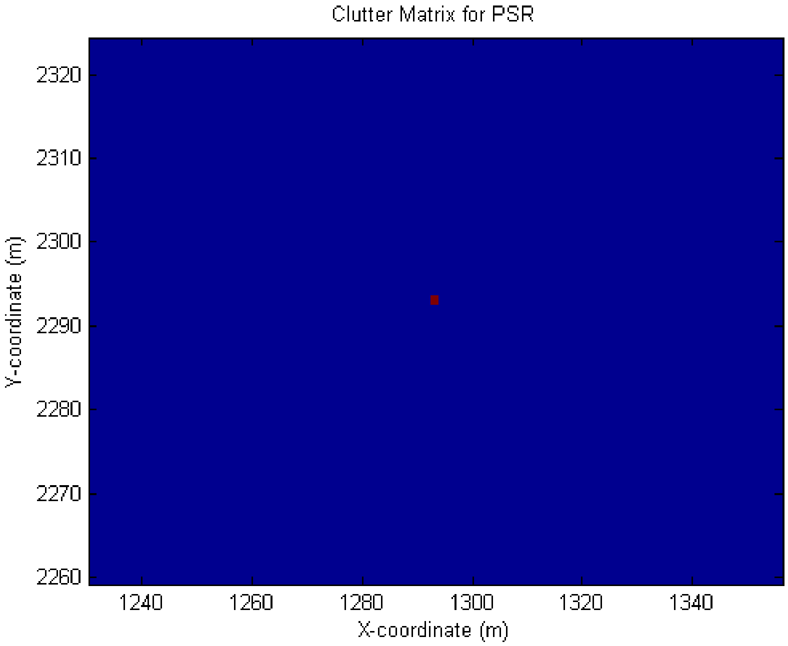


Figure 4.2: Clutter matrix for the PSR (zoomed in)

in the antenna pattern, and will always exist because of pulse repetition frequency (PRF) constraints [42]. The Doppler aliasing results in "ghost targets" that appear in the image because the matched filter perceives the aliased parts of the received signal's phase patterns to be identical [42].

In order to examine the effects of the LTE waveform, which has not been designed for radar usage, on radar performance, we performed a series of simulations on LFM pulses. If amplitude modulation is not employed in the waveform, its amplitude will be sustained at its maximum value, A, which results in a pulse energy of $A^2\tau$ [31]. This presents a challenge because higher pulse energies (longer pulses) improve estimation and detection performance (SNR), while increased resolution requires more bandwidth. Therefore, radar engineers employ pulse compression to improve both detection performance and resolution without compromising the other. Pulse compression is performed by frequency or phase modulating a simple

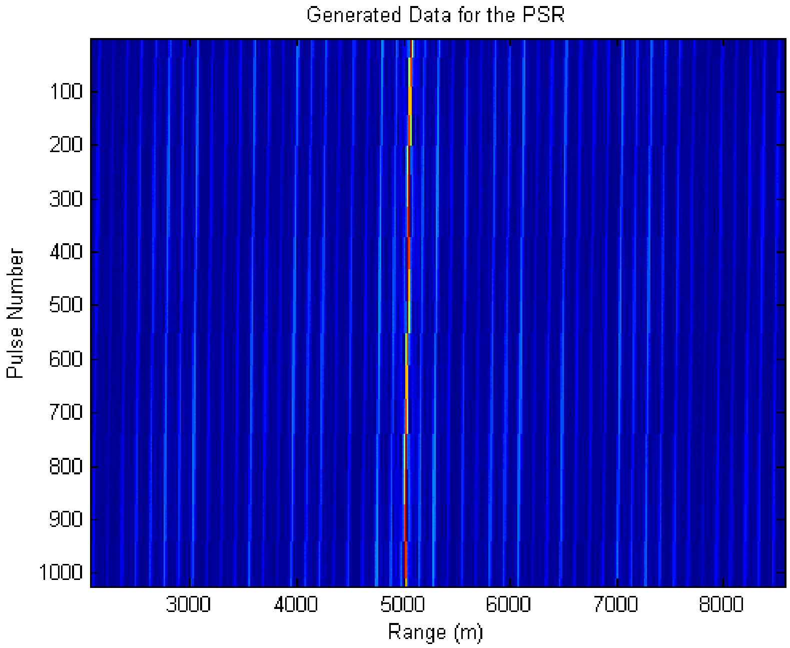


Figure 4.3: Generated pulse-compressed data for the PSR

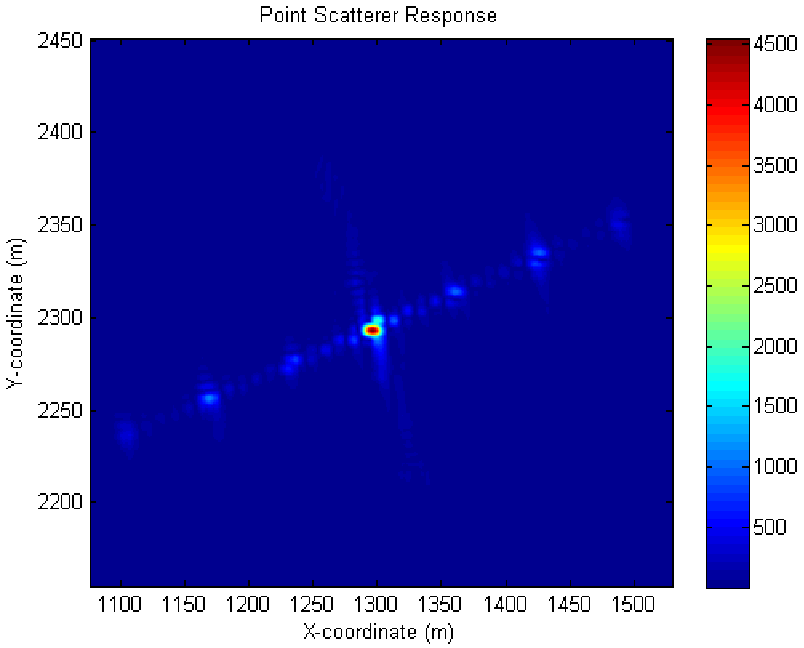


Figure 4.4: Zoomed in image of the PSR

pulse. The instantaneous frequency, $F_i(t)$, of an LFM waveform linearly sweeps across β Hz of bandwidth in τ seconds. The LFM pulse offers a signal processing gain by a factor of its compression ratio, the *time bandwidth product*, $\beta\tau$, in comparison to a non-modulated waveform with the same range resolution. Therefore, a radar waveform can be designed to offer both a long duration and a narrowly concentrated autocorrelation [31]. For our simulations with a pulse time of 1 ms, we choose a similar bandwidth as the standard LTE waveform and obtain a $\beta\tau$ of 30,000. The reverse backprojection algorithm is flexible enough to be used with any transmitted waveform, whether it is a conventionally used LFM pulse, another specifically-engineered radar pulse, an LTE waveform, or alternate communication waveform.

In order to determine how LTE waveform self-ambiguities affect simulation performance we compare the point scatterer response of the reverse and “forward” backprojection algorithms using LTE waveforms and conventional LFM waveforms. Using a the PSR from a single pulse can also be useful, since the image will not be focused in Doppler. Figures 4.5 and 4.6 illustrate the generated data and image when a single LFM pulse is employed. Because only one pulse is used, the image is not resolved in Doppler and is therefore the entire isorange contour corresponding to the range to P_0. The PSR resulting from simulations with a single pulse are also useful because they illustrate how features of the LTE waveform are translated into the pulse-compressed data output and final constructed image; more details about LTE waveform features observed in the PSR will be discussed in Section 4.3. In Figure 4.7, resulting from the same properties except using an LTE pulse instead, many range ambiguities are present because the LTE waveform has high cross-correlation with itself over time. When simulations are performed using more pulses over long apertures, these self-ambiguities are also evident in the im-

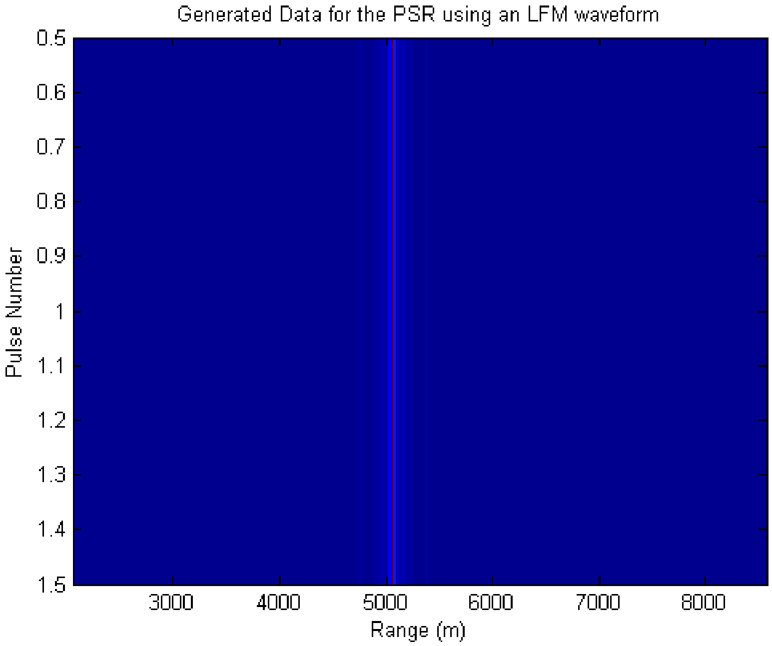


Figure 4.5: Generated pulse-compressed data for the PSR using an LFM waveform and only one pulse

age point spread responses. The data collected for a simulation utilizing an LFM pulse is featured in Figure 4.8, and Figure 4.9 features the image formed (PSR) using that data that has been upsampled by a factor of 10 for interpretation purposes. In contrast to the generated pulse-compressed data and image featured in Figures 4.3 and 4.4, which show ambiguities in range in the data, defocusing in range in the image, and multiple Doppler ambiguities, from Figure 4.9), it is evident that the image is tightly focused in range and Doppler, with almost no ambiguities in either dimension. Therefore, because of the self-ambiguities, using passive bistatic waveform sources such as LTE is not ideal for radar purposes and is not superior to more conventional waveforms such as LFM chirps.

