Terahertz Relative Positioning: An Alternative to GPS for Aircraft Flying in Formation

A Dissertation

Submitted by

John Scott Parker

In partial fulfillment of the requirements for the degree of

Doctor of Philosophy

In

Mechanical Engineering

Tufts University

August 2017

Advisor
Prof. Jason Rife (Tufts University Department of Mechanical Engineering)

Committee
Prof. Pratap Misra (Tufts University Department of Mechanical Engineering)
Prof. Usman Khan (Tufts University Department of Electrical Engineering)
Dr. Kevin Kremeyer (Physics, Materials, and Applied Mathematics Research)

ProQuest Number: 10232468

ProQuest 10232468

Abstract

A new terahertz-frequency (THz) relative positioning capability is developed and demonstrated in simulation for the formation flight of cargo aircraft. Formation flight is used by the military for a wide variety of applications. This work specifically considers the precision airdrop of personnel and supplies over hostile territory. During these operations, aircraft must maintain precise relative positioning; however, the current technology is flawed because radio signals are easy to detect and GPS is easily jammed. We propose using THz signals, the unique propagation properties of which provide both stealth and robustness to jamming. THz is a nascent technology with growing interest, particularly for high-speed wireless communications. To the author's knowledge, this is the first exploration of its potential for positioning. This dissertation develops methods for the measurement of range and bearing angle from the THz signal and a communication scheme to transmit altitude measurements over the THz link. These measurements are then fused together in a Kalman filter to estimate the aircrafts' relative positions. Results from an integrated simulation demonstrate that the THz system is capable of precisely measuring the position, with cross-track errors of 11 m, two-sigma, at separations of 2 km, well within the 50 m requirements for the precision airdrop application.

Acknowledgements

This dissertation would not have been possible without the assistance and support of a number of people.

First and foremost, I would like to thank my advisor Professor Jason Rife. You have served as an amazing mentor and role model to me over the last six years and I feel incredibly lucky to have had you as my advisor. You are always willing answer my questions, whether about research, school, or life. You are an amazing teacher, both in and out of the classroom. Also, you are an extremely gifted scientific writer, and I really appreciate the time that you have spent helping me with my sometimes less than stellar writing. I have learned a great deal from you, and can't thank you enough for all that you have done.

I would also like to extend special thanks to Professor Pratap Misra, with whom I have been fortunate enough to work both in research and teaching. I can't tell you how much I have appreciated getting to pop into your office on a weekly basis to ask random questions about GPS, signal theory, or teaching. You are always welcoming, offering me a snack, inviting me to come sit down, and taking the time to not only answer my question, but also to chat. Thank you so much for being a mentor. Also, thank you for giving me the opportunity to have one of my photos published in a book!

Next, I want to give many thanks to Professor Usman Khan, for serving on both this committee and my Master's committee, and for providing feedback on my work during CRISP group presentations. The insightful questions you asked have pushed me to explore new avenues in the research and to consider things from a different perspective. In addition, I really appreciate the supportive words you have given me as I worked on this project.

I also want to thank Dr. Kevin Kremeyer for all of the help I have gotten over the years on this project. Our conversations have yielded new insights and much deeper understanding of the concepts and application. In addition, you have been very supportive of my efforts on this project as I prepare to graduate, which I greatly appreciate.

Furthermore, I would like to extend many thanks to everyone at PM&AM for the work that they have done on this project and the assistance that they have given me. Your experience with the hardware and the amazing technical work that you have done is central to this dissertation. In particular, I would like to thank Pascal Mickelson and Jeremy Yeak.

In addition to the professors mentioned above, I would like to thank the faculty of the Tufts University Mechanical Engineering Department for all that they have taught me and all that they have done to support my growth. Academically, I have grown by leaps and bounds here thanks to your persistence and hard work. Professionally, you have taught me what it means to be scientifically rigorous. Many professors have taken time to meet with me individually and support or teach me in some way. Thank you for being such a crucial part of my education.

I would like to thank two professors in particular, Professor Robert White and Professor Marc Hodes. In addition to being phenomenal teachers, both have served as mentors and I greatly appreciate all the time they have taken outside of the classroom to advise and support me.

I also want to thank the department administrators who have helped me immensely with a wide array of issues, including with my transcript, stipend, and office space.

My time at Tufts has been punctuated by some amazing relationships with my fellow students, and I can't thank them enough for all that they have done. These

friends played a crucial role in every aspect of the experience. Whether it was late night or weekend homework sessions, studying for exams, organizing social events, giving feedback on presentations, or venting about some academic frustration, they made it fun and interesting and enjoyable. I have learned so much from them, maybe more than I learned in class, and I would not have made it to this point without them. I want to specifically mention the members of the CRISP and ASAR groups, both current and past, as well as Lisa Lam and David Wong.

I also want to give special thanks to Gabrielle String, who began her degree at the same time as me, and has been a classmate, officemate, co-president, and close friend of mine throughout my time at Tufts. You have attended countless practice presentations, assisted with homework and studying, talked through research questions, debugged code, and shared in good times and struggles. I am so thankful for all that you have done to help with classes, research, and life.

I would not have survived this process were it not for my amazing wife Julie, and I want to thank her most of all. You have tolerated my boring research stories, late night work, and the shirking of my chores for the last six years, and still remained supportive and incredibly caring. I especially appreciate your willingness to read my boring papers and listen to my long presentations. You are a warmhearted and intelligent person and I am so thankful to be married to you. You made this work possible by completely supporting me at home and I cannot thank you enough. I promise I will start cleaning the bathroom again soon.

I also want to thank my parents for supporting me not only in graduate school, but throughout my life. You taught me to work hard and persist through struggles, and challenged me to always learn more and better understand the world. You instilled in me the skills to get to this point, and I can't thank you enough.

Table of Contents

List of Figures

List of Tables

Nomenclature

A number of conventions are used in this dissertation to make the mathematics clear. Vectors and matrices are displayed in bold face, whereas scalars are displayed in italics (for example $\mathbf{x}$ is a vector and x is a scalar). Hats are used to indicate estimates (for example $\hat{\mathbf{x}}$ is an estimate of $\mathbf{x}$). Dots are used to indicate derivatives with respect to time (for example $\dot{x}$ is the time derivative of x).

Below is an alphabetical list is of variables frequently used in this dissertation and the quantities they represent. The integers $i, j,$ and m are used as indices.

a	Diffracting slit width	$\mathbf{R}$	Measurement noise covariance matrix
A	Area	t	Time
b	Data bit	T	Period
B	Amplitude constant	v	Signal
c	Speed of light	V	Amplitude model
d	Diffracting slit spacing	$\mathbf{V}$	Amplitude model vector
D	Interference distance	w	Width of detector
F	Frequency	x	Along-track position
$\mathbf{F}$	State matrix	$\mathbf{x}$	State vector
G	Gain	y	Cross-track position
$\mathbf{G}$	Process matrix	$\mathbf{y}$	Measurement vector
h	Relative altitude	z	Vertical position
$\mathbf{h}$	Linearized measurement vector	$\mathbf{z}$	Phasor
H	Absolute altitude	α	Dimensionless single-slit coordinate
$\mathbf{H}$	Measurement matrix	β	Dimensionless double-slit coordinate
I	In-phase integrator	γ	Probability
k	Time epoch	Γ	Reflection coefficient
K	Number of time epochs	Δ	Diffraction grating lateral position
ℓ	Distance	ζ	Internal diffraction angle
M	Phase ambiguity integer	θ	Phase shift
n	Measurement noise	λ	Wavelength
N	Number of quantity	μ	Atmospheric attenuation
p	Power density	ξ	Solid angle rotation
P	Power	ρ	Correlation
q	Process noise	$\boldsymbol{\rho}$	Correlation matrix
$\mathbf{q}$	Process noise vector	σ	Standard deviation
Q	Quadrature correlator	τ	Time-of-flight delay
$\mathbf{Q}$	Process noise covariance matrix	ϕ	Bearing angle
r	Range	ψ	Angle of incidence
$\mathbf{r}$	Measurement noise	ω	Angular frequency

Generic functions are expressed using the notation $f(u)$, where u is the independent variable. A couple of specific functions are also used. Normal Gaussian distributed variables are represented as $\mathcal{N}(u, \sigma^2)$, where u is the mean and σ^2 is the variance. The cumulative distribution function of a Gaussian is represented as Φ_{u,σ^2}.

In addition, a number of important acronyms are used in the dissertation

A/D	Analog to Digital
AC	Aircraft
AMC	Amplifier Multiplier Chain
BPSK	Binary Phase Shift Keying
CDF	Cumulative Distribution Function
CDMA	Code Division Multiple Access
CRC	Cyclic Redundancy Check
DAQ	Data Acquisition System
DC	Direct Current
DLL	Delay Locked-Loop
EKF	Extended Kalman Filter
GBAS	Ground-Based Augmentation System
GHz	Gigahertz
GPS	Global Positioning System
LS	Least Squares
MHz	Megahertz
PM&AM	Physics, Materials, and Applied Mathematics LLC
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
QED	Quantum Electrodynamics
RF	Radio Frequency
RFI	Radio Frequency Interference
SKE	Station Keeping Equipment
THz	Terahertz
WLS	Weighted Least Squares

Chapter 1

Introduction and Motivation

1.1   Motivation: Formation Flight

This dissertation describes a novel system using Terahertz (THz) signals to measure relative positions between aircraft flying in formation. Formation flight is used by the military for a variety of different applications. This work focuses specifically on the precision airdrop application; however, the methods developed here may be extended into a number of other applications including drag reduction [1]–[3], aerial refueling [4], [5], and civilian formation flight [6]–[9].

The military regularly uses precision airdrops to quickly deliver supplies and personnel to remote locations, often inside hostile territory [10], [11]. One recent example is the delivery of humanitarian aid to Yazidi civilians in Northern Iraq. During the summer of 2014, tens of thousands of members of this religious minority became trapped in a mountainous region near the border with Syria by the advance of Islamic State (IS) militants. In addition to the threat of violence posed by the militants, it was feared that many could die of hunger and dehydration without access to food and water. In response to this crisis, a group of C-17 and C-130 cargo aircraft from the United States airdropped hundreds of

thousands of pounds of food and water to the civilians, averting the immediate crisis and giving many the chance to evacuate [12], [13].

When performing precision airdrops, military aircraft typically fly in formation to ensure both precise package delivery and mutual protection from adversaries [14]. These formations are made up of three aircraft groupings called elements, with arbitrarily long formations formed from a sequence of elements, as shown in Figure 1 [6], [15]. Each element is composed of a lead and two following aircraft. The following aircraft have defined positions within the formation relative to the lead that they must maintain. This is achieved using Station Keeping Equipment (SKE), which provides relative position estimates.
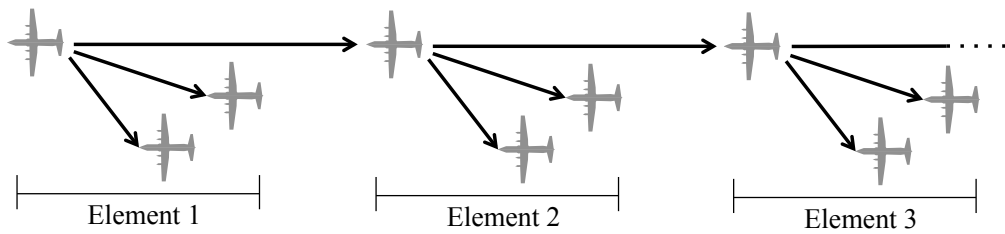
Figure 1 – Formation composed of multiple elements

Traditional SKE systems use radio frequency (RF) signals to measure the range and bearing angle between aircraft. These signals, however, can propagate long distances and are easily detected, potentially giving an adversary significant advance warning of a formation's approach [10], [14]. Alternatively, GPS can be used for relative positioning, but the vulnerability of GPS to jamming via radio-frequency interference (RFI) is a well-known weakness of such systems [16], [17]. As a result, there is demand for alternative relative positioning systems [18], [19], particularly ones that can operate in GPS-denied environments.

1.2   Proposed Solution: Terahertz (THz)

In order to address the limitations of conventional and GPS-based SKE technologies, this work proposes using signals from the underutilized THz-frequency band for aircraft relative positioning. The THz band lies between microwave and infrared radiation on the electromagnetic spectrum, roughly between 0.3 THz and 10 THz [20]–[22], as shown in Figure 2. This band is often referred to as the "THz gap," because of its historical difficulty to exploit [23], [24]. It marks the transition between optical and electronic frequencies, which each utilize very different hardware [25]. Adapting technology from these neighboring bands to work at THz frequencies has proven difficult. In addition, THz frequencies are absorbed by atmospheric gases, so signal attenuation can be significant near ground level, especially for frequencies at the higher end of the band [26]. As a result, the development of affordable THz hardware has been slow.
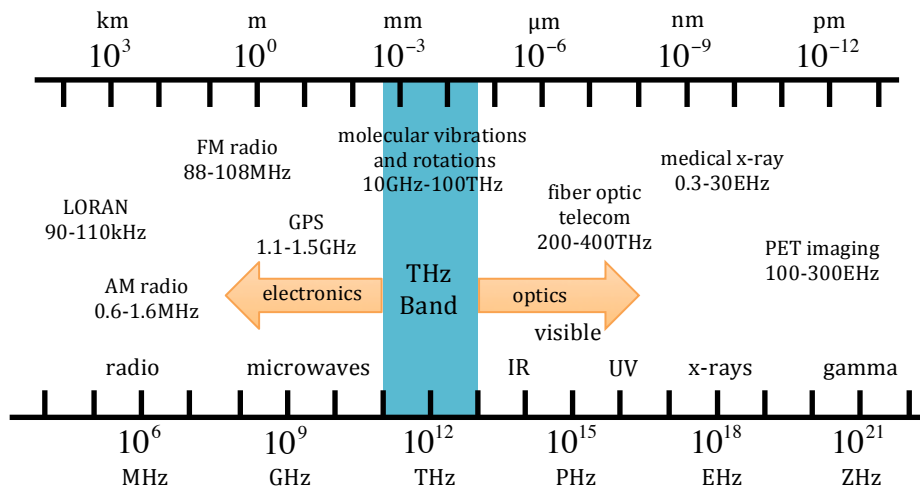
Figure 2 – THz band in the electromagnetic spectrum

Despite these limitations, interest in THz technology has grown significantly in recent years as the demands for high-speed wireless communications increase and unoccupied frequency bands are gobbled up. Technological advances have started

to close the "THz gap" over the last decade or so [24], [27]–[29], and THz transmitters and receivers have recently become commercially available [30]–[32]. With the increasing availability of hardware, researchers are now using THz signals for a wide variety of applications, including medical imaging [33]–[35], astronomy [36], [37], remote sensing of surfaces [38]–[40], synthetic aperture radar [41]–[46], airport and package security scanning [47], [48], and wireless communications [49]–[59].

The unique propagation properties of THz radiation offer a couple key advantages for formation flight applications: stealth and robustness to jamming. As mentioned above, THz signals are attenuated by atmospheric gases, severely limiting their detectability beyond a given propagation envelope. Importantly, this attenuation is highly altitude dependent. At aircraft cruising altitudes, the attenuation is low, and transmission distances of several kilometers can be readily achieved; whereas at low altitudes, the attenuation is orders of magnitude higher, dissipating the signal after it travels relatively short distances [26], [60], as shown in Figure 3.

Figure 3 – THz altitude-dependent attenuation

Because of this attenuation difference, a ground station cannot easily detect a formation's approach or jam its THz signals, as shown in Figure 4, providing both

stealth and robustness to jamming. These characteristics make THz signals a compelling alternative to current radio or GPS relative-positioning systems for military cargo aircraft. This dissertation envisions a THz relative positioning system for formation flight applications.
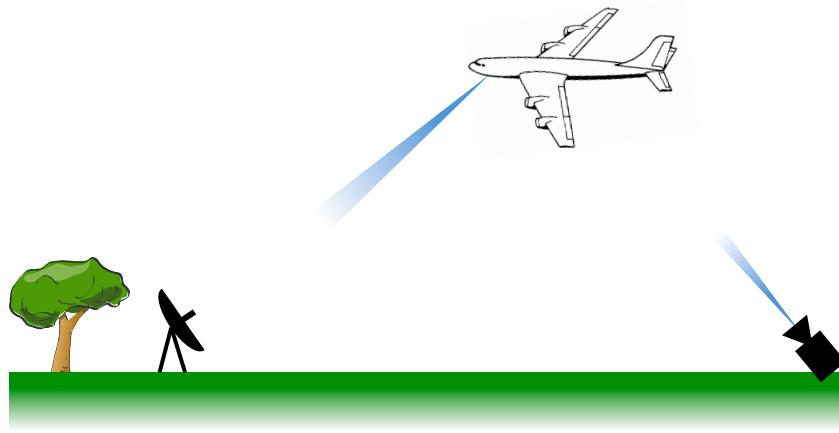
Figure 4 – Stealth and anti-jamming benefits of THz

1.3   Formation Flight Requirements

My concept for a THz positioning system is guided by application-specific requirements for precision airdrop. These requirements are derived from the need of trailing aircraft to stay clear of wake vortices from leading aircraft, which are dangerous to both the following aircraft and package parachutes [61], [62]. As a result, it is essential that following aircraft precisely maintain their positions relative to the lead during airdrop operations [15].

A set of error bounds, therefore, is applied to formation positions. Because the displacements of various expected formations are similar, the requirements are the same across a range of formation shapes. These requirements are defined in terms of a two-sigma rectangle, meaning that aircraft position must lie within the rectangle at least 95% of the time. The requirements are 50 m in the cross-track y direction, 150 m in the along-track x direction, and roughly 100 m in the vertical

z direction [15], [61]. The key requirement here is the cross-track requirement, because it is the tightest and also the dimension where errors are expected to be the largest. As a result, the work of this dissertation is directed towards minimizing cross-track position error.

Although different formations are possible, the error-bound requirements are considered to be independent of formation geometry. It is nonetheless important to define a representative formation for the purposes of analysis. A representative formation is shown in Figure 5. The particular element shown has an inverted-V shape. The element lead (AC1) is at the front. The first following aircraft (AC2) is positioned 1 km back in the along-track direction and 200 m offset in the cross-track direction. The second following aircraft (AC3) is positioned 2 km back in the along-track direction and 100 m offset in the cross-track direction.
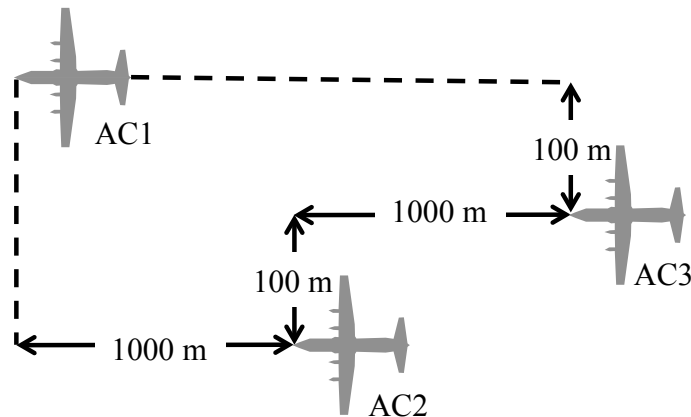
Figure 5 – Representative formation element

The representative formation helps to define two key variables that will be used later in analysis. One of these variables is the target separation between aircraft, which will be assumed to be in the 1 km to 2 km range. The second of these variables is the target bearing angle, which will be assumed to be in the 0 to 12° range, relative to the along-track axis. This bearing-angle range is a generalization

of the representative formation in Figure 5, where the largest bearing angle (between AC1 and AC2) is 11.3°.

In addition to the positioning requirements, some practical considerations are important for implementation. Overall system cost should be as low as possible. The device needs be compact and low profile, because weight and aerodynamic drag translate into inefficient use of fuel. The location and surface area (or footprint) of the device is particularly important for the cargo-aircraft application, noting that the nose-region of the aircraft is already densely populated with electronic equipment.

As a further constraint, it must be assumed that the full formation may consist of several sequenced elements, allowing for formations of more than three aircraft. The spacing between elements may be larger than 2 km, but it is assumed that precision relative-navigation is only required within each formation element, and not between elements. However, the system should be designed assuming that multiple transmitters are present in the extended formation. For a THz system, this implies a design that supports multiple access, with receivers being able to distinguish among multiple transmitters.

1.4   System Design Decisions

A design concept with the potential to address these requirements is one consisting of a THz transmitter fixed on the tail of the lead aircraft and a THz receiver fixed on the nose of each trailing aircraft. This section provides an overview of the design-concept architecture and identifies the key technical challenges that are addressed in the remainder of the dissertation.

The first key technical challenge is to make time-of-flight range measurements in the THz domain. The THz system is envisioned to make time-of-flight range measurements using commercial off-the-shelf THz equipment. A simplified block

diagram for proprietary hardware developed by PM&AM, LLC (Tucson, AZ) is shown in Figure 6. The transmitter, shown on the left, is composed of an arbitrary waveform generator regulated by an atomic clock, a frequency synthesizer, an amplifier multiplier chain and a diagonal horn antenna with a Teflon lens. The receiver, shown on the right, is composed of a Teflon lens and an antenna, an amplifier and detector, and a data acquisition system regulated by an atomic clock, which is synchronized to the transmitter clock. Like GPS, this system measures the time of flight of the THz signal to determine the range between the transmitter and receiver. While previous THz research has considered radar-like ranging, GPS-like ranging has not previously been demonstrated in the THz domain, and so error models must be developed for a realistic hardware implementation.
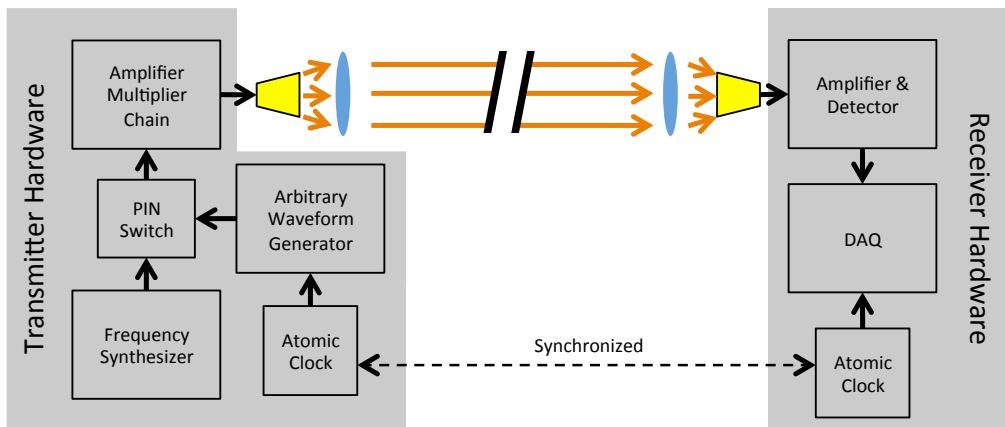
Figure 6 – Simplified system block diagram of THz hardware

Once range measurements have been made, the question remains how to convert those measurements into position estimates. There are many approaches to converting range into position. GPS uses trilateration, measuring the distance from multiple satellites with known positions and then finding the point where those distance measurements intersect. Similarly, trilateration could be used in the formation if each aircraft were equipped with both a transmitter and receiver for inter-aircraft ranging; however, the problems with this method are three-fold.

First, from a practical implementation perspective, the hardware is currently relatively expensive, on the order of $100,000 per component, so equipping every aircraft with both a transmitter and receiver is expensive. Second, this method does not support a minimal two-aircraft formation, as range measurements alone provide no information about the direction of the aircraft relative to one another. Finally, although range measurements between three or more aircraft can be used to determine the formation shape, without some other information, the solid angle rotation of the formation is unknown [10]. For example, consider the two formation geometries shown in Figure 7. Both formations have the same shape, and therefore the same range measurements, but the formation on the right is rotated by an angle ξ. Without some additional information, these two formations are indistinguishable from one another using range measurements alone. This problem is resolved in GPS because the absolute positions of the satellites are known. In this application, however, it is assumed that the aircraft operate in a GPS-denied environment and therefore have no knowledge of their absolute positions.
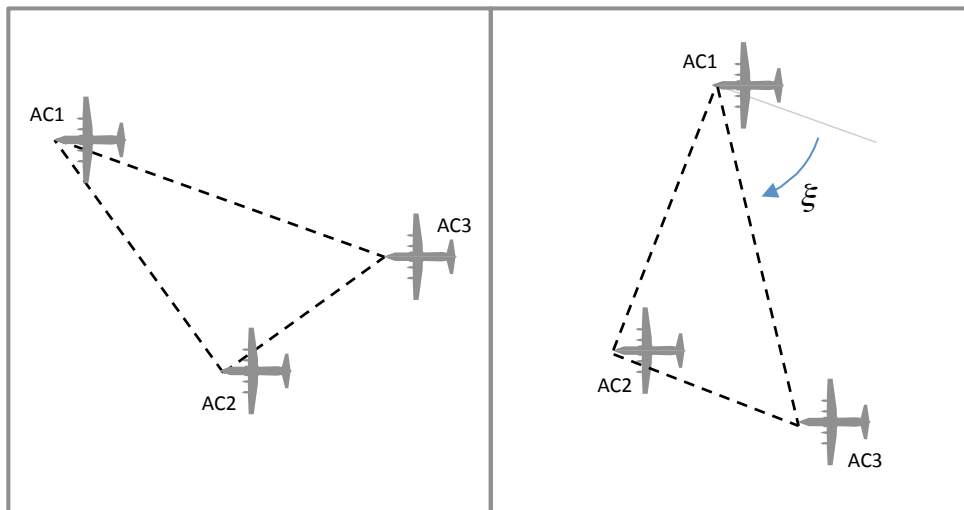
Figure 7 – Angle ambiguity in range-only relative positioning

To find the relative positions, therefore, consider measurements of the pointing vector that points from the receiver to the transmitter. When combined with the range measurement this provides full observability of the aircraft relative positions, including in the minimal two-aircraft formation case. In my previous work [63], [64], phased array methods were proposed to measure the bearing angle between the aircraft; however, these results relied on enhanced receiver hardware capable of carrier-phase tracking, which is not available with current off-the-shelf equipment. A method is needed, therefore, to measure the pointing vector using currently available hardware.

The second key technical challenge is to develop a method for measuring the bearing angle in order to supplement range measurements. This work proposes using planar bearing angle measurements made from a new interferometric receiver package, shown in Figure 8. This new device overcomes two key challenges of angle measurement with THz equipment. First, it uses a diffraction grating to convert the time-varying carrier signal into a spatial pattern, which can be probed by the detector, and allows the THz equipment to access otherwise inaccessible carrier-phase information for angle determination. Second, the diffraction grating moves, allowing a single detector to probe the pattern (instead of using a pixel-array), minimizing the cost of the system.
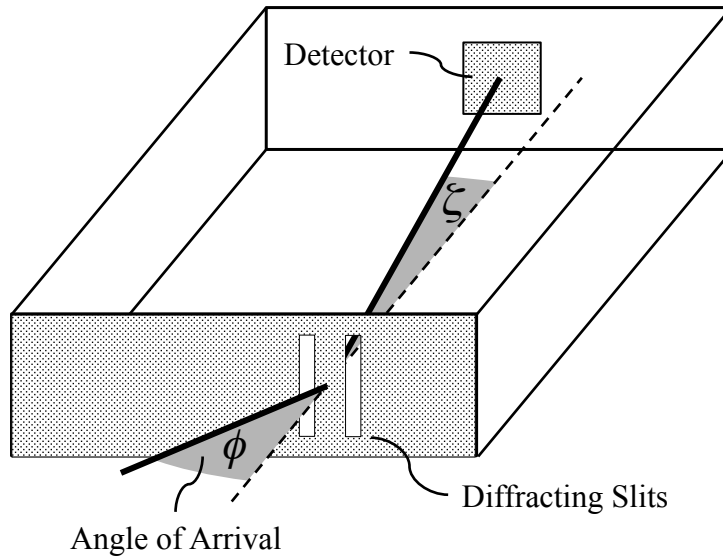
Figure 8 – THz interferometer concept

Although the moving diffraction grating has the potential to measure bearing angle observable, a number of key issues remain unresolved. First, the grating blocks a significant portion of the power before it can reach the detector, and diffraction spreads the remaining power over a wide area. It needs to be shown that this loss of power is not so significant that it makes ranging impossible. There is a large design space for the diffraction grating and a suitable grating design needs to be identified that allows for bearing observability while simultaneously allowing enough power through for ranging. This includes decisions about the number of slits in the grating, the width and spacing of the slits, and the position of the grating relative to the detector.

Once a grating design has been identified, the third key technical challenge is to develop method to leverage the time-varying power signal from the single-pixel detector to reconstruct the interference pattern and determine the bearing angle. Because measurements are noisy and the arriving power is uncertain, the interference pattern cannot be directly identified from individual signal measurements. Instead a time series is needed to recognize the pattern.

11

Furthermore, the pattern contains multiple spatially aliased peaks, which introduces ambiguity to snapshot measurements. A new algorithm is required to model time-varying power changes due to diffraction grating and infer bearing.

The fourth key technical challenge is to develop a method to enable effective communications even when the time-varying interference pattern strongly fades the signal power in regions of destructive interference. Low-rate communication is a key part of nearly any navigation system, transmitting key parameters. In GPS, for example, a host of parameters are communicated, including satellite ephemeris data, clock corrections, and satellite health. For formation flight, in addition to positioning parameters, it may also be beneficial to communicate control commands like formation maneuvers. Methods need to be developed to reliably pass data between aircraft when the signal strength is weak enough that communication bits are lost in the low-power destructive interference zones.

Architecturally, the proposed diffraction-grating system only measures a planar angle, not the pointing vector between the transmitter and receiver. In order to use the proposed moving-grating hardware, an alternate approach is needed for reconstructing the vertical separation. Rather than complicate the THz equipment design by adding a vertical angle capability, an alternative approach is suggested to use barometric altimeters to get a vertical separation measure. In this configuration the lead aircraft broadcasts its altitude to the trailing aircraft, exploiting the communication capability, and making a three-dimensional position solution possible.

Finally, the fifth key technical challenge is to demonstrate that this system is capable of producing position estimates with errors consistent with the requirements outlined in the prior section. It needs to be shown that the integrated system can fuse measurements of the range r, bearing angle ϕ, and altimeter h

together to produce precise position estimates and to verify that the THz system can perform the precision airdrop mission.

The measurements, it should be noted, do not map directly into the along-track x, cross-track y, and vertical z positions described in the requirements above, but instead form something close to cylindrical coordinates, as shown in Figure 9. The range measurement r is really a slant-range that includes a vertical component. When the relative altitude h equals zero, the range and slant-range are the same, but in general they will have slightly different values.
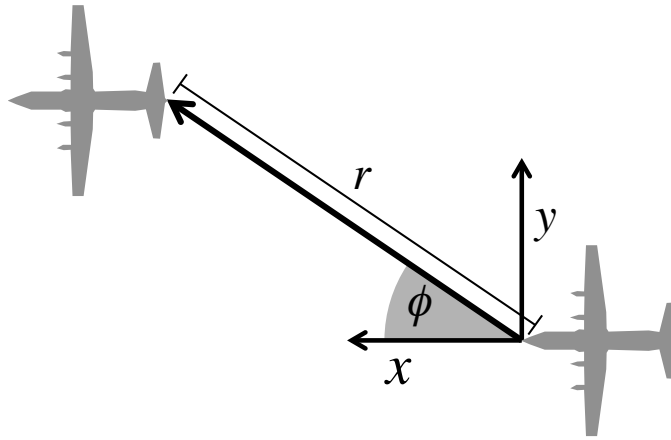
Figure 9 – Aircraft relative position and measurements

Although targeted to the precision airdrop application, the proposed system also has other potential applications. These include, military formation flight for drag reduction or aerial refueling, civilian formation flight, and possibly even short-range terrestrial applications like autonomous vehicles.

1.5   Contributions

The key contributions of this dissertation tackle the open questions identified in the prior section. These five contributions, enumerated below, are described in the five body chapters of the dissertation.

1.5.1 Leverage a hardware demonstration of THz ranging to identify differences from GPS and establish a baseline error model for algorithm development

This contribution addresses the first key technical challenge identified above, making time-of-flight range measurements in the THz domain. Working with partners at Physics, Materials, and Applied Mathematics LLC (PM&AM), ground-level tests were used to demonstrate GPS-like ranging using THz signals. This demonstration identified carrier phase tracking capabilities as a key difference between THz and GPS hardware. An error model was developed that describes observations to 100 m and that predicts errors at longer ranges. An important observation is that multipath effects differ from the typical signal delay seen in GPS multipath, and highlights the unique propagation properties of THz signals. The experimental data and associated error models are also used to quantify key equipment parameters for the baseline hardware that are used throughout the work. To the author's knowledge, this is the first analysis and demonstration of GPS-like ranging in the THz domain.

1.5.2 Quantify power loss from introducing a moving diffraction grating in the THz receiver, and present a method to mitigate the power loss using spatial aliasing

To address the second key challenge, a novel interferometric receiver package is proposed that uses a moving diffraction grating, and the resulting power loss is quantified and mitigated. The diffraction grating generates an interference pattern that makes the angle observable, but it results in significant power losses. To address this, a range of different geometric grating configurations are considered and a double-slit diffraction grating configuration is identified as a prime candidate to achieve observability with minimal power loss. This configuration uses spatial aliasing (the repetition of interference peaks) to make the bearing

angle observable over a wide field of view using a relatively narrow grating scan. The narrow scan allows the detector to integrate the signal over a longer period of time, mitigating the power loss while still maintaining a five-fold power difference between peaks and valleys in the pattern to ensure angle observability. To the author's knowledge, this is the first device to use a movable diffraction grating to make bearing angle observable in the THz range.

1.5.3 Propose a method to solve the inverse problem of obtaining bearing angle from a series of noisy power measurements modulated by the time-varying interference pattern gain

A method is presented to address the third key technical challenge, mapping the time-varying power signal measurements into the bearing angle. As the grating sweeps in front of the detector, a time-varying pattern of peaks and valleys in the received power is observed. This pattern is a function of the bearing angle and the known diffraction grating position. It is relatively easy to write an equation for the power at the receiver over time given a bearing angle, but inverting the problem to find solve for the bearing angle given a time series of power measurements is challenging. The problem is even harder when the noise level is high such that the received power is at an instant is not observable and a large number of measurements collected over time is needed to make an inference about the power. A modeling approach is proposed to solve the inverse problem and placed in the context of a least squares (LS) estimator to demonstrate the capabilities and provide an initial estimate of bearing angle measurement accuracy.

1.5.4 Propose a technique for matching the data-word repetition frequency to diffraction-grating scan frequency to enable reliable communication through intermittent periods of low gain

To resolve the fourth key technical challenge, communicating data through periodic dips in the received power due to the diffraction grating, a frequency-matching scheme is devised for the data transmission and diffraction-grating scan rates. In the noisy THz measurement environment, destructive interference caused by the diffraction grating can be severe enough to make communication data bits impossible to read. By contrast, in the constructive-interference regions of the pattern, communication is significantly easier because the signal-to-noise ratio is vastly improved. In order to make communication reliable at longer ranges, a clever solution is needed to design the communication broadcast to respect the motion of the diffraction grating, without knowing the phasing of the grating motion (because the diffraction pattern peak phasing varies with bearing angle). A solution is proposed that uses data repetition and matched phases to ensure that if a data transmission arrives when the diffraction grating gain is low, its repetition will arrive when the gain is high. Analysis shows that the proposed structure achieves reliable communication with missed data occurring only once every 2500 transmissions despite the interference pattern valleys.

1.5.5 Demonstrate that the system meets the requirements for precision airdrop by incorporating component analyses (hardware, signal propagation, and algorithms) into a three-dimensional simulation of the integrated THz system

Finally, the fifth key technical challenge is whether or not the full system as proposed can actually meet the error-bound requirements for precision airdrop (or any other application of interest). This is challenging because it requires combining analyses of hardware noise, signal propagation, signal design and signal processing, data modulation and communication, diffraction grating,

solving the inverse equation to get bearing, and combining range, bearing, and altitude data to get the desired relative position measurement. An Extended Kalman Filter (EKF) was used as the tool to bring these together models together, with the inverse problem being solved across multiple time steps. The end result is that the EKF cross-track accuracy is 11 m, two-sigma, at 2km, well within the 50 m requirement for precision airdrop.

1.6   Dissertation Overview

This dissertation is composed of five main body chapters, detailing each of the contributions listed above. Chapter 2 describes the development of the THz ranging error model from a ground-level demonstration. Chapter 3 introduces a novel THz angle sensor and quantifies and mitigates the power losses. Chapter 4 proposes an algorithm to solve the inverse problem and map the interference pattern to the bearing angle. Chapter 5 describes a signal and data structure for THz communication with matching diffraction grating scan frequency to minimize missed communications. Chapter 6 ties the range, bearing angle, and altitude measurements together in an algorithm and demonstrates that the THz system meets the requirements for precision airdrop applications. Finally, Chapter 7 summarizes the findings of the dissertation and comes to some conclusions. The beginning of some new work exploring diffraction patterns in the grating mid-field and their potential benefits for interferometry is described in Appendix 1. MATLAB code used in simulation is presented in Appendix 2.

Chapter 2

THz Range Measurement[1]

2.1   Introduction

Research on THz ranging has thus far primarily consisted of radar-like methods, which are inherently range limited due to spreading and scattering losses following signal reflection. For surface inspection and remote sensing, THz pulses are reflected off surfaces and the return echo signal is analyzed using time-of-flight measurements to determine distance and frequency content to determine surface properties [38]–[40]. THz synthetic aperture radars use a series of time-of-flight measurements from reflected THz pulses to construct a high resolution image of surface contours [41]–[44]. Similar THz radar applications have also been examined for automotive sensing and collision avoidance [45], [46].

This work considers a fundamentally different approach to range measurement in the THz band. Modeling on the great success of GPS [66], [67], this system utilizes one-way transmission of a signal between a transmitter and receiver that are not co-located and uses signal correlators to identify a modulated signal from

---

[1] This chapter was published under the title "Exploiting the Terahertz Band for Radionavigation" in the *Journal of Infrared, Millimeter, and Terahertz Waves* [65]. Some changes and restructuring of the text have been made for clarity and flow.

noisy measurements, allowing the system to function over significantly longer distances than radar-like systems.

This chapter addresses the technical challenge of making time-of-flight range measurement using THz signals and develops an error model to explain the results of ground-level tests. It describes three key results. First, it discusses THz hardware implementation and identifies relevant differences from conventional GPS hardware, which operates at radio frequencies (RF). Next, it presents experimental results from a ground-level demonstration performed on a compact outdoor test range. Lastly, it introduces a model for THz range-measurement error with an emphasis on multipath, since multipath plays a somewhat different role in THz-ranging systems as compared to conventional radio-frequency ranging systems. Simulations suggest the potential for obtaining ranging measurements over baselines of several kilometers.

2.2   Experimental Hardware

Similar to GPS, the THz ranging system works by measuring the signal's time of flight between the transmitter and receiver. The time-of-flight delay τ is simply the difference between the time the signal leaves transmitter t_t and the time the signal reaches the receiver t_r

$$\tau = t_r - t_t. \tag{1}$$

Because the signal travels at the speed of light, the distance r between the transmitter and receiver is equal to the time of flight τ multiplied by the speed of light c

$$r = c\tau. \tag{2}$$

Time-of-flight measurements were made using proprietary hardware developed by PM&AM, LLC (Tucson, AZ). A simplified block diagram for the hardware is shown in Figure 10. The transmitter hardware is illustrated on the left of the figure, and the receiver hardware on the right.
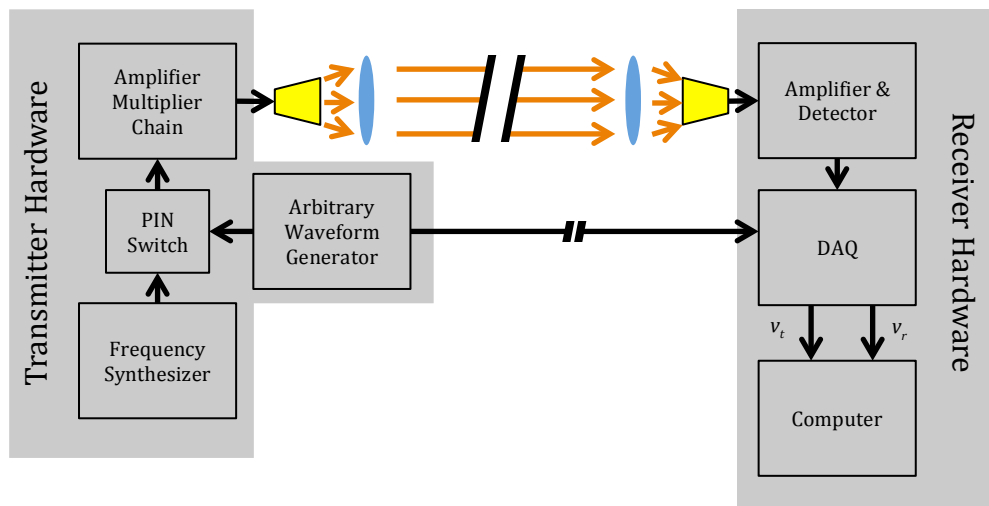
Figure 10 – Simplified system block diagram of PM&AM ground-level THz hardware

This experimental setup is similar to the hardware described above in Chapter 1, Figure 6; however, for simplicity and economics in the preliminary ground tests, the synchronized atomic clocks were eliminated. Instead, the clock in the DAQ serves as the common time standard, with the transmitter and receiver hardware hardwired to one another to simulate synchronization.

This system differs from previously published THz ranging systems [38], [41], [42], [45], [46], which use radar methods to measure the range. Modeled on GPS, the transmitter and receiver are placed at opposite ends. This significantly reduces transmission losses as compared to radar methods, because the signal travels half as far and does not require a reflection. In addition, it makes it possible for a

standalone receiver to measure range and perform trilateration position estimation, as in GPS.

The remainder of this section provides a rough overview of the transmitter equipment, the receiver equipment, and the ground-truth system used in subsequent experiments.

2.2.1   Transmitter

The primary components of the transmitter are a tunable frequency synthesizer and an amplifier-multiplier chain (AMC). In this system, the base frequency generated by the synthesizer is increased by a factor of 24 going through the AMC. For example, to generate 300 GHz (0.3 THz), the output of the tunable frequency synthesizer is set to 12.5 GHz so that after the AMC, a carrier frequency of 12.5 GHz x 24 = 300 GHz is obtained.

Simple on-off keying is applied to modulate the carrier signal. An arbitrary waveform generator controls a PIN switch with 200 ps rise/fall time, which toggles the transmitter on and off. In addition to controlling the PIN switch, the output of the waveform generator is used to trigger data collection on the personal computer.

Although the waveform generator will eventually be used to modulate a pseudo-random code sequence onto the THz carrier, the modulation in the following experiments was a simple 10 MHz square wave (chipping rate of 20 MHz). The modulated signal has a period of 100 ns and a wavelength of 30 m, as illustrated in Figure 11.
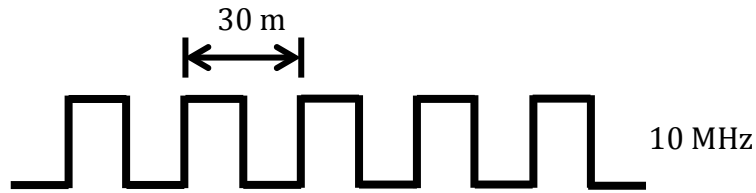
Figure 11 – Modulated on-off square wave

The assembled transmitter has an output power greater than 30 mW in the 300-325 GHz band. At the output of the transmitter, a diagonal horn antenna is used to transmit the 300 GHz carrier wave. To decrease spreading losses, a Teflon lens is used to partially collimate the beam, resulting in a signal divergence of roughly 2.5° (we reasonably treat the transmitter hardware as a point source in the following analysis).

The key difference between the THz transmitter and RF transmitters used in GPS is the modulation of the carrier signal. Ideally, binary phase-shift keying (BPSK) would be used to modulate code onto the THz carrier signal, as in GPS; however, as described below, current hardware limitations make carrier-phase tracking impossible. As a result, simple on-off modulation is used.

2.2.2   Receiver

The receiver equipment consists of an analog front end that demodulates the received signal, and a data acquisition system that digitizes the demodulated signal and sends it to a personal computer.

The first stage of the front end is a Teflon lens that concentrates the received signal on a 25-dB gain diagonal horn antenna. The beam divergence of the lens/antenna assembly is roughly 4°, and the lens area is 4.9×10^{-5} m^2. Attached to the antenna are a series of low-noise amplifiers that feed into a 300-330 GHz zero-bias Schottky-diode direct detector. The detector rectifies and accumulates the received signal. Rectification results in demodulation, mixing half the power

in the THz signal down to the baseband, centered at DC (and the other half of the power to twice the carrier frequency). The accumulation process acts likes a low-pass filter stage, with a bandwidth of approximately 1 GHz. In the process of accumulation, only the baseband signal is preserved. All higher-frequency data (e.g. carrier phase data) is lost during accumulation. The output of the detector is fed to a data acquisition system (DAQ), where an A/D converter digitizes the detector output at a sample rate of 1 Gsample/sec.

In order to determine range from time-of-flight, the DAQ sends the detector output to the personal computer for storage and post-processing. The DAQ also digitizes the waveform generator output, which is used to create the transmitted signal. In this manner, the detector and waveform generator outputs were both tagged with a common clock at the data-acquisition computer. The transmission and line delays were calibrated to correct the time-of-flight measurement (see below). Some residual clock error persists, where that error is a fraction of the DAQ's temporal resolution (e.g., a fraction of 1 ns).

As noted, the key difference between the THz and GPS receivers is their ability to track the phase of the carrier signal. Although this capability will likely be available in the future, currently affordable THz receivers rectify and accumulate the signal, resulting in a loss of carrier phase data. This prevents the use of BPSK on the THz carrier signal, as in GPS.

2.2.3   Ground Truth System

Ground truth was obtained by taking physical distance measurements (using a tape measure), in order to provide a basis of comparison for the ranging measurements obtained from the THz system.

In order to allow for variable separation between receiver and transmitter hardware, both units were placed on separate mobile carts. The equipment was

mounted on the carts approximately 1 m above the ground. The receiver cart was equipped with an uninterruptible power supply that allowed the operation of the receiver for over 8 hours without being tethered to a wall plug. Due to its higher power demands, the transmitter cart was powered via a wall plug. The separation between the transmitter and receiver carts was measured using markings on the ground, which were surveyed with a tape measure prior to the start of experimental data collection.

2.3   Range Measurement Methods

The objective of the experimental system is to precisely measure the signal's time-of-flight so that the range can be determined accurately via equation (2). A code-phase-based method was used to compare the offset between the transmitted and received signal (as described above, carrier-phase information is lost in signal rectification, so only code-phase is available for processing). The basic concept is to compare the received signal to the transmitted signal in order to determine the delay caused by the time-of-flight. The received signal should be delayed as compared to the transmitted signal, so finding the range entails finding the phase delay of the modulated signal (assuming that equipment delays have already been accounted for via the calibration described below), as illustrated in Figure 12. In the case shown, the phase difference between the received and transmitted signals is 0.7 cycles. The 10 MHz modulation signal has a period of 100 ns corresponding to a wavelength of $\lambda = 30$ m, so if the phase shift between the transmitter signal and the receiver signal is $\theta_r - \theta_t = 0.7$, the distance between transmitter and receiver is $(\theta_r - \theta_t)\lambda = 21$ m.

$\theta_r - \theta_t = 0.7$ cycles

Transmitted Signal

Received Signal
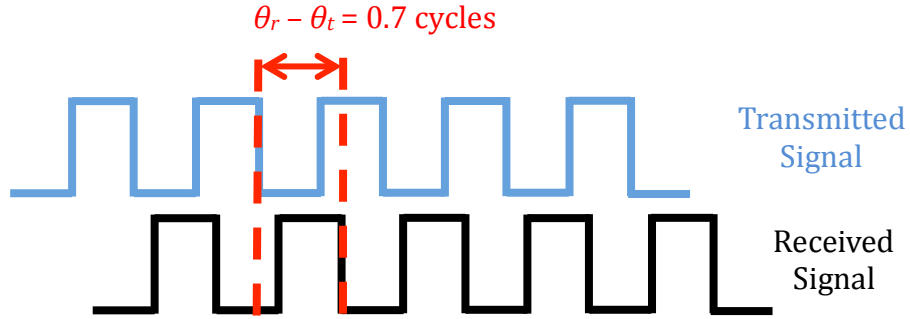
Figure 12 – Example phase-angle ranging

It should be noted that range measurements obtained in this way are subject to a cycle ambiguity since all chips have the same form. Because of the repetitive nature of the signal, it is impossible to tell the difference between a phase shift of 0.7 cycles and one of 1.7 cycles, 2.7 cycles, *etc*. For the purposes of this experiment, ambiguities were resolved procedurally, as described below (rather than via signal processing as, for example, in GPS).

To simplify signal processing, the phase shift was related to time-of-flight using only the fundamental frequency of the modulated signal. The fundamental frequency is the sine wave at the same frequency as the transmitted square wave. Decomposing the square wave sequence into a Fourier Series, the voltage component associated with the fundamental frequency is a sinusoidal wave at $\omega_{mod}/2\pi = 10$ MHz. For this sinusoidal wave, the phase shift can be related to time-of-flight τ from equations (1) and (2) as

$$\tau(k) = \frac{1}{\omega_{mod}} (\theta_r(k) - \theta_t(k) + 2\pi M) - T_{delay} + n_\tau, \quad (3)$$

where k is the time step (with $t(k) = kT_s$ where T_s is the sample period of 1 ns), θ_r is the received-signal phase shift and θ_t is the transmitted-signal phase shift (both are defined relative to the arbitrary clock reference), M is the cycle ambiguity, which is an arbitrary integer, T_{delay} represents the cable and hardware

delays, and n_τ a random variable representing corruption of the time-of-flight measurement by thermal noise and multipath.

Because the clock reference is common to the received and transmitted signals, the time reference can be set arbitrarily to zero at the midpoint of the first chip. The analytic expression v_t for the fundamental-frequency component of the transmitted signal voltage is given by

$$v_t = \cos(\omega_{mod}t). \tag{4}$$

Higher harmonic frequencies were not tracked in our experiments, so we will consider only the fundamental frequency here.

The fundamental frequency of the received signal is assumed to match that of the transmitted signal, as Doppler shift was assumed to be negligible for all experiments. Thus, the voltage v_r associated with the fundamental frequency of the received signal is modeled to be

$$v_r = \cos\left(\omega_{mod}\left(t - \tau - T_{delay}\right)\right), \tag{5}$$

A very simple means of extracting the phase shift at the fundamental frequency from the raw (square-wave) data is to correlate both transmitted and received signals with a complex sinusoid at the fundamental frequency. This procedure extracts the in-phase and quadrature components of the signal. The correlation operation was performed in software after digitization of the signals from the waveform generator (transmitter signal) and the detector (receiver signal). These correlations were performed over a sliding window of K data points, as described by the following equation. The in-phase and quadrature correlations at a particular

time step k, labeled $I(k)$ and $Q(k)$ respectively, are equivalent to discrete cosine and sine transforms at the fundamental frequency

$$I_i(k) = \sum_{m=k-K+1}^{k} \cos(\omega_{mod} m T_s) v_i(m) \quad \text{and} \tag{6}$$

$$Q_i(k) = \sum_{m=k-K+1}^{k} \sin(\omega_{mod} m T_s) v_i(m) . \tag{7}$$

The same correlation operation was applied to both the transmitted and received waveforms, hence the index i is used here to refer generically to either the transmitted or received signal, such that $v_i = \{v_t, v_r\}$ from equations (4) and (5).

The in-phase and quadrature signals can be combined to compute a phase angle at each time step as follows,

$$\theta_i(k) = \tan^{-1}\left(\frac{-Q_i(k)}{I_i(k)}\right). \tag{8}$$

The transmitted signal phase θ_t and received signal phase θ_r were obtained by applying equations (6), (7) and (8) separately to the transmitted and received waveforms v_t and v_r. Finally, the time-of-flight is computed by substituting these values of θ_t and θ_r into the time-of-flight equation, given by equation (3).

An example correlation is shown in Figure 13, for a small sample set of receiver data collected at an equipment separation of 4 m. The raw voltage measurements that make up the received signal v_r, shown in blue, form a square wave corrupted by noise. The on-off transitions of the square wave are clearly visible at 4 m separation. The purpose of the correlators from equations (6) and (7) is to extract

phase even when the transitions are not visible to the eye (as occurs at larger separation distances). The correlation function $\cos(\omega_{mod}t + \theta_r)$, shown in red, is a sinusoid at the fundamental frequency of modulation. In this case, the received signal's phase θ_r is roughly 290° as computed from equation (8), meaning the signal v_r is shifted by 290° as compared to the reference v_t. Assuming that the transmitted signal's phase is $\theta_t = 0°$, the ambiguity is $M = 0$, and the delay T_{delay} due to transmission through the cables and hardware is known to be 67 ns, by equation (3) the time-of-flight τ is 13.6 ns. Multiplying by the speed of light c, this gives a measured distance in this example of 4.07 m. It should be noted that only 300 samples are shown here for simplicity, whereas a window size of $K = 450{,}000$ samples is used in the experiments described below.
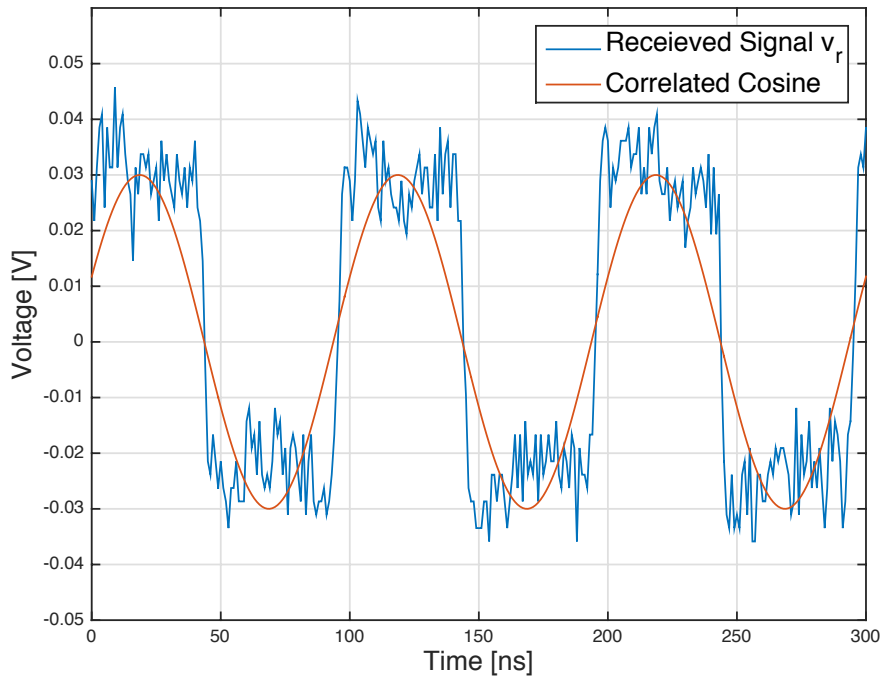
Figure 13 – An example correlation applied to a sample signal v_r received from a transmitter 4 m away

There are a couple of important differences between this THz system and GPS systems. First, for simplicity in this demonstration, simple square wave

modulation is used; however, more complex pseudorandom codes, like those used in GPS, could readily be applied to the input signal from the arbitrary waveform generator. Second, for demonstration purposes, this system uses a single clock in the DAQ tethered to both the transmitter and receiver as the time basis for measuring the delay caused by the signal's time-of-flight. The tether could be cut in one of two ways: either both the transmitter and receiver could both be equipped with synchronized atomic clocks, or, as in GPS, measurements from a constellation of four or more transmitters, each equipped with synchronized atomic clocks, could be used by the receiver to estimate the three position coordinates as well as the unknown time coordinate.

2.3.1   Ambiguity Resolution

For expediency, the ambiguity M in equation (3) was resolved manually. By starting the transmitter cart adjacent to the receiver cart, the initial ambiguity was known to be zero. In subsequent testing, when the receiver cart was slowly pulled away from the transmitter, jumps appeared in the phase measurements at each integer transition. These jumps were removed manually in post-processing so that θ_r was obtained as an accumulated phase (rather than an absolute phase angle).

Automatic resolution of the integer ambiguity will be possible in the future when the square wave signal is replaced with a pseudorandom code sequence. This implementation detail has been left to future work.

2.3.2   Calibration

Prior to experimental trials, a calibration procedure was performed to estimate the equipment delay T_{delay}. This was achieved by comparing a THz range measurement to a ground truth measurement at a short distance, where the signal-to-noise ratio is high. The calibration procedure is as follows:

1.   Position the transmitter and receiver close to each other

2. Make time-of-flight measurement τ using THz equipment and equations (3)-(8)

3. Make independent ground truth measurement and determine expected time-of-flight $\tilde{\tau}$ based on the speed of light c using equation (2)

4. Compare THz and ground truth measurements to determine offset T_{delay} using the following equation

$$T_{delay} = \tau - \tilde{\tau} \qquad (9)$$

The delay T_{delay} is stable over the length of a test (5-10 minutes), but appeared to have variability of roughly 1 ns (up to 30 cm) over the course of days. An average value of T_{delay} (equal to 69 ns) is used in all subsequent analysis, and then a detrend process is applied to remove any residual bias from the specific data set. This is done by finding the mean error (bias) for short-range measurements (2-30 m), where ground-bounce multipath error is not a factor, and subtracting that bias from each measurement in the data set.

2.4 Experimental Results

Experiments were conducted at the outdoor facilities of PM&AM Research, LLC in Tucson, AZ. Results are presented for static and dynamic tests. In static tests, measurements were made with transmitter and receiver held stationary. In dynamic tests, measurements were made while range was continuously changed. The atmospheric conditions during these experiments were as follows: 103 $^{\circ}$F (39.4 °C, 313 K), 6% relative humidity, 923 millibars (92,300 Pa), and a dew point of 25.5 $^{\circ}$F (-3.6 °C, 270 K). Though the measurements were made at ground level, it should be noted that the local terrain was at 2500 ft (762 m) above sea level.

2.4.1 Thermal Noise Measurement

The noise spectral power P_{N_0} was not well characterized for the experimental system, and so it was necessary to use experimental data to evaluate this parameter for use in the range error model developed in the following section. Noise was assessed by running the experimental system at a short range (4 m), where multipath was expected to be negligible. At this range, the receiver data were digitized and a power spectrum was evaluated using $K = 450{,}000$ points. Note that exactly 100 data points were captured over each wavelength of the 10 MHz fundamental frequency, and that K is an integer multiple of this period.

The received signal can be modeled as a square wave corrupted by noise,

$$v_r(k) = v_0 \operatorname{square}(\omega_{mod} kT_s + \theta) + n_v(k)\,, \qquad (10)$$

where v_0 is an amplitude scaling factor that comes from the link budget and n_v is the noise on the signal, which is assumed to be a zero mean Gaussian random variable

$$n_v = \mathcal{N}(0, \sigma_v^2)\,, \qquad (11)$$

where σ_v is the standard deviation. From the 4 m data, an portion of which is shown in Figure 13, the amplitude v_0 is found to be 26.7 mV and it is corrupted by random noise with a standard deviation σ_v of 7.7 mV.

The signal amplitude v_0 is a manifestation of the signal intensity at the detector, and is defined here as

$$v_0 = G_R \sqrt{P_C} \, , \tag{12}$$

where P_C is the received signal power, calculated from the link budget described below in equation (18), and G_R is a receiver gain. The receiver gain relates the power incident on the receiver to the signal amplitude observed. From the 4 m data, its value is found to be $6.81 \text{ V}/\sqrt{\text{W}}$.

The power spectra at 4 m range are shown in Figure 14. Power is plotted on the vertical axis in arbitrary units and the frequency is shown along the horizontal axis in hertz. The plot on the left shows the power magnitude for the half spectrum through the Nyquist frequency (at 500 MHz). The plot on the right zooms in on the fundamental frequency and first harmonic (the range 5-25 MHz).
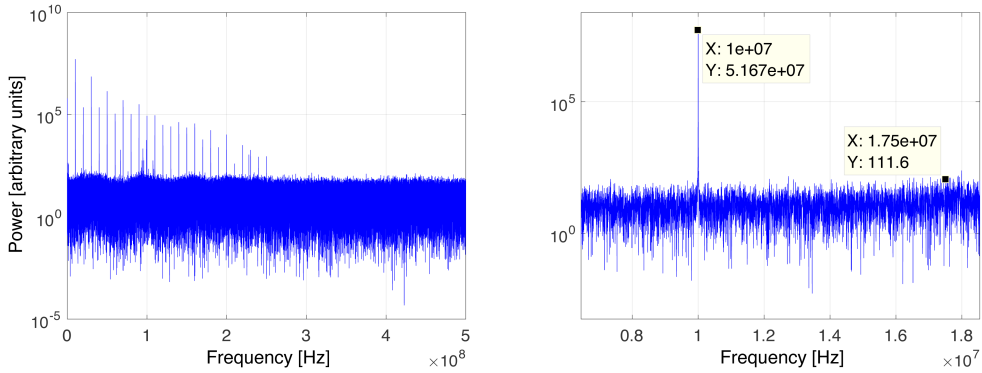
Figure 14 – Signal power over a wide range of frequencies (left) and zoomed in on the main peak at 10 MHz (right)

The figures confirm that the noise spectrum is essentially flat, justifying the assumption that thermal noise is white. Code power P_C and noise power P_N were obtained in the right hand plot (with values of $P_C = 5.2 \times 10^7$ and $P_N = 110$, in arbitrary units). Taking multipath to be zero at this range, the ratio of these two values can be used to estimate P_C/P_N. Evaluating P_C at 4 m from equation (15), as described below, it is possible to estimate P_N from the data, and hence to estimate

N_0 from Equation (16). The resulting value of the P_{N_0} parameter is listed in Table I.

2.4.2   Range Error from Static Tests

Static tests are so named because neither the transmitter nor the receiver carts were moving relative to one another. The advantages of static tests are two-fold: first, because the carts are not moving, the range between the transmitter and receiver (i.e. the ground truth measurement) can be determined precisely using a tape measure. Second, multiple data points can be recorded to assess the spread of range estimates around each transmitter-receiver separation.

In static tests, the transmitter and receiver carts were set up so that the antennas were approximately aligned. Experiments were run at several fixed distances of roughly 1 m, 10 m, 20 m, and every subsequent 10 m interval out to 100 m. The results of these static range measurements are shown in Figure 15. The THz measurement errors, relative to the ground truth, are plotted in blue as a function of the true transmitter-receiver separation. To provide context, the figure also shows two types of one-sigma error bound: one without multipath (black line) and one with multipath (gray box). These models will be introduced and discussed in more detail below.
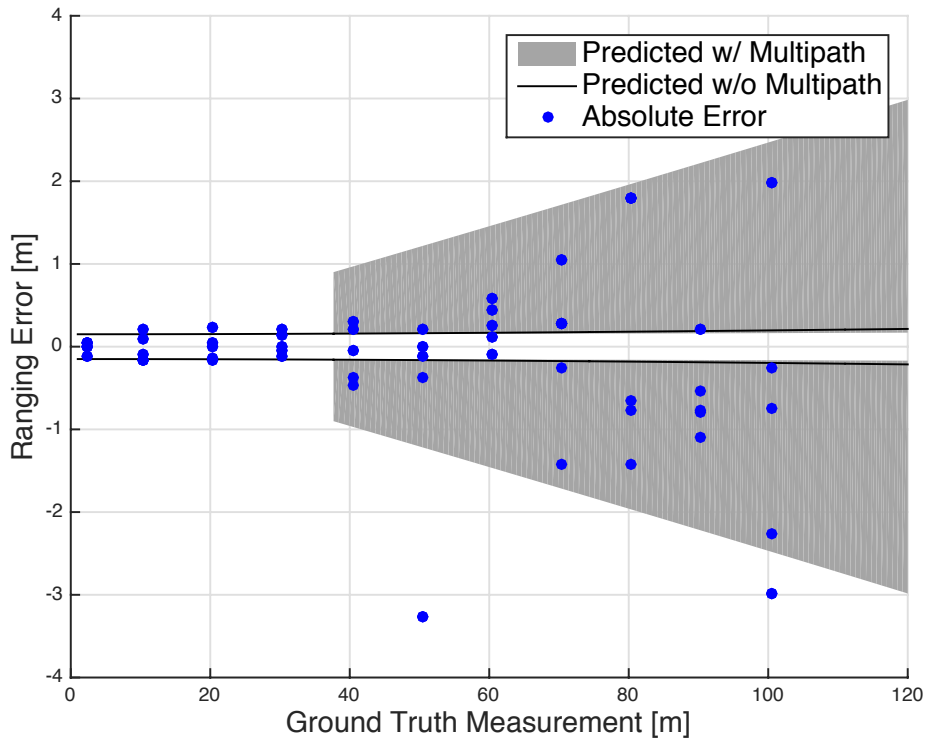
Figure 15 – Static trial range measurement error

2.4.3 Dynamic Tests

In contrast to static tests, dynamic tests involved moving the receiver cart continuously over the entire range of interest. While the receiver cart was manually pulled away from the transmitter, data were recorded to the computer RAM in real time. The transmitted and received signals were periodically stored to disk. In the trial, the receiver cart covered a distance of almost 100 m over the course of 200 s, at roughly constant speed (~0.5 m/s).

Concurrently, the ground truth measurements were logged independently of the THz measurements using distance markers on the ground and a stopwatch. Each time the receiver cart passed one of the distance markers, the stopwatch lap timer was used to record the time that had elapsed since the previous marker. While there is some timing error associated with this low-tech solution, the receiver

moved slowly enough that the resulting record of position versus time was sufficiently accurate to provide a good ground truth for the THz ranging system.

Figure 16 shows the results from this trial. The cart was initially stationary, and then began moving at roughly constant speed over the length of the test range before coming to a stop at the end. The THz ranges are plotted as blue circles, while the ground truth is plotted as a red line. A linear spline is used to interpolate a continuous ground truth model from the set of ground marker and lap time pairs.
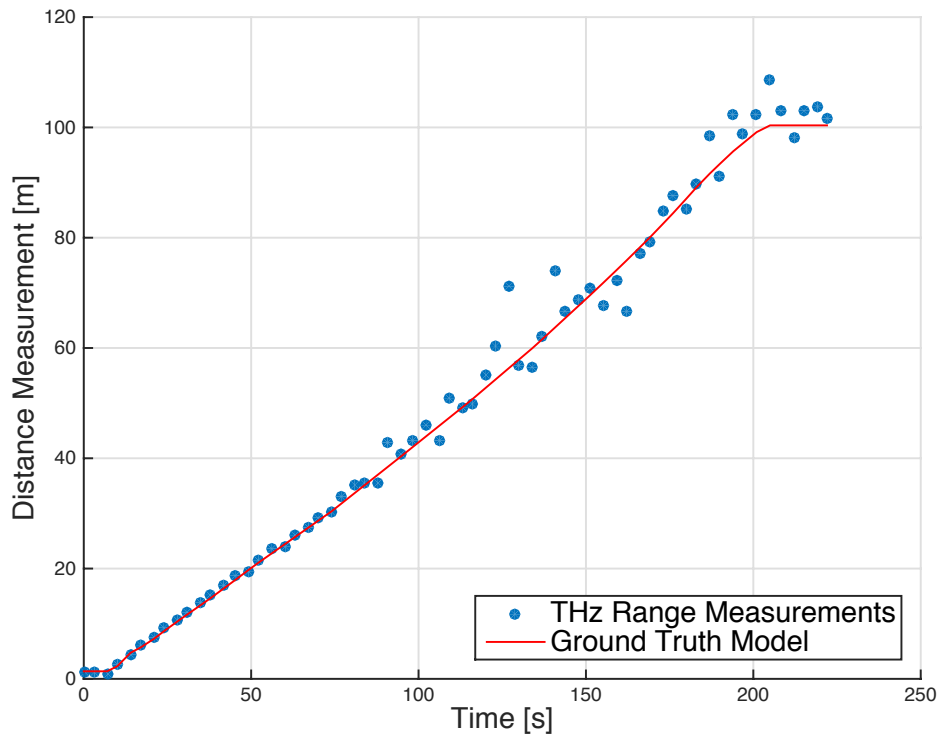
Figure 16 – Dynamic trial range measurement time series

The measurement residuals (measurement minus ground truth model) are plotted in Figure 17. Individual ranging error measurements are shown as blue circles, and the one-sigma error bounds are again shown in black with the gray boxes indicating the uncertainty once multipath signals come into play. It should be noted that the errors in this trial are significantly higher than in the static trial.

35
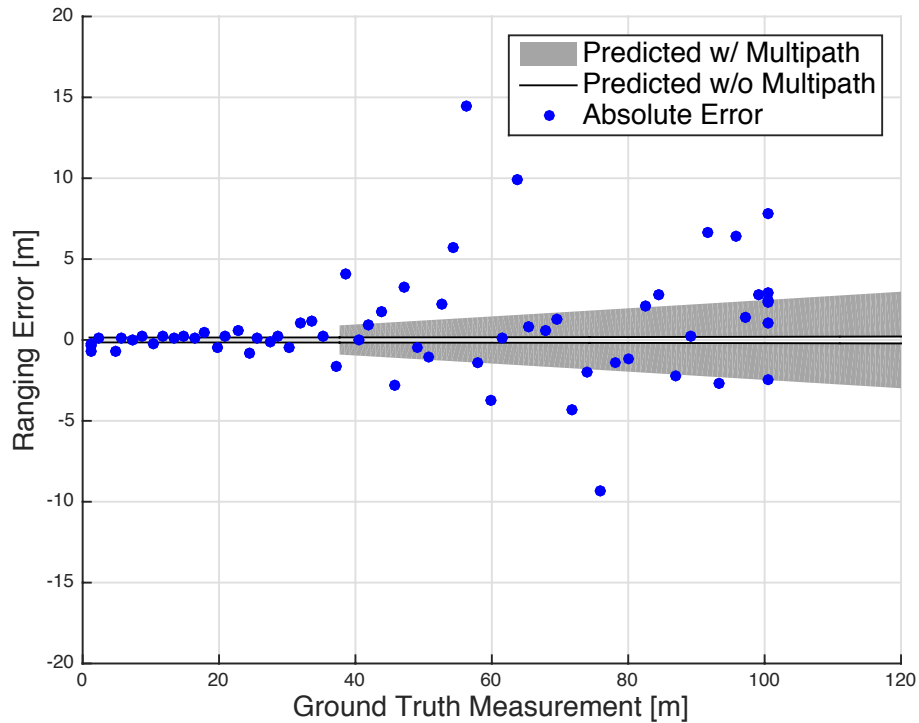
Figure 17 – Dynamic trial range measurement error

2.5   Range Error Model

The key contribution of this chapter is the development of error models to describe the phenomena observed in experimental trials and to predict the performance of the equipment over distances larger than supported by the test facility.