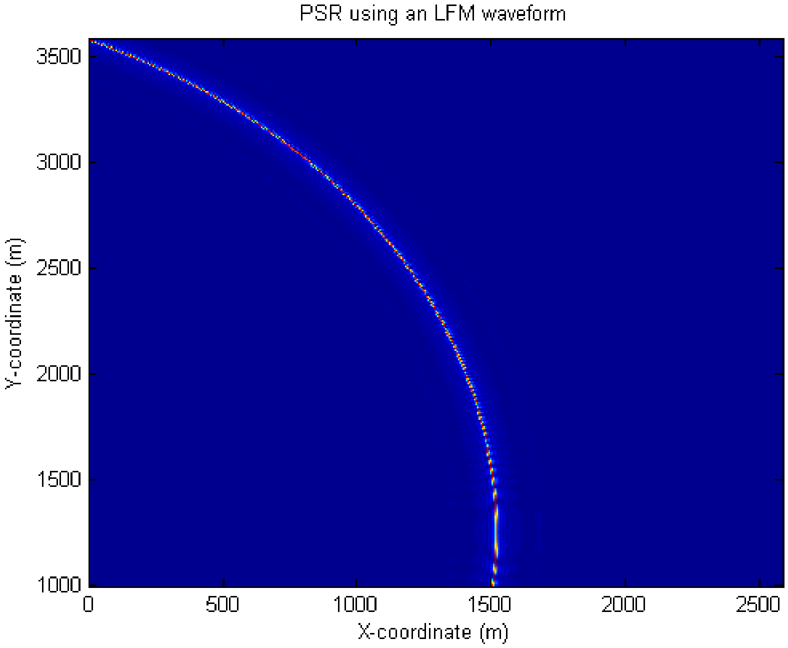


Figure 4.6: Image of the PSR using an LFM waveform and only one pulse

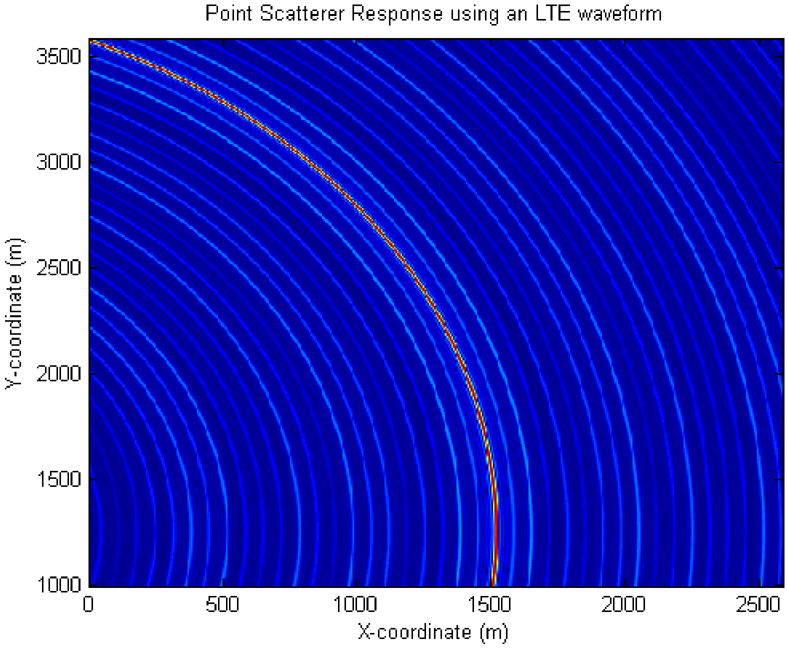


Figure 4.7: Image of the PSR using an LTE waveform and only one pulse

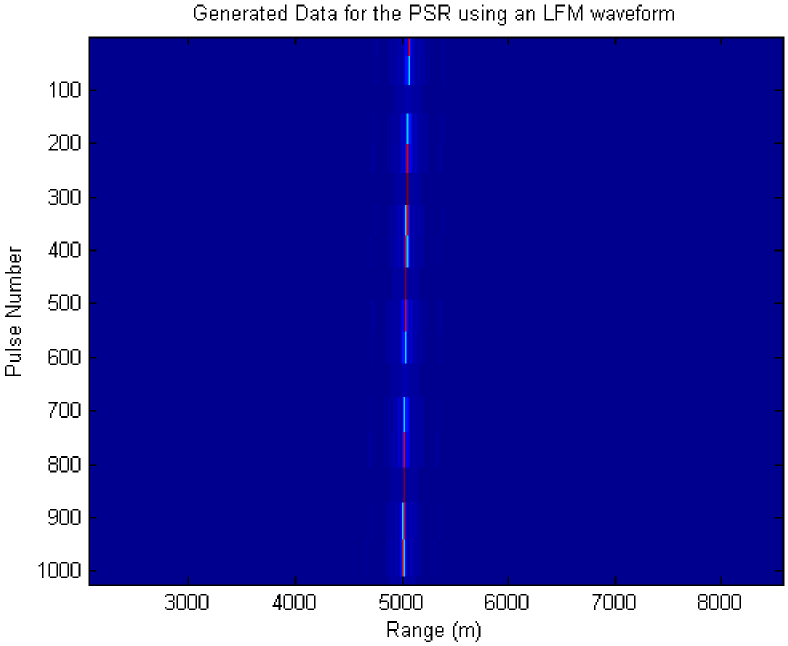


Figure 4.8: Generated pulse-compressed data for the PSR using an LFM waveform and 1,024 pulses

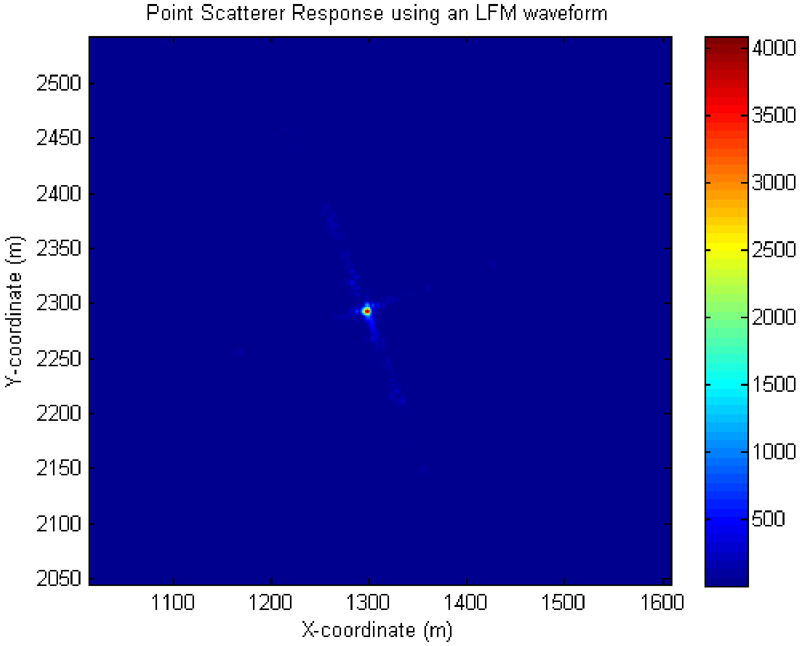


Figure 4.9: Zoomed in image of the PSR using an LFM waveform and 1,024 pulses

4.2 LTE Waveform Self-Ambiguities

As discussed in Chapter 2, LTE waveforms contain deterministic features necessary to their usage as communication waveforms that cause self-ambiguities in the autocorrelation that can be unfavorable for usage as radar waveforms. LTE waveforms also contain nulls between bursts of data; if a "pulse" is extracted during one of those nulls, we will be unable to perform matched filtering with that pulse. An example of an LTE waveform plotted over time is given in Figure 4.10, which features a single radio frame of a waveform with [25 25 25 25 25 25] resource blocks; nulls (periods of time with zero amplitude) are evident around the middle of the transmission. As discussed in previous work in [2], different LTE pulses will also contain different amounts of energy, making normalization required for coherent processing. In Figure 4.11, which features a single radio frame with [25 25] resource blocks, the LTE waveform exhibits "bursts" of data through parts of the transmission, with visible separation between and similar nulls nearing the middle of the transmission; accordingly, pulses extracted from the "bursts" will contain much more energy than pulses extracted from guard periods or nulls. Deterministic features that can cause ambiguities include the CP, PSS, SSS, and CSREs. As Evers and Jackson discussed in [8], self-ambiguities caused by the PSS, SSS, and CSREs are so low energy that they do not have a significant effect on the performance of LTE waveforms for radar applications; however, their effects are still visible in simulated pulse-compressed data. Because the cyclic prefix has a higher amount of energy than the other deterministic features, self-ambiguities resulting from symbols containing a cyclic prefix do influence the waveform's autocorrelation and subsequent data generation and image formation.

We generated a set of data to visualize how extracting pulses from different

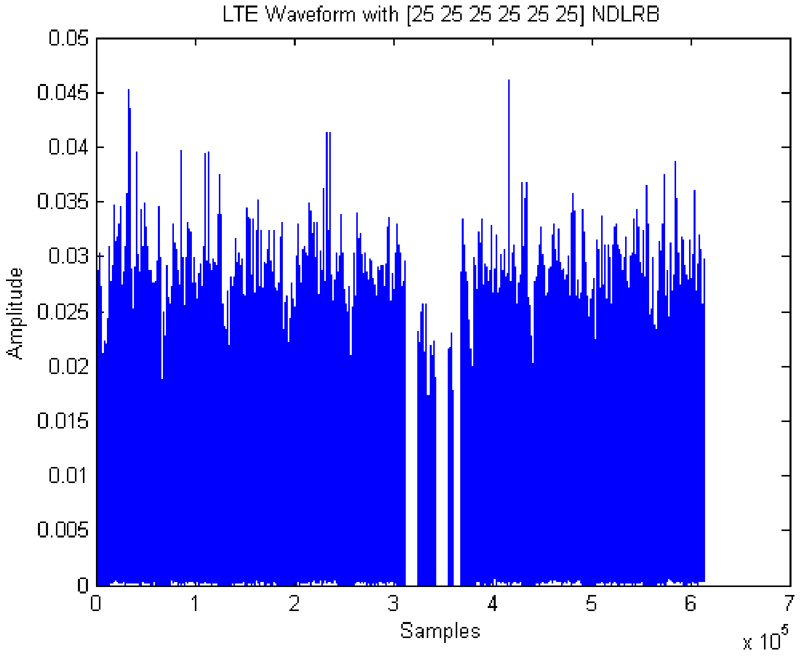


Figure 4.10: LTE waveform radio frame when NDLRB = [25 25 25 25 25 25]

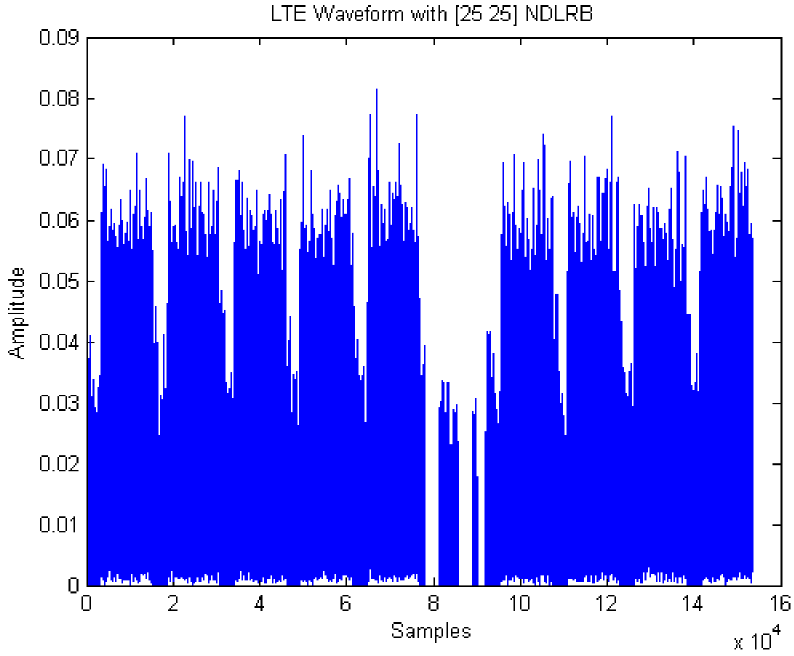


Figure 4.11: LTE waveform radio frame when NDLRB = [25 25]

PRI	1 ms	1 ms
Platform velocity	200 $\frac{m}{s}$	200 $\frac{m}{s}$
LTE NDLRB	[50]	[25 25]
Bandwidth	10 MHz	9.8 MHz
Carrier Frequency	728 MHz	728 MHz
Platform positions	1	1
Pulse length	676 samples	676 samples
Quantization levels	3	3
Scene size levels	7.65x7.65 km^2	7.65x7.65 km^2

Table 4.2: Simulation Properties

placements, ν, within the waveforms affect the waveform autocorrelation and generated pulse-compressed data; for this illustration, we will deviate from the standard simulation by using waveforms with [25 25] NDLRB. Figures 4.12 and 4.13 show the generated pulse-compressed data for the PSR for one pulse extracted from two different placements: one at .422 through the waveform and another at .500 through the waveform; this data allows us to visualize autocorrelation characteristics, since the generated data is made up of shifted and scaled copies of the waveform autocorrelation. Compared to the pulse used to produce the data in Figure 4.12, the pulse used to produce the data in Figure 4.13 exhibits stronger artifacts from self-ambiguities in the autocorrelation. These artifacts cause sidelobes that are evident in the image, shown in Figure 4.15, whereas the image formed from the data with low self-ambiguities given in Figure 4.14 has very low sidelobes.

One parameter that influences radar performance is the pulse bandwidth. Because the LTE waveforms transmitted by a tower are not controlled by radar engineers, performance may vary based on the tower's current bandwidth demand. We performed various simulations employing waveforms with different bandwidths and channel structures. Each pulse transmitted within each simulation is a different randomly generated waveform with the specified structure. The "standard" waveform with 6 channels containing 25 resource blocks each has a bandwidth of 29 MHz. In LTE waveforms, waveform bandwidth is determined by the number of allocated resource blocks (NDLRB). However, it is important to note that two waveforms with a different structure albeit identical bandwidths, will produce two different images because of that structure. Below we compare the structure and resulting images from two LTE waveforms: one with one channel containing 50 resource blocks ([50]) and another with two channels containing 25 resource blocks each ([25 25]). A full list of simulation parameters is given in Table 4.2. If we compare Figures 4.16 and 4.17, we can see that both waveforms have the same bandwidth because they have the same number of resource blocks. The LTE waveform in Figure 4.16 has a slight guard band between the two orthogonal frequency channels, but the occupied bandwidth is the same as the signal shown in Figure 4.17: each waveform has approximately 8 MHz of occupied bandwidth. We performed simulations with one pulse to generate pulse-compressed data from the PSRs for the same waveforms. In Figure 4.22, which compares the autocorrelation function between the two waveforms, it is evident that both waveforms' autocorrelation functions have similar structures because of the way the data are modulated, but that the two-channel waveform has higher sidelobes due to the guard interval between the channels; these differences are also evident in the pulse-compressed data from the PSRs visible in Figures 4.18 and 4.19. These sidelobes are also evident in the images - the image in Figure 4.20 has more visible sidelobe effects than the image in Figure 4.21. Therefore, bandwidth and NDLRB are not enough by themselves to completely characterize the autocorrelation function of a particular LTE waveform.

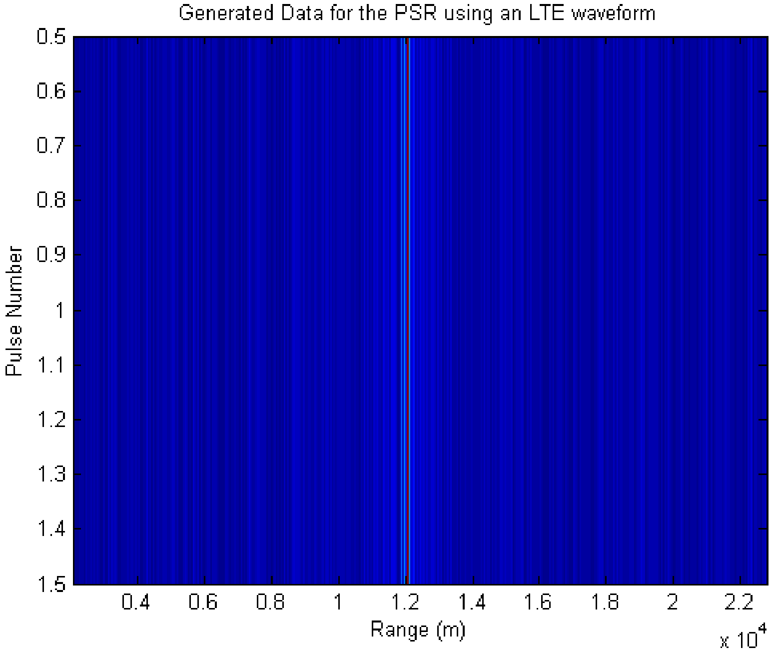


Figure 4.12: Generated pulse-compressed data for the PSR when NDLRB = [25 25] and .422

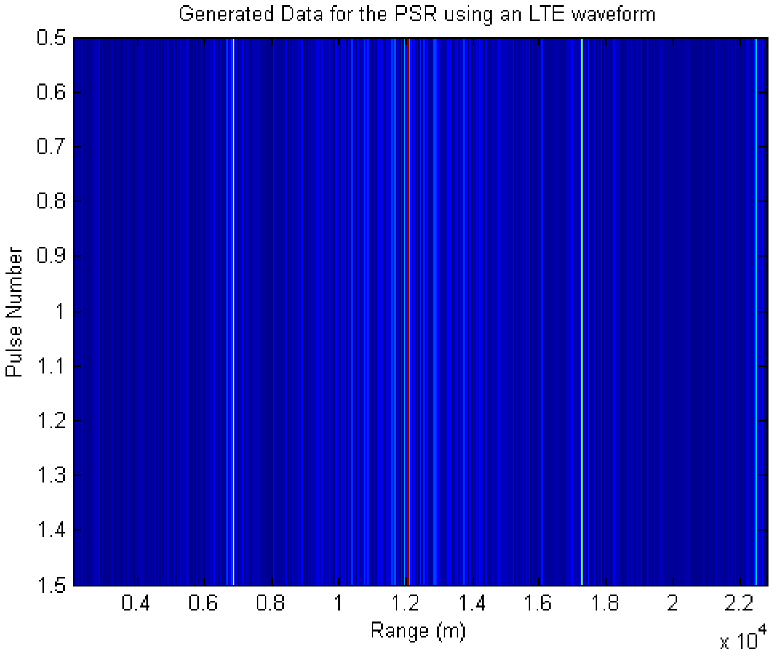


Figure 4.13: Generated pulse-compressed data for the PSR when NDLRB = [25 25] and .500

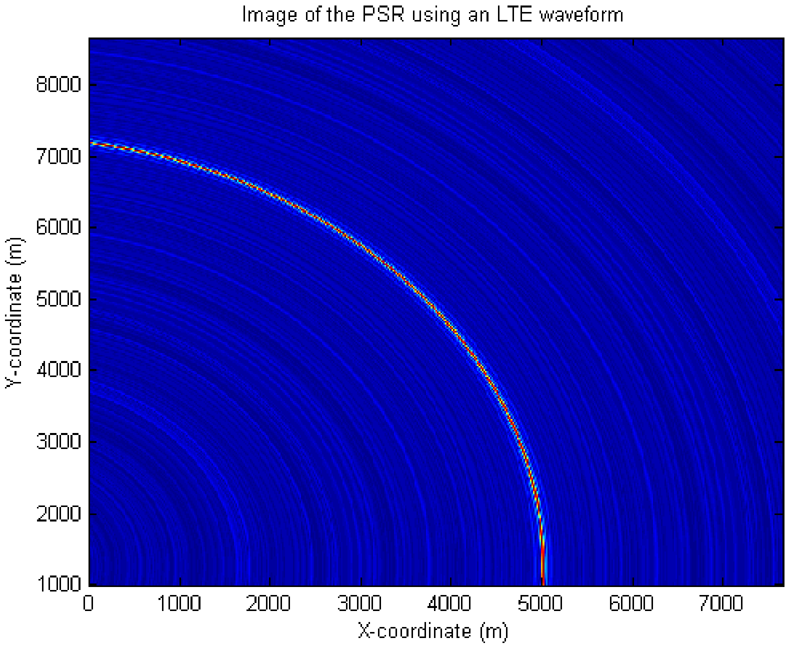


Figure 4.14: Image for the PSR when NDLRB = [25 25] and .422

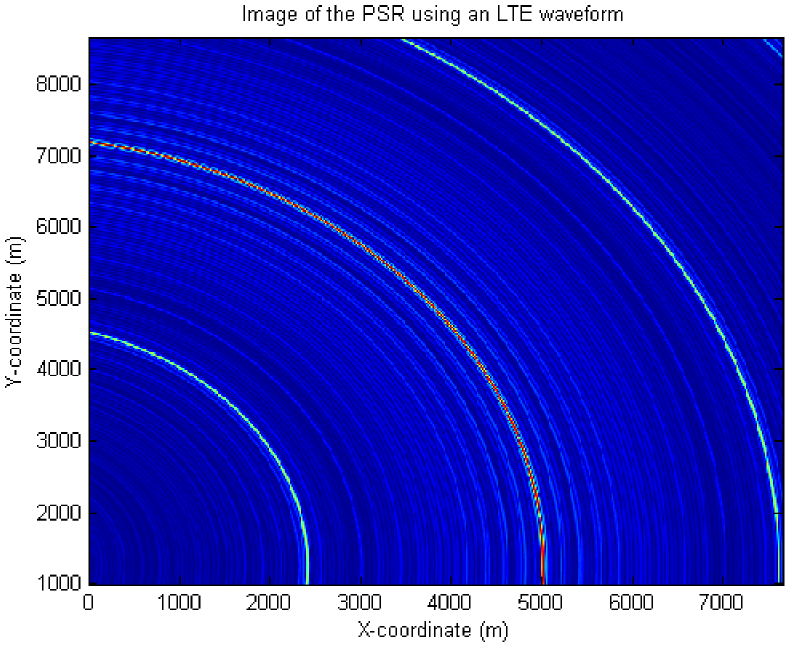


Figure 4.15: Image for the PSR when NDLRB = [25 25] and .500

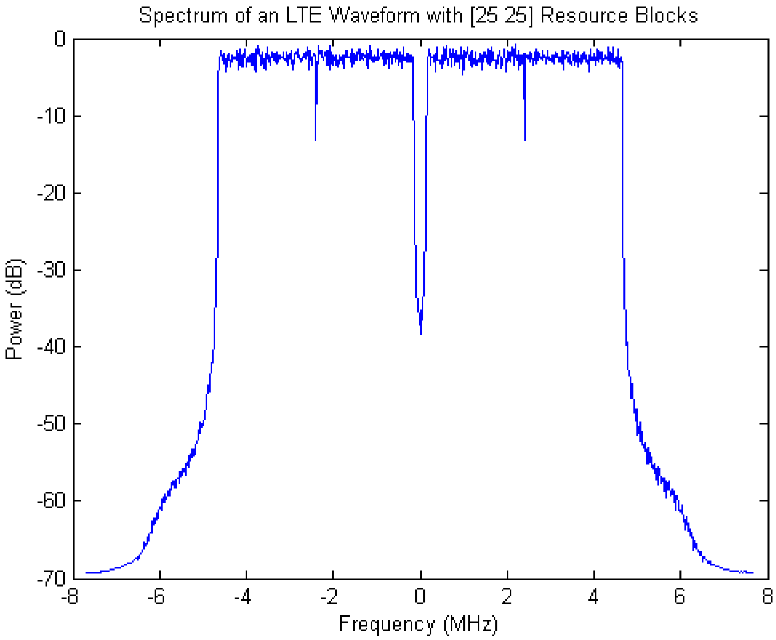


Figure 4.16: Spectrum of LTE waveform when NDLRB = [25 25]

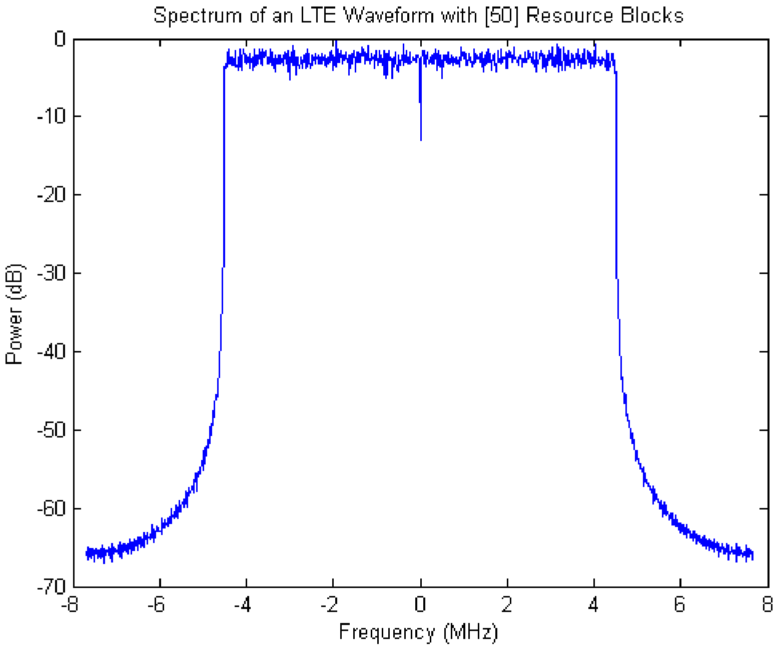


Figure 4.17: Spectrum of LTE waveform when NDLRB = [50]

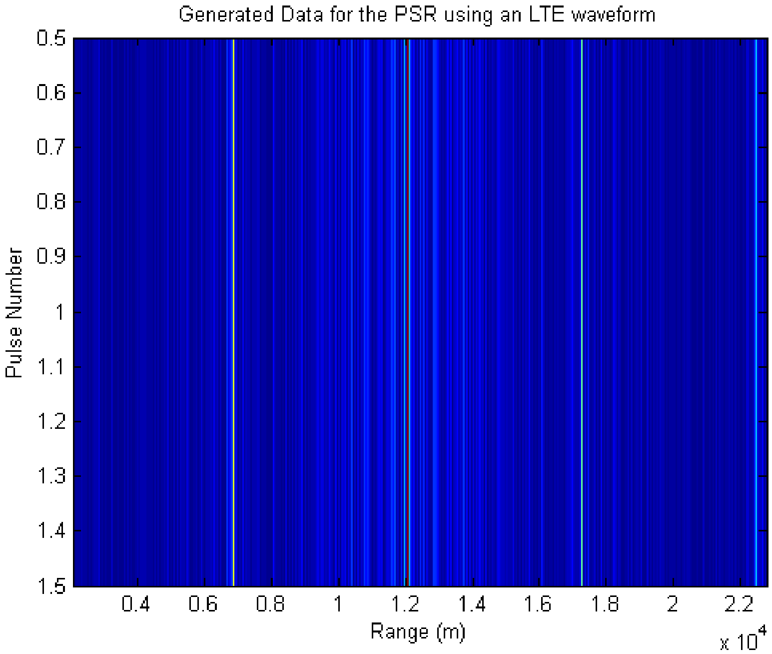


Figure 4.18: Generated pulse-compressed data for the PSR when NDLRB = [25 25] and .500