Range measurement error is assumed to be composed primarily of two independent sources: timing error from imperfect clocks and tracking error from noise and multipath signals. Assuming Gaussian errors, the total one-sigma range measurement error σ_r is the root of the sum of the squares of the one-sigma clock error $\sigma_{r,clk}$ and the one-sigma tracking error $\sigma_{r,tr}$

$$\sigma_r = \sqrt{\sigma_{r,clk}^2 + \sigma_{r,tr}^2}. \tag{13}$$

2.5.1 Clock Error

Clock error results from uncertainty in the limited time resolution of DAQ, which acquired samples from both the arbitrary waveform generator clock and the detector. The sample interval was 1 ns, and sampling was phase locked with the modulation frequency. Because of discrete sampling, the relative phase of the transmitter and receiver signals cannot be determined with a resolution of better than half a sample. Taking clock error to be 0.5 ns one-sigma, and multiplying by the speed of light c results in a clock ranging error $\sigma_{r,clk}$ of 15 cm, one-sigma, in equation (13).

2.5.2 Tracking Error

In addition to clock error, tracking error is a significant contributor to the total range measurement error. The one-sigma tracking error $\sigma_{r,tr}$ primarily depends on the ability of the integrators in equations (6) and (7) to pick out the signal over the background noise. In other words, accuracy is a function of the received code-to-noise power ratio P_C/P_N. A model of the one-sigma error $\sigma_{r,tr}$ in equation (13), as a function of the code-to-noise ratio P_C/P_N, is derived in the chapter appendix. The primary result of this derivation, where c is the speed of light and ω_{mod} is the angular frequency of the modulated signal, is

$$\sigma_{r,tr} = \frac{c}{\omega_{mod}} \sqrt{\frac{1}{2(P_C/P_N)}}. \tag{14}$$

The code-to-noise ratio P_C/P_N is the key parameter in this expression. It compares the power of the desired signal P_C to the power of the background noise

P_N. It varies with the distance between the transmitter and receiver as the strength of the signal decreases. To develop a representative model of the error, we consider two distinct regions: short distances where only the direct signal and thermal noise reach the receiver, and long distances where indirect multipath signals (i.e. signals that reflect off the ground) must also be considered (as shown in Figure 18).
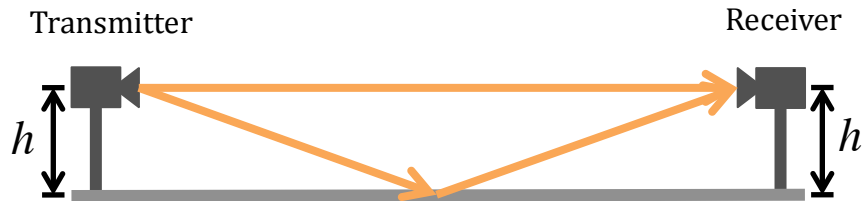
Figure 18 – Direct and indirect multipath signals

2.5.2.1 Direct Signal

To develop a model of the power incident on the receiver P_R in the absence of multipath, consider the system link budget. After accounting for all losses, the received signal power P_R is only a fraction of the initial transmitted power P_T. The dominant energy loss is due to beam spreading. Spreading losses are calculated as the ratio of the area of the receiver lens A_{lens} to the area $2\pi r^2 (1 - \cos\phi_t)$ of the spherical cap subtended by the beam (the spatial propagation pattern of the beam is spherical), which is a function of range r from the transmitter and beam spreading angle ϕ_t. In the THz regime, the beam is also scattered as it travels through the atmosphere. Scattering losses are exponential as a function of range, as expressed in terms of an atmospheric attenuation coefficient μ. Finally, additional cosine losses are incurred for large angles of incidence ψ on receiver lens i.e. when receiver is not pointed directly at the transmitter and the signal comes in at an angle. Combining these losses, and treating the transmitter as a far field point source such that the waves can be

assumed to travel as concentric spherical caps, the received code power is expressed as

$$P_R = P_T \cdot G_{pr} \left(e^{-\frac{\mu r}{10} \ln 10} \right) \frac{A_{lens}}{2\pi r^2 (1 - \cos \phi_t)} \cos \psi \ . \tag{15}$$

Signal processing incurs additional losses, as accounted for by G_{pr}. The on-off modulation of the transmitted signal shifts only half of the transmitted power to the baseband. Of the remaining signal power, only a fraction $8/\pi^2$ resides in the fundamental frequency of the transmitted square wave. The rectification process acts like multiplication by a square wave in the time domain (or a convolution in the frequency domain), such that the fundamental frequency power is again spread through the harmonics of a square wave, again reducing power in the fundamental frequency (post-rectification) by $8/\pi^2$. Combining these three multipliers, the net result is a signal processing loss G_{pr} equal to $32/\pi^4$ (or about 0.33).

The code power from the direct signal is balanced against the noise power P_N at the fundamental frequency in equation (14). Background noise comes from environmental radiation and from the receiver electronics. Based on our assumption that the noise is white, the noise spectral density P_{N_0} is essentially uniform across a broad range of frequencies. The total noise power received is the product of the noise spectral density and detector bandwidth B_N, but this power is divided among K discrete frequencies when frequency-domain analysis is applied to K points. Hence the noise in the discrete band associated with the fundamental frequency is P_N, where

$$P_N = \frac{B_N P_{N_0}}{K}.$$

In our experiments, P_C/P_N was determined directly from the data, as described in the results section above. For a known value of code power computed by equation (15), the noise power at each discrete frequency was determined experimentally to be $P_N \approx 1.1 \times 10^{-11}$ W.

2.5.2.2   Multipath Effects

Though multipath was not a factor at short ranges, multipath became important at ranges beyond about 37 m. To understand this sudden onset of multipath, consider the equipment geometry shown in Figure 19. Both the transmitter and receiver have relatively narrow beam angles over which they can transmit or receive signal. Assume that the lenses give each beam pattern a sharp rolloff, going quickly from a maximum value to zero near the edge of the beam. As a result of this assumption, the signal from the transmitter that reaches the ground is essentially zero out to a distance ℓ_t from the transmitter base. The distance ℓ_t depends on the transmitter height H and the beam angle ϕ_t.
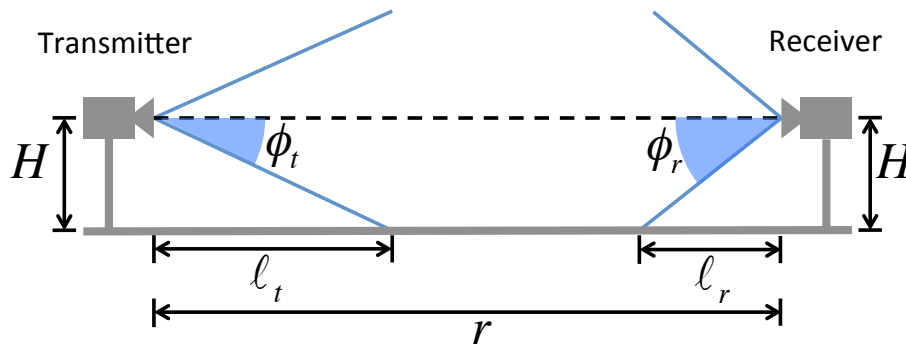
Figure 19 – The transmitter and receiver geometry

Similarly, the closest distance over which the receiver can see a signal from the ground is a distance ℓ_r away from its base. Thus for this model, no indirect

multipath signal is visible at the receiver for equipment separations shorter than the sum $\ell_t + \ell_r$. For distances greater than the sum $\ell_t + \ell_r$, on the other hand, both the direct signal and the indirect multipath signal reflected from the ground reach the receiver. This means that for long distances, the power of the transmitted signal must be modified to account for indirect multipath. The sum $\ell_t + \ell_r$ is approximately 37 m for the parameters of the hardware system used in this work.

The power P_M of the multipath signal is modeled in a similar manner to the direct signal power shown in equation (15), with two key differences. First, the distance r_M that the multipath signal travels is slightly longer than the distance r that the direct signal travels. This difference in path length is small, and therefore results in a negligibly small change in the power. The second difference is that not all of the power will be reflected by the surface, as encompassed by the coefficient of reflectivity Γ^2. The multipath signal power is thus given as

$$P_M = P_T \Gamma^2 G_{pr} \left(e^{-\frac{\mu r_M}{10} \ln 10} \right) \frac{A_{lens}}{2\pi r_{mp}^2 (1 - \cos \phi_t)} \cos \psi \,. \tag{17}$$

Because of the relative infancy of THz technology, the coefficient of reflectivity Γ^2 has not been well characterized for most materials. A study [68] of THz reflections off common (relatively smooth) building materials including glass, plaster, and wood found coefficients of reflectivity approaching 1 as the angle of incidence approached 90° (near grazing). Considering that the angle of incidence is slightly less than 90° in our case and the asphalt surface has roughness on the order of the wavelength (1 mm) making reflections a little more diffuse, we selected a reflectivity slightly less than this ideal value of unity (namely, $\Gamma^2 = 0.9$, a value which agrees reasonably well with our experimental results).

41

In typical radionavigation applications like GPS, the main impact of multipath signals is to bias distance measurements due to the additional path length of the reflected signal. In our case, the maximum additional path length of a multipath signal is about 5.7 cm. This results in a maximum bias of only 2.85 cm, which is significantly less than the clock error of 15 cm, and therefore not a significant contributor to the overall error.

The multipath signal, however, has another effect. Consider the two-ray model [55], [69]. In this model, the primary impact of the multipath signal is to interfere, either constructively or destructively, with the carrier waves of the direct signal. This interference alters the signal strength, either increasing or decreasing the signal-to-noise ratio. This means that depending on the precise geometry of the equipment and the resulting carrier phase difference, the multipath signal can improve or degrade the accuracy of the range measurements.

Because of the short wavelength of the THz carrier signal (1 mm), very small changes in the path length will result in significant changes in the phase difference between the multipath and direct signals. In addition, the point of reflection can vary significantly because the ground surface, an asphalt parking lot, is composed of many smooth surfaces oriented at various angles. Any change in the equipment position due to bumping, wind, or wobbling may result in a significant change in the relative path lengths, causing a transition between constructive and destructive interference. Similarly, sub-mm shifts in the phase center of the transmitting and/or receiving antenna due to thermal effects can also affect the interference pattern. Because of this high sensitivity, we model the amount of interference between the direct and indirect signals as essentially random between two extremes (fully constructive and fully destructive interference). Fully constructive interference results in an effective boost to the signal code power at the receiver, while destructive interference results in an

effective decrease of the signal power. Equation (18) computes these upper and lower bounds, $P_{C,max}$ and $P_{C,min}$, noting that power is amplitude squared

$$P_{C,max} = \left(\sqrt{P_R} + \sqrt{P_M}\right)^2$$

$$P_{C,min} = \left(\sqrt{P_R} - \sqrt{P_M}\right)^2.$$

(18)

 Figure 20 shows the effects of multipath on P_C/P_N. The modeled signal-to-noise ratio for the direct signal (black line) is high at short ranges r but decreases as r increases, since code power is a function of r while thermal noise is not. There are no multipath signals at short range; however, at approximately 37 m, the multipath signals come into play, and constructive and destructive interference are both possible. This is represented by the gray region. If the level of multipath interference is essentially random, then the signal-to-noise ratio can take any value in the gray region. In the constructive interference case (top of the gray region), P_C/P_N increases slightly relative to the direct signal. In the case of destructive interference (bottom of the gray region), P_C/P_N is significantly decreased.
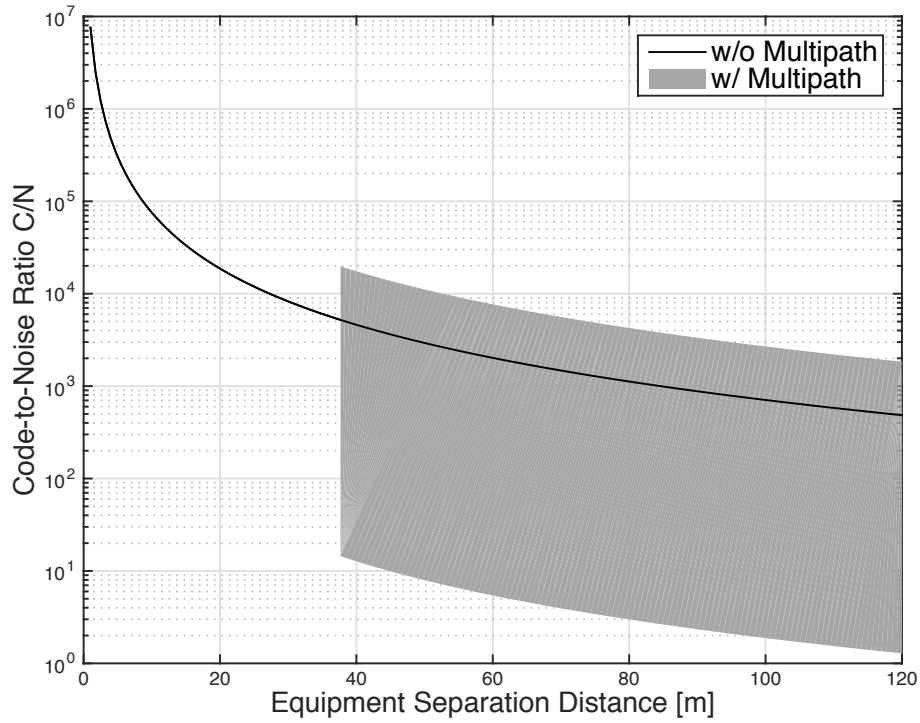
Figure 20 – Signal-to-noise ratio model

The P_C/P_N values plotted in Figure 20 were generated using parameter values listed in Table I. The thermal noise N_0 was determined from experimental data, as described in the Thermal Noise Measurement section above. We did not have experimental data for ground reflectivity ρ, so it was used as a tuning parameter to set the noise level in the static trial. All other parameters were obtained from system specifications or, for parameters describing the natural world, from standard tabulated data.

Table I – THz system parameters

Carrier frequency	$F_c = 0.30$ THz
Carrier wavelength	$\lambda = 1$ mm
Speed of light	$c = 3.0 \times 10^8$ m/s
Atmospheric attenuation	$\mu = 3$ dB/km
Ground Reflectivity	$\Gamma^2 = 0.9$
Processing gain	$G_{pr} = 32/\pi^4$
Receiver gain	$G_R = 6.81$ V/$\sqrt{W}$
Thermal noise	$P_{N_0} = 5.9 \times 10^{-15}$ W/Hz
Noise bandwidth	$B_N = 1$ GHz
Angular frequency	$\omega_{mod} = 2\pi \times 10$ MHz
Transmitted power	$P_T = 30$ mW
Lens area	$A_{lens} = 4.9 \times 10^{-5}$ m^2
Lens width	$w = 7$ mm
Transmitter spreading angle	$\phi_t = 2.5°$
Receiver spreading angle	$\phi_r = 4°$
Equipment height	$H = 1$ m
Buffer length	$K = 450,000$

2.6 Discussion

The primary challenge in adapting GPS ranging techniques to THz frequencies is the current inability of THz hardware to track the signal's carrier phase. As a result, simple on-off keying is used to modulate the carrier signal instead of BPSK, resulting in a power loss as half of the power drops to DC. As THz technology matures, it is expected that carrier phase tracking may become available. In addition to BPSK, this capability would enable carrier phase smoothing methods used in GPS, like Hatch filtering [70]. Because of their short wavelength, THz signals could provide extremely precise range measurements using these methods.

The trials described above demonstrate accurate THz range measurements at the limit of our current test range. In the static case, even with relatively severe multipath interference, measurements of up to 100 m are demonstrated with only a couple meters of error (one-sigma). The significant increase in error in the dynamic trial is likely the result of small bumps and rotations of the transmitter

cart as it was manually pulled backward. These resulted in slight misalignments of the transmitter and receiver equipment. Because of the tight beam patterns, even small misalignments could result in significant decreases in the code-to-noise ratio, and corresponding degradation of the measurement accuracy. The dynamic trials may also be affected by Doppler shift, which was assumed to be zero for this system, or by secondary multipath.

We also note the particular sensitivity of THz signals to multipath interference. When ground-bounce multipath is present, the short wavelength of THz signals combined with the rough asphalt surface of the test range cause the received signal to transition somewhat unpredictably between regions of constructive and destructive interference for even very small changes in equipment configuration. This phenomenon is both a mathematical consequence of our error modeling and a reasonable description of our experimental observations, where measurement errors jump suddenly at the distances where multipath is active (40 m).

The multipath model presented in this paper describes a single reflection from the horizontal ground surface; however, signal reflections from vertical reflectors on the edges of the test range (e.g. cars or sign posts in the parking lot) may have resulted in additional secondary multipath interference. For certain equipment geometries, these additional paths could act to effectively concentrate interference on the receiver. Although efforts were taken to avoid secondary multipath, this may explain the outlier data points observed in both the static and dynamic trials at roughly 50 m separation.

The ground-bounce multipath model presented has a couple of interesting consequences. First, as long as multipath signals are a factor, simply increasing the transmitter strength will not significantly improve the signal-to-noise ratio. As the direct signal strength increases, the multipath signal strength will increase alongside it. Second, the model also suggests that a thoughtful system design

modification (e.g. increasing the height of the equipment, as would be the case for formation flight, or narrowing the transmitter beam) could significantly mitigate multipath. Such mitigation could enable accurate THz ranging over much longer distances than those of our experimental demonstrations. The error models, given by equations (13)-(16), predict that in the absence of multipath interference range measurements over a 1 km baseline would result in measurement errors of only 1.76 m one-sigma. Experimentally verifying this prediction is an important topic for future work.

Other system design features might be adjusted to increase accuracy. A key user defined design parameter is the integrator buffer length K in equations (6) and (7), which can be tuned to mitigate noise. Larger values of K provide more filtering and hence more noise mitigation. In the longer term, enhanced THz electronics could also improve system accuracy, particularly if it were feasible to employ Phase Shift Keying (PSK) and Quadrature Amplitude Modulation (QAM) as is common in radio communications.

2.7   Summary

This paper introduces a novel THz ranging system, modeled on GPS systems, which uses a 300 GHz carrier signal modulated with a simple 10 MHz square wave. Experiments successfully obtained measurements with decimeter accuracy up to 40 m range, and with approximately 10 m accuracy at 100 m range. An error model was developed that well characterized experimental results. The error model suggests that tracking errors due to thermal noise are very low at the ranges tested. For the system described herein, clock sampling errors were dominant below distances of about 40 m, and multipath errors were dominant at longer distances.

2.8　Appendix

This appendix provides a derivation of the range measurement error expression given in equation (14). It begins with a model of the primary sinusoidal component of the continuous time signal detected by the receiver, where P_C is the signal power and $n_v(t)$ is noise

$$v_r = \sqrt{P_C}\cos(\omega_{mod}t + \theta) + n_v(t). \tag{19}$$

The goal is to find the phase shift θ. To do this, the signal is compared to a replica signal $\hat{v}_r$ in the integrators shown in equations (6) and (7),

$$\hat{v}_r = \cos(\omega_{mod}t). \tag{20}$$

Discretizing the signal model and plugging in, the in-phase integrator from equation (6) is given as

$$I_i(k) = \sum_{m=k-K+1}^{k} v_r \hat{v}_r = \tag{21}$$

$$\sum_{m=k-K+1}^{k} \left(\sqrt{P_C}\cos(\omega_{mod}mT_s + \theta) + n_v(m) \right) \cos(\omega_{mod}mT_s).$$

With the help of some trigonometric identities, this expression can be re-arranged as follows,

$$I_i(k) = \sum_{m=k-K+1}^{k} \frac{\sqrt{P_C}}{2} \cos\theta + \sum_{m=k-K+1}^{k} \frac{\sqrt{P_C}}{2} \cos(2\omega_{mod}mT_s + \theta)$$

$$+ \sum_{m=k-K+1}^{k} n_v(m) \cos(\omega_{mod}mT_s). \tag{22}$$

The first term of equation (22) is simple to evaluate. Because the terms of the summation do not depend on the summation variable m, it is the summation of K equal terms and the summation can be replaced by multiplication by K. The second term will be zero, because the buffer size used in this work is an integer multiple of the number of samples per modulation period, leaving

$$I_i(k) = \frac{K\sqrt{P_C}}{2} \cos\theta + \sum_{m=k-K+1}^{k} n_v(m) \cos(\omega_{mod}mT_s). \tag{23}$$

For the final term, the noise n_v must be modeled per Fourier frequency bin. For white noise, the power is uniformly distributed across the bins, so the power in any one bin is P_N. As a result, we have the following discrete time model of the noise per bin $n_k(m)$, where $\tilde{\theta}$ is the random phase of the noise signal

$$n_v(m) = \sqrt{P_N} \cos(\omega_{mod}kT_s + \tilde{\theta}_m). \tag{24}$$

Now, this expression is plugged into equation (23) and again a trigonometric identity is applied to give

$$\sum_{m=k-K+1}^{k} n_m \cos(\omega_{mod} mT_s) =$$

(25)

$$\sqrt{P_N}\left(\sum_{m=k-K+1}^{k} \frac{1}{2}\cos\tilde{\theta} + \sum_{m=k-K+1}^{k} \frac{1}{2}\cos\left(2\omega_{mod} mT_s + \tilde{\theta}\right)\right).$$

Using the same arguments as above, the summation on the first term can be replaced by multiplication by K, and the second summation goes to zero leaving

$$\sum_{m=k-K+1}^{k} n_m \cos(\omega_{mod} mT_s) = \frac{K}{2}\sqrt{P_N}\cos\tilde{\theta}.$$

(26)

The in-phase component of the signal is thus reduced to the equation below. Likewise, the quadrature component is found using the same process except that sine is used in the replica signal instead of cosine (i.e. $\hat{v}_r = \sin(\omega_{mod} t)$)

$$I_i(k) = \frac{K\sqrt{P_C}}{2}\cos\theta + \frac{K}{2}\sqrt{P_N}\cos\tilde{\theta}$$

(27)

$$Q_i(k) = -\frac{K\sqrt{P_C}}{2}\sin\theta - \frac{K}{2}\sqrt{P_N}\sin\tilde{\theta}.$$

To find the error in the phase angle estimate $\hat{\theta}$ relative to the actual angle θ, shift the angle coordinate (without loss of generality) and apply the inverse tangent operation

$$\tan\left(\hat{\theta} - \theta\right) = \frac{\sqrt{P_N}\sin\tilde{\theta}}{\sqrt{P_C} + \sqrt{P_N}\cos\tilde{\theta}}. \tag{28}$$

The geometric interpretation of this is shown graphically in Figure 21.
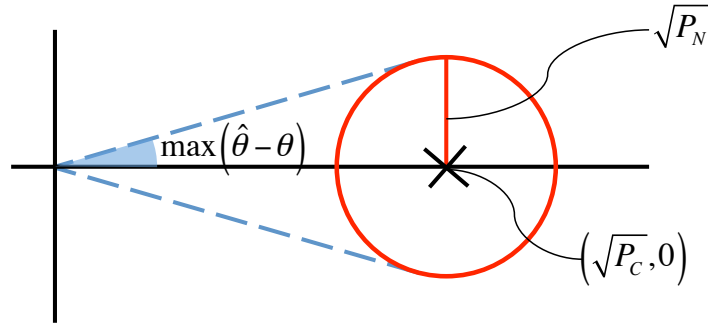
Figure 21 – Geometric interpretation of integrator error

Because white noise is assumed, the unknown phase $\tilde{\theta}$ is evenly distributed and so (28) represents a circle around the point $\left(\sqrt{P_C}, 0\right)$ with radius $\sqrt{P_N}$.

Assuming that $\sqrt{P_C} \gg \sqrt{P_N}$ and making the small angle approximation, the equation can be simplified to

$$\hat{\theta} - \theta = \sqrt{\frac{P_N}{P_C}}\sin\tilde{\theta}. \tag{29}$$

The variance of a uniform distribution $\tilde{\theta}$ mapped through the function $\sin\tilde{\theta}$ is ½. Hence the phase-jitter has variance σ_θ^2, which can be computed as the variance of equation 29,

$$\sigma_\theta^2 = \text{var}\big(\hat{\theta} - \theta\big) = \frac{1}{2}\frac{P_N}{P_C}. \tag{30}$$

Converting from phase-jitter to range measurement error in units of distance requires scaling by the speed of light c and the angular frequency of the modulation signal ω_{mod}

$$\sigma_r = \frac{c}{\omega_{mod}}\sigma_\theta = \frac{c}{\omega_{mod}}\sqrt{\frac{1}{2(P_C/P_N)}}. \tag{31}$$

Chapter 3

Diffraction-Based Interferometer for Angle Measurement[2]

3.1   Introduction and Motivation

As described in Chapter 1, bearing angle measurements are necessary in addition to range measurements to determine aircraft relative positions. Range-only position estimates are subject to a solid angle rotation, as shown in Figure 7. Complementing a THz ranging measurement with a THz bearing measurement (and relative altimeter data) provides a direct means of constructing the relative position vector between a transmitter and receiver; this is particularly important to support minimal or damaged formations, which may consist of as few as two aircraft. THz bearing measurements also benefit larger formations, as the long and narrow geometry of cargo aircraft formations (hundreds of meters wide and several kilometers long) is unfavorable for trilateration.

Angle of arrival measurements are used in location estimation problems for everything from aircraft to automobiles to cell phones [72]–[76]. The precision of angle measurements is a key limiting factor to accuracy in these problems. When projected over a long baseline, even small angle measurement errors can result in large position estimate errors. As a result, very precise angle of arrival measurements are necessary in relative positioning applications.

Various phased array approaches, similar to those used in radio frequency (RF) applications, were examined in previous work [63], [64] for measuring the signal's angle of arrival. The high frequency and relative infancy of THz equipment, however, present some unique challenges that make phased arrays impractical with current THz equipment. First, the equipment is expensive, making it desirable to use as few detectors as possible. Second, currently available THz detector equipment is large relative to the wavelength (detectors are several wavelengths in diameter), making it impossible to tightly pack multiple detectors. Finally, currently affordable detectors are not capable of tracking the phase of the carrier signal, making it impossible to compare the phases from multiple detectors in electronics as is common in RF phased arrays. This chapter, therefore, investigates a novel method for measuring angle of arrival using a movable diffraction grating and a single detector.

To the author's knowledge, this is the first example of a movable diffraction grating being used to make angle of arrival measurements on THz signals. There has been some recent work investigating THz frequency diffraction patterns [77] and diffraction gratings [78], and movable diffraction gratings have been used in other frequency bands to allow a single detector to scan an entire diffraction pattern [79]; however, these two elements have not previously been combined. The combination is useful because, as will be described, the diffraction grating acts in an analogous manner to an RF phased antenna array, inducing interference to find the signal's angle of arrival. This approach is particularly well suited to

THz signals because it allows the device to access otherwise inaccessible carrier phase information and does it using a single detector.

This chapter describes a new method for making angle of arrival measurements in the THz frequency domain, and details efforts to mitigate a key technical challenge, power loss due to the diffraction grating. A novel device is proposed that uses a movable diffraction grating to mimic phased antenna arrays used at RF. The diffraction grating provides spatial observability of the THz carrier phase, since the sample rate of existing hardware is too slow to allow for temporal resolution of the carrier phase. The proposed diffraction-grating system does not provide a direct measurement of relative-bearing between a receiver and a transmitter; instead, the system provides an interference pattern which is indirectly related to bearing angle.

3.2   Brief Design Summary

The proposed system is composed of two primary hardware components, a THz transmitter package and a THz interferometric receiver package, as shown in Figure 22. In this work, it is assumed that the geometry is two-dimensional, in other words that altitude differences are negligible, and position can be determined from range and angle measurements.
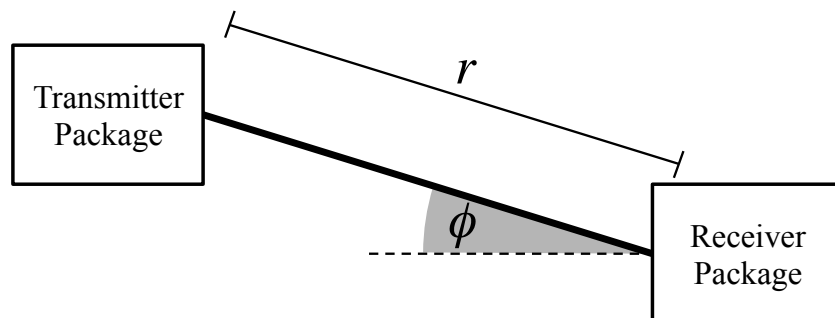
Figure 22 – THz system concept

3.2.1 THz Transmitter Package

The transmitter package is composed solely of the baseline transmitter equipment, discussed in the previous chapter. It broadcasts the THz signals to the receiver package, and can modulate code and data onto the signal for communications and range measurement. The transmitter package is composed of a frequency synthesizer, arbitrary waveform generator, amplifier multiplier chain, and a diagonal horn antenna.

3.2.2 THz Interferometric Receiver Package

The THz receiver package is the novel component of the system, and is assembled from three different elements: a diffraction grating, an actuator, and a detector, as shown in Figure 23.
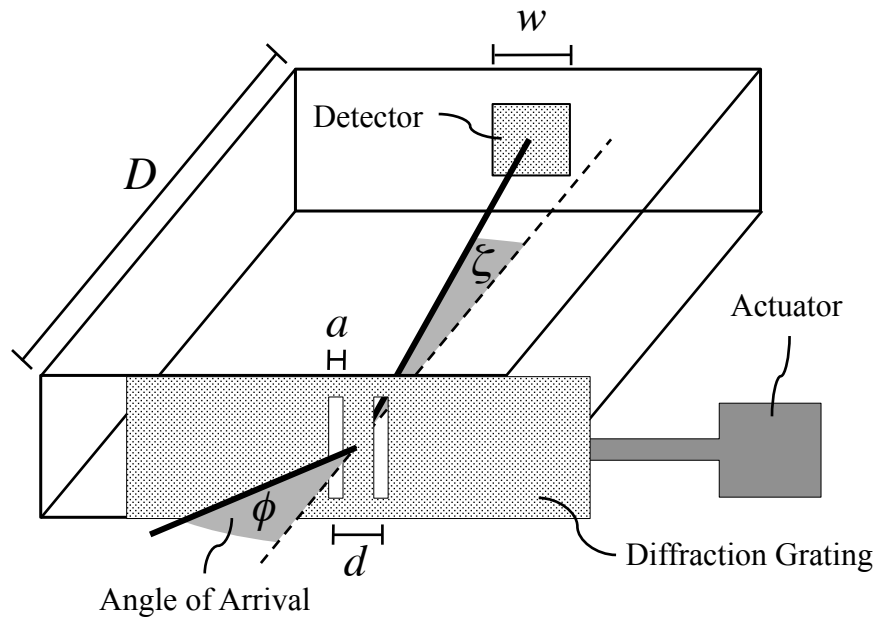
Figure 23 – THz interferometric receiver package with movable diffraction grating

3.2.2.1 Diffraction Grating

The diffraction grating is a thin opaque sheet with optically small, uniformly spaced openings that cause the THz signal to diffract as it passes though. The diffraction causes the signal to interfere with itself inside the receiver package, generating an interference pattern on the back wall. The pattern of interference depends on the size and spacing of the openings, and the signal's angle of incidence on the grating, as described in a following section.

Conceptually, any number of slits could be used. This chapter considers two grating configurations: one with a single slit, and one with two slits. It is possible to use a many-slit grating and that configuration is briefly discussed in the dissertation appendix, but is not the focus of this chapter.

3.2.2.2 Actuator

An actuator is used to sweep the detector through the interference pattern. One logical way to achieve this is by moving the detector back and forth through the pattern, but in this case, it is simpler to move the grating because it is not composed of sensitive electronics and does not require electrical connections, as the detector does. For clarity, Figure 23 illustrates a mechanically actuated diffraction grating, where it is assumed that the grating sits on a track and the actuator slides it from side to side across the front of the receiver package. In practice, a solid state grating generated and modulated on an LCD [80]–[82], metamaterial [83], or electromechanical [84] screen will likely be more effective.

3.2.2.3 Detection Equipment

The detector element, like the transmitter, is the same as the baseline detector equipment from the previous chapter. A single THz detector is used to make measurements of the signal. It is composed of an antenna, an amplifier and

detector, and a data acquisition system. The detector is stationary at the back of the interferometric receiver package, and takes measurements of the THz signal as the grating is swept in front of it.

3.3   Analogy to RF Technology

Two equipment configurations are considered in this paper, one with a single-slit grating and one with a double-slit grating. Analogies can be drawn between these two equipment configurations and existing techniques used in the RF range, namely radar and phased antenna arrays.

In the single-slit configuration, the slit acts basically like a window, blocking the signal when it is not aligned between the transmitter and detector, and allowing it to pass through when it is aligned. As a result, the measured power is highest when the slit is aligned, and low when it is not. The angle, therefore, is found by actuating the grating and identifying the point of highest power. This is analogous to rotating antenna or radar systems used in RF applications. Both are scanning over a range of angles and identifying the point of highest power as the signal's angle of arrival. The only difference is the field of view is limited to less than $180°$ for the single-slit grating.

In the double-slit configuration, an interference pattern is generated on the back wall of the receiver package due to slight differences in the distance the signal travels as it passes through different slits on its way to the detector. Again, an analogy can be drawn to RF technology, this time to phased antenna arrays. In these arrays, a different phase is observed by each antenna in the array due to slight differences in the distance travelled by the signal. The phases are then compared in electronics to determine the angle of arrival. The double-slit configuration is effectively doing the same thing, except the interference is happening in space instead of in electronics.

3.4   Objective and Hypothesis

The goal of this chapter is to identify a receiver package configuration that can achieve continuous or near continuous tracking of multiple transmitters distributed over a wide field of view in order to perform positioning and maintain communications. This is important for the intended application of precision airdrop, where timely, precise, and stealthy relative positioning and communications are necessary between multiple aircraft in a formation.

Two equipment configurations are presented and compared, the single-slit and double-slit configurations. It is hypothesized that the double-slit diffraction pattern will provide spatial aliasing that can be exploited to minimize the grating sweep range, allowing continuous or near continuous tracking of multiple targets with little impact on the power and accuracy.

3.5   Detailed Diffraction Analysis

Diffraction is the flaring or spreading of waves as they pass by obstacles or through narrow openings. It is one of the wave-like properties of photons and its occurrence is described by the theory of quantum electrodynamics (QED) [85]. Diffraction gratings are engineered to induce diffraction, and can take a number of different forms. For this application, a simple grating is assumed, made from a thin sheet of opaque material with one or more thin vertical slits cut out. As photons pass through the slit(s), they diffract, and assuming the signal is coherent, their interference will generate a diffraction pattern on the back wall of the receiver package. Detailed derivations of the patterns can be found in [86]. This section provides a quick summary and presents the key equations.