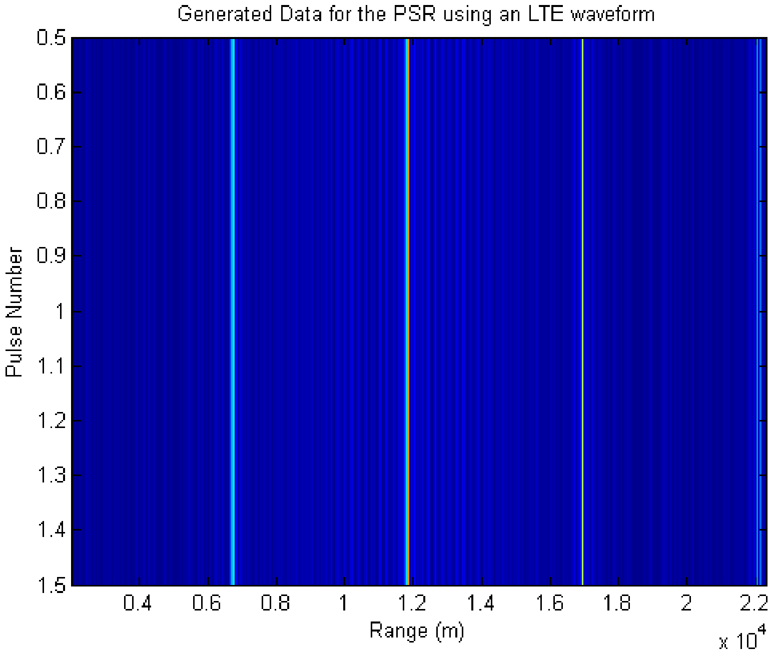


Figure 4.19: Generated pulse-compressed data for the PSR when NDLRB = [50] and .500

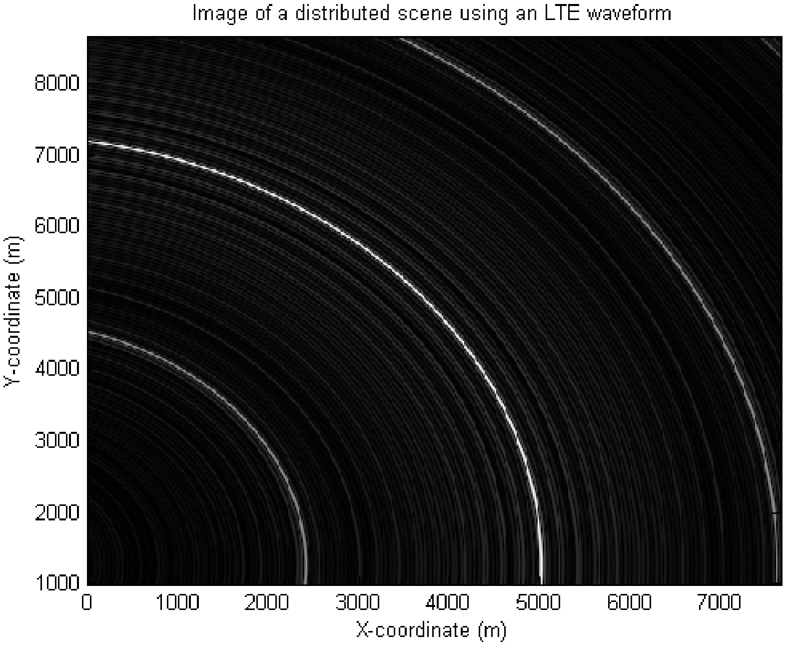


Figure 4.20: Image for the PSR when NDLRB = [25 25] and .500

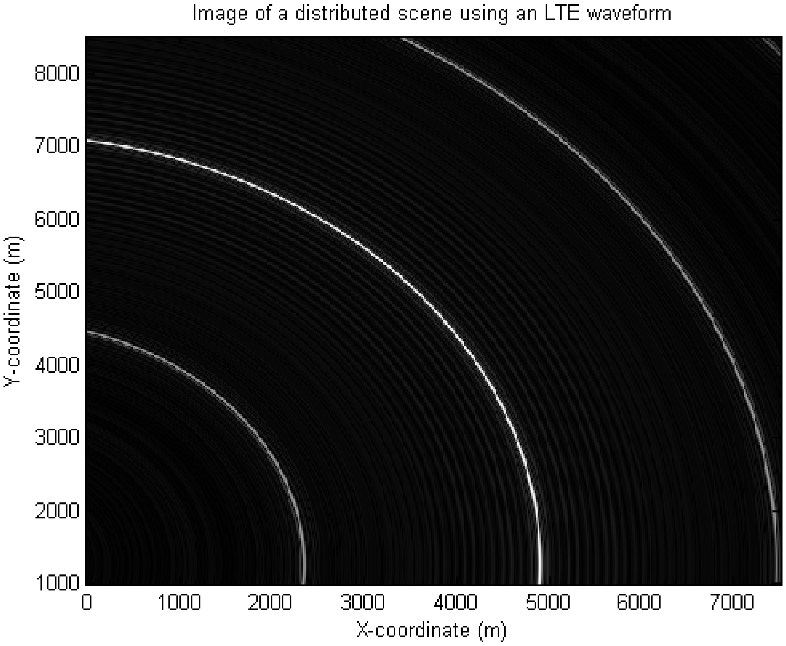


Figure 4.21: Image for the PSR when NDLRB = [50] and .500

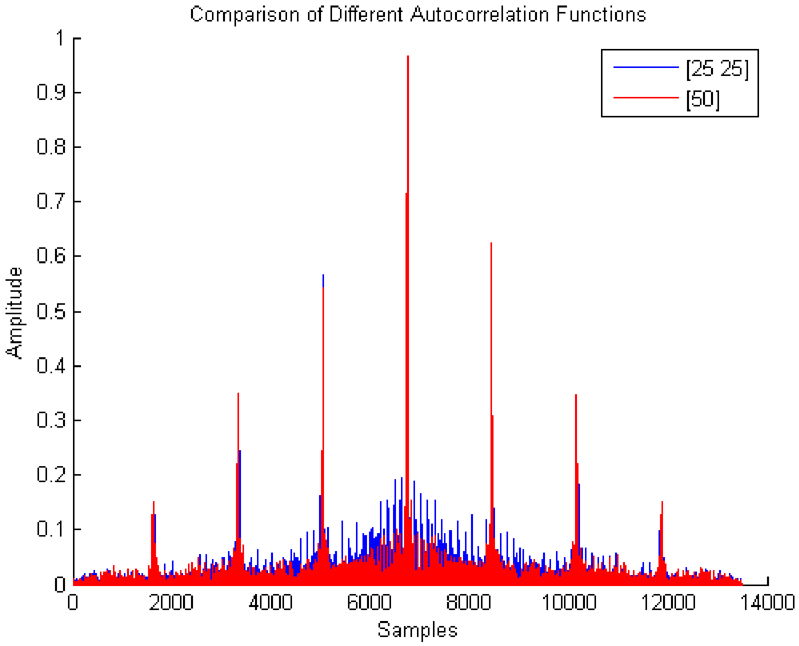


Figure 4.22: Comparison of autocorrelation matrices when NDLRB = [25 25] and [50]

4.3 LTE Bistatic SAR Range Resolution

In order to more directly compare range resolution between simulations using pulses from LTE waveforms with [25 25] resource blocks and [25 25 25 25 25 25] resource blocks, we performed side-looking simulations, with parameters given in Table 4.3, positioning the transmitter at the same location as a point scatterer, with an x-coordinate located in the middle of the platform's trajectory and a y-coordinate located in the midpoint of the range swath in y. This geometry allows us to visualize the Doppler sidelobes; the width of the mainlobe will be proportional to the range resolution because most of the energy from the imaged pixel will be observed in the main lobe. Figure 4.23 compares the intensity of the image for both waveforms. From a zoomed in perspective in Figure 4.24, we can see that the width of the mainlobe in the [25 25] case spans about 20 m and the width of the mainlobe in

PRI	1 ms	1 ms
Platform velocity	200 $\frac{m}{s}$	200 $\frac{m}{s}$
LTE NDLRB	[25 25 25 25 25 25]	[25 25]
Bandwidth	29 MHz	9.8 MHz
Carrier Frequency	728 MHz	728 MHz
Platform positions	1024	1024
Pulse length	628 samples	676 samples
Quantization levels	3	3
Scene size levels	2.59x2.59 km^2	7.65x7.65 km^2

Table 4.3: Range Resolution Simulation Properties

the standard case spans roughly 8 m, corresponding to theoretical minimum range resolutions of 16 and 5 m, respectively. Therefore, LTE waveforms with higher bandwidths will form images with finer range resolution. Therefore, our simulation tool validates that finer resolution is attained as a result of using waveforms with higher bandwidths.

4.4 Results from Distributed Scenes

Now that we have examined PSRs formed with data from the reverse backprojection algorithm, we can understand and interpret images formed from simulated distributed scenes - scenes where all clutter patches have unique RCS constants like those to be observed in real-time SAR data collection. We will examine sets of simulations performed on two different distributed scenes: one swath with an “OU” logo, a zoomed in piece of which is features in Figure 4.27, and another with a clutter pattern of a flower seen in Figure 4.32. Each of these clutter patterns was formed on a scattering grid of 2001x2001 pixels; the pixels corresponding to the “OU” logo have an RCS value of 500 and the pixels corresponding to the flower have values mapped to an 8-point gray scale. All pixels with non-zero RCS values in both clutter patterns have a random phase component. Data formed from

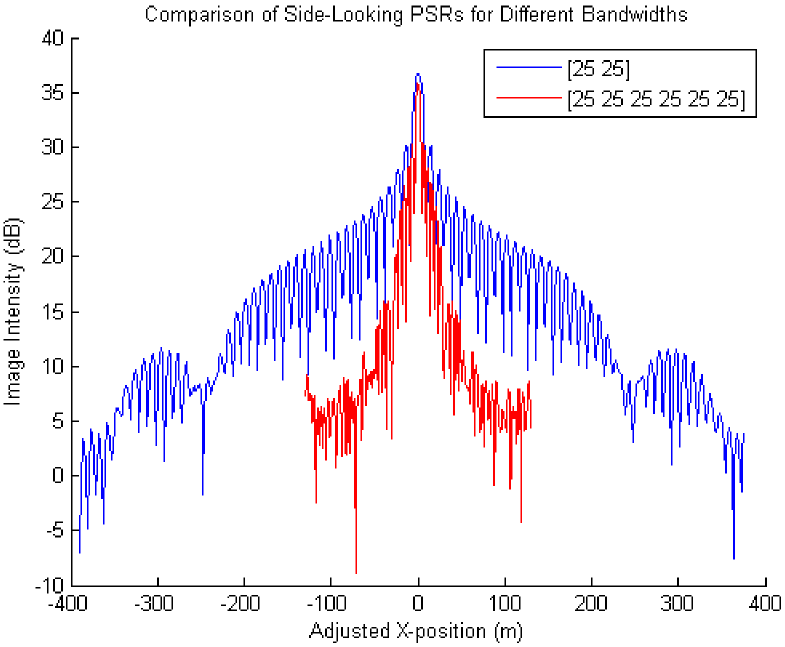


Figure 4.23: Comparison of side-looking Doppler cuts when NDLRB = [25 25] and [25 25 25 25 25 25]

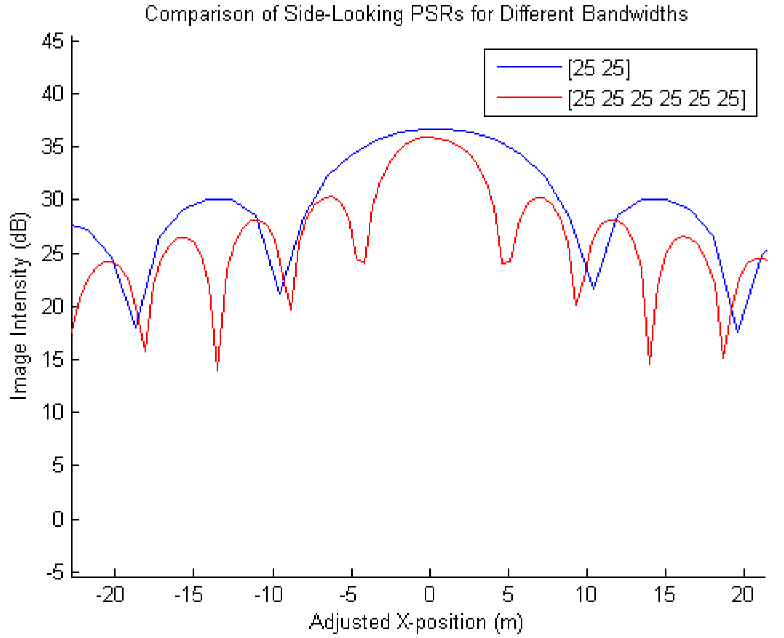


Figure 4.24: Zoomed in comparison of side-looking Doppler cuts when NDLRB = [25 25] and [25 25 25 25 25 25]

both patterns is projected onto 501x501 imaging grids with spacing corresponding to that simulation's range resolution. For both sets, we will be primarily interested in examining the outputs from the backprojection imaging algorithm, since imaging performance is not easily interpreted from visualizing the sampled radar data created with methods described in Chapter 3.

Because the algorithm presented in Chapter 3 utilizes shifted, scaled copies of the waveform $\rho_{xx}(q, \tau_m; n)$, our ability to construct a high-quality image will depend on self-ambiguities in the captured pulses. Even though the simulations employed capture a pulse, sized to the range swath, from different LTE waveforms, because the radio frame generated has the same resource block structure and the pulses are extracted from the same placement within the waveforms for each platform position, the pulses extracted for each platform position will exhibit similar autocorrelation behavior. Features (sidelobes, etc) are therefore present in images formed from shifted, scaled $\rho_{xx}(q, \tau_m; n)$ featuring self-ambiguities.

In order to visualize the effects that the waveform bandwidth and autocorrelation self-ambiguities have on image formation capabilities, we performed four simulations: one set of two involving waveforms with [25 25] structure, and another with waveforms with [25 25 25 25 25 25] structure. Within both sets, we performed simulations utilizing a pulse beginning halfway through the waveform and another at a random starting place (chosen to be .422) through the waveforms. For each of these sets we will reference Figures 4.12, 4.13, 4.25, and 4.26, which feature the generated pulse-compressed data from simulations with one pulse for all four of the aforementioned sets. The comparison within that set is intended to show the effect of cross ambiguities, while the second set of simulations are intended to show how increasing the waveform bandwidth yields finer image resolution.

The first set of images we will examine are the returns from simulations imag-

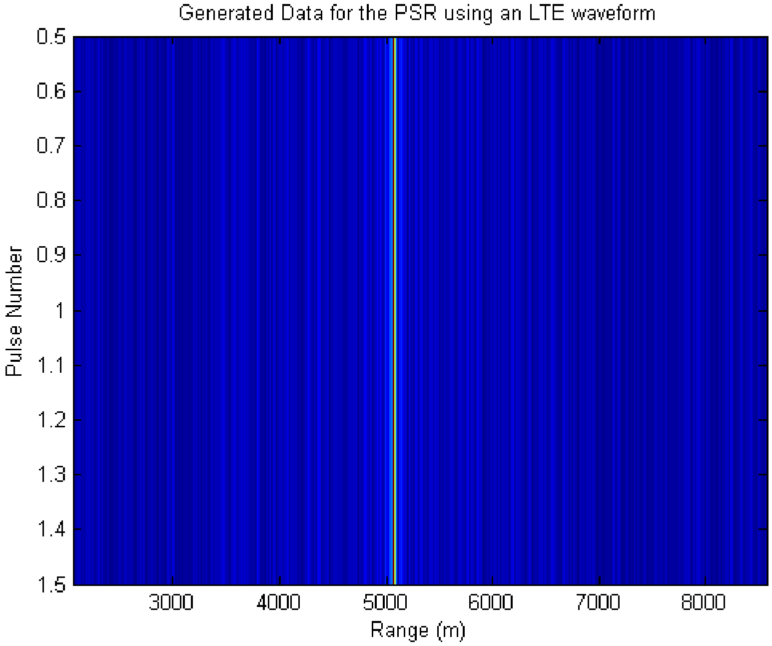


Figure 4.25: Generated pulse-compressed data for the PSR when NDLRB = [25 25 25 25 25 25] and .422

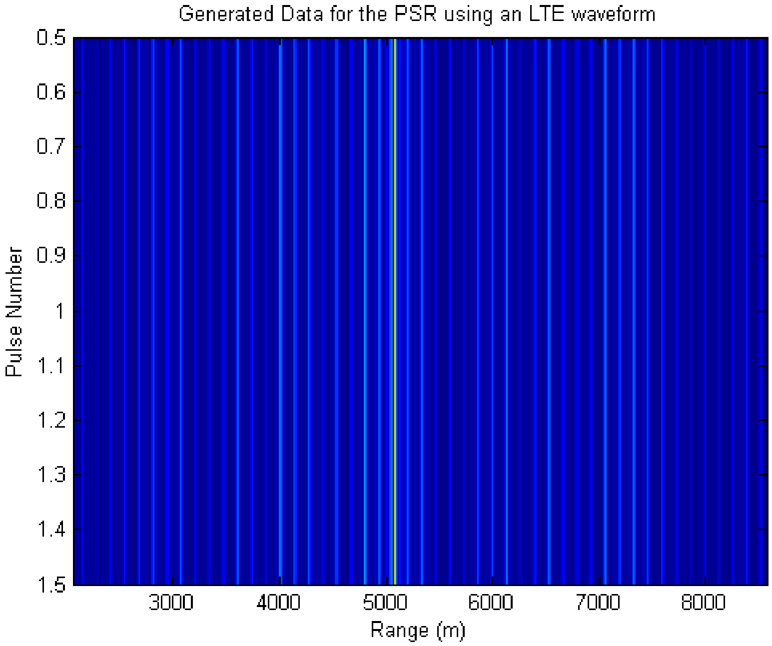


Figure 4.26: Generated pulse-compressed data for the PSR when NDLRB = [25 25 25 25 25 25] and .500

ing the "OU" pattern in Figure 4.27 using 1,024 pulses and 3 quantization levels. Figures 4.28 and 4.30 show images taken using pulses with low self-ambiguities. Because data is captured over 1,024 pulses and has low sidelobes, the images appear to be well-focused in both range and Doppler and the "OU" pattern is clearly visible. Because the simulations with [25 25] and [25 25 25 25 25 25] where the pulse is taken .500 through the radio frame result in received data with higher sidelobes, the images generated from that data, featured in Figures 4.25 and 4.29 is not nearly as focused in Doppler. Because more of the energy in the pulse-compressed data output is located outside of the peak corresponding to the target delay, that energy is interpolated by the backprojection algorithm as belonging at a different range entirely - as a result, the images look much more unfocused in Doppler. Because the data featured in Figure 4.26 contains low sidelobes at many ranges outside the peak, the image in Figure 4.31 does not contain any prominent ghost targets, but rather many faint ghost targets. In comparison to the data in Figure 4.26, the data in Figure 4.13 contains prominent sidelobes corresponding to bistatic ranges approximately 5 and 5.5 km away from the target's true bistatic range. These prominent sidelobes cause prominent ghost targets in the image in Figure 4.29; anyone interpreting the image will be able to see the two prominent "OU" pattern repeating itself at incorrect ranges. Therefore, the chosen segment of waveform matters - choosing pulses from placements yielding low sidelobes from self-ambiguities will result in clearer, more focused images.

In order to visualize imaging capabilities for a more complex distributed scene, we performed simulations using the clutter matrix featured in Figure 4.32, which will be referred to as the "flower" image. The 8-point gray-scale values of the image were used to represent the real part of it's RCS values, σ, with each image pixel containing four scattering patches, $p(x, y)$ with RCS values σ and random phases,

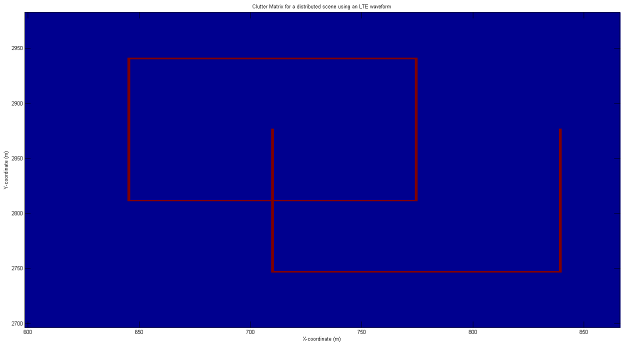


Figure 4.27: Clutter matrix of the “OU” distributed scene

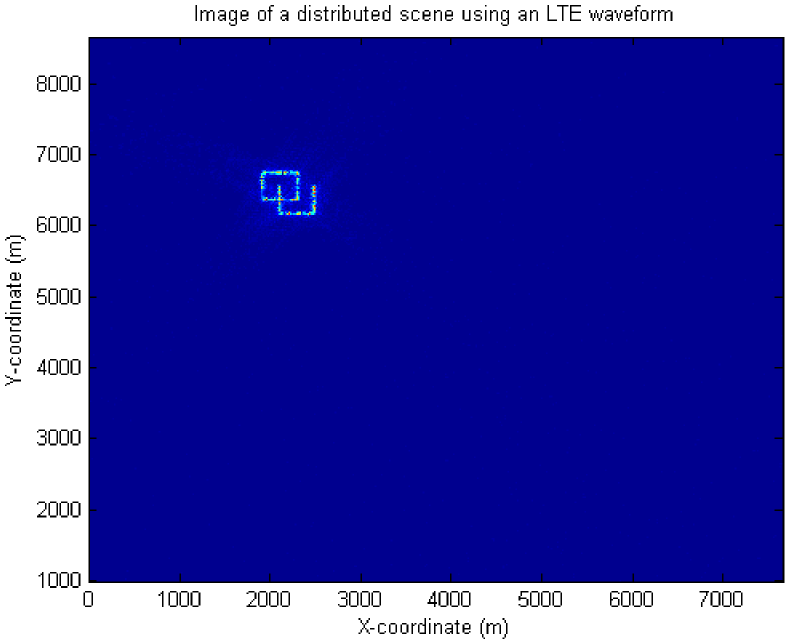


Figure 4.28: Image for the “OU” distributed scene when NDLRB = [25 25] and .422

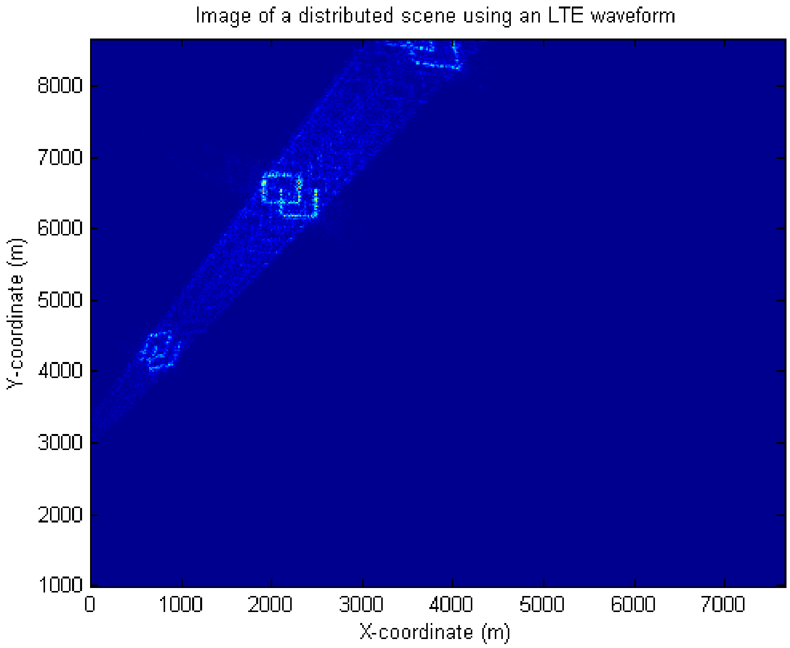


Figure 4.29: Image for the “OU” distributed scene when NDLRB = [25 25] and .500

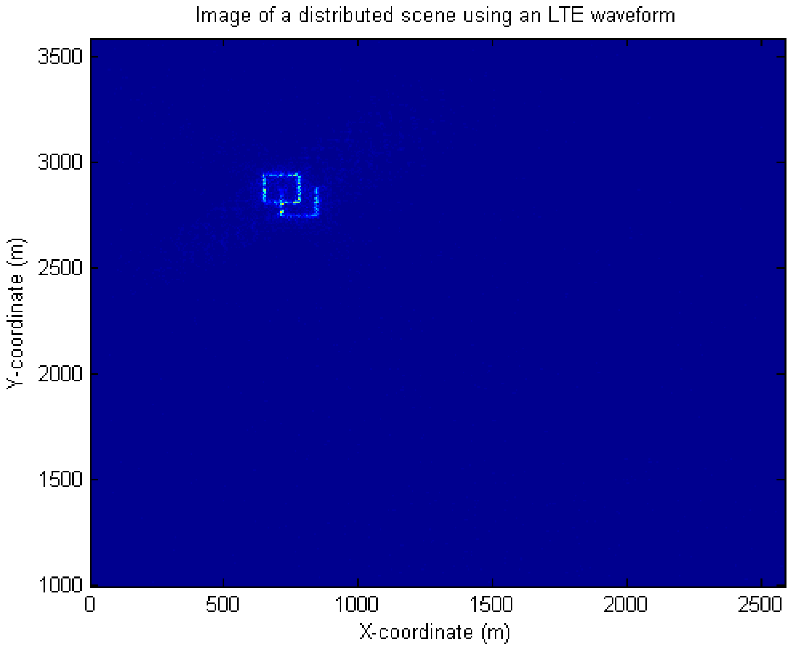


Figure 4.30: Image for the “OU” distributed scene when NDLRB = [25 25 25 25 25 25] and .422