3.5.1 Single-Slit Diffraction

Figure 24 depicts a simple single-slit diffraction grating configuration, as viewed from above. Photons from the transmitter may take any one of a number of paths through the slit on their way to a point O on the wall, passing through the top, middle or bottom of the opening. Because the path lengths are different for each of these paths, the photons will have different phases when they meet at the detector. This results in interference.
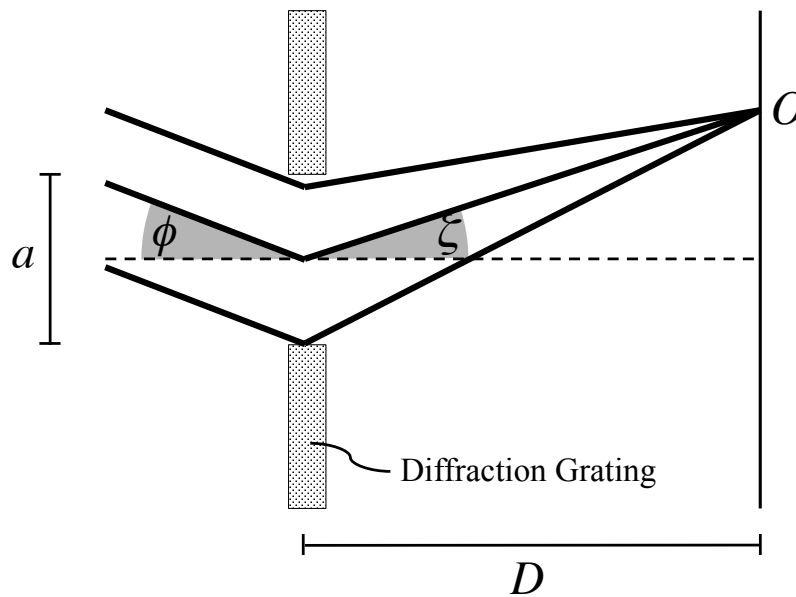
Figure 24 – Single-slit diffraction geometry

In QED, each path and corresponding phase is represented by a phasor, a unit vector that rotates over time. If the distance to the wall is much greater than the size of the opening ($D \gg a$), the various paths from the slit to the detector can be treated as essentially parallel. This drastically simplifies the phasor equations, and by integrating over the width of the slit, the single-slit power distribution is found to be

$$p(\zeta) = p_m \operatorname{sinc}^2 \alpha \tag{32}$$

where p is the power distribution, p_m is the maximum or peak power, and the argument α is defined as

$$\alpha = \frac{\pi a}{\lambda} (\sin \zeta + \sin \phi) \tag{33}$$

where a is the width of the slit, λ is the wavelength of the signal, ζ is the angle from the center of the slit to the point on the wall, and ϕ is the signal's angle of arrival [86]. The cardinal sine is defined here as $\operatorname{sinc}(x) = \sin(x)/x$.

The shape of the single-slit diffraction pattern is shown in Figure 25 for a few different values of the ratio a/λ (an angle of arrival $\phi = 0°$ is assumed). The pattern is composed primarily of one central peak. Technically there are some small local maximums in the tails of the pattern, but they are so small relative to the central peak that they are not particularly useful and can be ignored. The width of the central peak varies from roughly $30°$ to greater than $180°$ in the figure with the slit width a. Narrow slits result in greater diffraction of the signal and a broader central peak. Wide slits result in less diffraction and a narrower central peak. The ratio a/λ, therefore, is the key parameter controlling the width of the peak.
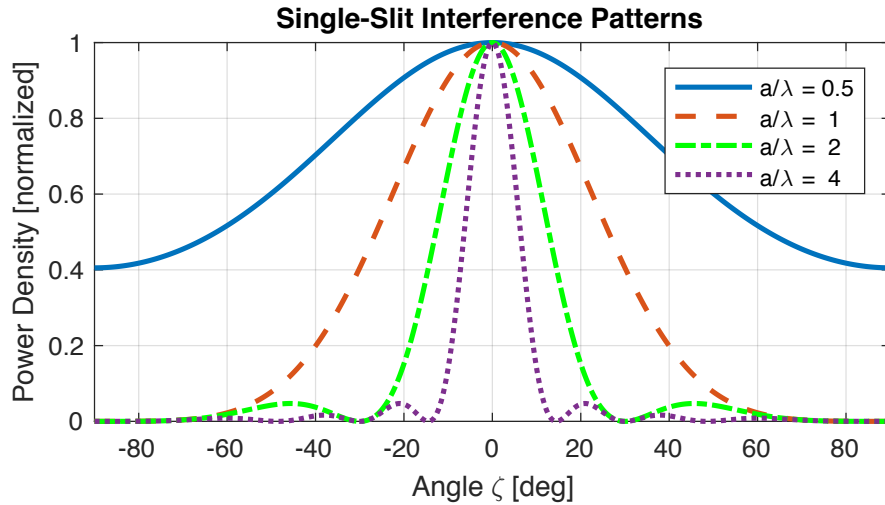
Figure 25 – Example single-slit diffraction patterns

It is important to note that the peak power p_m is normalized in Figure 25 to highlight the difference in the width of the peaks. In reality, the peak power is also a function of the slit width a. Narrow slits allow less power through than wide ones, resulting in a reduction in the peak power. In addition, they spread the power out over a wider area. As a result, the peak power of a narrow slit configuration is significantly less than the peak power of a wide slit configuration.

3.5.2 Multi-Slit Diffraction

Figure 26 depicts a simple multi-slit diffraction grating configuration, as viewed from above. In the particular case shown there are three slits ($N = 3$), but the analysis below is generic and may be applied to a grating with an arbitrary number of slits.
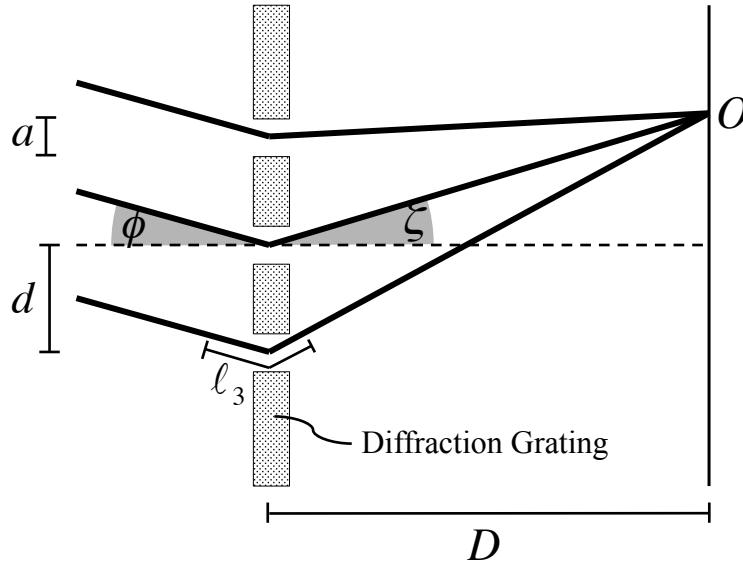
Figure 26 – Multi-slit diffraction geometry

Similar to the single-slit case, there are multiple paths a photon may take to get from the transmitter to a point O on the wall. In addition to the infinite number of paths contained within any one individual slit, there are multiple slits that a photon may pass through, and each of these slits has a different path length associated with it, resulting in a phase shift and interference. Like before, different paths can be represented by phasors. For the multi-slit analysis, however, there are a finite number of slits, so a summation is used instead of an integral. The complete interference pattern thus can be written as

$$p(\zeta) = p_m \, \mathrm{sinc}^2 \, \alpha \left\| \frac{1}{N} \sum_{n=1}^{N} \mathbf{z}_i(\zeta, \phi) \right\|^2 \tag{34}$$

where N is the number of slits, i is an index used to label the slits sequentially from 1 to N, and $\mathbf{z}_i$ is the phasor associated with the ith slit. The phasor $\mathbf{z}_i$ is defined as

$$\mathbf{z}_i = \begin{bmatrix} \cos(2\pi\ell_i(\zeta,\phi)/\lambda) \\ \sin(2\pi\ell_i(\zeta,\phi)/\lambda) \end{bmatrix} \tag{35}$$

where $\ell_i(\zeta,\phi)$ is the additional distance the photon needs to travel to pass though the ith slit. As an example, ℓ_3 is shown in Figure 26. Here, the top slit has been labeled $i = 1$ with each subsequent slit iterating by one. The additional path length for the first slit thus is $\ell_1 = 0$. If it is again assumed that the distance to the wall is large relative to the diffraction grating area $(D \gg (N-1)d + a)$, the paths can be assumed to be parallel, and the path difference ℓ_i is defined as

$$\ell_i = d(i-1)(\sin\zeta + \sin\phi) \tag{36}$$

where d is the spacing between the slits.

Equation (34) is composed of two parts multiplied together. The first portion represents the contribution of single-slit diffraction, and is equivalent to Equation (32). The second portion represents the role of multi-slit diffraction and comes from the phasor analysis.

The single-slit pattern described by Equation (32) is a special case of Equation (34). When there is only one slit $(N = 1)$, the summation in the second term is dropped and the term becomes the norm of a unit vector, which is one. This leaves only the first term.

3.5.3 Double-Slit Diffraction

The double-slit configuration $(N = 2)$ is also a special case of Equation (34). In this case, the summation of phasors can be simplified [86], and the equation becomes

$$p(\zeta) = p_m \operatorname{sinc}^2 \alpha \cos^2 \beta \qquad (37)$$

where the argument β is defined as

$$\beta = \frac{\pi d}{\lambda}(\sin \zeta + \sin \phi) \qquad (38)$$

Figure 27 shows an example of a double-slit diffraction pattern broken down into its two components: the single-slit interference term and the double-slit interference term. The parameters a/λ and d/λ have been arbitrarily set here to 1 and 2, respectively, to make the figure clear. The single-slit term (the first term of Equation (37)) is shown as a green dash-dot line, the double-slit term (the second term of Equation (37)) is shown as a red dashed line, and the complete pattern, which is composed of the two terms multiplied together, is shown as a solid blue line.
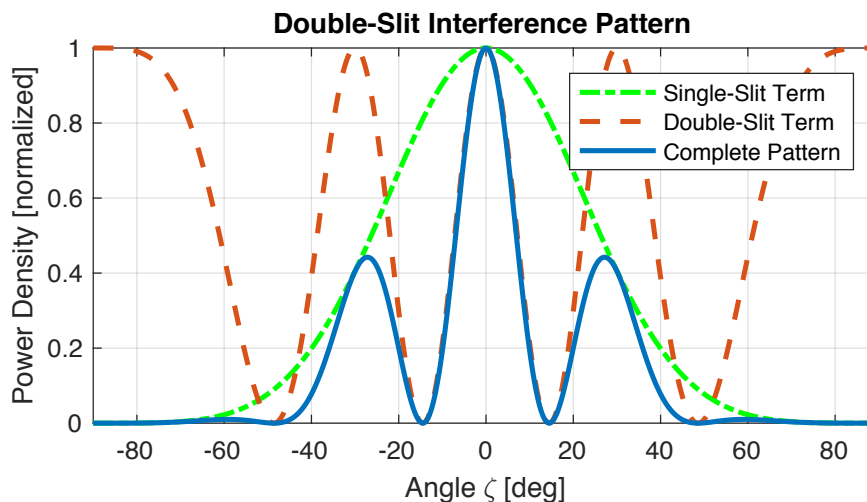
Figure 27 – Example composition of double-slit pattern

The double-slit term is a cosine function, and this gives the pattern its multiple peaks. The parameter d/λ controls the spacing of the peaks. Large slit spacing d

65

results in a pattern with many closely spaced peaks, and small slit spacing d results in a pattern with only a few peaks.

The single-slit term is a cardinal sine function, and this gives the pattern its bell shape. As described above, the width of the bell is controlled by the parameter a/λ. Small values of the slit width a result in a wide bell, and large values of the slit width a result in a narrow bell.

The complete double-slit pattern is the two terms multiplied together. The key difference to notice between the double-slit and single-slit patterns is the presence of multiple peaks in the double-slit pattern. These peaks can serve as additional markers for identifying the angle of incidence ϕ. This is the advantage of the double-slit pattern, and, as described below, can be utilized to allow the device to maintain a large field of view while scanning only a small subset of angles.

It is important to note the role of the single-slit term (the bell curve term) in limiting the width of the pattern. The tail of the bell suppresses side peaks in the double-slit pattern, as shown in Figure 27. The width of the pattern, therefore, is controlled by the slit width a, a fact that will be important in the discussion of field of view below.

3.5.4　Detector Modeling

Equations (32), (34), and (37) describe the power density (or intensity, measured in Watts per radian) of the signal at a point O along the back wall. The detector, however, has width, as shown in Figure 28 (not to scale) and as a result integrates the power density over a swath of internal angles ζ.
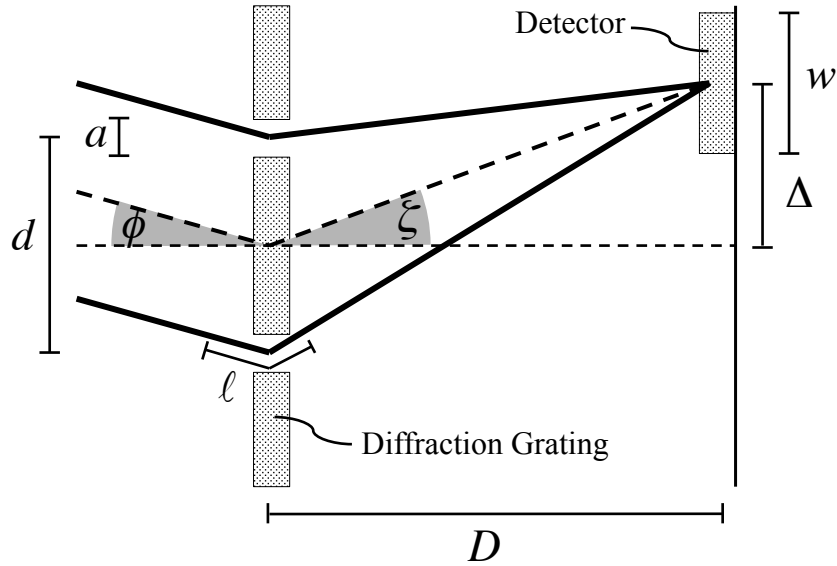
Figure 28 – Double-slit diffraction geometry with detector

The internal angle ζ is indirectly related to the lateral translation Δ of the diffraction grating. Specifically, the translation required to obtain a particular angle ζ is

$$\Delta = D \tan \zeta . \tag{39}$$

To find the power incident on the detector, therefore, the width of the detector w must be taken into account by integrating the intensity. For a given grating position Δ, the detector spans between angles $\zeta_{det,+}$ and $\zeta_{det,-}$

$$\zeta_{det,\pm} = \tan^{-1}\left(\frac{\Delta \pm w/2}{D}\right). \tag{40}$$

The signal power at the receiver P_R (measured in Watts) can thus be found by integrating the intensity p (measured in Watts per radian) from equation (32), (34), or (37) between these limits

$$P_R(\Delta, \phi) = \int_{\zeta_{det,-}}^{\zeta_{det,+}} p(\zeta, \phi) \, \delta\zeta \, . \tag{41}$$

The power P_R incident on the receiver is a function of the grating position Δ, via the integration limits and the angle of incidence ϕ, via the power density p. The primary impact of the integration is to smooth the pattern. In the single-slit case, this results in a slight broadening of the central peak seen in Figure 25. In the double-slit case, this lowers the peaks and lifts the valleys seen in Figure 27. The amount of smoothing depends on the width of the detector w, with a narrow detector resulting in minimal smoothing and a wide detector resulting in significant smoothing.

3.5.5   Power Link Budget

The power incident on each slit of the diffraction grating is calculated as

$$P_{slit} = P_T \times 10^{-r\mu/10} \times \frac{A_{slit}}{2\pi r^2 (1 - \cos\phi_t)} \times \cos\psi \, , \tag{42}$$

which comes from the link budget given in equation (15) from Chapter 2 with the processing gain term removed, and the area set to the relevant area of the slit. The angle of incidence ψ differs slightly from the planar bearing angle ϕ. When the aircraft fly at the same altitude, the angle of incidence ψ on the grating is equivalent to the planar bearing angle ϕ, but in general, ψ contains a vertical component in addition to the planar component, whereas ϕ lies exclusively in the plane. It is assumed that the slits are very tall, such that any diffraction in the vertical direction is negligible, and the rays can be assumed to only diffract in the horizontal direction. As such, the relevant slit area is defined as the slit width

times the detector height $A_{slit} = aw$, where the detector height is equal to the detector width w.

The peak power density p_m is calculated by dividing the total power through the slits by the integral of the normalized power distribution. The peak power is thus

$$p_m = \frac{NP_{slit}}{\int_{-\pi/2}^{\pi/2} \frac{p(\zeta,\phi)}{p_m} d\zeta} \ . \tag{43}$$

where the integral on the bottom is the same as the integral in equation (48) except for the integration limits.

3.6   Design Considerations

Design decisions were made to achieve the objective: near continuous tracking of multiple signals from transmitters distributed over a wide field of view. Specifically, the slit width a, slit spacing d, and distance to the detector D in the interferometric receiver package were set based on considerations of the field of view, minimum grating sweep range, and power.

3.6.1   Field of View and Slit Width a

A wide field of view is necessary to observe multiple transmitters spread over a wide area, as in formation flight. The field of view of the receiver package is fundamentally limited to less than $180°$ by the geometry, and in practice will be further limited because very little power will reach the detector at high angles of incidence. As a result, a reasonable target field of view for this work is $90°$.

In the single-slit configuration the field of view is determined by the sweep of the grating. Because the pattern is composed of one main peak, the angle can only be

observed if the peak is observed. This means that the receiver package's field of view is equivalent to the range of angles swept by the grating.

In the double-slit configuration, on the other hand, there are multiple peaks in the pattern. Because these additional peaks can also serve as markers in the pattern, the angle can be observed even if the central peak lies outside of the range of angles swept by the grating, assuming the ambiguity can be resolved. As a result, the field of view in the double-slit case is related to the width of the pattern and the number of observable side peaks. A wide pattern with multiple side peaks allows a wide field of view, while a narrow pattern with only a few side peaks limits the field of view. Thus, the key design parameter is the width of the slits $a_{N=2}$. As described above and shown in Figure 27, narrow slits result in more diffraction and a wide field of view, and wide slits result in minimal diffraction and a narrow field of view. The relationship between the slit width $a_{N=2}$ and the field of view is given by

$$a_{N=2} = \frac{\lambda}{\sin \zeta_{FOV}}, \tag{44}$$

where ζ_{FOV} is the location of the first dark fringe in the cardinal sine (sinc) term of the double-slit pattern from Equation (37).

Because the desired field of view is $90°$, the angle ζ_{FOV} is set to $70°$ in this case, which, accounting for the roll off towards the tail, comfortably provides a field of view of roughly $\pm45°$ to either side of center. From Equation (44), this gives a slit width of $a_{N=2} \approx 1.06\lambda$. Given that the wavelength λ of THz signals is on the order of 1 mm, this yields a slit size that is practical and easy to generate.

3.6.2 Minimum Grating Sweep and Slit Spacing d

Continuous or near continuous tracking of the signal is necessary in the precision airdrop application to allow for uninterrupted positioning and communications. As a result, it is desirable to utilize a small grating sweep range to minimize the amount of time spent scanning dead space in the pattern.

In the single-slit configuration, the lone peak in the pattern can only be observed if it lies within the sweep area. This means that the grating sweep must cover the entire field of view to observe the signal's angle of arrival. Because the desired field of view is 90°, this means that the grating sweep must span $\zeta_{lim} = \pm 45°$ to either side of center.

In the double-slit configuration, on the other hand, the presence of multiple peaks in the pattern allows the angle to be determined even if the central peak lies outside the range of the grating sweep. As long as at least one peak is observed over the sweep, the signal's angle of arrival can be estimated, assuming the ambiguity can be resolved. The minimum range that the grating must sweep, therefore, depends on the spacing of the peaks, with closely packed peaks allowing for a small grating sweep, and sparse peaks requiring a large grating sweep. An excessively small peak spacing, however, can make the ambiguity resolution challenging, so in this case the desired minimum grating sweep was set to one fifth the field of view, or 18°. The grating sweep, therefore, covers the subset of angles between $\pm\zeta_{lim}$, where the limiting angle $\zeta_{lim} = 18°$, by actuating the grating position $\varDelta$ is between two extremes

$$-D \tan \zeta_{lim} < \varDelta < D \tan \zeta_{lim} \, , \tag{45}$$

The peak spacing in the double-slit configuration, and therefore the minimum sweep range, is governed by the slit spacing $d_{N=2}$. A small slit spacing results in

large peak spacing, and a large slit spacing results in small peak spacing. The relationship between the slit spacing $d_{N=2}$ and the peak spacing is given by

$$d_{N=2} = \frac{\lambda}{\sin \zeta_{PS}}, \tag{46}$$

where ζ_{PS} is the location of the peak next to the central peak in the double-slit pattern.

To achieve the desired minimum grating sweep of 18°, the angle ζ_{PS} is set to 15°, leaving 3° buffer to account for the non-linearity in Equation (37). From Equation (46), this gives a slit spacing of $d_{N=2} \approx 3.86\lambda$. Again, given that the wavelength λ of THz signals is roughly 1 mm, this corresponds to an easily attainable slit spacing.

3.6.3   Power and Detector Distance D

After passing through the grating, the THz signals are diffracted and spread out as they travel toward the back wall of the receiver package. To minimize spreading power losses, it is desirable to position the detector as close to the grating as possible; however, as mentioned above, the interference pattern equations assume that the distance to the detector is much greater than the size of the diffracting area ($D \gg (N-1)d + a$). As a result, the detector distance D is set to 10 times the size of the diffracting area,

$$D = 10\big((N-1)d + a\big). \tag{47}$$

In the double-slit case, based on the parameters defined above, equation (47) gives a detector distance of $D_{N=2} = 49.3\lambda$. For THz signals this corresponds to a receiver package that is roughly 5 cm in size.

In the single-slit case, to make a fair comparison, the receiver package is assumed to be the same size as in the double-slit configuration ($D_{N=2} = D_{N=1} = D = 49.3\lambda$). To maximize the power then, the slit width $a_{N=1}$ is made as large as possible while still maintaining the assumption that the distance to the detector is much greater than the size of the diffracting area ($D \gg a$). Using Equation (47) the maximum acceptable slit width is found to be $a_{N=1} = 4.93\lambda$.

Table II – Diffraction grating design parameters

Double-slit	Slit width	$a_{N=2} = 1.06\lambda$
	Slit spacing	$a_{N=2} = 3.86\lambda$
	Sweep width	$\zeta_{lim,N=2} = 9°$
Single-slit	Slit width	$a_{N=1} = 4.93\lambda$
	Sweep width	$\zeta_{lim,N=1} = 45°$
Both	Detector distance	$D = 49.3\lambda$

3.7   Simulations Overview

Simulations were performed in MATLAB to compare the single-slit and double-slit diffraction grating configurations and determine whether either can achieve the objective laid out in this paper. Ultimately, the goal is to extract the signal's angle of arrival from measurements of the pattern. The following simulations explore the shapes of the measured patterns as a first step toward this and provide some preliminary insights into how this might be achieved. The simulations presented here do not consider noise, as they are only focused on the shapes of the patterns. A brief overview of the simulations is presented in this section.

3.7.1   Assumptions

It is assumed that the baseline transmitter and receiver hardware is the THz equipment from the previous chapter, with specification given in Table I

Table III summarizes assumptions about the locations of the equipment and corresponding physical constants. It is assumed that the transmitter and receiver

packages are both mounted on aircraft flying in formation at an altitude H of 10 km and a separation distance r of 1 km. Note that the attenuation coefficient μ of the THz signal at that altitude is 3×10^{-3} dB/km [26], three orders of magnitude less than in the ground level tests described above.

Table III – Simulation assumptions

Altitude H	10 km
Measurement range r	1 km
Attenuation coefficient μ	3×10^{-3} dB/km

3.7.2　Pattern Simulations

First, two simulations model the power distribution on the back wall of the receiver package for both the single-slit and double-slit case (without taking into account the effects of the detector). These come directly from Equation (32) and Equation (37). The first simulation shows the case where the angle of arrival ϕ is $0°$, and the second simulation shows the case where the angle of arrival ϕ is $20°$. The peak power density p_m is calculated from equation (43).

3.7.3　Measurement Simulations

Next, the power incident on the detector (accounting for its width) is simulated. This is calculated from equation (49), using numerical integration to calculate the diffraction grating gain term G_D. Again, the first simulation shows the case where the angle of arrival ϕ is $0°$ and the second simulation shows the case where the angle ϕ is $20°$.

In both the single-slit and double-slit case, it is assumed that the grating moves at a constant linear speed and completes one pass in 0.09 s. From the hardware specifications, this results in a total of 200 integrator measurements per sweep between $\zeta_{lim,N=1} \pm 45°$ in the single-slit case. Because the double-slit sweep covers one-fifth the distance (stretching between $\zeta_{lim} \pm 9°$), the same angular

resolution can be achieved with one-fifth the number measurements (i.e. 40 measurements per sweep). This makes it possible to integrate the signal five times longer in the double-slit case, resulting in a five-fold boost in the power. As a result, the power calculated from equation (49) is multiplied by 5 in the double-slit simulation.

In both cases, it is assumed that any smoothing or smearing of the pattern due to the motion of the grating during integrator measurements is negligible.

3.8   Results

Results from the THz interferometer simulations comparing the single-slit and double-slit configurations relative to the objective are presented in this section.

3.8.1   Pattern for $\phi = 0°$

Figure 29 shows the interference pattern simulation results for the case where the angle of arrival ϕ is $0°$. The power density incident on the back wall is plotted as a function of the internal angle ζ. Note that this is a power distribution and has units of Watts per radian. The power density for the single-slit pattern, shown as a solid blue line, is composed of a single central peak, spanning roughly $11°$ to either side of center, with a high maximum power density. The double-slit power density pattern, shown as a red dashed line, on the other hand, is composed of five similar-sized peaks, separated laterally by roughly $15°$, with low peak power.
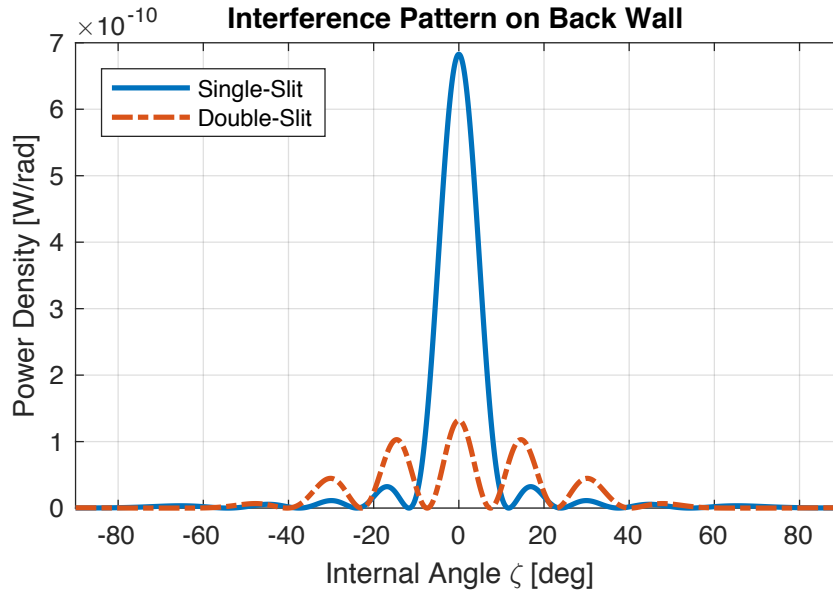
Figure 29 – Single and double-slit power distribution for $\phi = 0°$

3.8.2   Pattern for $\phi = 20°$

Figure 30 shows the interference pattern simulation results for the case where the angle of arrival ϕ is $20°$. The power density, measured in Watts per degree ζ, is again shown as a function of the internal angle ζ. The single-slit pattern is shown as a solid blue line and the double-slit pattern is shown as a dashed red line. The patterns here are almost identical to the patterns in Figure 29, except shifted by $20°$ to the right. The other difference to note is the slight asymmetry of the patterns around the central peak. Both patterns appear slightly stretched on the right side, with larger peak separations than those on the left side. This is due to the nonlinear argument $\sin \zeta$ that appears inside the terms of Equation (32) and Equation (37).
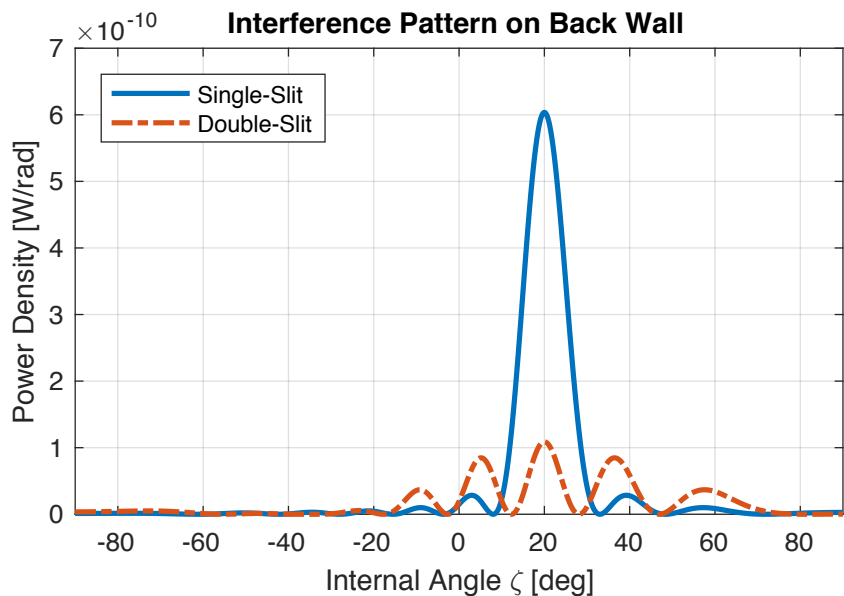
Figure 30 – Single and double-slit power distribution for $\phi = 20°$

3.8.3 Measurements for $\phi = 0°$

Figure 31 shows the pattern of power incident on the receiver when the signal's angle of arrival ϕ is $0°$. Note that this plot shows power in Watts, not power density measured in Watts per radian, as above. It is a subtle distinction, but these are incident power measurements made by the detector, as opposed to the pattern of intensity along the back wall. The measurements of the single-slit pattern, shown as a solid blue line, span $\pm 45°$, the range of angles swept by the grating in that case. The measurements of the double-slit pattern, shown as a dashed red line, span a smaller range between $\pm 9°$, because, as discussed above, only this range needs to be swept in the double-slit case. The decrease in sweep range allows longer integration times, which effectively boost the power, making the two peaks similar in size. In the single-slit case, the peak power measurement is roughly 45% of what it would be if there were no grating, and in the double-slit case it is roughly 41%.
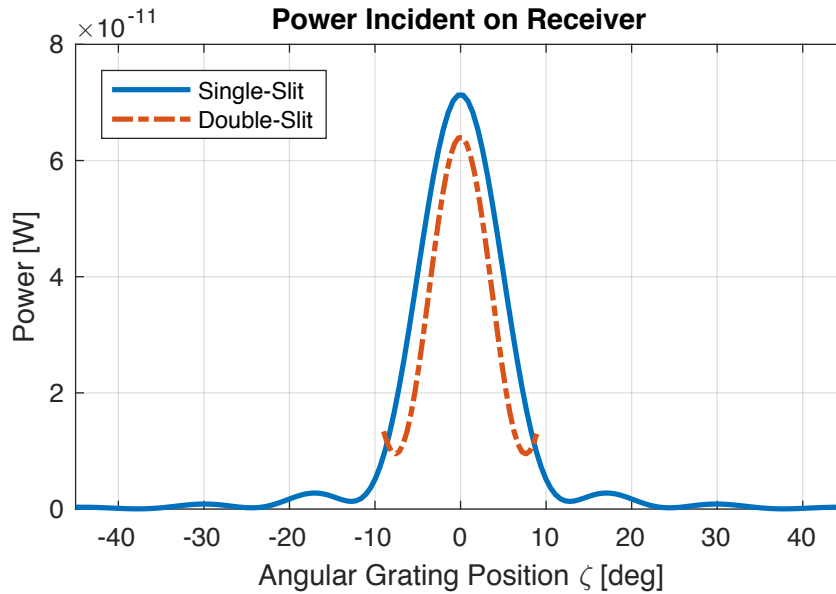
Figure 31 – Single and double-slit power incident on the receiver for $\phi = 0°$

3.8.4   Measurements for $\phi = 20°$

Figure 32 shows the measured interference patterns for the case where the signal's angle of arrival ϕ is 20°. Like the previous plot, the single-slit measurements, shown as a solid blue line, span $\pm 45°$, while the double-slit measurements, shown as a red dashed line, span $\pm 9°$. The main difference to note between this simulation and the one shown in Figure 31 is that the central peak in both cases as shifted to the right. In the single-slit case, the central peak is still visible, but in the double-slit case it has moved outside the window scanned by the grating. Despite this, a peak is still visible in the double-slit pattern, the first side peak. The single-slit peak is roughly 39% of what would be measured with no grating, and the double-slit peak is roughly 28%.
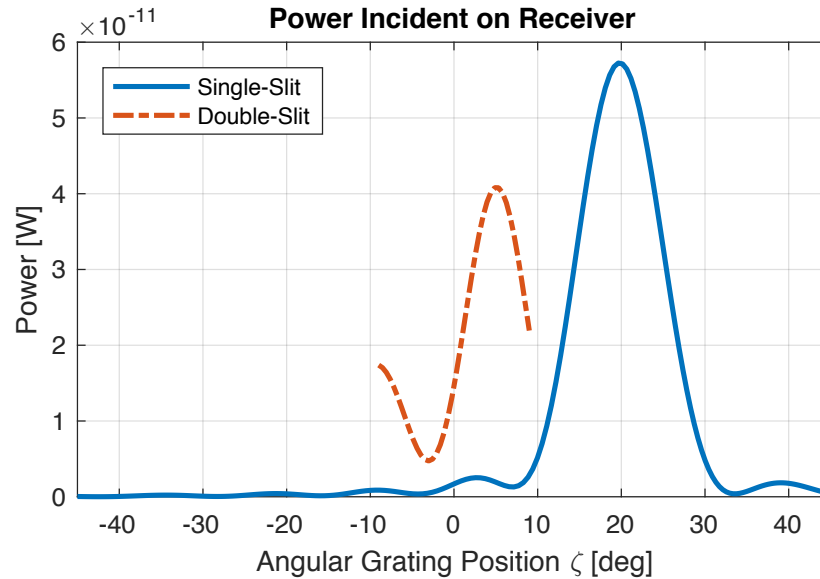
Figure 32 – Single and double-slit power incident on the receiver for $\phi = 20°$

3.9   Discussion

A number of differences between the single-slit and double-slit configurations are evident in the results shown above, which suggest that the double-slit configuration is capable of providing continuous or near continuous tracking of multiple transmitters distributed over a wide field of view, as is necessary for the precision airdrop application.