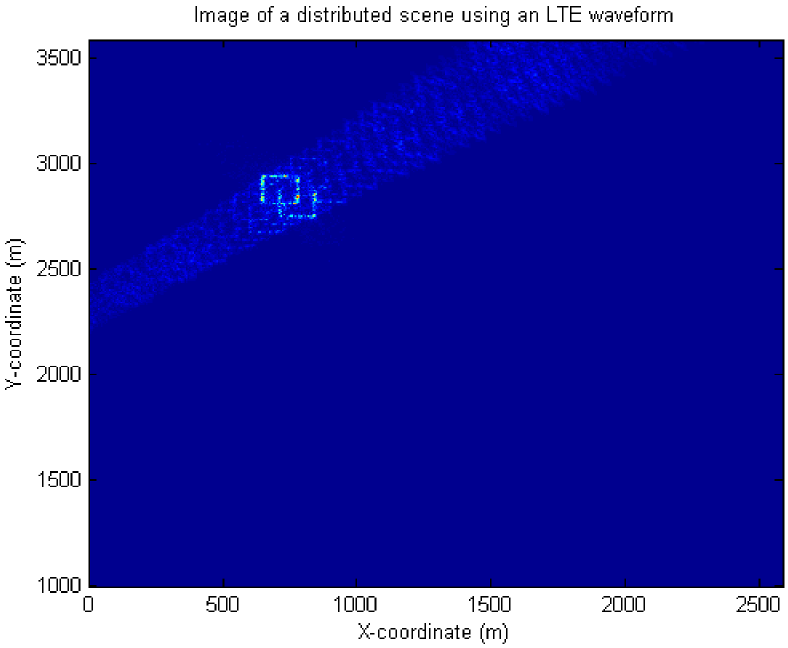


Figure 4.31: Image for the “OU” distributed scene when NDLRB = [25 25 25 25 25 25] and .500

$e^{(-j*\phi_{rand})}$, where ϕ_{rand} is a uniformly-distributed random number in the interval $[0, 2\pi]$. This particular image was chosen because it is a high contrast photo - the flower in the center is bright against a relatively dark foreground. The flower petals feature shading, while its pistil contains many fine details. The background is also a useful photo for determining image formation performance because it features straight lines and other more circular-shaped objects. Like in the set with the “OU” pattern, simulations performed with pulses taken from segments of the radio frame with low self-ambiguities yield superior images than those with pulses taken from segments with higher self ambiguities. Because each image grid pixel contains four scattering patches, the complex phases of the scattering patches combine and the images all appear *speckled*. Upon examining Figure 4.33 and Figure 4.35, we can detect many characteristics of the original image. In both images, the petals and other auxiliary shapes are distinguishable from the background, petal shading is

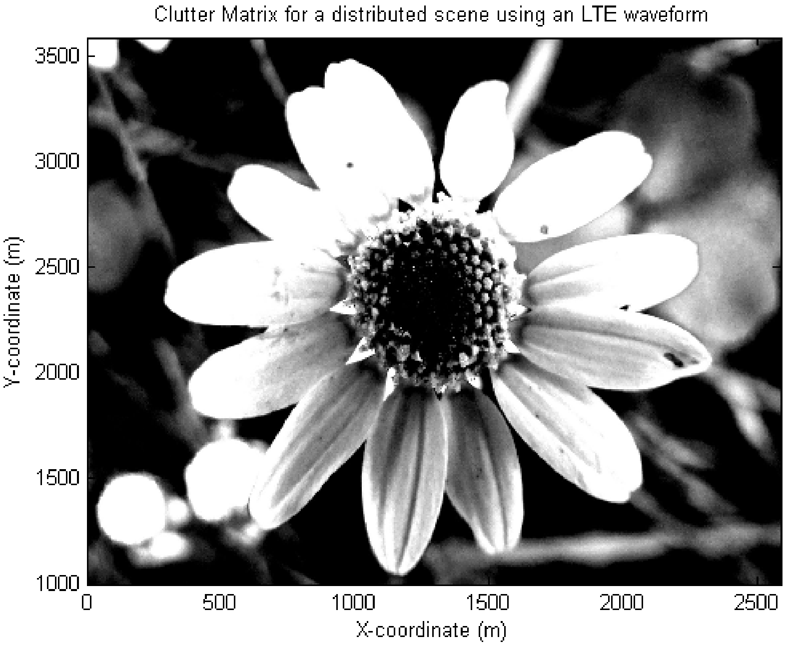


Figure 4.32: Clutter matrix of the ’flower’ distributed scene

visible, and some artifacts of the pistil are well-formed. When we contrast the two images, the image formed using the higher bandwidth waveform (with [25 25 25 25 25 25 25 25] NDRLB) is considerably brighter against the dark background, with smaller details more well-defined because of the finer range resolution. Likewise, simulated pulse-compressed data formed with pulses containing higher self-ambiguities construct less well-formed images. The flower foreground in Figures 4.34 and 4.37 is still distinguishable from the background, but the image quality has been degraded by misplacing energy from the self-ambiguity sidelobes in the pulse-compressed data. As is the case with the images formed from $\nu = .500$, smaller details within the image are more distinguishable in the image formed with the higher bandwidth waveform. Therefore, radio frames with higher bandwidth and pulses extracted from regions with lower self-ambiguity sidelobes produce better quality images.

In order to further examine the role of pulse extraction timing within the radio

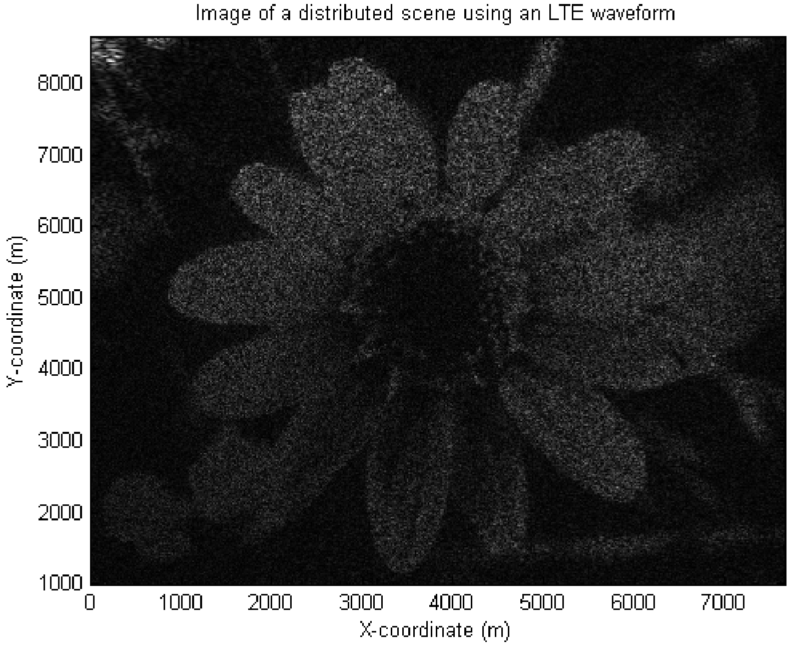


Figure 4.33: Image for the 'flower' distributed scene when NDLRB = [25 25] and .422

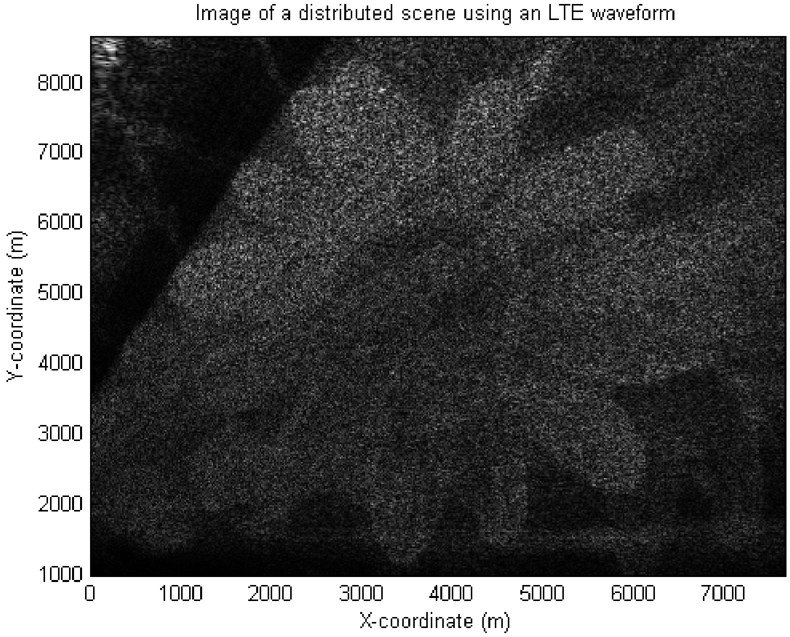


Figure 4.34: Image for the 'flower' distributed scene when NDLRB = [25 25] and .500

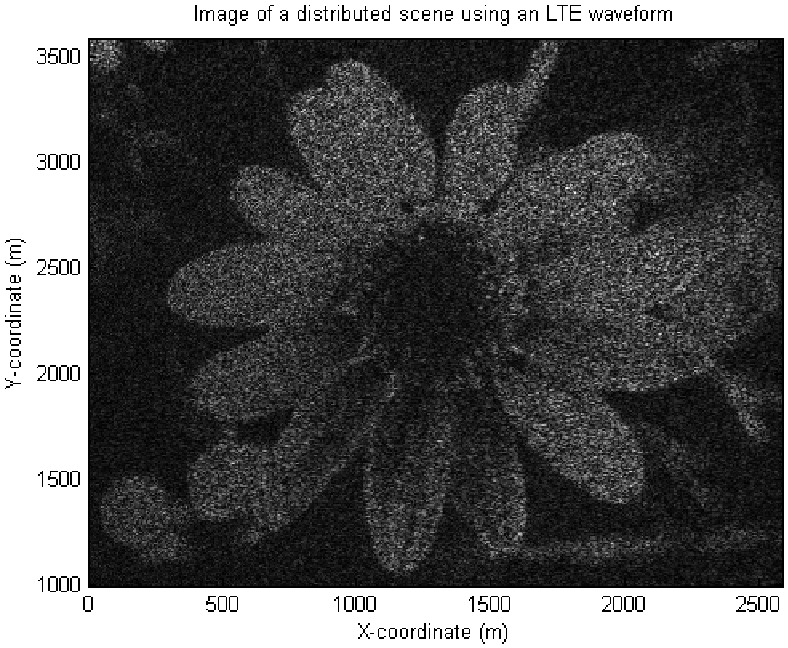


Figure 4.35: Image for the 'flower' distributed scene when NDLRB = [25 25 25 25 25 25] and .422

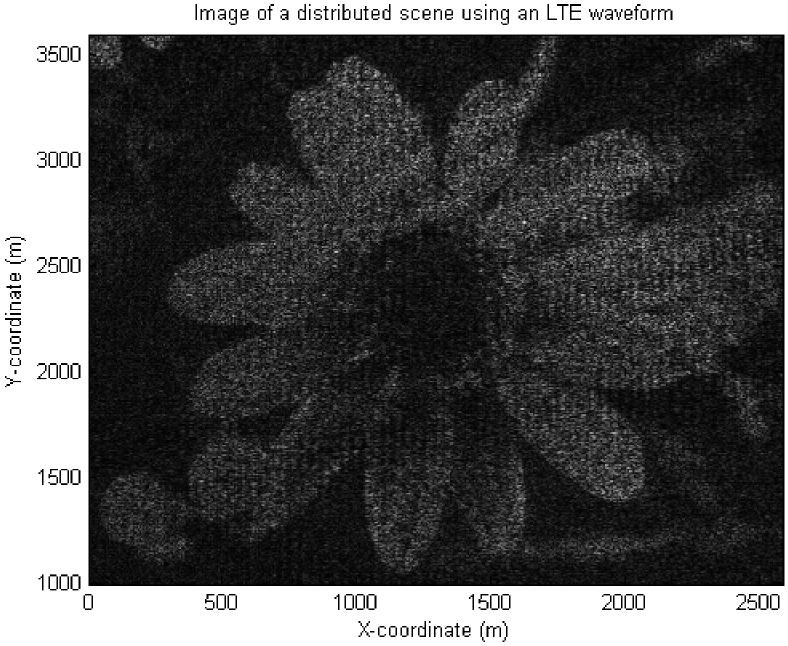


Figure 4.36: Image for the 'flower' distributed scene when NDLRB = [25 25 25 25 25 25] and .422, upsampled for range resolution comparison

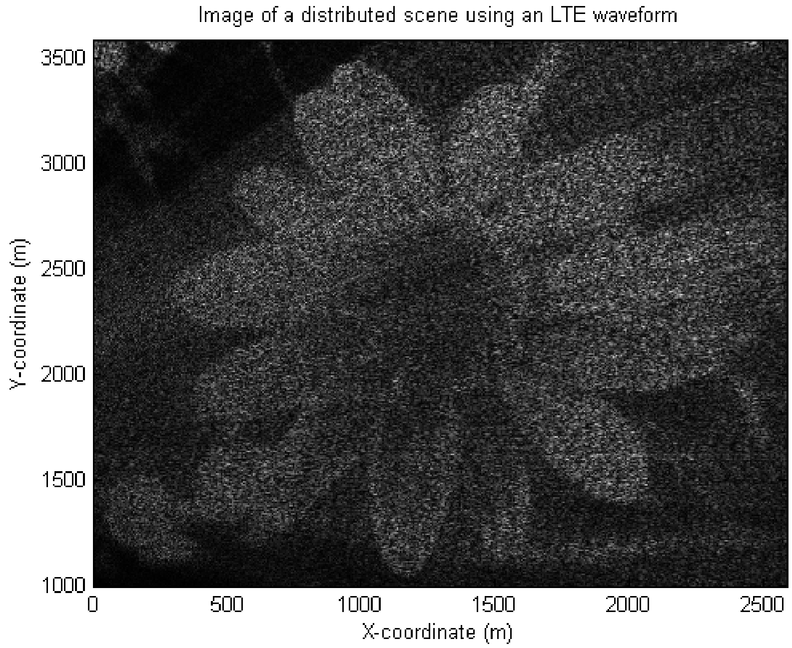


Figure 4.37: Image for the 'flower' distributed scene when NDLRB = [25 25 25 25 25 25] and .500

frame, we performed simulations extracting pulses starting from random indices within the radio frame. Because some pulses will inevitably be extracted from nulls or other regions with low energy, we disregard data generated from low-energy regions - since that data will have a significantly negative effect on image quality. The data would then be fed into the backprojection imaging algorithm to form an image. The energy-level threshold, which we will call η, which determines whether or not a pulse is fit to contribute to the data simulation is up to the engineer's discretion. Upon examining 1,000 sets of energy levels of 100 pulses extracted from random locations within radio frames, 65-70% on average were determined to be within regions with sufficiently high energy level. Figures 4.38 and 4.39 are images resulting from simulations where a [25 25 25 25 25 25] NDLRB is used with random pulse extraction with η selected to be .02 and .05, respectively. The general shape of the flower foreground is distinguishable from the background, but the image features

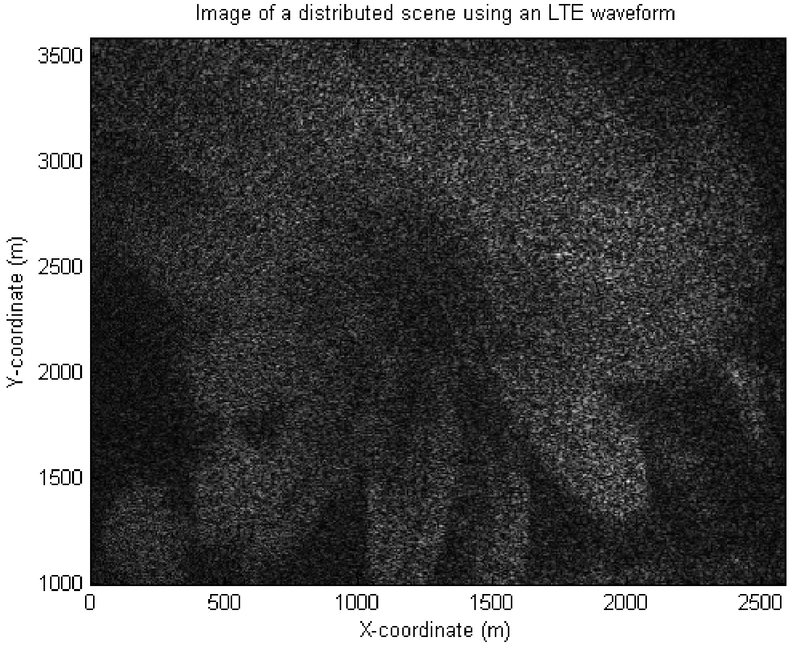


Figure 4.38: Image for the 'flower' distributed scene when NDLRB = [25 25 25 25 25 25] and random pulse placement, with η = .02

are not well formed. When the same approach is applied to the "OU" pattern, the results are less favorable, the pattern is not recognizable against its background in Figure 4.40. This is because a considerable amount of the pulses are discarded for not meeting the necessary energy threshold - the phase history from pulse-to-pulse is too fragmented in the generated data to form a high fidelity image.

4.5 Quantization Effects

We also performed simulations to investigate the effects of the number of quantization levels on image formation. In order to more easily visualize comparisons between various numbers of quantization levels, we performed a series of side-looking simulations - placing the scatterer in the middle of the receiver's trajectory and the transmitter in the same along-track location as the scatterer - so that we

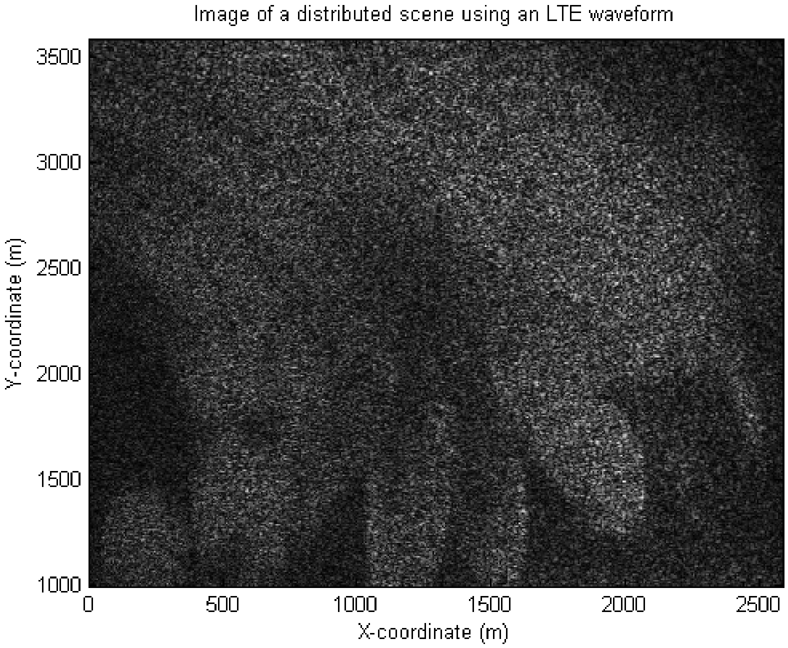


Figure 4.39: Image for the ’flower’ distributed scene when NDLRB = [25 25 25 25 25 25] and random pulse placement, with η = .05

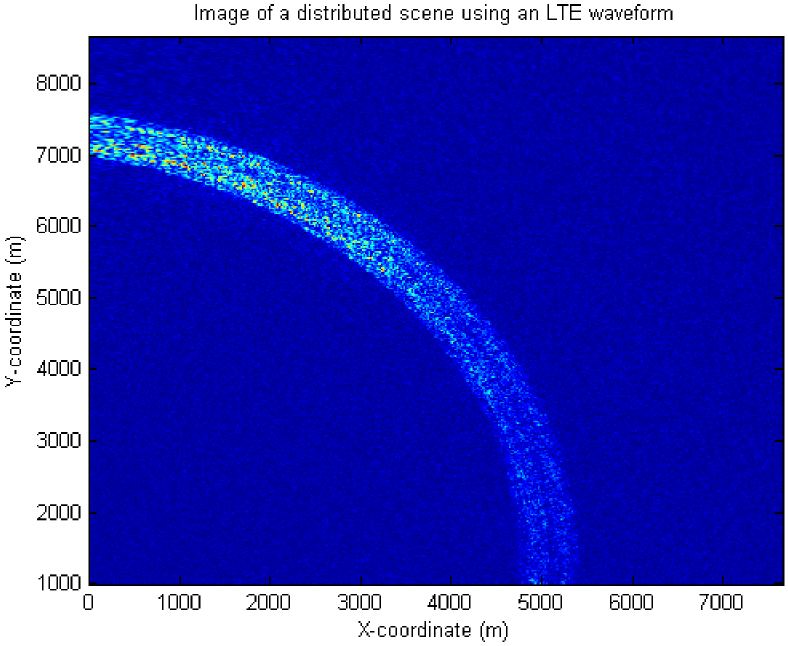


Figure 4.40: Image for the ’OU’ distributed scene when NDLRB = [25 25] and random pulse placement, with η = .35

PRI	1 ms
Platform velocity	200 $\frac{m}{s}$
LTE NDLRB	[25 25 25 25 25 25]
Bandwidth	29 MHz
Carrier Frequency	728 MHz
Platform positions	256
Pulse length	628 samples
Quantization levels	various
Scene size levels	2.59x2.59 km^2

Table 4.4: Quantization Level Simulation Properties

could view the Doppler sidelobes in the resulting image since the differences are not evident to the human eye in the final images. A full list of parameters is given in Table 4.4. We performed simulations with no quantization, 3 levels, 5 levels, 35 levels, and 105 levels. The results are presented in Figure 4.41. From Figure 4.41, we can see that compared to not employing any quantization, utilizing the quantized approach lowers Doppler sidelobes and results in a narrower main beam. Without quantization, the mainlobe to sidelobe levels vary by approximately -15.6 dB, with three levels they differ by approximately -16.4 dB, and five or more levels results in a difference of -16.9 dB; using five or more quantization levels yields results that are nearly indistinguishable from each other. From Figure 4.41 we can conclude that utilizing more than five quantization levels has a negligible improvement on image formation; for the sake of computational efficiency, we have used three levels for most of our simulations. This concludes our discussion of simulation results employing the reverse backprojection method.