3.9.1   Continuous Tracking

Continuous or near continuous signal tracking is important for uninterrupted communications and precise positioning. For communications, long outages significantly increase the risk of missed data bits and to avoid this, messages must be repeated many times, significantly slowing communication speeds. For positioning, frequent extended outages can limit the integration and correlation times used to make accurate measurements and filter noise, resulting in imprecise measurements.

The simulations above show that the single-slit pattern has large regions of dead space, where the power is nearly zero. Because it has only one peak, the grating must scan the entire field of view to observe the angle of arrival. As a result, continuous tracking and a wide field of view cannot simultaneously be achieved in the single-slit case. Conceptually, a peak-following control algorithm could be implemented to keep the grating locked onto the peak, but this presents a problem for multiple access scenarios, as discussed below.

The double-slit pattern, on the other hand, is composed of multiple peaks and therefore has minimal dead space within the field of view. Because the peaks are repeated (spatial aliasing), only a small region of the pattern needs to be scanned to maintain a large field of view, assuming that the ambiguity can be resolved, perhaps through initialization. From the measurement simulations in Figure 31 and Figure 32, it can be seen that this results in a large portion of the scan time being spent at or near peak power, allowing near continuous tracking of the signal. Furthermore, because of the closely packed peaks and the averaging effect of the detector width, even the valleys in the measured pattern are not compete dead zones; they have some power. As a result, fully continuous tracking may be possible depending on the power threshold of the system.

3.9.2 Multiple Access

The ability to simultaneously track signals from multiple transmitters at different locations is essential for multiple access communications and positioning. This is important for formation flight applications, where it is necessary to locate and communicate with a number of aircraft scattered across a wide field of view.

Simulations show that in the single-slit case, the entire field of view must be swept by the grating to identify the angle of arrival, resulting a significant amount of time scanning dead space. Conceptually, the device could lock onto the peak

and follow it, allowing continuous signal tracking of one signal; however, in a multiple access scenario where multiple transmitters are scattered at various angles, this would necessarily reject the other signals. As a result, the single-slit configuration cannot simultaneously achieve continuous signal tracking and a wide field of view for multiple access scenarios.

In contrast, the spatial aliasing of peaks in the double-slit pattern means that signals coming from different directions may have peaks that overlap. Simulations show the signal can be nearly continuously tracked using a narrow grating sweep, even if the central peak in the pattern lies outside the range of the sweep. As a result, multiple signals can be simultaneously tracked over a wide field of view with minimal signal interruption in the double-slit configuration.

3.9.3   Power

In order to maximize the accuracy of measurements and ensure reliable communications, it is necessary to maximize the power of the received signal.

The pattern simulations above clearly show that the single-slit configuration provides significantly higher power densities than the double-slit configuration. The difference in power density between the two cases in Figure 29 and Figure 30 is two-fold: first, because the double-slit grating blocks more of the signal, and second, because it spreads the power over multiple peaks.

In the measurement simulations, however, the power difference is almost completely eliminated by the increased integration time in the double-slit configuration granted by the shorter sweep area, as shown in Figure 31 and Figure 32. Because the grating covers one fifth the number of points, it can integrate power five times longer, making the received signal powers almost equal in the single-slit and double-slit cases. The trade-off, though, is noise. In addition to increased signal power, the longer integration time also means increased noise

power. This means that while the raw signal power is quintupled by the increased integration time, the signal-to-noise ratio is only improved by a factor of $\sqrt{5}$.

It should be noted that both the single-slit and double-slit gratings result in a greater than 50% decrease in the received power as compared to no grating at all. This is to be expected since both gratings block some portion of the signal headed towards the receiver and also cause it to diffract and spread out.

3.9.4 Device Size

For aerodynamic reasons, the size of the receiver package is a major concern, particularly its cross-section. Since it is mounted on the nose of the aircraft, it must be minimally protruding. From the parameters given in Table II, the device dimensions are on the order of 5 cm, giving a cross-section area of roughly 25 cm^2, a tiny fraction of the cross-section of a cargo aircraft.

3.10 Conclusions

The THz double-slit interferometer provides a compelling option for relative positioning during precision airdrop operations. The compact interferometer design presented in this chapter allows angle measurements to be made on the THz signal using currently available and affordable equipment.

Simulations show that the double-slit configuration allows near continuous tracking of multiple transmitters distributed over a wide field of view, while the single-slit configuration can only achieve either continuous tracking or a wide field of view. In addition, because of the increased integration time in the double-slit configuration, the signal-to-noise ratio is improved, largely making up for the apparent difference in power between the two cases.

For the precision airdrop application, where high precision angle measurements are necessary and high data rate communications may be advantageous, the double-slit THz interferometer therefore is the better choice.

Chapter 4

Snapshot Bearing Angle Estimation from THz Interferometer Measurements[3]

4.1   Introduction and Motivation

The preceding chapter introduced a novel concept for THz angle of arrival sensor. This device uses a moving diffraction grating to convert the time-varying carrier signal into a spatial pattern, which can be probed by the detector. This allows the receiver to access otherwise inaccessible carrier phase information for angle determination.

In the previous chapter, equations were developed that give the interference pattern for a given bearing angle. The technical challenge here is solving the inverse problem, going from measurements of the interference pattern to bearing angle estimates. The challenges of this problem are three-fold. First, the equations are not readily inverted and numerical methods must be used. Second, because the received power is not known ahead of time, individual measurements do not provide any information about the bearing angle, and a time-series as the

---

[3] This chapter was published under the title "Relative Position Estimates from Terahertz Observables" in the Proceedings of the 2017 International Technical Meeting of the Institute of Navigation (ION ITM 2017) [87]. Some changes and restructuring of the text have been made for clarity and flow.

diffraction grating sweeps in front of the detector is needed. Finally, the spatial aliasing of the peaks produces ambiguity in the pattern.

This chapter presents a method for solving the inverse problem to map noisy signal measurements from the THz interferometer to the bearing angle. A least squares (LS) algorithm is applied to batches of signal measurements to produce bearing angle estimates. Simultaneous range estimation, using the methods described in Chapter 2, provides the basis for two-dimensional position estimation. Analysis demonstrates the capabilities and provides and initial estimate of bearing angle measurement accuracy.

4.2   Bearing Angle Requirements

As mentioned above in Chapter 1, the key positioning requirement for the airdrop application is the cross-track requirement (50 m, two-sigma). Because bearing angle errors project primarily in the cross-track direction, precise bearing angle measurements are necessary to meet this requirement. At 2 km, the length of a typical formation element, this necessitates bearing angle accuracy of 1.44°, two-sigma, or better.

4.3   Relating Bearing Angle to Interference Peaks

The diffraction grating on the front of the THz interferometric receiver generates an interference pattern of bright and dark lines on the back wall of the package, as described in Chapter 3. By identifying the locations of these lines, the bearing angle ϕ can be inferred.

4.3.1   Interference Pattern

The interference pattern for the double-slit configuration is described by the power density given in Chapter 3, equation (37). In order to visualize the pattern, it is useful to consider an example of the signal power arriving at the backplane

after passing through the grating (where $a/\lambda = 1.06$, $d/\lambda = 3.86$). The right-hand side of Figure 33 plots power as a function of the internal angle ζ for three different angles-of-arrival ϕ, {0°, -10°, -20°}. The left-hand side depicts the corresponding aircraft formations (not to scale). The most salient feature of each interference pattern is the largest peak, which shifts along the backplane as the relative bearing angle between the transmitter and receiver changes.
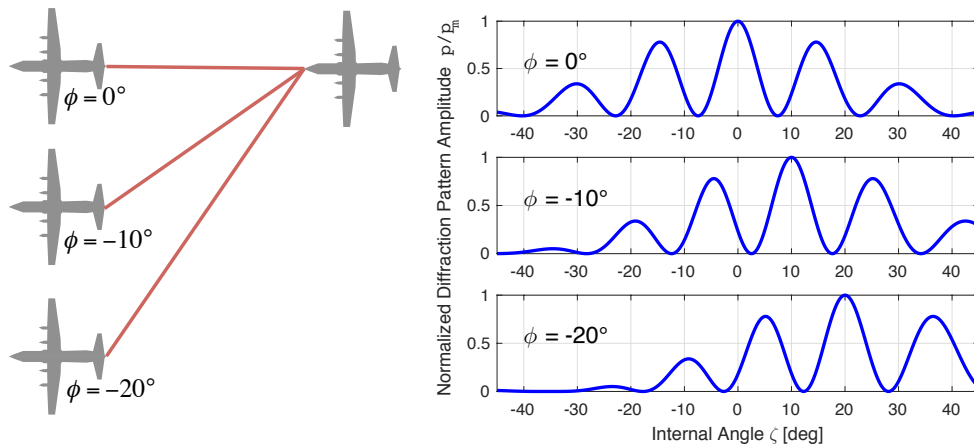
Figure 33 – Example aircraft formations and corresponding double-slit diffraction patterns

As described in Chapter 3, the pattern is probed by sweeping the diffraction grating in front of the detector. The grating sweeps a limited set of angles between $\pm\zeta_{lim}$, where $\zeta_{lim} = 9°$ as described in equation (45) and Table II, and, therefore, only observes a limited portion of the pattern, resulting in ambiguity.

4.3.2 Integer Ambiguity

In addition to the large main peak in the patterns in Figure 33, spatial aliasing generates a number of significant side peaks. Because the peak power p_m is not known a priori, it is not immediately obvious when analyzing a partial interference pattern whether or not any particular peak is the main peak or a side peak.

One possible solution is to scan the entire pattern and compare the peak heights; however, this is not desirable, because, as described above, (1) the width of the sweep must be large to accommodate multi-target tracking over a wide formation; (2) it requires scanning dead space in the pattern, which makes data communications challenging; (3) the necessary device cross section quickly grows as the breadth of the grating sweep increases; and (4) the additional peaks in the pattern are redundant and provide no new information about the bearing angle ϕ.

Instead, the THz interferometer (shown in Figure 23 in Chapter 3) scans only a small portion of the pattern, limiting the amount of dead space scanned, maintaining a small device cross section, and acquiring minimal redundant information (assuming at least one peak in the pattern is visible). This however introduces an ambiguity (which peak of the pattern is observed). This ambiguity can readily be resolved via an initialization process and then continuously tracked, analogously to carrier-phase ambiguity resolution in GPS [67], [88]. Figure 34 shows the same cases as Figure 33, with the minimal scan range highlighted at the center of the plots. In each of the cases shown, at least one of the peaks is visible in the scan range, allowing for bearing-angle observability.
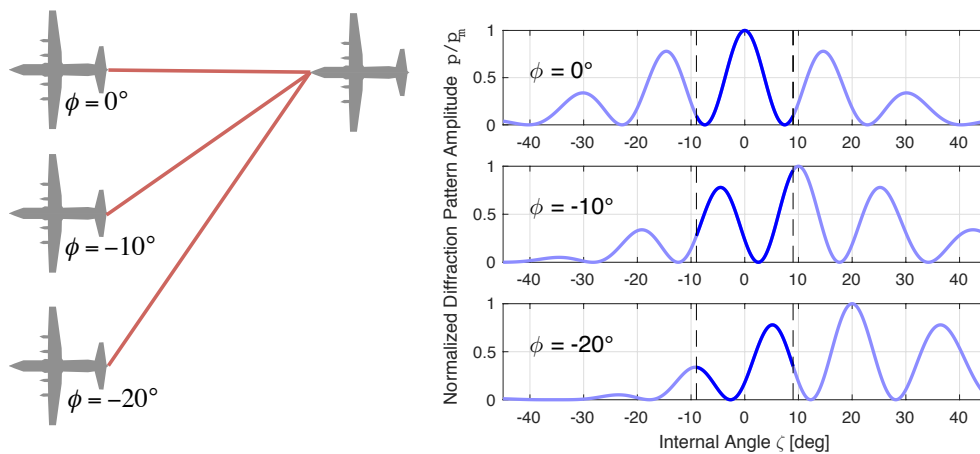
Figure 34 – Example aircraft formations and corresponding double-slit diffraction patterns with diffraction grating scan identified

From signal measurements of the partial pattern, it is possible to estimate the bearing angle ϕ. Angle measurements can then be fused with range and relative altitude measurements to produce position estimates. The remainder of this chapter describes a state-estimation algorithm for simultaneous range and bearing angle estimation.

4.3.3 Diffraction Grating Gain

As described in Chapter 3, the width of the detector serves to smooth this pattern. In order to solve the inverse problem of converting measurements of the interference pattern into bearing angle estimates, it will be convenient to define a diffraction grating gain term G_D. This gain is taken from equation (41) by normalizing the power to the peak intensity p_m,

$$G_D(\Delta, \phi) = \int_{\zeta_{det,-}}^{\zeta_{det,+}} \frac{p(\zeta, \phi)}{p_m} \delta\zeta \ . \tag{48}$$

The power incident on the receiver for a given grating position Δ and bearing angle ϕ then is

$$P_R(\Delta, \phi) = p_m G_D(\Delta, \phi) \tag{49}$$

4.4 Signal Model

For this preliminary investigation, the simple signal structure from the range demonstration described in Chapter 2 is assumed, where binary on-off keying is used to modulate a 10 MHz square wave onto the THz carrier signal. The next chapter considers a more complex signal structure utilizing spread spectrum methods to allow for multi-target tracking and communication.

Updating equation (10) from Chapter 2 to include the diffraction grating gain, the raw signal measurements are modeled as

$$v(k) = v_0 \sqrt{G_D(\Delta, \phi)} \, \text{square}(\omega_{mod} k T_s + \theta) + n_v(k) \,, \tag{50}$$

where the G_D is the diffraction grating gain given in equation (48).

The time-of-flight phase delay θ results from the fact that the signal travels a distance r between the transmitter and receiver at the finite speed of light c, and is defined as

$$\theta = 2\pi \omega_{mod} \frac{r}{c} \,. \tag{51}$$

Equation (50) has three key unknowns, v_0, ϕ, and θ (assuming synchronized atomic clocks and good actuator encoders, the T_s parameter, the modulation frequency ω, and the grating position Δ are known). These three key states – two of which (ϕ and θ) relate directly to the position estimation problem depicted in Figure 9 and one (v_0) is a nuisance parameter – can be estimated using batches of multiple samples, as described in the algorithm below.

4.5   Digital Signal Processing

The raw signal measurements are processed through two correlators, to find the in-phase I and quadrature Q components of the modulated signal, as described above in Chapter 2, equations (6) and (7), which are reproduced here.

$$I_i(k) = \sum_{m=k-K+1}^{k} \cos(\omega_{mod} m T_s) v_i(m) \quad \text{and} \qquad (6)$$

$$Q_i(k) = \sum_{m=k-K+1}^{k} \sin(\omega_{mod} m T_s) v_i(m) . \qquad (7)$$

These components can then be used to find the phase delay θ, as described by equation (8) from Chapter 2.

A model of the raw signal measurements is given in equation (50) in Chapter 4, and reproduced here

$$v(k) = v_0 \sqrt{G_D(\Delta, \phi)} \, \text{square}(\omega_{mod} k T_s + \theta) + n_v(k) . \qquad (50)$$

Plugging this signal model into the integrators from equations (6) and (7), the I and Q values can be modeled as

$$I(k) = \frac{2K}{\pi} v_0 \sqrt{G_D(\Delta, \phi)} \cos\theta + n_I \quad \text{and} \qquad (52)$$

$$Q(k) = \frac{2K}{\pi} v_0 \sqrt{G_D(\Delta, \phi)} \sin\theta + n_Q , \qquad (53)$$

where n_I and n_Q are independent zero mean random Gaussian noise variables with a standard deviation of

$$\sigma_{IQ} = \sigma_v \sqrt{\frac{K}{2}}. \tag{54}$$

For a detailed derivation of equations (52), (53), and (54) see chapter Appendix 1.

The proposed angle estimation algorithm operates on the I and Q values. These values relate to the three unknown states ϕ, θ, and v_0 (noting that the angle of arrival ϕ comes into the equations through the diffraction grating gain term G_D and that the displacement of the grating $\Delta(t)$ is a known function of time). Thus equations (52), (53), and (54) are the key measurement equations for the estimator.

Though the integrator values I and Q are derived from the same set of raw measurements $v((k - K + 1) < m < k)$, the noise values n_I and n_Q are independent (see chapter Appendix 1). This independence is significant to the subsequent design and noise analysis of the proposed state-estimation algorithm.

There are minor effects from the discretization of the signal and from clock drift error in the I and Q measurements, which are discussed in chapter Appendix 2 but otherwise neglected in our analysis

4.6   Algorithm

This section introduces a new state-estimation algorithm that infers the bearing angle ϕ from the I and Q outputs from signal processing. In addition, the range r and the nuisance power variable v_0 are also estimated and used to produce two-dimensional position estimates. In this snapshot algorithm, the I and Q values are accumulated in a batch spanning a relatively short time (90 ms, allowing for 40 I and 40 Q samples) and processed together, providing snapshot estimation of the relative position between the transmitter and receiver.

Within the batch, each pair of instantaneous I and Q measurements is collected at a specific grating position corresponding to a location in the diffraction pattern. Each measurement, therefore, is tied to a diffraction grating position Δ and corresponding internal angle ζ. Figure 35 provides a visualization of this. The blue line shows the segment of the diffraction pattern scanned by the grating sweep (in this case the bearing angle ϕ is -10°, same as the middle plot in Figure 34). The red circles mark locations in the pattern where I and Q measurements are made. In all there are 40 red circles in the diagram, corresponding to the 40 epochs of data processed in the batch. This number was selected to ensure that enough data points are collected to identify the pattern shape and minimize distortion of the pattern, but not so many as to slow the grating motion to the point where the relative positions can no longer be assumed constant over the length of one sweep.
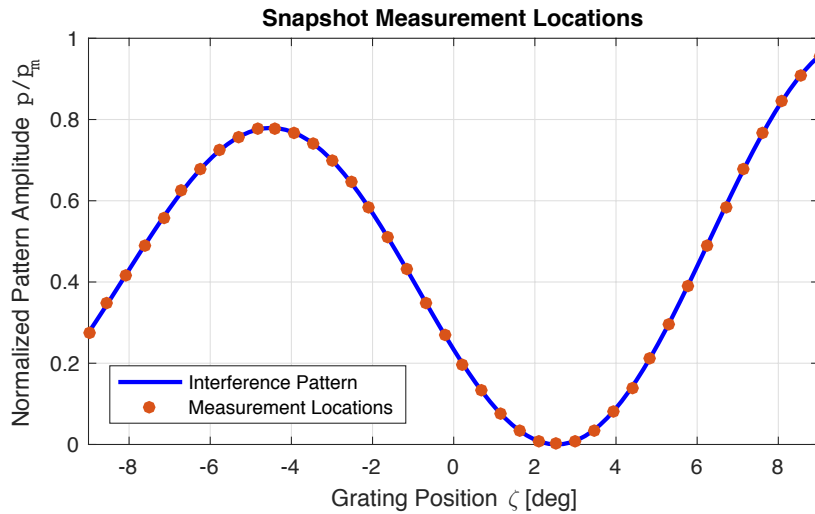
Figure 35 – Locations of instantaneous pattern measurements

The measurement vector $\mathbf{y} \in \mathbb{R}^{2N \times 1}$ comprises all I and Q measurements in the batch, with

$$\mathbf{y} = [I_{\Delta_1} \quad \cdots \quad I_{\Delta_N} \quad Q_{\Delta_1} \quad \cdots \quad Q_{\Delta_N}]^T, \tag{55}$$

and $N = 40$.

The unknown states sought by the algorithm are likewise assembled into a state vector $\mathbf{x} \in \mathbb{R}^{3 \times 1}$,

$$\mathbf{x} = [\phi \quad \theta \quad v_0]^T. \tag{56}$$

Measurement batches $\mathbf{y}$ are then processed by the algorithm using the Newton-Raphson method [89], [90] to iteratively solve the nonlinear measurement equations (52) and (53) for the unknown state vector $\mathbf{x}$. Following this procedure, the system of $2N$ equations is linearized around the initial state estimate $\hat{\mathbf{x}}_0$, taking the form

$$\delta \hat{\mathbf{y}} = \mathbf{H}_{\hat{\mathbf{x}}_0} \delta \hat{\mathbf{x}}, \tag{57}$$

where $\mathbf{H}_{\hat{\mathbf{x}}_0} \in \mathbb{R}^{2N \times 3}$ is the linearized measurement model, and then inverted to find the state estimate update. Equation (57) is inverted to find $\delta \hat{\mathbf{x}}$, which is used to update the state estimate $\hat{\mathbf{x}}$. Following Newton-Raphson, this procedure is repeated iteratively until the algorithm converges, with the norm of the state update $|\delta \hat{\mathbf{x}}|$ less than the tolerance (10^{-3}). In simulations, convergence is assumed to fail if it does not reach the tolerance within the maximum number of iterations (50). See Appendix 3 for a derivation of $\mathbf{H}_{\hat{\mathbf{x}}_0}$.

Finally, the distance estimate $\hat{r}$ is calculated from the phase delay estimate $\hat{\theta}$ using equation (51), and then combined with the bearing angle estimate $\hat{\phi}$ to find

the along-track x and cross-track y position estimates. These estimates are computed as

$$\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = \begin{bmatrix} \hat{r} \cos \hat{\phi} \\ \hat{r} \sin \hat{\phi} \end{bmatrix}. \qquad (58)$$

Simulations suggest that the nonlinear measurement equations (52) and (53) generate a cost function that is locally convex, but is composed globally of a number of minimums. This is the result of spatial aliasing in the pattern that gets projected into the bearing angle estimate and integer ambiguity in the phase delay estimate. Given a sufficiently close initial guess (for this case, bearing angle $\hat{\phi}_0$ within ~5° and phase delay $\hat{\theta}_0$ within ~90°) and tolerable signal-to-noise levels, the algorithm reliably converges to an accurate position estimate. It is assumed that a good guess is generally available from the prior time step since the time scale of the estimator algorithm is much shorter than the time scale for relative dynamics between cargo aircraft. As a result, the initial guess is really an acquisition problem – similar in nature to a GPS tracking loop. This chapter does not discuss acquisition.

4.7 Performance Analysis

This section considers the algorithm performance. A probabilistic analysis is used to predict the errors over a wide range of possible formations. A statistical analysis is then applied to a specific case to corroborate the probabilistic results.

It is assumed that the double-slit diffraction grating described in Chapter 3 is mounted on the demonstration equipment from Chapter 2. The relevant equipment parameters are assumed to be those detailed in Table I and Table II with one difference: to accommodate the longer distances associated with formation flight, it is assumed that the transmit power is 5 times greater than that

of the ground-level system ($P_T = 150$ mW). In addition, it is assumed that the integration is performed over the longer period, covering $K = 2{,}250{,}000$ samples and resulting in an integration time of 2.25 ms. It is assumed that both aircraft fly straight and level at a predetermined altitude, such that altitude measurements are not needed in this case. The diffraction grating is assumed to be an LCD screen, with the slits moving at constant speed across the front of the device and then abruptly jumping back and starting the motion over again. The diffraction grating is assumed to scan at a rate of $F_g = 11.1$ Hz, allowing for 40 independent measurements of the integrators.

4.7.1 Probabilistic Analysis

The expected error in state estimates can be approximated using tools from linear least-squares regression analysis. Assuming that the nonlinear equations are locally well behaved, linear methods are applied. Since all of the measurements have independent noise and the same standard deviation, the covariance $\boldsymbol{\Sigma}$ of the state estimates can be written as

$$\boldsymbol{\Sigma} = \sigma_{IQ}^2 (\mathbf{H_x}^T \mathbf{H_x})^{-1}\,, \tag{59}$$

where σ_{IQ} is the standard deviation on measurements of I and Q and $\mathbf{H_x}$ is the measurement model matrix linearized around the true states $\mathbf{x}$. Picking off the diagonal elements of the covariance $\boldsymbol{\Sigma}$, state-error variances are found. These predicted variances depend on aircraft configuration, manifesting from changes in the linearization states $\mathbf{x}$.

Figure 36 shows the expected two-sigma errors on range estimates from equation (59) as a function of the bearing angle ϕ. Range estimate errors are largest at long distances and wide angles, where the signal-to-noise ratio is lowest, as calculated from the link budget in equations (42) and (43) from Chapter 3. The ranging error

95

scales linearly with increased distance over the span of ranges used for the precision airdrop application. As a result, the 2D error surface representing the assortment of possible formation configurations (r, ϕ) can be collapsed to this single curve, which represents the amount of ranging error per kilometer between the transmitting and receiving aircraft.
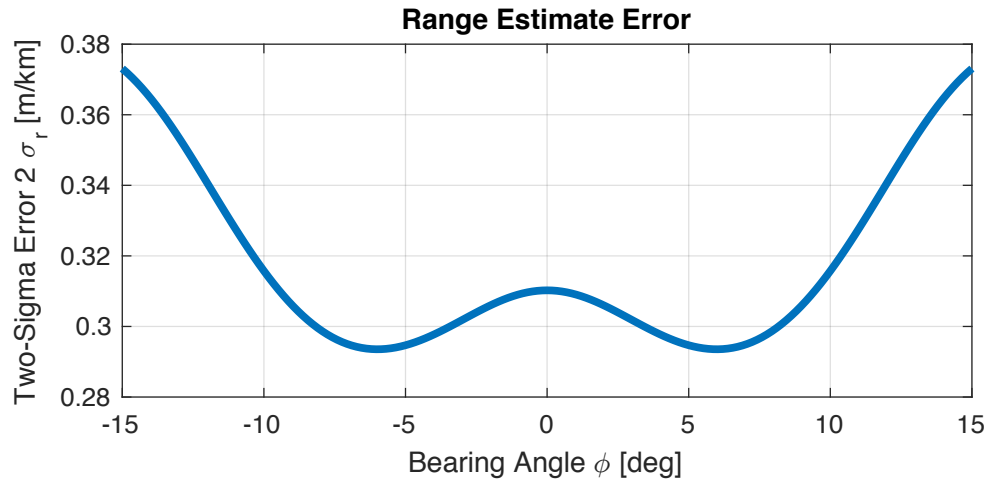
Figure 36 – Batch LS range error, two-sigma

Similarly, bearing estimate errors are also largest for long distances and wide angles, although the shape of the error curve is different. Figure 37 shows the expected two-sigma errors on the bearing angle estimates from equation (59). Again, over the span of formations considered, the error scales linearly with range, so the error surface is collapsed onto a single curve, representing the bearing estimate error per kilometer of distance.
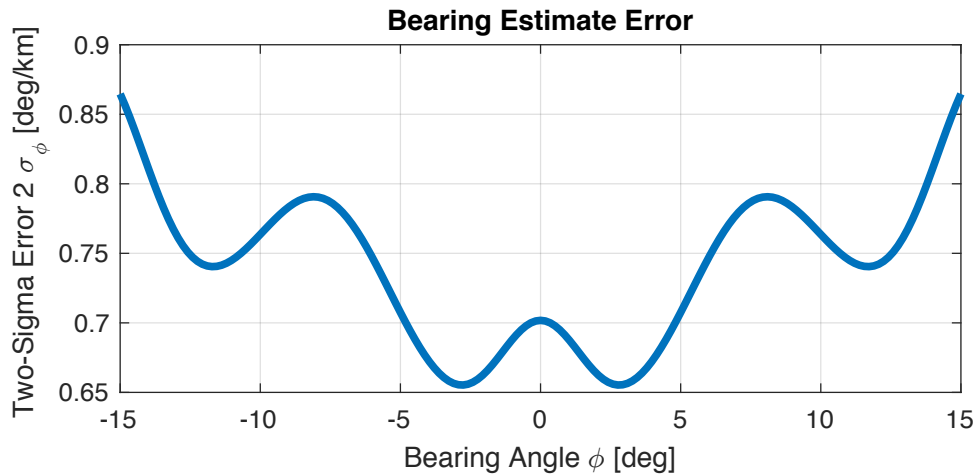
Figure 37 – Batch LS bearing angle error, two-sigma

Both of these plots appear to be composed of two components: a convex bowl shape overlaid with a wavy curve approximately sinusoidal in nature. The bowl shape comes from decreases in the signal-noise-ratio as the signal becomes more glancing and the large central peak (which contains a significant fraction of the signal power) drifts out of view leaving only the side peaks. The waviness comes from the features of the interference pattern (peaks and valleys of power) moving across the detector.

4.7.2   Statistical Analysis

A Monte Carlo simulation is used to validate the linear probabilistic analysis. A representative formation is assumed where the aircraft fly together with no relative motion and where the transmitter is positioned at a range r of 1000 m and

a bearing angle ϕ of 10°. Measurements are simulated using equations (52) and (53) with random noise as described by (54) and power calculated from the link budget in equations (42) and (43) from Chapter 3. Position estimates are then calculated from the noisy measurements using the algorithm described above. Acquisition is assumed to be accurate and the algorithm is initialized at the first time step with true states and then with the previous solution at each subsequent step.

Figure 38 shows position estimation results for the Monte Carlo simulation. The plot on the right shows the distribution of position estimates with the true position of the aircraft as a black dot, and the estimated positions as green crosses. The plot on the left shows the error in the bearing angle estimate as a function of time in blue, with the predicted two-sigma error bound from equation (59) shown as red dash-dot lines. For this simulation, the two-sigma error from the data is calculated to be 0.81° for the angle estimate, and 0.31 m for the range, over 3333 epochs.
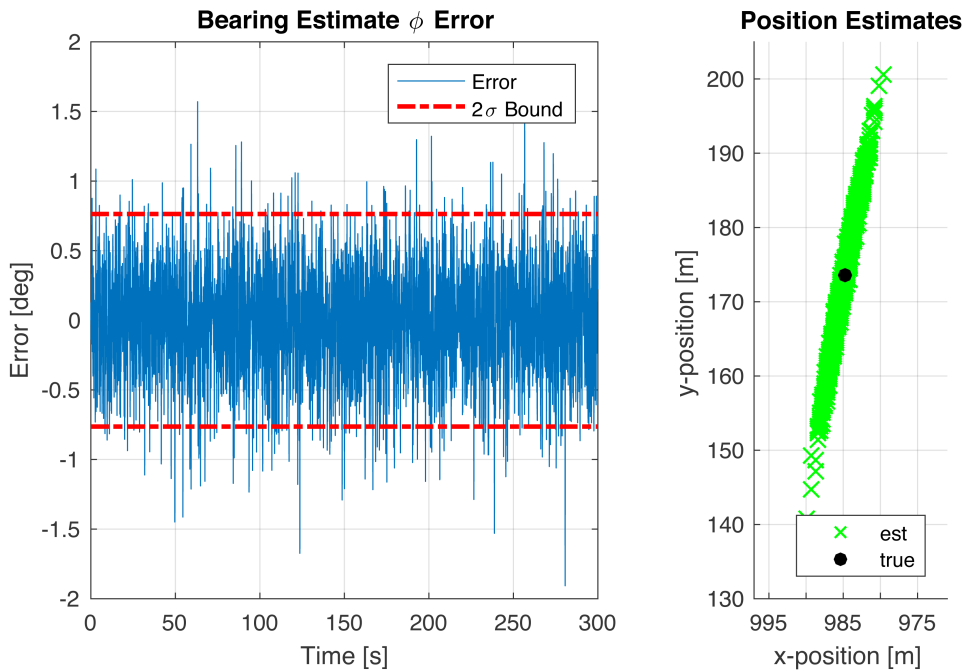
Figure 38 – Batch LS simulation results

4.8   Discussion

The simple snapshot algorithm presented in this chapter is capable of estimating the bearing angle using measurements of the THz signal taken by the interferometric receiver package. Results of the Monte Carlo simulations from the statistical analysis agree reasonably well with the probabilistic analysis. For the case examined ($r = 1000$ m and $\phi = 10°$), the predicted and observed errors were close ($2\sigma_r = 0.31$ m and $2\sigma_\phi \cong 0.8°$).

Though the analysis presented results for 1 km separation of aircraft, those results can easily be generalized to the nominal operational conditions for the formation flight scenarios that motivated the work. Since errors are proportional to range, errors for 2 km aircraft separation are approximately twice the values for 1 km separation. In other words, 2-sigma errors for 2 km separation are 0.64 m (range) and roughly 1.4°-1.6° (bearing). The ranging error is more than sufficient to meet the requirements for the precision airdrop application; however, the bearing error, which is the key source of error, is only just sufficient to meet the requirements for small angles (~5° or less). To accommodate larger angles and provide higher accuracy, Bayesian methods are introduced in the full position estimation algorithm discussed in Chapter 6.

To achieve these results, it should be noted that an increase in the transmitter power was assumed as compared to the demonstration equipment used in prior experiments. The five-fold increase in power corresponds with the significant increase in distance (an order of magnitude) relative to our ground-level tests. It is believed that the desired power amplification is representative of future hardware that will become available as the technology matures.

4.9 Conclusion

This chapter presents the first algorithm to make bearing angle estimates from THz interferometer measurements. Using measurements taken over the span of a diffraction grating sweep, the simple snapshot algorithm is able to accurately estimated the bearing angle to the transmitter. In addition, range estimates were considered in this work to allow for simple two-dimensional position estimation. Both probabilistic and statistical analyses were compared and corroborated, demonstrating the system's accuracy. Predicted accuracy in the range dimension (0.64 m two-sigma or less out to 2 km) easily meets the requirements for along-track positioning, and predicted accuracy in the bearing dimension (~1.4° two-sigma or less out to 2 km) is just sufficient to meet the tight cross-track positioning requirements for formations with small bearing angles (~5° or less). This work is expanded in the full positioning algorithm described in Chapter 6, using Bayesian methods to smooth estimates over time and significantly improve accuracy.

4.10 Appendix 1: *I* and *Q* Measurement Equation Derivation

The appendix in Chapter 2 derived a relationship between the raw signal measurement noise and the correlator noise in the frequency-domain. This appendix derives a similar result in the time-domain. The results are obtained by applying relatively standard analytical techniques and are presented merely for completeness.

In this analysis, we consider only the fundamental frequency of the square wave,

$$v(k) \approx v_0\sqrt{G_D}\frac{4}{\pi}\cos(\omega_{mod}kT_s + \theta) + n_v(k). \tag{60}$$

It is assumed that the integration time of the correlators is sufficiently small that the grating can be assumed stationary over the length of the integration. Plugging this into equation (6) from Chapter 2, the in-phase correlator can be written as

$$
\begin{aligned}
I(k) = \frac{4}{\pi}v_0\sqrt{G_D}\sum_{m=k-K+1}^{k} & \cos(\omega_{mod}mT_s + \theta)\cos(\omega_{mod}mT_s) \\
& + n(m)\cos(\omega_{mod}mT_s)
\end{aligned} \tag{61}
$$

Using trig identities, this can be re-written as

$$
\begin{aligned}
I(k) = \frac{4}{\pi}v_0\sqrt{G_D}\sum_{m=k-K+1}^{k} & \frac{1}{2}(\cos(\theta) + \cos(2\omega_{mod}mT_s + \theta)) \\
& + n(m)\cos(\omega_{mod}mT_s).
\end{aligned} \tag{62}
$$

Because the sum is over an even number of modulation cycles, the term $\cos(2\omega_{mod}mT_s + \theta)$ sums to zero, leaving

$$I(k) = \frac{2K}{\pi}v_0\sqrt{G_D}\cos(\theta) + \sum_{m=k-K+1}^{k} n(m)\cos(\omega_{mod}mT_s). \tag{63}$$

For the noise term, the expected value of cos is 0 and its variance is $\frac{1}{2}$, so it can be reduced as follows

$$I(k) = \frac{2K}{\pi} v_0 \sqrt{G_D} \cos(\theta) + \mathcal{N}\left(0, \frac{K\sigma_v^2}{2}\right). \tag{64}$$

A similar approach can be used to find the quadrature component

$$Q(k) = \frac{2K}{\pi} v_0 \sqrt{G_D} \sin(\theta) + \mathcal{N}\left(0, \frac{K\sigma_v^2}{2}\right). \tag{65}$$

An important observation is that the noise terms on these two measurements are independent. This can seem counterintuitive because they are derived from the same initial set of random numbers n_v. Remember however, that due to Nyquist, a set of N random numbers maps into $N/2$ complex numbers (with independent real and imaginary parts) in the frequency domain. The in-phase correlator samples the real parts of each complex number; the quadrature correlator samples the (independent) imaginary parts of each complex number. Hence the random errors on the in-phase and quadrature correlators are independent.

4.11 Appendix 2: Other Errors

In addition to random error propagated from the raw measurement noise, there is also some systematic error in the in-phase and quadrature measurements.

Consider discretization error. This arises from the fact that the modulation is actually a square wave, not a single sinusoid. For a perfect square wave with sharp edges, there are only two points per cycle that provide any information about the phase of the signal, the transitions between peaks and valleys. If the measurement and modulation frequency are locked multiples of each other (as they are in this case), the transition may happen at a time that is biased.

Figure 39 shows an example of this, where the signal is shown as the blue square wave, the measurements are shown as red dots, and the fitted (and biased) sinusoid is shown in yellow. This is an extreme example where the measurement frequency is only 4 times the modulation frequency, and the measurements are biased as far as possible toward the leading edge of the signal. Because of the systematic shift in the location of the measurements, the resulting cosine fit ends up biased by 45° relative to the square wave.
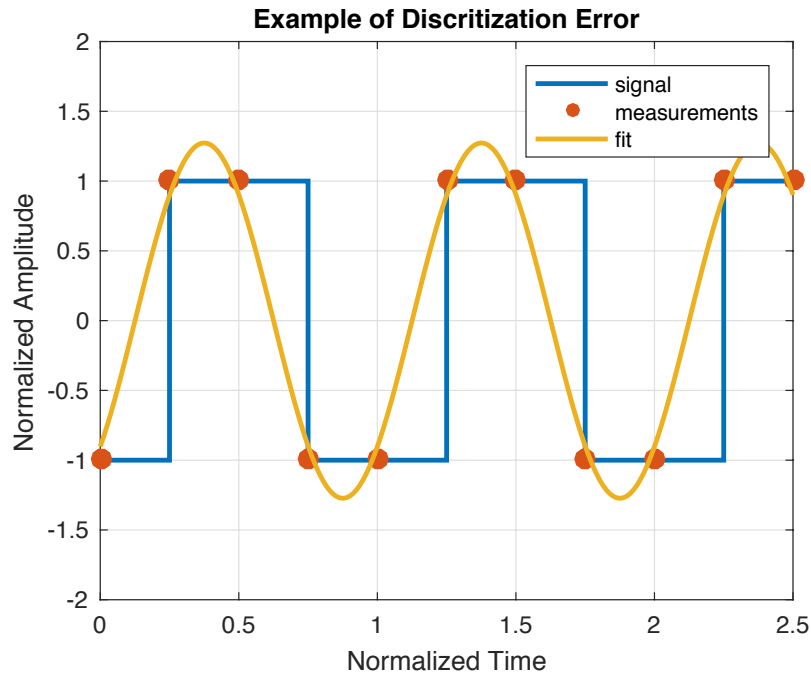
Figure 39 – Example of discretization error

Discretization error probability is uniformly distributed between the extremes $\pm\pi F_{mod}/F_s$. In this case, the ratio of measurement frequency to modulation frequency is 100, so the maximum possible discretization error is $\pm1.8°$. That equates to about 15 cm of ranging error for the 10 MHz modulation. This effect is non-negligible relative to the random error modeled in the simulations; however, the bias has no effect on bearing-angle estimation, and so was not considered in the simulations of this paper.

Note that, in the future, using a different modulation scheme with smoother transitions, like a sinusoid, could hypothetically mitigate the discretization error.

Another secondary error source is clock drift. In one-way communication, it is necessary for the transmitter and receiver to carry synchronized atomic clocks to perform ranging. Over a mission lasting many hours, these clocks will drift. Assuming that high quality atomic clocks are used, a drift on the order of 0.5 ns is possible over a few hours, resulting in 15 cm of additional error. Again, this effect was not considered in the simulations in this paper because it has no effect on bearing angle estimation.

4.12 Appendix 3: Measurement Equation Linearization

Measurement equations (52) and (53) are functions of the state vector

$$\mathbf{x} = \begin{bmatrix} \phi & \theta & v_0 \end{bmatrix}^T . \tag{66}$$

They can be linearized around a nominal state estimate $\mathbf{x}_0$ as

$$\delta I = \begin{bmatrix} \dfrac{\partial I}{\partial \phi}\bigg|_{\mathbf{x}_0} & \dfrac{\partial I}{\partial \theta}\bigg|_{\mathbf{x}_0} & \dfrac{\partial I}{\partial v_0}\bigg|_{\mathbf{x}_0} \end{bmatrix} \delta \mathbf{x} \quad \text{and} \tag{67}$$

$$\delta Q = \begin{bmatrix} \dfrac{\partial Q}{\partial \phi}\bigg|_{\mathbf{x}_0} & \dfrac{\partial Q}{\partial \theta}\bigg|_{\mathbf{x}_0} & \dfrac{\partial Q}{\partial v_0}\bigg|_{\mathbf{x}_0} \end{bmatrix} \delta \mathbf{x} , \tag{68}$$

where δI and δQ are the differential measurements, $\delta \mathbf{x}$ is the differential state vector, and the partial derivatives are

$$\frac{\partial I}{\partial \phi} = \frac{v_0 K}{\pi \sqrt{G_D}} \frac{\partial G_D}{\partial \phi} \cos \theta \ , \tag{69}$$

$$\frac{\partial I}{\partial \theta} = -v_0 \frac{2K}{\pi} \sqrt{G_D} \sin \theta \ , \tag{70}$$

$$\frac{\partial I}{\partial v_0} = \frac{2K}{\pi} \sqrt{G_D} \cos \theta \ , \tag{71}$$

$$\frac{\partial Q}{\partial \phi} = \frac{v_0 K}{\pi \sqrt{G_D}} \frac{\partial G_D}{\partial \phi} \sin \theta \ , \tag{72}$$

$$\frac{\partial Q}{\partial \theta} = v_0 \frac{2K}{\pi} \sqrt{G_D} \cos \theta \ , \text{and} \tag{73}$$

$$\frac{\partial Q}{\partial v_0} = \frac{2K}{\pi} \sqrt{G_D} \sin \theta \ , \tag{74}$$

with $\delta G_D / \delta \phi$ defined as

$$\frac{\partial G_D}{\partial \phi} = \int_{\zeta_{det,-}(\Delta)}^{\zeta_{det,+}(\Delta)} 2 \operatorname{sinc} \alpha \cos \beta \frac{\pi}{\lambda} \cos \phi \left[a \cos \beta \left(\frac{\cos \alpha}{\alpha} - \frac{\sin \alpha}{\alpha^2} \right) \right.$$
$$\left. - d \operatorname{sinc} \alpha \sin \beta \right] \delta \zeta \ . \tag{75}$$

The measurement matrix $\mathbf{H}_{\mathbf{x}_0}$ linearized around the point $\mathbf{x}_0$ for a batch of N measurements is thus written as

105

$$\mathbf{H_{x_0}} = \begin{bmatrix} \left.\dfrac{\partial I_1}{\partial \phi}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial I_1}{\partial \theta}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial I_1}{\partial v_0}\right|_{\mathbf{x_0}} \\ & \vdots & \\ \left.\dfrac{\partial I_N}{\partial \phi}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial I_N}{\partial \theta}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial I_N}{\partial v_0}\right|_{\mathbf{x_0}} \\ \left.\dfrac{\partial Q_1}{\partial \phi}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial Q_1}{\partial \theta}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial Q_1}{\partial v_0}\right|_{\mathbf{x_0}} \\ & \vdots & \\ \left.\dfrac{\partial Q_N}{\partial \phi}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial Q_N}{\partial \theta}\right|_{\mathbf{x_0}} & \left.\dfrac{\partial Q_N}{\partial v_0}\right|_{\mathbf{x_0}} \end{bmatrix}. \tag{76}$$

Chapter 5

Signal and Data Structure for Communication and Navigation Using a THz Interferometer[4]

5.1   Introduction and Motivation

A preliminary signal structure was used in Chapter 2 and Chapter 4, where a simple on-off square wave was modulated onto the carrier signal. While this naïve structure was sufficient for ranging and bearing determination between a single transmitter and receiver, it did not allow for multi-target tracking and was not designed for communications.

Low-rate communication of key parameters is a vital part of most navigation systems. GPS satellites, for example, communicate a host of parameters describing their orbits so that receivers can precisely determine their position, their clock behaviors so that receivers know the time exactly, and even satellite health information so that receivers know when to ignore their measurements. For

---

[4] This chapter was accepted to be published under the title "Signal and Data Structure for Navigation with a Terahertz Interferometer" in the Procedings of the 30th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2017) [91]. Some changes have been made to the text for clarity and flow.

formation flight, a number of parameters and measurements may be useful. For example, communicating the compass heading measurement and/or any planned maneuvers would be helpful for flight control. For positioning, as described below, altimeter measurements are key, and the communication scheme described in this chapter is designed with this data in mind.

As mentioned earlier, range measurements can be combined with measurements of the pointing vector to find the relative position. The previous two chapters detailed methods for measuring the planar bearing angle between the aircraft, but to fully describe the pointing vector in three-dimensions an independent measurement of the vertical component is necessary. Conceptually, the planar bearing angle could be supplemented with measurements of the elevation angle to describe the pointing vector between the aircraft. One strategy would be to use a second THz interferometer, oriented in the vertical direction, to measure the elevation angle; however, because of the high cost of the equipment this makes the system significantly more expensive. Another strategy would be to use a diffraction grating with a two-dimensional pattern to provide observability in the vertical direction; however, because the interference pattern will be two-dimensional, this requires significantly increased scanning speeds to cover the space, cutting down on integration time and degrading the signal-to-noise ratio.

Instead, this work proposes using barometric altimeters to measure the vertical displacement of the aircraft. Because altimeters are cheap, this does not add to the overall cost of the system, and it does not require any modifications to the THz hardware. In this scheme, the lead aircraft broadcast its altimeter measurements to the following aircraft. The following aircraft can then calculate the relative altitude from a difference of altimeter measurements.

A custom GPS-like signal structure is introduced for navigation and communication at THz frequencies. It leverages spread spectrum methods to

enable simultaneous communication and positioning with multiple transmitters in a formation.

The technical challenge here is overcoming the signal fading in the valleys of the interference pattern to ensure reliable communication. As the diffraction grating scans in front of the detector, the received signal power necessarily fluctuates between high and low. This time-varying signal power presents a challenge for data transmission, as bits received during periods of low power have a high likelihood of being missed.

This chapter proposes frequency matching for the diffraction grating scan and data bit repetition to compensate for lost bits during periods of low gain due to the diffraction grating. All data bits are repeated, and then this method sets the frequencies such that if a data bit falls during a period of low power, its repetition will fall during a period of high power. This significantly decreases the risk of missed communications, making reliable communication possible.

5.2 Designing Signal and Data Structure

5.2.1 Problem Description

The moving diffraction grating generates peaks and valleys in the received signal power over time. The peaks and valleys of the diffraction pattern are advantageous because they make it possible to infer bearing, but they impose a unique communication challenge during the intermittent periods when the gain is low. In these valleys of the gain pattern, any data bits modulated on the THz signal may be lost or corrupted.

This chapter focuses specifically on the double-slit diffraction grating equipment configuration identified in the last chapter. The diffraction grating gain G_D for the double-slit grating is defined as

$$G_D(\Delta, \phi) = \int_{\zeta_{det,-}}^{\zeta_{det,+}} \text{sinc}^2 \, \alpha \, \cos^2 \beta \, \delta\zeta \; . \tag{77}$$

Two example interference patterns are shown in Figure 40 (assuming the equipment parameters defined in Table I). The figure illustrates the diffraction grating gain $G_D(\Delta, \phi)$, where Δ is the lateral distance moved by the grating, shown on the x-axis, and ϕ is the aircraft bearing angle. The solid blue line shows the diffraction pattern for the case when the bearing angle is 0°, and the dashed red line shows the case when the angle is 10°. The locations of the peaks and valleys in these two patterns distinguish them from each other, allowing for angle estimation. The valleys, however, can result in communication gaps.
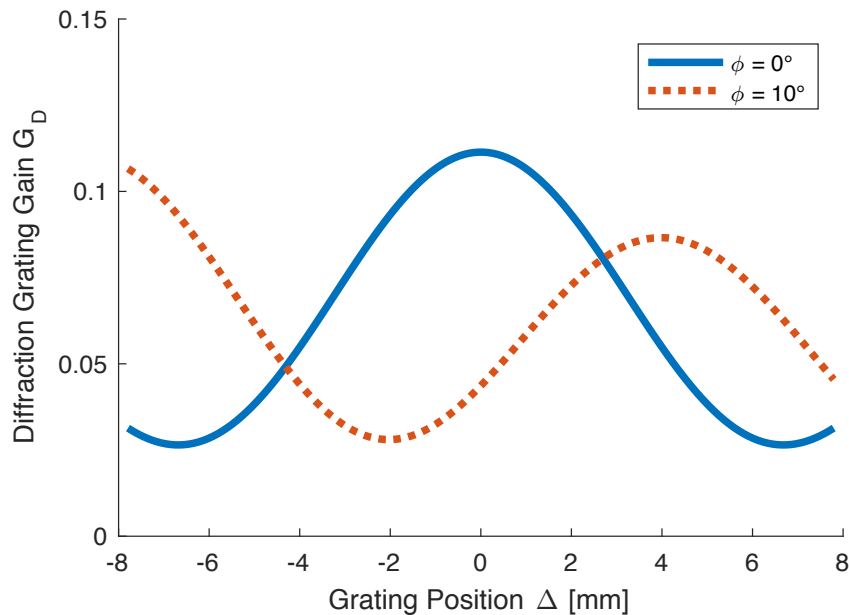
Figure 40 – Example diffraction patterns

Modulation is needed to perform three-dimensional relative positioning. First, a code must be modulated on to the carrier signal in order for the receiver to estimate its range relative to the transmitter. Second, data bits must be modulated on to the code, in order to communicate the transmitter's altimeter and altimeter

110

rate measurements. Together, bearing, range, and relative altimetry can be combined to estimate three-dimensional relative position between transmitter and receiver. For flight applications, the relative-position estimate should be computed at about 1 Hz. To ensure integrity of this relative-positioning signal, the communication probability of correctly decoding bits must be high and the latency of the communication must be low. Because of aircraft dynamics, in this work a requirement is imposed that the altimetry measurements must arrive no more than 0.2 s from the time they were sampled.

Much like in GPS, communication in our proposed THz navigation system is one-way. For large formations, a single receiver may potentially observe signals from more than one transmitter. Accordingly, multiple access considerations must be considered in designing the THz system.

This section describes a signal and data bit structure that has been customized to meet these requirements while addressing the specific issues of working in the THz domain and working with a diffraction grating. This structure addresses some of limitations of THz hardware and mitigates the problem of missed bits due to the diffraction grating. It allows for communication of measurements, and provides extra bandwidth for additional communications.

5.2.2   Signal Structure

The proposed THz signal structure, modeled on GPS, is composed of three main components overlaid on top of each other: carrier, code, and data, as shown in Figure 41. On the bottom is the carrier THz signal. Next, the transmitter modulates a uniquely identifying Gold code onto the carrier using simple on-off modulation. Finally, data bits are modulated on top of the code using Binary Phase Shift Keying (BPSK).
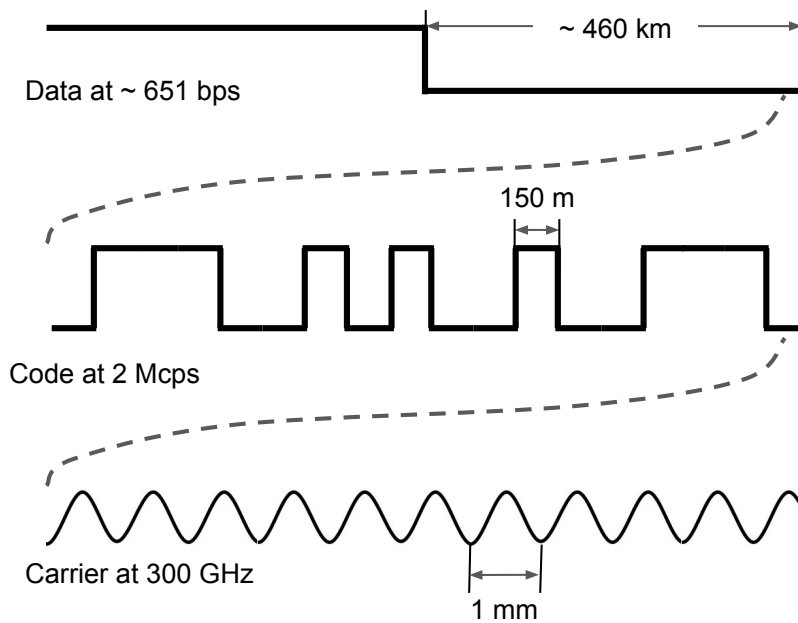
Figure 41 – THz signal structure modeled on GPS

Table IV lists the frequencies and multiplying factors that define the relationships between the layers of the THz signal structure. The proposed THz signal is transmitted on a 300 GHz carrier (note, higher THz frequencies may be used in the future for added stealth and anti-jamming benefits). The receiver hardware rectifies and collects the signal, somewhat like an optical device, effectively integrating the power at a sample rate of 1 GHz. This means that there are 300 carrier waves per signal sample. The code is modulated onto the carrier at a chipping rate of 2 Mcps, resulting in 500 signal samples per chip. The code is a length-1023 Gold code, giving a code frequency of ~1.96 kxps. Finally, data bits span 3 code repetitions, resulting in a data bit rate of ~651 bps.

Table IV –THz signal structure parameters

	Frequency	Multiplying Factors	
Carrier	300 GHz		
Sampling	1 GHz	Number of carriers per sample	$N_{ws} = 300$
Chipping	2 Mcps	Number of samples per chip	$N_{sc} = 500$
Code	~1.96 kxps	Number of chips per code	$N_{cx} = 1023$
Data	~ 651 bps	Number of codes per data bit	$N_{xd} = 3$

There are two key differences between the THz and GPS signals. The first is simply the difference in carrier wavelength, with GPS broadcast at 1.57542 GHz and the THz signal broadcast at 300 GHz. This higher frequency carrier provides the main stealth and anti-jamming advantages of the THz system.

The second key difference is the mode of carrier modulation. In GPS, BPSK is used to modulate the code on to the carrier signal. Because of the high frequency of THz signals and the relative infancy of the hardware, carrier phase tracking is not currently available for off-the-shelf THz receivers. Rather, standard receiver hardware rectifies and integrates the signal over a nanosecond, wiping out any phase information, and leaving only a measurement of the carrier signal amplitude. As a result, BPSK cannot be used on the carrier in the THz domain with current technology. Instead, the THz system uses binary amplitude modulation, namely on-off modulation, to modulate the code onto the carrier (BPSK can still be used for modulation of the data bits on to the code, however). As a result, half of the power is lost, because the transmitter is off half of the time.

5.2.3   Data Bit Structure

To determine altitude relative to the THz transmitter, the THz receiver needs barometric altimeter measurements broadcast from the transmitter. In addition, it is beneficial to broadcast altimeter rate measurements. This allows the receiver to accurately predict the relative altitude for a period of time and coast through brief outages in the data.

Conceptually, a standard radio link could be used for communication; however, this risks degrading the stealth and anti-jamming advantages of the THz system. Alternatively, separate THz navigation and communication channels could be set up using a dedicated communication receiver that does not have a diffraction

grating, and therefore does not suffer from periodic power reductions, but this is undesirable given the current high cost of the hardware. Our proposed design, in which communication bits are modulated on to the code, is a strong alternative, providing an adequate data rate while maintaining stealth and anti-jam, with no additional hardware costs.

A simple data structure is proposed for communicating the lead aircraft's measurements to the follower, shown in Figure 42. The data is broken into two words, both 20 bits long. The first is composed of the barometric altimeter measurement and a 3-bit cyclic redundancy check (CRC) value. The barometric altimeter measurement is allotted 17 bits, giving roughly 11 cm resolution over a range from 0 to 15 km, with one unique value reserved as a flag in case the altitude measurement lies outside this range. The second word is composed of the barometric altimeter rate measurement, a compass measurement, which is useful for flight control but is not used in this paper, and a 3-bit CRC value. The altimeter rate measurement is allotted 8 bits, giving roughly 8 cm/s resolution over a range from -10 to +10 m/s, again with one unique value reserved as a flag. Finally, the compass is allotted 9 bits, allowing for roughly 0.7° resolution from 0 to 360°.

Word 1

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Altimeter H_{lead}																	CRC		

Word 2

21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Altimeter Rate $\dot{H}_{lead}$								Compass									CRC		
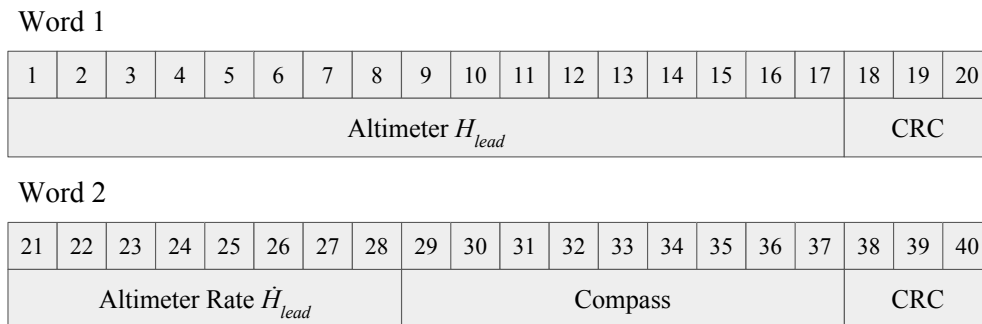
Figure 42 – Data bit structure

The two data words are transmitted one right after the other and then, as a defense against missed communication, they are repeated a second time, giving the

receiver two chances to receive the data uncorrupted. Complete transmission of the data, therefore, requires 80 bits, which takes roughly 0.12 s, well under the 0.2 s requirement for promptness.

Because altitude measurements are needed only once a second, the current signal design contains 560 blank bits between transmitted measurements. This is shown graphically in Figure 43, where the 80 bits that make up the two data words and their repetitions, shown in dark grey, are separated by 560 blank bits in the data stream. While these blank spaces do not serve a particular purpose in the current algorithm other than for positioning, they could hypothetically be used as pseudo-pilot channel for navigation or to support additional communications.
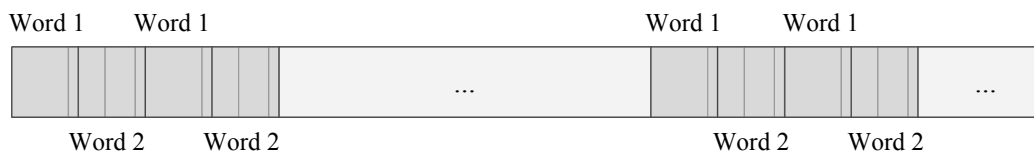
Figure 43 – Data stream

5.2.4   Diffraction Grating Timing

The timing of the diffraction grating sweep relative to the word repetition has a significant impact on the likelihood of missed data bits. As mentioned above, the diffraction grating generates peaks and valleys in received signal power as it sweeps in front of the detector. If a data word arrives during a period of low power in the pattern, it has a significantly higher likelihood of being corrupted by noise. As a result, it is desirable to time the diffraction grating sweep to the arrival of data. Note that for this work, if a word is corrupted by a bit error and the CRC values do not match, then the entire word is thrown out. Bit error correction is left to future work.

Conceptually, the diffraction grating sweep could be controlled, and the following aircraft could use its knowledge of the transmitter position and the diffraction

pattern to time the diffraction grating so that the data arrives during a peak in the pattern; however, this is only possible if it is tracking a single transmitter. In a multi-access scenario with multiple transmitting aircraft, locking the diffraction grating sweep to one of the transmitters would result in higher likelihood of missed communications from the other transmitters.

Instead, fixing the diffraction grating sweep frequency relative to the data transmission rate is suggested, which produces a low probability of missed data that is robust to a variety of different bearing angles and grating positions. To demonstrate this, consider three example cases (inspired by the fairy tale "Goldilocks and the Three Bears"): one that is too slow, one that is too fast, and one that is just right.

Start by considering the two suboptimal cases (Mama and Papa bear), shown in Figure 44. On the left is a case with a diffraction grating sweep that is too slow ($F_g = 2$ Hz) and on the right is a case with a diffraction grating sweep that is too fast ($F_g = 15$ Hz). Time is plotted on the x-axis and diffraction grating gains are plotted along the y-axis. The solid black curves show the diffraction grating gains for two different angles of incidence ($\phi = 0°$ and $8°$). The blue boxes show the timing of the first word (w1) and its repetition, and the red boxes show the timing of the second word (w2) and its repetition. Note that we are assuming that a solid-state screen like an LCD is used as the diffraction grating, with the slits sweeping from left to right across the screen and then abruptly jumping back and starting over.
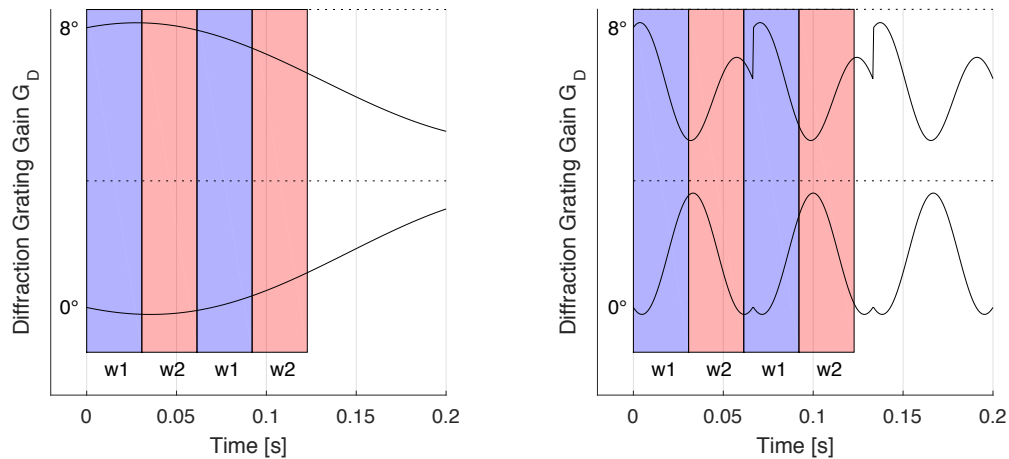
Figure 44 – Suboptimal (Mama and Papa bear) grating sweep frequencies: (left) too slow, (right) too fast

In the "too slow" case on the left, the pattern changes slowly. This appears beneficial for the 8° bearing angle case shown on top, because all of the data is received at a time of relatively high gain. The 0° bearing angle case, however, tells the opposite story, with all of the data arriving at a time of low gain. The feast or famine nature of this configuration results in a number of configurations where the likelihood of corrupted bits is high for both repetitions of the data words, resulting in a high likelihood of loosing the data entirely.

In the "too fast" case on the right, the pattern changes quickly. Here each word in both the 0° and 8° bearing angle cases contain some bits at high gain and some at low gain. As a result, every word will have some bits that are highly likely to be corrupted by noise, and the overall likelihood of losing data will again be high.

To minimize the likelihood of lost data, the grating frequency should be set such that if one of the words lands on a valley in the pattern, its repetition will land at a peak. Figure 45 shows two example alignments for this (baby bear) grating frequency. In both cases, whenever a word falls in a valley in the pattern, its repetition falls on a peak. As a result, if a data word is missed on the first pass, it is highly likely to be caught the second time around.

117
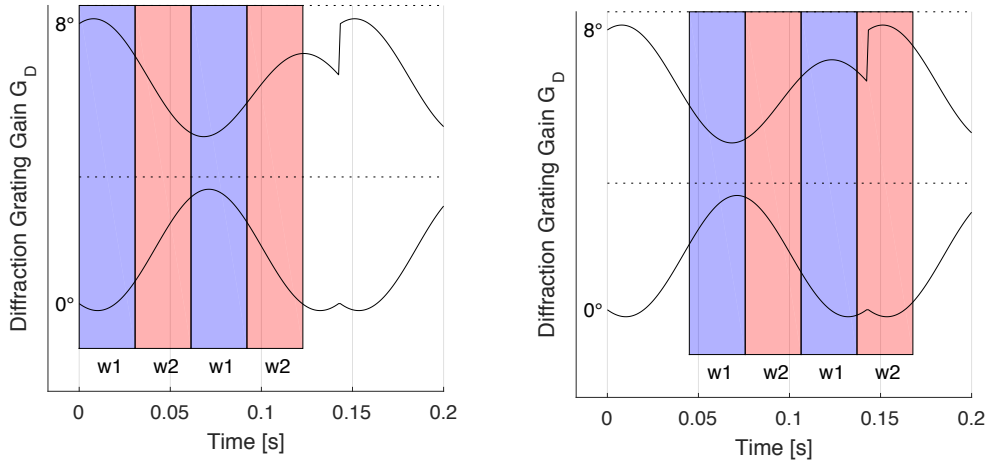
Figure 45 – Baby bear (just right) grating sweep frequency for two different alignments

The proposed system, therefore uses a Goldilocks grating frequency of $F_g = \sim 7$ Hz, which is just right.

To compute the Goldilocks grating frequency, consider that each displacement Δ of the diffraction-grating pattern is associated with a different internal angle ζ, as illustrated in Figure 28. The relationship is $\Delta = D \tan \zeta$, where D is the depth of the package (i.e. the perpendicular distance between the grating and the detector). The maximum displacement of the diffraction-grating pattern corresponds to an internal angle of $\pm\zeta_{max}$ (on either side of the neutral configuration), so the entire span of the internal angles is $2\zeta_{max}$. An angle can also be associated with the spacing between interference peaks generated along the back wall. This can be computed from the internal angle associated with the peak spacing ζ_{PS} from equation (46) in Chapter 3. The peak to valley spacing thus is roughly half this $\zeta_{PV} \cong \frac{1}{2}\zeta_{PS}$. The diffraction grating frequency F_g that matches the peak and valley spacing with the word spacing thus is calculated by scaling the word repetition frequency F_{wr} by the ratio of the peak to valley spacing ($\frac{1}{2}\zeta_{PS}$) to the sweep span ($2\zeta_{max}$),

$$F_g = \frac{\zeta_{PS}}{4\,\zeta_{lim}} F_{wr} \quad . \tag{78}$$

For our proposed device ζ_{lim} is $9°$, ζ_{PS} is $15°$, and $F_{wr} = \frac{\sim 651 \text{ bps}}{40 \text{ bpwr}} = \sim 16.3$ wrps, resulting in $F_g = \sim 7$ Hz.

5.3   Data-Bit Estimation

A simple data-bit estimation algorithm is used in this work. It is depicted graphically in Figure 46. Correlator measurements of the signal are processed in batches through a weighted least squares (WLS) algorithm to produce estimates of the data-bit. This section describes bit estimation, starting with a model of the measurements, followed by the details of the algorithm.

Figure 46 – Data-bit estimation algorithm

5.3.1   Measurement Model

The algorithm inputs are signal correlations I, which come from correlating the received signal v from equation (50) with a replica code $\overline{v}$, as in GPS [67], [92],

$$I = \sum_{m=k-N_{sc}N_{cx}+1}^{k} v_m \overline{v}_m \quad , \tag{79}$$

Here the number of samples per chip is N_{sc} and the number of chips per code is N_{cx}; assuming the correlation is performed over the length of a single code, their product $N_{sc}N_{cx}$ is the number of signal samples processed by the correlator. The

correlation is high when the signal and replica are aligned and nearly zero when they are not. The correlation peak for a noise-free signal is shown in Figure 47. The correlation peak is located at the time-of-flight delay τ and has a value of $N_{sc}N_{cx}v_0\sqrt{G_D}$. The term v_0 is the amplitude of the received signal, and G_D is the gain between the grating and the detector. The peak is two chipping periods T_c wide. The value of the correlation at the edge of the peak is nearly zero (with a precise value of $-N_{sc}v_0\sqrt{G_D}$).
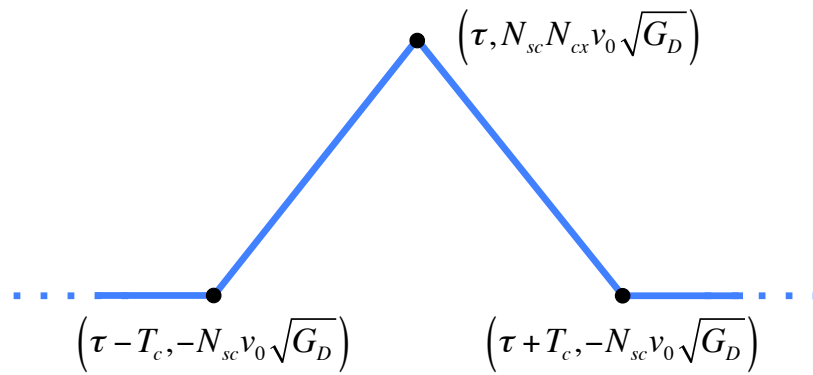
Figure 47 – Correlation peak model

The peak is triangular, with linear sides. For this triangle peak shape, it is straightforward to show that the correlation I for any time shift $\Delta t = (t - \tau) \in [-T_c, T_c]$ is

$$I = N_{sc}v_0\sqrt{G_D}\left(N_{cx} \pm \frac{N_{cx} + 1}{T_c}\Delta t\right) + n_I \,, \tag{80}$$

where the plus is used on the left side of the peak with positive slope, where the minus is used on the right side with negative slope, and where n_I is the noise on an integrator measurement. The noise n_I is the sum of the $N_{sc}N_{cx}$ Gaussian distributed sample voltage noises n_v from equation (50),

$$n_I = \sum_{m=k-N_{sc}N_{cx}+1}^{k} n_{v,m} \overline{v}_m \qquad (81)$$

Assuming independent, Gaussian-distributed samples, the standard deviation σ_I of the integrator noise n_I is

$$\sigma_I = \sqrt{N_{sc}N_{cx}}\sigma_v , \qquad (82)$$

where σ_v is the standard deviation of the sample noise n_v. Note σ_v for this work is treated as an equipment parameter, given as 7.7 mV from Figure 13. Because the diffraction grating is used here, the signal amplitude is $v_0 G_D$ (as described by equation (50) in Chapter 4) and the amplitude constant v_0 from equation (12) in Chapter 2 is updated to

$$v_0 = G_R \sqrt{p_m} , \qquad (83)$$

where G_R is the receiver gain from Table I and the peak power density p_m is calculated from equations (42) and (43) from Chapter 3.