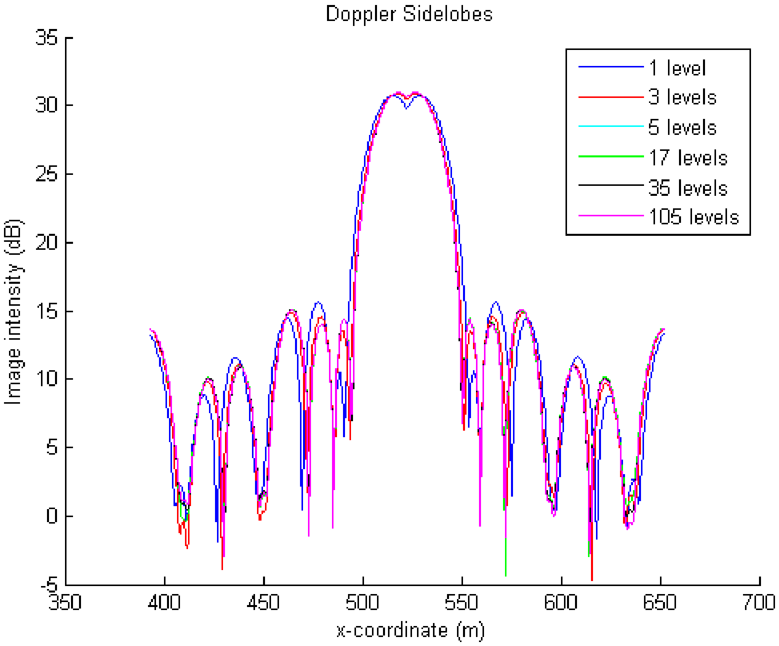


Figure 4.41: Doppler sidelobes for various Q

Chapter 5

Conclusion

In this thesis we introduced the concepts behind the reverse backprojection algorithm for passive bistatic SAR. An introduction to passive bistatic radar and a discussion of waveform selection was presented in Section 1.2. Pertinent background information regarding SAR and the geometry of passive bistatic SAR systems - particularly the two-part bistatic range - was described in Sections 2.1 and 2.2, respectively. In Section 3.1 we explained the need for more efficient simulation methods, and the increased computational burden inherent in passive bistatic SAR simulations originating from the variability of waveforms throughout the aperture length; because it would be computationally impractical to calculate the delay, in a continuous set of numbers, of each echo from the range swath, we define the scattering grad and divide the range swath into discretized "patches". We discussed our intentions to exploit favorable characteristics of the backprojection imaging algorithm *in reverse* in order to model passive bistatic SAR data. In the following section (Section 3.2), we reviewed SAR imaging algorithms used in practice, giving the advantages and disadvantages of using backprojection. We examined the origins of backprojection, from tomographic imaging formation methods based on the projection slice theorem, from their first implementations in computer-aided tomography (CAT) medical imaging in the 1970s. We traced the algorithm's evolution over the decades and described approaches taken to improve its efficiency, since backprojection is still considered to be computationally expensive despite its flexibility and wide-scale usage in SAR imaging. Lastly, we outlined the implementation of the backprojection method in software and roughly calculated the algorithm's computational burden, since we employed "forward" backprojection to validate the data generated in our simulation.

In Section 3.3 we explain the role of the autocorrelation function in matched filtering. We discuss how we can model the received echoes on a finite set of delays that in agreement with the number of samples necessary to receive the echoes from the entire range swath. We define the shifted autocorrelation matrix for each pulse as the autocorrelation function repeated for an arbitrarily-selected number of quantization levels that has been shifted by sub-sample amounts that can more accurately represent a scatter's delay if the scatterer is not located in the center of the patch it lies within. We explain our methods for identifying which patches correspond to each quantized delay (contour), and which patches within the contour correspond to each row of the shifted autocorrelation matrix. We then describe how we extract the RCS constants for each of those patches, and scale the chosen copies of the autocorrelation matrix to adequately represent the contributions of each patch in the resulting generated raw data. To conclude our discussion in Section 3.3 we characterize the total computational burden of the algorithm, which we found to primarily depend on the number of simulated scattering patches.

In Chapter 4 we analyze the results from our simulations. In Section 4.1, we present comparisons between simulations performed with LFM and LTE pulses to show how self-ambiguities in the LTE waveform are evident in the image's point spread response; these self-ambiguities confirm that LTE waveforms are not as promising for radar usage as conventional chirp waveforms. In Section 4.2 we examine how waveforms with higher bandwidths yield finer range resolution. In Section 4.3, we present results from simulations performed over distributed scenes and discuss the effects of self-ambiguities in the image. Because of the presence of deterministic elements in the LTE waveforms, image quality is dependent on how correlated the waveform is with itself; waveforms that are less correlated over time produce clearer images. Because each waveform also has a different amount of energy, it is also necessary to normalize the data and only consider returns from pulses with a sufficient energy level. When pulses are selected from random indices within an LTE waveform throughout a CPI, image quality is also degraded and further processing methods will be necessary to produce clearer images. In Section 4.4 we compared the image intensity when various numbers of quantization levels were used and found that our simulation yields diminishing improvements when more than five quantization levels are employed. We conclude that the reverse backprojection method can be used to simulate raw data collection for passive bistatic SAR, though self-ambiguities present challenges that may require additional processing methods when employing LTE waveforms as illuminators of opportunity.

Over the course of this work, we developed a simulation method that utilizes the backprojection algorithm in reverse to model received raw data from a passive bistatic SAR system using LTE as an illuminator of opportunity. The novel simulation method presented is shown to be computationally faster than existing methods, including the matched filter algorithm. Additionally, the simulation method accurately accounts for artifacts in waveforms resulting from deterministic features and nuances in waveform structure from non-cooperative sources. Future work will explore further processing methods to create high fidelity images to combat complications resulting from self-ambiguities in digital waveforms.

References

[1] (). Remote sensing, [Online]. Available: `https://en.wikipedia.org/wiki/Remote_sensing`.

[2] T. Brumble, *Passive bistatic radar utilizing LTE as illuminators of opportunity*, University of Oklahoma Honors Thesis, 2014.

[3] N. Willis, *Bistatic Radar*. Scitech, 2004.

[4] M. Cherniakov, *Bistatic Radars: Emerging Technologies*. Scitech, 2004.

[5] I. W. et. al, "Bistatic SAR processing and experiments", *IEEE Trans. Geosei. Remote Sens.*, vol. 44, no. 10, pp. 2710–2717, 2006.

[6] S. R. Stevens and J. A. Jackson, "Emitter selection criteria for passive multistatic synthetic aperture radar imaging", *IET Radar, Sonar and Navigation*, vol. 8, no. 9, pp. 1267–1279, 2014.

[7] H. Griffiths, "Where has all the spectrum gone?", in *IEEE Intl. Conf. on Radar*, IEEE, Adelaide, Australia, 2013, pp. 1–5.

[8] A. Evers and J. A. Jackson, "Experimental passive SAR imaging exploiting LTE, DVB, and DAB signals", in *IEEE Radar Conference*, IEEE, Cincinnati, OH, 2014, pp. 680–685.

[9] J. E. P. et al., "DVB-T passive radar signal processing", *IEEE Trans. Signal Process.*, vol. 61, no. 8, pp. 2116–2126, 2012.

[10] A. Evers and J. A. Jackson, "Analysis of an LTE waveform for radar applications", in *IEEE Radar Conference*, IEEE, Cincinnati, OH, 2014, pp. 200–205.

[11] J. R. G. del Arroyo and J. A. Jackson, "Collecting and processing WIMAX ground returns for SAR imaging", in *IEEE Radar Conference*, IEEE, Ottawa, Canada, 2013.

[12] D. W. O. et al., "A multi-frequency hybrid passive radar concept for medium range air surveillance", *IEEE Aerospace and Electronic Systems Magazine*, vol. 27, no. 10, pp. 6–15, 2012.

[13] H. Griffiths and C. Baker, "Measurement and analysis of ambiguity functions of passive radar transmissions", in *IEEE International Radar Conference*, IEEE, 2005.

[14] ——, "Passive coherent location radar systems. part 1: Performance prediction", *IEE Proc.-Radar Sonar Navig.*, vol. 153, no. 3, pp. 153–159, 2005.

[15] A. Evers and J. A. Jackson, "Cross-ambiguity characterization of communication waveform features for passive radar", *IEEE Trans. Aerosp. Electron. Syst.*, vol. 51, no. 4, pp. 3440–3455, 2015.

[16] *Digital video broadcasting (DVB); framing structure, channel coding and modulation for digital terrestrial television (DVB-T)*, European Telecommunications Standards Institute, 1997.

[17] H. H. et al., "Understanding the signal structure in DVB-T signals for passive radar detection", in *IEEE Radar Conference*, IEEE, Arlington, VA, 2010, pp. 532–537.

[18] *ATSC digital television standard - part 2: RF/transmission system characteristics*, Advanced Television Systems Committee, 2011.

[19] W. C. Barott and J. Engle, "Single-antenna ATSC passive radar observations with remodulation and keystone formatting", in *IEEE Radar Conference*, IEEE, Cincinnati, OH, 2014, pp. 159–163.

[20] W. C. Barott and B. Himed, "Simulation model for wide-area multi-service passive radar coverage predictions", in *IEEE Radar Conference*, IEEE, 2013.

[21] F. Stremler, *Introduction to Communication Systems*. Scitech, 2004.

[22] J. G. N. Henderson, "ATSC DTV receiver implementation", *Proc. IEEE*, vol. 94, no. 1, pp. 119–147, 2006.

[23] *LTE: Evolved universal terrestrial radio access (E-UTRA), physical channels and modulation*, European Telecommunications Standards Institute, 2011. [Online]. Available: `http://www.etsi.org/deliver/etsi_ts/`

`136200_136299/136211/09.01.00_60/ts_136211v090100p.pdf`.

[24] *LTE resource guide*, Anritsu, 2009. [Online]. Available: `http://www.anritsu.com/en-GB/Media-Room/Newsletters/files/anritsu%5C_te%5C_guide.pdf`.

[25] S. Haykin and M. Moher, *Communication Systems*. John Wiley & Sons, 2009.

[26] *LTE system toolbox*, Mathworks, 2017. [Online]. Available: `https://www.mathworks.com/products/lte-system.html.`.

[27] A. G. et al., *Fundamentals of LTE*. Prentice Hall, 2011.

[28] E. D. et. al, *3G Evolution: HSPA and LTE for Mobile Broadband*. Elsevier Ltd, 2008.

[29] [Online]. Available: `http://www.fcc.gov/maps`.

[30] L. Gorham and L. Moore, *Sar image formation toolbox for MATLAB*, Air Force Research Laboratory, Sensors Directorate.

[31] M. Richards, *Fundamentals of Radar Signal Processing*. McGraw Hill, 2014.

[32] M. Jackson, “The geometry of bistatic radar systems”, *IEE Proceedings on Communicatons, Radar, and Signal Processing*, vol. 133, no. 7, pp. 604–612, 1986.

[33] T. T. et al., “Ambiguity function for a bistatic radar”, *IEEE. Trans. on Aerospace and Electronic Systems*, vol. 33, no. 3, pp. 1041–1051, 1997.

[34] M. D. Desai and W. K. Jenkins, “Convolution backprojection image reconstruction for spotlight mode synthetic aperture radar”, *IEEE Trans. Image Process.*, vol. 1, no. 4, pp. 505–517, 1992.

[35] A. F. Yegulalp, “Fast backprojection algorithm for synthetic aperture radar”, in *IEEE Radar Conference*, IEEE, Waltham, MA, 1999, pp. 60–65.

[36] H. J. Scudder, “Introduction to computer aided tomography”, *Proc. IEEE*, vol. 66, no. 6, pp. 628–637, 1978.

[37] D. C. M. et al., “A tomographic formulation of spotlight-mode synthetic aperture radar”, *Proc. IEEE*, vol. 71, no. 8, pp. 917–925, 1983.

[38] J. L. Bauck and W. K. Jenkins, “Tomographic processing of spotlight-mode synthetic aperture radar signals with compensation for wavefront curvature”, in *International Conference on Acoustics, Speech, and Signal Processing*, IEEE, New York, NY, 1988, pp. 1192–1195.

[39] W. G. C. et. al, *Spotlight Synthetic Aperture Radar Signal Processing Algorithms*. Artech House, 1995.

[40] B. D. Rigling and R. L. Moses, “Polar format algorithm for bistatic SAR”, *IEEE Trans. Aerosp. Electron. Syst.*, vol. 40, no. 4, pp. 1147–1159, 2004.

[41] S. R. Stevens and J. A. Jackson, “Emitter subset selection for passive multistatic synthetic aperture radar”, in *IEEE Radar Conference*, IEEE, Cincinnati, OH, 2014, pp. 686–691.

[42] I. G. Cumming and F. H. Wong, *Digital Processing of Synthetic Aperture Radar Data: Algorithms and Implementation*. Artech House, 2005.