For range estimation, described in the following chapter, the goal is to identify the location of the peak τ, which is the time-of-flight delay. This requires multiple correlator measurements to provide some information about the shape of the peak at a given moment. In GPS, it is typically done using a Delay-Locked Loop (DLL) with two correlator taps, an early and a late. For both, the same raw signal data v is correlated in equation (79), but in once case the replica $\overline{v}$ is shifted slightly toward the early side, and in the other it is shifted slightly toward the late side. This results in two correlator measurements, which are located just to the left

and right of the correlation peak. By taking the difference of these two correlators, the algorithm can determine whether the time-of-flight estimate $\hat{\tau}$ is too short or too long and adjust accordingly.

For both the data-bit estimation, as described below, and the bearing-angle estimation, as described in the following chapter, the goal is to determine the height of the correlation peak $N_{sc}N_{cx}v_0\sqrt{G_D}$. Conceptually, the height could be found from a single correlator tap taken at the time shift $\Delta t = 0$; however, because of uncertainty in the time of flight estimate $\hat{\tau}$, this correlator will not actually lie at the peak, introducing additional error. Instead, this work proposes using a set of correlator taps located at positions $t = \left[t_1, t_2, \dots t_{N_t}\right]$, where N_t is the number of correlator taps. These taps are scattered around the peak, and they can then be used to try to fit the model to data. A minimum of $N_t = 2$ taps are needed to provide observability, but conceptually any number of taps could be used. As described below, a larger number of taps will help to elucidate the peak in the presence of noise.

It should be noted, that correlator taps that are located within one chip T_c of each other will have partially correlated noise. Consider first the case of two correlator taps (i and j) taken at distant locations ($\left|t_i - t_j\right| > T_c$). The noise values n_v on the signal v are random numbers, and they are being multiplied by a pseudorandom code $\bar{v}$ in equation (81), so although both correlator values are derived from the same set of raw measurements, the noises n_{I_i} and n_{I_j} will be uncorrelated. Now, consider the case where both correlators are co-located ($t_i = t_j$). In this case, the random noise values n_v will be multiplied in equation (81) by the same pseudorandom code values $\bar{v}$ each time, resulting in equivalent output noise $n_{I_i} = n_{I_j}$, so they are perfectly correlated. Finally, consider the case where the taps close but not co-located ($\left|t_i - t_j\right| < T_c$). In this case, some of the raw signal noise values n_v will overlap and be multiplied by the same pseudorandom numbers in both correlators, and the rest will not. The correlation, in this case,

122

will depend on the amount of overlap, which varies linearly with the separation distance $|t_i - t_j|$. The noise correlation between two correlators located at t_i and t_j thus is described by the constant ρ_{ij}, given as

$$\rho_{ij} = \begin{cases} 1 - \left| \dfrac{t_i - t_j}{T_c} \right| & |t_i - t_j| \leq T_c \\ 0 & \text{otherwise} \end{cases} , \qquad (84)$$

where the minimum value of ρ_{ij} is 0 and the maximum value is 1.

5.3.2   Bit Estimation Algorithm

As mentioned above, BPSK is used to modulate the data bits onto the code. For a positive bit, the code is multiplied by +1, and for a negative bit the code is multiplied by -1, flipping all of the chips. As a result, the correlation peak flips in the negative direction for negative data bits, which is equivalent to setting the v_0 parameter to be negative. Data-bit estimation, therefore, comes down to estimating the sign of the signal amplitude v_0.

Because the bits are transmitted twice, there are two separate instances of each individual bit. Two bit estimation methods are considered. One approach is to fuse the measurements from both instances into a single estimation algorithm. This leverages all of the available information, and therefore is expected to minimize the likelihood bit corruption, but requires that the diffraction grating gain term be estimated from the positioning algorithm, since the two instances occur at different points in the grating sweep. Another approach is to handle each instance separately, and then individually check each word for errors. In this method, fewer measurements are used in each case, which is expected to result in a higher likelihood of bit corruption; however, because the data-bit rate (~651 bps) is nearly 100 times faster than the diffraction grating sweep rate (7 Hz), the diffraction grating can be assumed stationary over the length of a single bit. As a

result, the diffraction grating gain term can be treated as a constant and bit estimation can be reduced to finding the sign of the power-scaling factor $B = v_0\sqrt{G_D}$. For simplicity in this work we consider the second option.

In the proposed algorithm, multiple correlator taps are used, where the correlation is calculated for different alignments of the signal and replica code at each time step. They form a group of correlator taps $t = [t_1, t_2, \ldots, t_{N_t}]$, surrounding the correlation peak, which can be used to estimate its location and height. An example is shown in Figure 48, where the blue line represents the correlation peak and the red dots represent the correlator tap locations.
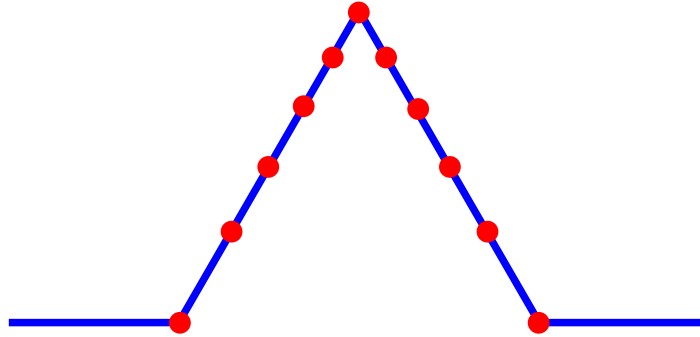
Figure 48 – Example correlator tap locations

In the following algorithm, we will use 15 correlator taps, evenly spaced between $\Delta t = \pm 0.8T_c$. This is many more taps than are typically used in GPS, but with the readily achieved in a software-defined receiver. This set of correlators is chosen to ensure reasonable coverage of the correlator peak. Optimization of the number and spacing of correlator taps is left to future work.

Note, in this chapter, it is assumed that accurate estimates $\hat{\tau} \approx \tau$ are available. These estimates come from the positioning algorithm, which is performed in parallel with data-bit estimation and described in the next chapter. Relaxation of this assumption is considered in the discussion section below.

The set of correlator measurements at a given time step k are assembled into a vector $\mathbf{I}_k \in \mathbb{R}^{N_t \times 1}$

$$\mathbf{I}_k = [I_{k,1} \quad \cdots \quad I_{k,N_t}]^T , \tag{85}$$

where the component $I_{k,i}$ is the measured correlator value at the epoch k for the ith correlator tap and N_t is the number of correlator taps. The noise covariance $\mathbf{\Sigma}_I \in \mathbb{R}^{N_t \times N_t}$ on these measurements is given by

$$\mathbf{\Sigma}_I = \sigma_I^2 \mathbf{\rho} . \tag{86}$$

where the matrix $\mathbf{\rho} \in \mathbb{R}^{N_t \times N_t}$ represents the correlation between the measurements from the taps $t = \left[t_1, t_2, \dots, t_{N_t}\right]$ and is assembled from the individual correlation values ρ_{ij} from equation (84).

In the absence of noise, the measurement $I_{k,i}$ is directly proportional to the data value B_{k_b}, with a constant of proportionality

$$V_i = N_{sc} \left(N_{cx} \pm \frac{N_{cx} + 1}{T_c} \Delta t_i \right), \tag{87}$$

where again Δt_i is position of the ith tap (located at t_i) relative to the peak and the plus is used for the region with positive slope and the minus for the region with negative slope, as described by equation (80). Just like the correlator measurements in equation (85), these can be assembled into a vector $\mathbf{V}_k \in \mathbb{R}^{N_t \times 1}$,

$$\mathbf{V}_k = [V_{k,1} \quad \cdots \quad V_{k,N_t}]^T .\tag{88}$$

Each data bit is composed of multiple repetitions of the code at different times k. For data-bit estimation, all of the measurements $\mathbf{I}_k$, sampled at different times within the same data bit, are assembled into a single measurement vector $\mathbf{y}_{k_b} \in \mathbb{R}^{(N_t N_{xd}) \times 1}$, defined as

$$\mathbf{y}_{k_b} = \left[\mathbf{I}_{k_b - N_{xd} + 1}^T \quad \cdots \quad \mathbf{I}_{k_b}^T\right]^T ,\tag{89}$$

where k_b is the epoch marking the end of a data bit and N_{xd} is the number of code repetitions per data bit, in this case 3, as described in Table IV.

As mentioned above, it is assumed that the diffraction grating barely moves over the length of a code chipping period T_c, and so the problem can be reduced to estimating the power-scaling factor B_{k_b} at the epoch k_b. The measurement model relates this factor to the measurements $\mathbf{y}_{k_b}$ via

$$\mathbf{y}_{k_b} = \mathbf{H}_{k_b} B_{k_b}\tag{90}$$

Where $\mathbf{H}_{k_b} \in \mathbb{R}^{(N_t N_{xd}) \times 1}$ is the normalized measurement model vector, which is assembled from the individual vectors $\mathbf{V}_k$,

$$\mathbf{H}_{k_b} = \left[\mathbf{V}_{k_b - N_{xd} + 1}^T \quad \cdots \quad \mathbf{V}_{k_b}^T\right]^T .\tag{91}$$

Note that the constant of proportionality in equation (87) does not depend on the epoch k. As a result, all of the vectors $\mathbf{V}_k$ are equivalent and the subscript k can be dropped, making the normalized measurement model vector just $\mathbf{H}$.

To find an estimate $\hat{B}_{k_b}$ of the data value, the measurement equation (90) is solved using weighted least squares (WLS),

$$\hat{B}_{k_b} = (\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{R}^{-1} \mathbf{y}_{k_b} \,, \tag{92}$$

where $\mathbf{R} \in \mathbb{R}^{(N_t N_{xd}) \times (N_t N_{xd})}$ is the measurement covariance matrix, which is assembled from N_{xd} diagonal repetitions of the matrix $\mathbf{\Sigma}_I$. Note that because $\mathbf{H}$ is a vector, the term in the parenthesis here is a constant. Furthermore, because $\hat{B}_{k_b}$ is a constant, this is essentially a weighted average of the measurements $\mathbf{y}_{k_b}$, where the weights are $(\mathbf{H}^T \mathbf{R}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \mathbf{R}^{-1}$.

The data-bit estimate $\hat{b}_{k_b}$ is found from the sign of the amplitude estimate $\hat{B}_{k_b}$,

$$\hat{b}_{k_b} = \text{sign}(\hat{B}_{k_b}) \,. \tag{93}$$

Once all of the bits in a data sequence have been estimated, the bits are then assembled into words and polynomial division is used to check the CRC values for corrupted bits [93], [94]. If corrupted bits are detected in the first pass of the data word, then the corresponding second pass is checked. Only if both passes are corrupted is the word thrown out and the data considered missed. One advantage of the CRC is that it provides information about the location of the errant bit, which may be used to correct the error. This is left to future work.

Note, that it is assumed in this paper that both receiver and transmitter are connected to atomic clocks that have been perfectly synchronized either on the ground or midair in the presence of GPS (and they have negligible drift over time for lengthy missions in GPS-denied environments). The assumption of perfectly synchronized clocks means that, unlike in GPS, the THz receiver maintains its own precise time standard and knows both the current time and the signal's

prearranged time of transmission exactly. This allows the receiver to independently determine the time-of-flight delay τ and to precisely align the correlators for code correlation and data-bit estimation. Again, a relaxation of this assumption is explored in the discussion section below.

5.4   Performance Analysis

Probabilistic analysis is used to predict the data-bit error rate and data integrity risk for a representative THz system. It is assumed that the double-slit diffraction described in Chapter 3 is mounted on the demonstration equipment from Chapter 2. The relevant equipment parameters are assumed to be those detailed in Table I and Table II with one difference: just as in Chapter 4, it is assumed that the transmit power from the ground-level tests is boosted by a factor 5 ($P_T = 150$ mW) to accommodate the longer distances. The diffraction grating is assumed to be an LCD screen, with the slits moving at constant speed across the front of the device and then abruptly jumping back and starting the motion over again. It is also assumed that a total of 15 correlator taps are utilized, located at positions t_i evenly spaced between $\tau \pm 0.8T_c$.

5.4.1   Data-Bit Error Rate

Data-bit estimation is performed using a WLS algorithm, as described by equation (92). The expected variance σ_B^2 of the Gaussian distributed data-bit value $\hat{B}$ is the constant

$$\sigma_B^2 = (\mathbf{H}^\mathsf{T}\mathbf{R}^{-1}\mathbf{H})^{-1}. \tag{94}$$

The bit error rate can be computed by first finding the probability of a flipped bit (positive/negative) at epoch k_b. For a positive bit, the probability of a bit error γ_{be,k_b} is the probability that the estimated value $\hat{B}_{k_b}$ is negative

$$\gamma_{be,k_b} = \mathrm{P}\big(\hat{B}_{k_b} < 0\big). \tag{95}$$

This probability can be found by evaluating the cumulative distribution function (CDF),

$$\gamma_{be,k_b} = \Phi_{v_0\sqrt{G_D},\sigma_{\hat{B}}^2}(0), \tag{96}$$

where $\Phi_{v_0\sqrt{G_D},\sigma_{\hat{B}}^2}$ is the CDF of a Gaussian distribution with the mean $v_0\sqrt{G_D}$, which is the expected value of the amplitude estimate $\mathrm{E}\big[\hat{B}_{k_b}\big]$, and standard deviation σ_B. The CDF is evaluated at 0, the point where the bit flips from positive to negative. Note that the probability of a negative bit flipping is the same as that of a positive bit.

The probability of an error is highest when the distance between the receiever and transmitter is high. This dependence occurs because the transmitter factors into v_0, as per the link budget from equations (83) and (41) from Chapter 3.

The probability of an error is also increased for certain bearing angles between the receiver and transmitter, as the angle ϕ affects the the diffraction grating gain G_D. When the diffraction grating is positioned in a region of destructive interference, the gain G_D is low, and the probability of a bit error is high.

The probability γ_{we,k_w} that the word at epoch k_w is corrupted by errant bits is the product of the 20 individual bit error probabilities associated with that word. Because the correlator windows do not overlap (such that the noise values at different times are assumed to be independent), the word error probability is

$$\gamma_{we,k_w} = 1 - \prod_{k_b=k_w-19}^{k_w} \left(1 - \gamma_{be,k_b}\right). \tag{97}$$

This is the probability that the word contains corrupted bits, but because the word is repeated twice, the receiver gets two chances to receive it error free. If the first repetition of the word is corrupted, then the algorithm goes on to the second.

The probability $\gamma_{mw,l}$ that the word is missed completely, therefore, is the probability that both word repetitions contain corrupted bits. It is the product of the two individual word probabilities

$$\gamma_{mw,l} = \prod_{k_w=l,l+2} \gamma_{we,k_w}, \tag{98}$$

where l is the index of the word, and where the product considers a pair of repeated words separated by an interval in which a different word is broadcast (see Figure 45). Note, that if both bit repetitions were incorporated into a single bit-estimation algorithm, equation (98) would be rendered irrelevant.

The value $\gamma_{mw,l}$ is the risk that a transmitted measurement will contain an error and be unusable. This risk can be thought of as similar to the continuity risk used in analyzing safety-of-life aviation systems (such as GBAS [95], [96]).

This risk is plotted in Figure 49 for a variety of different bearing angles and message-grating alignments. The x-axis is the bearing angle ϕ to the transmitter. The y-axis is the position Δ of the diffraction grating at the time the first bit in the message is received. Blue regions represent low probabilities of a missed word due to corrupted bits, and yellow represents high probability. It is assumed that the transmitter is 2 km away, which is a worst case scenario representing roughly

the maximum distance separating aircraft in a typical formation element, and the grating frequency is the Goldilocks frequency $F_g = 7$ Hz.
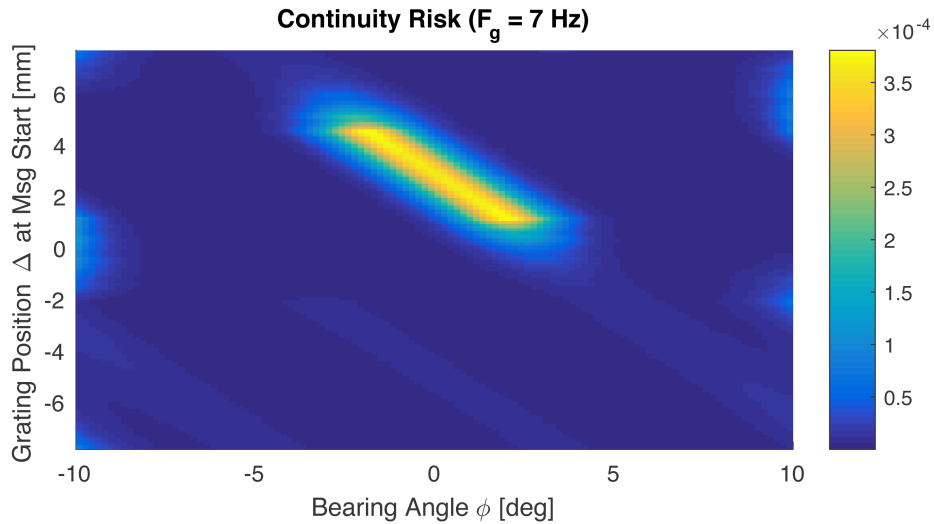
Figure 49 – Probability of errant word at $r = 2$ km

The highest missing-word probability observed over this range is roughly 0.04%, meaning that there is for a worst-case configuration of the receiver and transmitter a roughly 0.04% chance that at a given communication epoch data will be lost in transmission. Because the grating frequency and message frequency are not locked, the grating position Δ at the time of message arrival will drift over time. As a result, the probability of missing a data word over time for a particular bearing angle can be calculated as the mean of the vertical slice at that angle. For $\phi = 0°$, this gives a 0.005% chance of missed words.

It should be noted that the maximum value is observed only over a limited subset of bearing angles ϕ centered on 0°. From Figure 45, we see that the transition from one grating sweep to the next results in an extended period of low gain in the 0° case. When the message aligns unfavorably, there is a higher likelihood of missed words because both repetitions fall during periods of low gain. This extended low-gain region exists because the grating sweep range $\pm\zeta_{max}$ was conservatively defined in Chapter 3, equations (45) and (46), to account for

131

increased peak spacing at large bearing angles. This results in the extended tails seen in this case and has a clear impact on communication. For operations limited to small bearing angles, it would likely be advantageous to select a tighter grating sweep range to reduce the likelihood of missed data bits. This is left to future work.

For comparison, Figure 50 shows the risk of missed words for the two sub-optimal grating frequencies ($F_g = 2$ Hz and $F_g = 15$ Hz, aka Mama and Papa bear). Again, the x-axis is the bearing angle ϕ and the y-axis is the position Δ of the diffraction grating when the first bit arrives. Blue is low probability of missed detection and yellow is high probability. In both cases, it is clear the peak probability of error is significantly larger than it was in the baby bear case ($F_g = 7$ Hz), just short of 1% in the 2 Hz case and over 0.3% in the 15 Hz case. Taking the mean of the vertical slices at 0°, both produce expected bit error rates of roughly 0.12%, more than an order of magnitude greater than the Goldilocks case.
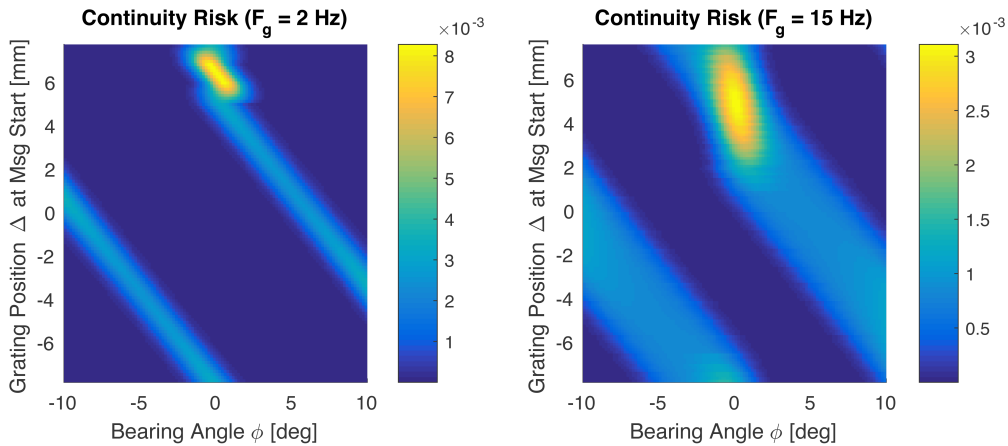
Figure 50 – Probability of errant word at $r = 2$ km for the too slow case (left) and too fast case (right)

5.4.2 Data Integrity Risk

There is also a corresponding integrity risk, in which a word is corrupted but the aircraft is not aware of the problem, because the CRC fails to catch the error. The

132

CRC is only guaranteed to catch single-bit errors, or error burst that span fewer than three bits (the length of the CRC). To determine the risk of missed detection, then, start by considering the risk of there being two bit errors in a single word. We will conservatively consider all of these as a risk for missed detection, even though the CRC is guaranteed to catch the small fraction of cases where the errors occur within three bits of each other. The probability γ_{w2e,k_w} of two bit errors is

$$\gamma_{w2e,k_w} =$$

$$\sum_{k_{b,1}=k_w-19}^{k_w} \left(\sum_{k_{b,2}=k_w-19 \neq k_{b,1}}^{k_w} \left(\frac{\frac{\gamma_{be,u}\gamma_{be,v}}{\left(1-\gamma_{be,k_{b,1}}\right)\left(1-\gamma_{be,k_{b,2}}\right)}}{\prod_{j=k_w-19}^{k_w} \left(1-\gamma_{be,j}\right)} \right) \right) \tag{99}$$

where $k_{b,1}$ and $k_{b,2}$ are the indices summed over. Similarly, it is possible to calculate the probability of three or more bit errors in a word. The full probability of missed detection comes from the summation of all of the higher order missed bit probabilities, but the series is dominated by the first term given in equation (99), so that is sufficient as an approximation.

For these multi-bit errors, the CRC will catch a large fraction of them, but is not guaranteed to catch them all. For words containing multiple widely spaced errant bits, the CRC will miss detection $1/(1-2^{-n}) = 12.5\%$ of the time, where $n = 3$ is the length of the CRC [97]. The probability that a corrupted word goes undetected γ_{cw} (i.e. the integrity risk) thus is calculated as

$$\gamma_{cw,k_w} = 0.125 * \gamma_{w2e,k_w} \, , \tag{100}$$

Because the data words are repeated, some corrupted words that slip by the CRC may still be detected by comparing the two repetitions. This would further suppress the integrity risk, but it left to future work. For now, we consider the value given in equation (99) as a conservative model.

Figure 51 shows the integrity risk as calculated from equation (99) at 2 km for the grating frequency $F_g = 7$ Hz. Again, the x-axis is the bearing angle ϕ and the y-axis is the position Δ of the diffraction grating when the first bit arrives. Blue is low probability of missed detection and yellow is high probability.
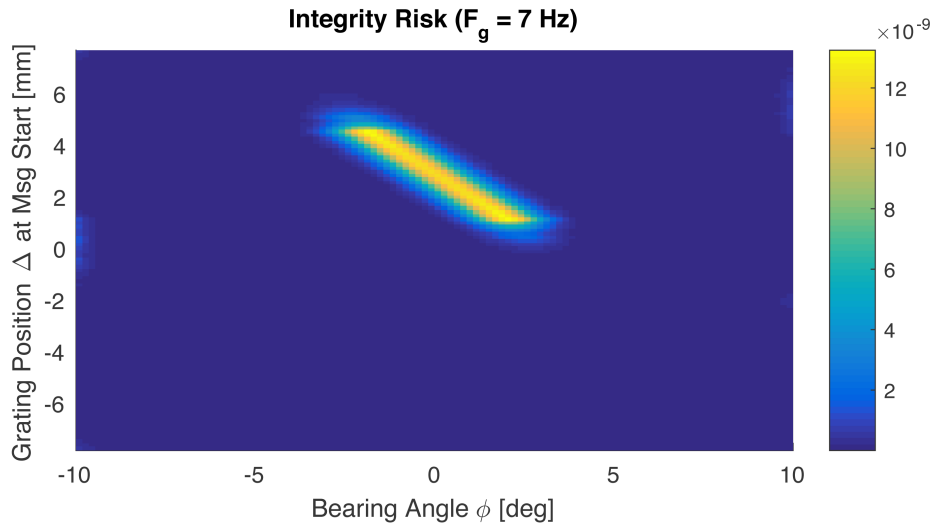
Figure 51 – Probability of undetected bit error at $r = 2$ km

It is immediately obvious that the integrity risk is much lower than the continuity risk. Again, the highest probability is observed at around 0°, with missed detection peaking at a little over 1 in 10^8. Because it is so low, the integrity risk is not considered a primary concern in this work.

5.5   Discussion

A key assumption in this work is that accurate time-of-flight estimates $\hat{\tau} \approx \tau$ are available from the positioning algorithm. As a result, the correlator tap positions are constant relative to the peak. Relaxing this assumption, the tap locations may slide from side to side relative to the peak. If they slide too far, the peak may pass out of the field of view; however, as described in the chapter below, this is not expected. Even small shifts, though, can result in mismatches between the measurement model and the measurements, which could cause higher rates of data-bit estimation errors. Results from the positioning algorithms in the following chapter show that range estimates are highly accurate, with errors less than 5 m, two-sigma. Given that a code chip is 150 m long, this results in the peak shifting of roughly 3% or less, which will have minimal impact on the data-bit estimation.

Another key assumption is that the aircraft are equipped with perfectly synchronized atomic clocks, and that the code sequence and data bit broadcasts are scheduled at predetermined times. As a result, the following aircraft knows exactly when each code sequence was sent allowing it to precisely compute the signal's time-of-flight. In addition, it knows exactly when each data bit was sent, allowing it to integrate the correlators across whole data bits, maximizing the energy received, unlike GPS, where the data boundary is unknown, limiting the integration times.

Relaxing this assumption, consider the case where the clock synchronization is imperfect and clock drift is present. This will result in bias between the clocks that may grow over time. Assuming that the clocks are initially synchronized from GPS, errors of roughly 10 ns are possible. In addition, inter-clock bias growth of roughly 5 ns can be expected from commercially available Cesium frequency standards [98] over a 6-hour period.

A 15 ns clock bias will result in a roughly 4.5 m bias in the position estimate from the positioning algorithm, but should have no effect on the bit estimation algorithm as long as tracking is maintained This is because bit estimation does not depend on the peak location, only its height, so it can tolerate peak shifts from clock bias. Potential problems could arise if ambiguity slip were a threat. Because the code is the same every time, the algorithm has no way to tell the difference between the repetitions, and in theory it is possible to slip between codes, resulting in a misalignment of the data-bit integration. This, however, is not a threat to the current system, as the period of each code is over 500 μs, which corresponds to a code length of over 150 km, well beyond the range of the THz transmitter. As a result, there is no ambiguity and likewise no risk of ambiguity slip, even for relatively large clock biases.

5.6   Conclusion

A customized GPS-like signal and data bit structure are presented for THz communication and relative positioning using the interferometric receiver. This structure provides the capability for simultaneous communication and navigation with multiple transmitters mounted on aircraft flying in formation, and is specifically designed to overcome the limitations of THz hardware and unique challenges of working with the diffraction grating. Probabilistic analyses show that this structure and the corresponding bit estimation algorithm are capable of reliably transmitting and decoding messages, with measurements lost only about once every 2500 transmissions at separations of 2 km. Furthermore, the communications are shown to have high integrity with very low probability of corrupted communications going undetected.

Chapter 6

Integrated THz Relative Positioning Simulation

6.1 Introduction and Motivation

The preceding chapters present methods for measuring the range r and bearing angle ϕ from the THz signal, as well as methods for communicating altimetry data over the THz link to make the relative altitude h observable. This chapter fuses those measurements together to produce 3D position estimates. The chapter then shows the resulting positioning accuracy is compatible with the requirements for formation flight applications.

Chapters 2-5 established a THz time-of-flight ranging capability and developed an error model for THz measurements, introduced a new diffraction-based sensor to make bearing angle observable and designed the device to mitigate power losses, developed methods for solving the inverse problem and mapping the interference pattern to the bearing angle, and proposed a signal structure and a frequency matching scheme to enable reliable communication of key parameters despite intermittent signal fading due to the diffraction grating. The technical question then that remains to be addressed is whether or not these disparate elements can be fused into a system capable of meeting the requirements for precision airdrop.

This chapter integrates all of the components discussed so far in the dissertation into an algorithm and uses a single unified simulation to verify the THz system's capabilities. The sensor-fusion algorithm used in this chapter is a conventional Bayesian filter with a sensor model customized to the available data. Using the measurements of range, bearing angle, and relative altitude described above, an Extended Kalman Filter (EKF) produces precise relative position estimates. It utilizes a simple kinematic model, propagating the velocities to predict changes in the position. Simulations encapsulating the signal propagation, measurement error modeling, and both the data-bit and position estimation algorithms provide performance analysis of the complete system, and demonstrate that the system is easily able to meet the requirements for the precision airdrop application.

6.2   Kalman Filter Positioning Algorithm

6.2.1   Problem Statement

The goal of the navigation algorithm presented in this chapter is to estimate the three-dimensional position of the transmitting aircraft relative to the following aircraft using measurements of the THz signal and communicated altimetry measurements. As mentioned above, this dissertation focuses on the position estimation problem for an individual following aircraft, as shown in Figure 9, and the key positioning requirement is the cross-track requirement of 50 m, two-sigma.

The positioning algorithm estimates seven dynamic states: the position states x, y, and z, their derivatives $\dot{x}$, $\dot{y}$, and $\dot{z}$, and the signal amplitude state v_0, expressed as a state vector

$$\mathbf{x} = [x \quad y \quad z \quad \dot{x} \quad \dot{y} \quad \dot{z} \quad v_0]^T \, . \tag{101}$$

Note that in this dissertation, it is assumed that the aircraft fly straight and level. As a result, the body-fixed and gravity-fixed coordinate frames are aligned, and z points along the axis of the gravity vector. As a result, the bearing angle measurement lies in the horizontal plane and is perpendicular to the altitude measurements. A relaxation of this assumption is considered in the discussion.

6.2.2   Algorithm

Position estimation and bit estimation are performed independently of each other. Bit estimation is performed as described in in Chapter 5 and shown in Figure 46. A batch WLS process is still used here, because the duration of the bit so short it is essentially instantaneous relative to the motion of both the aircraft and the diffraction grating. Once all of the bits in a data word have been process, the message CRC value is checked using polynomial division, as described above. If the value matches, then altimetry measurements are available for that epoch and incorporated into the positioning algorithm. If it does not, then the corresponding measurement is thrown out.

Figure 52 graphically depicts the Kalman-filter positioning algorithm. THz correlator measurements, which are tied to known grating positions, are fused with communicated altitude and altitude rate measurements. These integrated measurements are processed through an Extended Kalman Filter (EKF) [90], [99], to produce estimates of the aircraft position states.
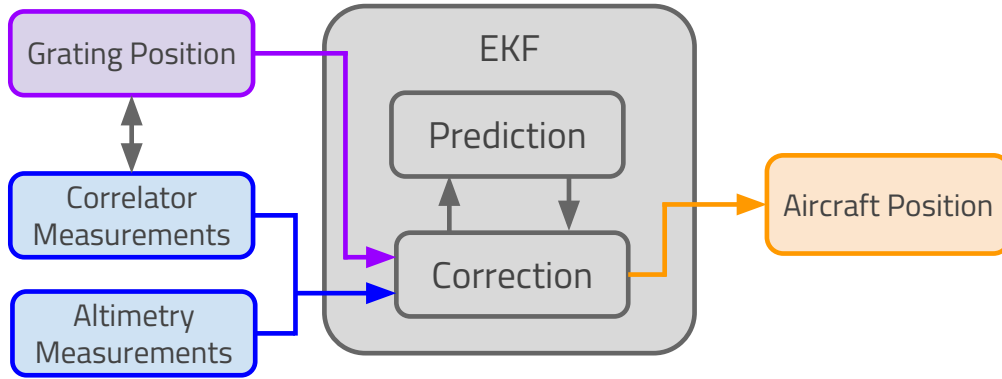
Figure 52 – Relative positioning algorithm

In the prediction phase, the EKF uses a system model (described below in equations (102) and (103)) to propagate the states forward in time. This allows the algorithm to account for aircraft motion over time as it filters the state estimates. The uncertainty in this propagation is captured by the process noise, which is assumed to be a random uncorrelated Gaussian process. In reality, disturbances in the aircraft position have some correlation over time. As a result, the process noise in the following simulations is artificially inflated to make the system robust to aircraft dynamics. Modeling of the process noise as correlated in time is left to future work.

In the correction phase, the EKF compares the measurements to the measurement model (described below in equations (112) and (113)) to determine the accuracy of the current state estimate. The known diffraction grating positions are used to calculate the measurement model.

It should be noted that the THz integrator and communicated altimetry measurements have different sample rates, and that the altimetry sample rate may be variable if data communications are unreliable. Integrator measurements are taken for every code that is received, and so they are made available to the algorithm every filter period T_x. Altimetry measurements, on the other hand, are only available following a successful transmission. The data-bit structure

described in Chapter 5, sets the transmission rate at roughly once a second (strictly every $N_{xd}N_{dmsg}T_x = {\sim}0.98$ s, where $N_{dmsg} = 640$ is the number of data bits between messages). When both THz integrator and communicated altimetry measurements are available, they are fused together in the correction phase. When altimetry measurements are not available, however, the correction only applies integrator measurements, leaving the vertical position unobserved. As a result, the algorithm continues to propagate the state forward based on its model and accumulating uncertainty, until an altimeter measurement becomes available.

6.2.3 System Dynamic Model

The states $\mathbf{x}$ are propagated using a simple kinematic model, where the velocities and the nuisance parameter are assumed to be constants subject to some random uncorrelated Gaussian process noise $\mathbf{q} = \left[q_{\dot{x}}, q_{\dot{y}}, q_{\dot{z}}, q_{v_0}\right]^T$. This simple model is feasible because the relative positions of cargo aircraft are not expected to change quickly, with movements happening over a couple seconds. For a more dynamic formation, a precise dynamic model is necessary. In this case, the discrete time model is described by

$$
\mathbf{x}_{k+1} =
\begin{bmatrix}
1 & 0 & 0 & T_x & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & T_x & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & T_x & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\mathbf{x}_k +
\begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\mathbf{q}_k,
\tag{102}
$$

where T_x is the period of code integration, the first matrix is the state matrix $\mathbf{F}$, and the second matrix is the process noise matrix $\widetilde{\mathbf{G}}$. Note, it is assumed that the EKF has the same duration T_x of one code length, because that is the rate at which the receiver is assumed to make correlator measurements.

A key assumption of your proposed algorithm is that the relative position of the aircraft doesn't change fast (over some time period – a second or so?). That allows you to use a simple kinematic model to propagate the EKF, which in turn allows for longer effective integration times as compared to the batch algorithm introduced in chapter 4.

Each of the elements of the process noise $\mathbf{q}_k$ are assumed to be independent, with covariance described by

$$\mathbf{Q} = \begin{bmatrix} \sigma_{\dot{x}}^2 & 0 & 0 & 0 \\ 0 & \sigma_{\dot{y}}^2 & 0 & 0 \\ 0 & 0 & \sigma_{\dot{z}}^2 & 0 \\ 0 & 0 & 0 & \sigma_{v_0}^2 \end{bmatrix} \tag{103}$$

where $\sigma_{\dot{x}}^2$, $\sigma_{\dot{y}}^2$, and $\sigma_{\dot{z}}^2$ are the variances of the velocity process noise $q_{\dot{x}}$, $q_{\dot{y}}$, and $q_{\dot{z}}$, $\sigma_{v_0}^2$ is the variance of the amplitude state process noise q_{v_0}. The process noise variances $\sigma_{\dot{x}}^2$, $\sigma_{\dot{y}}^2$, $\sigma_{\dot{z}}^2$, and $\sigma_{v_0}^2$ represent the propagation uncertainty, and they act as tuning parameters for the filter. For a system with well-characterized dynamic behavior, tight bounds can be set for the process noise values, which tell the algorithm to trust the prediction, effectively increasing the integration time and suppressing measurement error. For a system whose dynamics are more uncertain, conservative process noise values should be selected, telling the algorithm to listen more to measurements, which increases robustness to modeling errors, but decreases the integration time resulting in noisier position estimates. The assumption of uncorrelated Gaussian noise is an over simplification of the true aircraft dynamics, because disturbances in the aircraft states from wind gusts, engine thrusts, etc. are correlated in time.

As a result, conservative process noise values are selected for the simulation below, as shown in Table V. In the simulations described below, random aircraft

motion results in accelerations on the order of 1 m/s². Given that the period T_x of the filter is roughly 5×10^{-4} s, this results in velocity changes between time steps of roughly 5×10^{-4} m/s; however, akin to real-life disturbances, the simulation motion model results in correlated processes over a period of roughly 6 s. To accommodate this, a value of 0.2 m/s is selected for the velocity state process noise. Correlated noise, likewise, is an issue for the amplitude states process noise. For the aircraft configuration and motion model described below, the absolute amplitude state v_0 is roughly 1×10^{-4} V, and has a maximum change over the course of the simulation of roughly 3×10^{-8} V. As a result, a value of 10^{-8} V is selected for the amplitude state process noise.

Table V – Selected process noise values

Velocity process noise	$\sigma_{\dot{x}} = \sigma_{\dot{y}} = \sigma_{\dot{z}} = 0.2$ m/s
Amplitude state process noise	$\sigma_{v_0} = 10^{-8}$ V

6.2.4 System Measurement Model

Two types of measurements are integrated into the positioning algorithm: THz integrator measurements and communicated altimetry measurements. As described in the chapters above, the range r and bearing angle ϕ are both made observable by the integrator measurements I taken of the THz signal. The model for the measurement I_i corresponding to the ith integrator tap can be written as

$$I_i = N_{sc} v_0 \sqrt{G_D(\phi, \Delta)} \left(N_{cx} \pm \frac{N_{cx} + 1}{T_c} \left(t_i - \frac{r}{c} \right) \right) + n_I ,$$
(104)

as described in equations (80) and (85) from Chapter 4, where t_i is the location of the ith integrator tap, and the time of flight is $\tau = r/c$, where r is the range and c is the speed of light. This equation is a nonlinear function of the transmitter position (x, y, z), the nuisance parameter v_0, and the known grating position Δ

143

(i.e $I = f(x, y, z, v_0, \Delta)$). The position dependence comes in via the range r and the bearing angle ϕ, where

$$r = \sqrt{x^2 + y^2 + z^2} \quad \text{and} \tag{105}$$

$$\phi = \tan^{-1}\left(\frac{y}{x}\right). \tag{106}$$

Here, the EKF solves the inverse problem of angle estimation sequentially instead of as a batch process, as in Chapter 4, with the assumption that state changes are small and driven primarily by process noise.

A simple scheme is

As described above, the communicated altimetry measurements include the lead aircraft's barometric altimeter H_{lead} and altimeter rate $\dot{H}_{lead}$. The following aircraft uses these measurements to calculate the relative altitude h and altitude rate $\dot{h}$ from the difference with its own altitude H_{follow} and altitude rate $\dot{H}_{follow}$,

$$h = H_{follow} - H_{lead} \quad \text{and} \tag{107}$$

$$\dot{h} = \dot{H}_{follow} - \dot{H}_{lead}, \tag{108}$$

where the relative altitude h and altitude rate $\dot{h}$ measurements are related to the vertical position and velocity states as

$$h = z + n_h \quad \text{and} \tag{109}$$

$$\dot{h} = \dot{z} + n_{\dot{h}}, \tag{110}$$

respectively, where n_h is the noise on the relative altimeter measurement and $n_{\dot{h}}$ is the noise on the relative altimeter rate measurement and both are assumed to be zero-mean random Gaussian variables, with standard deviations $\sigma_h = 1$ m and $\sigma_{\dot{h}} = 0.2$ m/s, shown in Table VI, chosen as reasonable approximations of barometric altimeter precision.

<div align="center">Table VI – Barometric altimetry errors</div>

Barometric altimeter error	$\sigma_h = 1$ m
Barometric altimeter rate error	$\sigma_{\dot{h}} = 0.2$ m/s

Note that barometric altimeter measurements H typically have low noise, but are often subject to large bias due to unpredictable and changing atmospheric conditions. In this case, because both aircraft are relatively close, they should be subject to roughly the same atmospheric conditions; therefore, the bias should mostly subtract out in the difference. As a result, relative altitude h and altitude rate $\dot{h}$ measurements are expected to be unbiased with relatively low noises n_h and $n_{\dot{h}}$. Some digitization error will be incurred due to the limited number of bits used, but that is not considered in the following simulation.

These measurements are assembled into a measurement vector,

$$\mathbf{y}_k = \left[\left| I_{k,1} \right| \quad \cdots \quad \left| I_{k,N} \right| \quad h_k \quad \dot{h}_k \right]^T, \tag{111}$$

where k is an index associated with the given epoch and the correlator value $I_{k,i}$ is the measurement associated with the ith tap location and epoch k.

Note that altimetry measurements h and $\dot{h}$ are not available at every epoch k like integrator measurements I. They are only available following the successful transmission from the lead about once a second. For epochs k where they are not available the measurement vector $\mathbf{y}_k$ is truncated by removing the bottom two rows, leaving only integrator measurements. In cases where one of the data words has been corrupted and only one of the altimetry measurements is available, the relevant measurement is kept, while the other is removed.

Also, note that bit wipeoff is applied to the correlator values $I_{k,i}$ using an absolute value. As described in Chapter 5, the signal amplitude v_0 flips positive or negative depending on the data bit. This presents a problem for measurement modeling in the EKF as the bits are not known a priori. The absolute value is a simple means of bit wipeoff that makes all of the correlator peaks positive. The impact of this method and alternative methods are examined in the discussion section.

The integrator measurement model given by equation (104) is nonlinear, so it must be linearized for the EKF. Linearizing the model around the state estimate $\hat{\mathbf{x}}_k^-$ (see chapter appendix for details), the linearized model is given as

$$\delta\mathbf{y}_k = \mathbf{H}_k \delta\hat{\mathbf{x}}_k^+ + \mathbf{r}_k \,, \tag{112}$$

where $\delta\mathbf{y}_k$ is the differential measurement found from the difference between the measurement vector $\mathbf{y}_k$ and the predicted measurements $\hat{\mathbf{y}}_k$ made from the position estimate $\hat{\mathbf{x}}_k^-$, $\mathbf{H}_k \in \mathbb{R}^{(N_t+2)\times 7}$ is the navigation measurement matrix, $\delta\hat{\mathbf{x}}_k^+$ is the differential position estimate, and $\mathbf{r}_k$ is the measurement noise. The measurement noise covariance is given as

$$\mathbf{R} = \begin{bmatrix} \boldsymbol{\Sigma}_I & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_h^2 & 0 \\ \mathbf{0} & 0 & \sigma_{\dot{h}}^2 \end{bmatrix},$$

(113)

where $\boldsymbol{\Sigma}_I$ is the integrator covariance matrix given in equation (86) from Chapter 5, σ_h^2 is the altitude measurement variance, and $\sigma_{\dot{h}}^2$ is the altitude rate measurement variance. Again, for epochs k where altimetry data is not available, the measurement matrix $\mathbf{H}_k$ and the measurement covariance $\mathbf{R}$ are truncated by removing the bottom rows corresponding to the altitude and altitude rate measurements.

Note that the terms of measurement matrix $\mathbf{H}_k$ (given in the appendix) are functions of the known diffraction grating position Δ_k via the diffraction grating gain G_D. It is assumed that an LCD screen produces the diffraction grating slits, and as a result their position is known exactly. The impact of uncertainty in the grating position is left to future work.

6.3   Statistical Performance Analysis

Monte Carlo simulations are used to test the effectiveness of the EKF position estimation algorithm and to assess the system performance relative to the requirements.

6.3.1   Simulation Assumptions

Relative positioning is simulated for both aircraft in the representative formation geometry described in Chapter 1 and shown in Figure 5. For the following aircraft in the AC2 position, the aircraft control position is set to $[x, y, z] = [1000\text{ m}, 200\text{ m}, 0\text{ m}]$, and for the aircraft in the AC3 position it is set to $[x, y, z] = [2000\text{ m}, 100\text{ m}, 0\text{ m}]$. The simulation is run for 100 s.

In both cases, it is assumed that disturbances like wind gusts and turbulence perturb the aircraft positions, and that a suitable control system works to maintain position, resulting in some position "wobble" around the desired position. This is simulated by randomly choosing waypoints around the control position and connecting them using a high order spline. In the following simulations, waypoints are selected every 6 s within 10 m box around the control position and connected with a 6th order spline. An example is shown in Figure 53. On the left is the aircraft track, with "X" marking the initial position of the aircraft, "O" marking the final position, and the black line delineating the track of the aircraft over time. On the right is a plot of the position, velocity, and acceleration in the x-dimension for this motion.
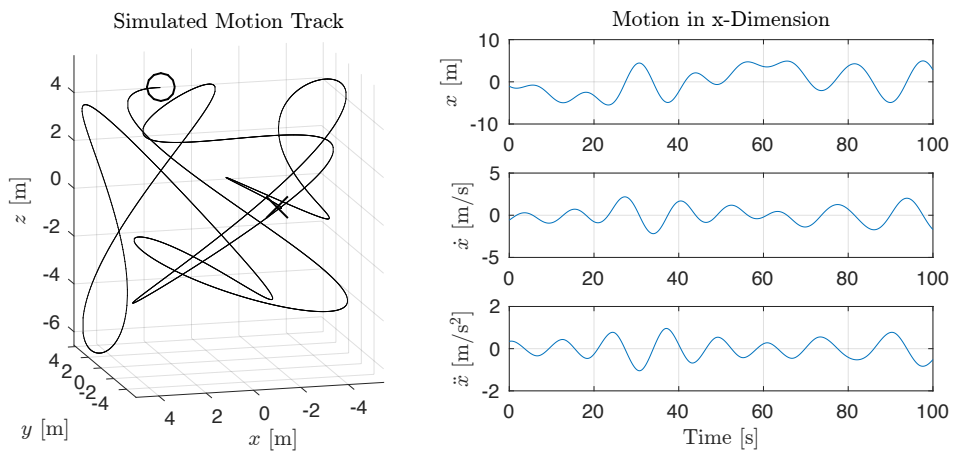
Figure 53 – Example simulated aircraft motion

This motion, while random and related to the Markov noise model, is not identical because the disturbances in the velocity states are correlated over time. As a result, the EKF is slightly detuned, with conservative process noise values selected to ensure robustness to correlated noise, as described above.

It is assumed that the THz and diffraction grating hardware are those described above in Chapter 2 Table I and Chapter 3 Table II, with one exception. As in Chapter 4 and Chapter 5, to accommodate the longer distances associated with

formation flight, it is assumed that the transmit power is 5 times greater than that of the ground-level system ($P_T = 150$ mW). The signal and data structure assumed are the those from Chapter 5, as described in Figure 41 and Table IV.

THz correlator measurements I are simulated at each Kalman-filter time step using the measurement model defined in equation (104), with the signal amplitude v_0 calculated from the link budget as described in Chapter 5 by equation (83), and the measurement noise n_I calculated as a correlated Gaussian distributed random variable with covariance described by equation (86). Again, $N_t = 15$ correlator taps are used, evenly spaced between $\tau \pm 0.8T_c$ to ensure good coverage of the peak.

Relative altitude h and altitude rate $\dot{h}$ measurements are calculated as described by equations (109) and (110). Their noises n_h and $n_{\dot{h}}$ are calculated as zero-mean Gaussian distributed random variables with standard deviations given in Table VI.

In position estimation, the simulated THz and altimetry measurements are integrated together in the EKF. The prediction phase uses the dynamics model from equations (102) and (103) to propagate the estimate and covariance forward. The correction phase compares the measurements to the predicted measurements based on the measurement model given in equations (112) and (113).

Bit estimation occurs in parallel. The simulated THz measurements are processed through the WLS algorithm described in Chapter 5. Equations (92) and (93) are used to generate bit estimates.

It is assumed that the initialization of the position estimate is performed midair via GPS. The filter is initialized in the simulation with position and velocity estimates that contain one-sigma errors conservatively set to $\sigma_{pos,0} = 3$ m in the position states and $\sigma_{vel,0} = 0.5$ m/s in the velocity states. The initial amplitude state estimate is assumed to be roughly calculated from a link budget, with an

initial one-sigma error σ_{v_0} is set to 20% of the true value in simulation. Once initialization is complete, the system runs without GPS.

6.4 Results

The simulation shows that the following aircraft in the AC2 position $[x, y, z] = [1000 \text{ m}, 200 \text{ m}, 0 \text{ m}]$ achieves high precision position estimation using the methods described in this dissertation, as shown in Figure 54. The top plot shows the along-track x position error, the middle plot shows cross-track y position error, and the bottom plot shows the vertical z position error. The blue curve represents the position estimate error and the black lines are the two-sigma error bounds from the Kalman filter algorithm.

Figure 54 – Position estimate error for AC2 simulation $[1000 \text{ m } 200 \text{ m } 0 \text{ m}]$

The two-sigma error bounds for the AC2 relative position estimate are well within the requirements for precision airdrop and contain more than 95% of the position estimate errors, as shown in Figure 55. In addition, data transmission was successful every time in this simulation, with 0 missing measurements over 102 transmissions, as expected from the analysis above.
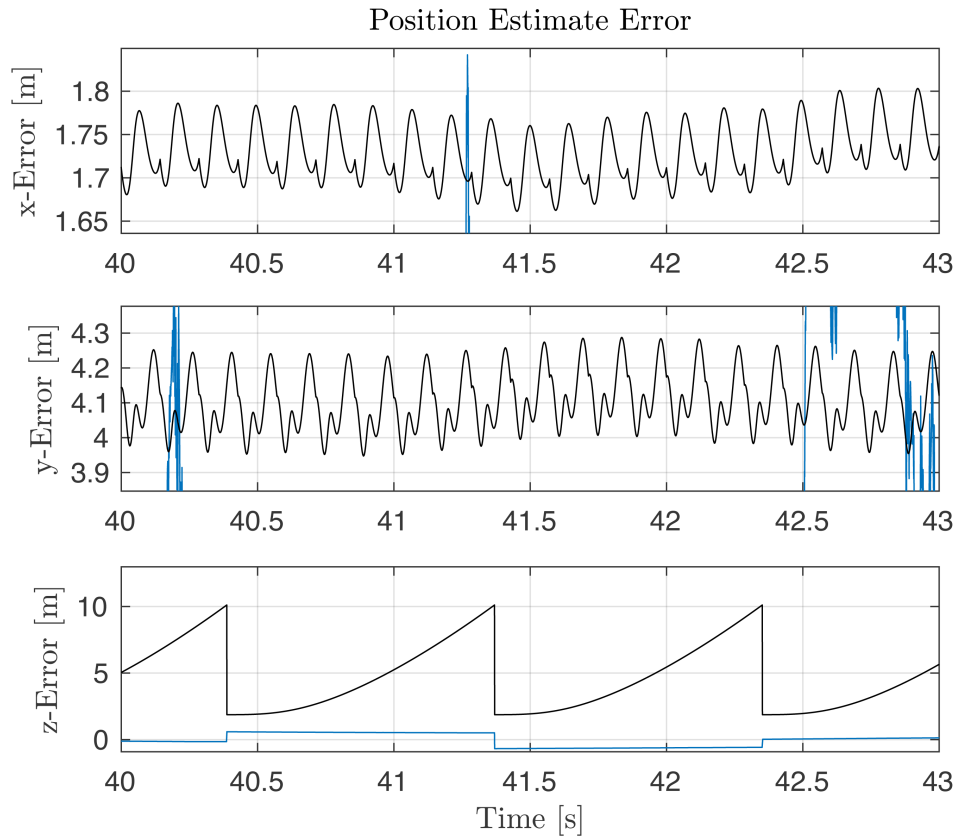
Figure 55 – Zoom in on error bounds for AC2 simulation [1000 m 200 m 0 m]

A periodic, sinusoidal-like pattern is observed in the x and y error bounds with a dominant frequency of 7 Hz (matching that of the diffraction grating sweep $F_g = 7$ Hz). Remember that the power incident on the receiver varies with time as the diffraction grating sweeps in front of the detector, which translates into increases and decreases in the signal-to-noise ratio of the signal and causes the error bounds to fluctuate.

The z error bound exhibits a saw tooth pattern, where errors grow as the algorithm makes successive predictions until a new measurement becomes available, which provides fresh information to the algorithm and clamps down the error. In this case, the saw tooth is particularly pronounced because the altimetry measurements are infrequent (roughly 1 Hz) and the conservative process noise model results in fast model uncertainty growth.

The cross-track y estimate error is roughly 4.1 m, two-sigma, well within the 50 m requirement for precision airdrop applications. Likewise, the along-track x error is roughly 1.7 m, two-sigma, and the vertical z error is roughly 5 m, both well within their requirements of 150 m and 100 m, respectively.

In addition to the AC2 results, the simulation also demonstrates high precision position estimation for the following aircraft in the AC4 position $[x, y, z] = [2000 \text{ m}, 100 \text{ m}, 0 \text{ m}]$, as shown Figure 56. Once again, the top is the x position error, the middle is the y position error, and the bottom is the z position error, with blue representing error and black representing the error bounds.
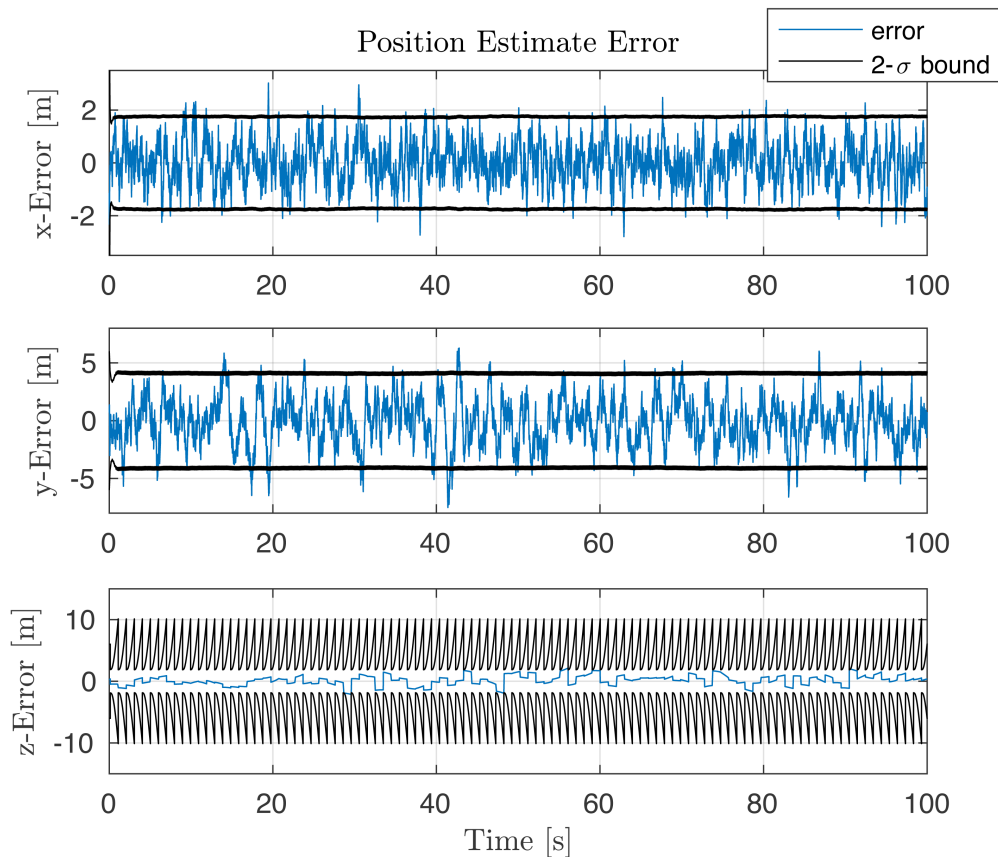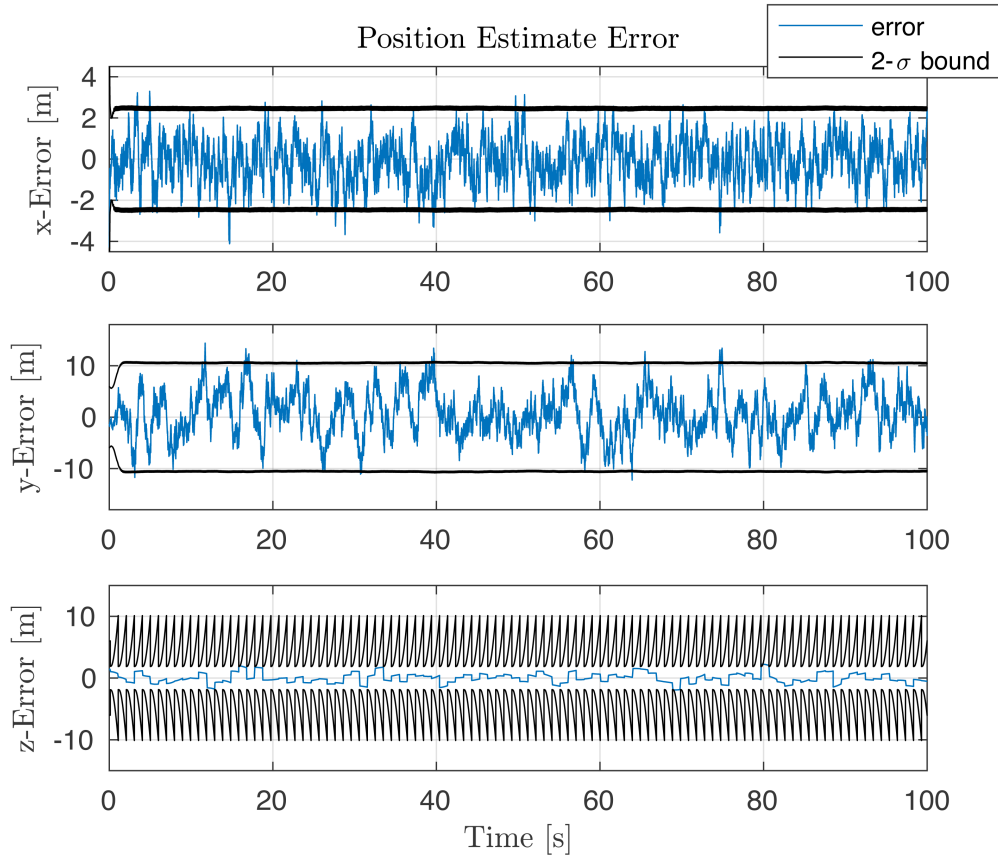
Figure 56 – Position estimate error for AC3 simulation [2000 m 100 m 0 m]

Like above, the two-sigma error bounds for the AC3 relative position estimate are well within the requirements and contain more than 95% of the errors, as shown in Figure 57. Also, like before, zero transmission failures were observed over 102 transmissions, as expected.
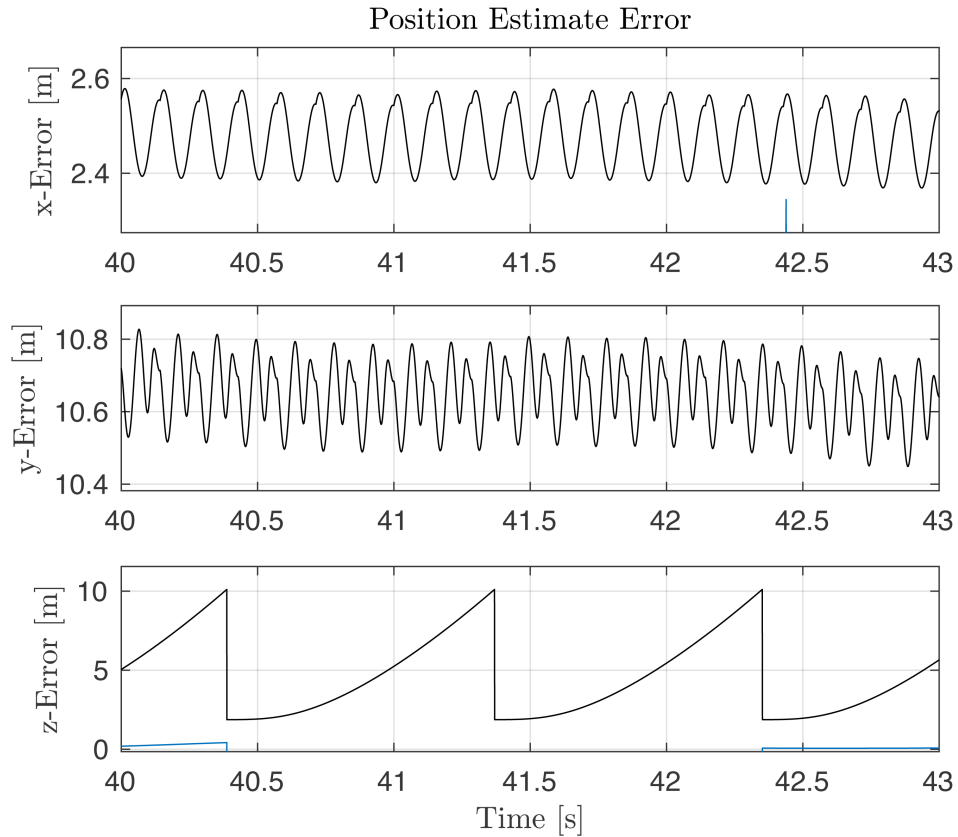
Figure 57 – Zoom in on error bounds for AC3 simulation [2000 m 100 m 0 m]

Although the shapes are different in this case, similar cyclical behavior is observed in the shape of the x and y error bounds due to the diffraction grating sweep, and the same saw tooth pattern is observed in the vertical z error bound. In this case, the cross-track y estimate error is roughly 10.7 m, two-sigma, which although larger than the AC2 case is still well within the 50 m requirement for precision airdrop. The along-track x error is roughly 2.5 m, two-sigma, and the vertical z error is roughly 5 m, also, both well within the requirements.

6.5   Discussion

6.5.1   Position Estimation Accuracy

The position estimate errors in both cases easily meet the requirements for precision airdrop. In the long distance case (AC3), cross-track positioning error of roughly 11 m, two-sigma is observed, well within the 50 m requirement for precision airdrop. This high precision demonstrates the capabilities of the system for precision airdrop applications, and may open the system up to more demanding applications like drag reduction and aerial refueling. In addition, the results justify the assumption that the along-track and vertical positioning requirements are not primary concerns, with both estimate errors well over an order of magnitude less than the requirements.

As the distance roughly doubles going from AC2 to AC3, the positioning errors also roughly double, as described in Chapter 4, due primarily to spreading losses. From Figure 55 and Figure 57, when moving from the AC2 position (~1 km) to the AC3 position (~2 km), the along-track x error goes from roughly 1.7 m to 2.5 m, and the cross-track y error goes from roughly 4.1 m to 10.7 m. It should be noted that in this particular case these errors did not exactly double: the along-track error increased by less than a factor of two and the cross-track increased by more than a factor of two. This can be explained primarily by geometry. The largest component of error is in the bearing angle direction, which projects more into the along-track direction in the AC2 case than it does in the AC3 case.

The vertical z errors, on the other hand, remain constant as the distance increases, at roughly 5 m. This is because the precision of barometric altimeter measurements does not depend on the link budget. As the signal strength weakens with distance, the altitude is unaffected. It is only threatened when reliable communication is threatened.

6.5.2   Distortion of the Measurement Noise

An important part of the algorithm described above is the absolute value applied to integrator measurements before position estimation. This is done to make the correlation peak positive for all bits, but can result in non-Gaussian error distributions when noise levels are high. Consider first the correlator peak for a positive bit. For high signal-to-noise ratio measurements, distortion of the correlation peak will be minimal and integrator values should all be positive. For low signal-to-noise ratio cases, on the other hand, peak distortion will be significant, and there is a high likelihood that at least some of the integrator taps will produce values that are negative. In this case, the absolute value will flip these measurements positive, distorting the measurement error distribution and influencing the estimation process. For the AC3 simulation, the best-case (interference peak) signal to noise ratio is $N_{sc}^2 N_{cx}^2 v_0^2 G_D / \sigma_I^2 \cong 9$ at the correlator peak, and the worst-case (interference valley) ratio is $N_{sc}^2 N_{cx}^2 v_0^2 G_D / \sigma_I^2 \cong 2$. Despite this relatively low signal-to-noise ratio, however, the algorithm still produces high precision, unbiased position estimates.

Although the noise distortion did not harm position estimation in the simulations shown here, it may result in bias or increased estimation error for longer aircraft separations or different formation geometries. For those cases, a couple of strategies could be implemented to mitigate the noise distortion from the absolute value. One simple approach is to increase the length of the correlator integration. By integrating across the entire data bit, all of the N_{xd} codes are assessed at once and the signal-to-noise ratio can be improved by a factor of N_{xd}^2. Another simple strategy is to utilize the blank space in the data stream shown in Figure 43 as a sort of pilot channel. Because there is no data modulated onto this portion of the code, the absolute value does not need to be applied here, so there will be no distortion of the noise distribution. Finally, the data-bit estimation could be used to determine whether a given bit is positive or negative, and then that value could

be used in the position estimation algorithm to eliminate the need for the absolute value. While this method would allow the algorithm to completely eliminate the absolute value, it introduces a new potential source of error when data bit estimation is wrong.

6.5.3   Straight and Level Assumption

A key assumption in this chapter is that following aircraft flies straight and level. As a result, the body-fixed and gravity-fixed coordinate frames are aligned and THz bearing angle measurements are therefore orthogonal to altitude measurements. Relaxing this assumption requires differentiating between body-fixed and gravity-fixed coordinates and applying rotations based on gyroscope measurements. For non-zero roll and pitch angles, the bearing angle and altitude measurements are no longer orthogonal. For small angles, this does not significantly change the algorithm and minimally impacts the performance. For large angles, however, observability deteriorates, with fully degenerative cases when either the roll or pitch angle are 90°, resulting in large uncertainty in the position estimate. Because precision airdrop is performed by large cargo aircraft not fighter jets, it is assumed that the roll and pitch angles are generally small.

6.5.4   Synchronized Clocks

Like Chapter 5, another key assumption is that the aircraft are equipped with perfectly synchronized atomic clocks. As a result, the following aircraft knows exactly when each code was transmitted allowing it to precisely compute the signal's time-of-flight. As described above, relaxing this assumption, it is reasonable to expect clock bias of up to 15 ns between clocks. The difference between the clocks will manifest as range estimate error (altitude and bearing angle are unaffected). A 15 ns inter-clock difference would result in 4.5 m of ranging bias. Because most formations are long, with relatively small bearing

angles, this error will project primarily into the along-track direction, where the positioning requirements are loose with plenty of room to spare. Furthermore, the high precision of the positioning algorithm leaves plenty of room in cross-track direction to absorb the small component of the clock bias error the projects in that direction.

6.6   Conclusion

Measurements of range, bearing angle, and relative altitude are integrated together in an EKF estimation algorithm to produce high precision position estimates. The methods described in Chapters 2 - 5 are fused together into a unified simulation to verify the performance of the THz system. Simulations show that this system is capable of achieving cross-track errors of 11 m, two-sigma, for a representative formation geometry, easily meeting the 50 m cross-track requirement for precision airdrop applications, and potentially opening the door to more demanding applications like drag reduction or aerial refueling.

6.7   Appendix: Linearized Measurement Model

The correlator measurement model described by equations (104), (105), and (106) is linearized around a nominal state vector $\mathbf{x}_0$. The partial derivatives of the measurement equation are calculated as follows

$$\frac{\delta I_i}{\delta x} = N_{sc} \frac{v_0}{\sqrt{G_D}} \left(\frac{1}{2} \frac{\delta G_D}{\delta x} \left(N_{cx} \pm \frac{N_{cx} + 1}{T_c} \left(t_i - \frac{r}{c} \right) \right) \mp G_D \frac{N_{cx} + 1}{cT_c} \frac{\delta r}{\delta x} \right) \quad (114)$$

$$\frac{\delta I_i}{\delta y} = N_{sc} \frac{v_0}{\sqrt{G_D}} \left(\frac{1}{2} \frac{\delta G_D}{\delta y} \left(N_{cx} \pm \frac{N_{cx} + 1}{T_c} \left(t_i - \frac{r}{c} \right) \right) \mp G_D \frac{N_{cx} + 1}{cT_c} \frac{\delta r}{\delta y} \right) \quad (115)$$

$$\frac{\delta I_i}{\delta z} = N_{sc} v_0 \sqrt{G_D} \left(\mp \frac{N_{cx} + 1}{cT_c} \frac{\delta r}{\delta z} \right) \quad (116)$$

$$\frac{\delta I_i}{\delta v_0} = N_{sc} \sqrt{G_D} \left(N_{cx} \pm \frac{N_{cx} + 1}{T_c} \left(t_i - \frac{r}{c} \right) \right) \quad (117)$$

$$\frac{\delta I_i}{\delta \dot{x}} = \frac{\delta I_i}{\delta \dot{y}} = \frac{\delta I_i}{\delta \dot{z}} = 0 \quad (118)$$

where the partial derivative terms are defined as

$$\frac{\delta r}{\delta x} = \frac{x}{r} \quad (119)$$

$$\frac{\delta r}{\delta y} = \frac{y}{r} \quad (120)$$

$$\frac{\delta r}{\delta z} = \frac{z}{r} \quad (121)$$

$$\frac{\delta G_D}{\delta x} = \int_{\zeta_{det,-}(\Delta)}^{\zeta_{det,+}(\Delta)} 2 \operatorname{sinc} \alpha \cos^2 \beta \left(\frac{\cos \alpha}{\alpha} - \frac{\sin \alpha}{\alpha^2} \right) \frac{\delta \alpha}{\delta x}$$
$$- 2 \operatorname{sinc}^2 \alpha \cos \beta \sin \beta \frac{\delta \beta}{\delta x} \delta \zeta \quad (122)$$

$$\frac{\delta G_D}{\delta y} = \int_{\zeta_{det,-}(\varDelta)}^{\zeta_{det,+}(\varDelta)} 2 \operatorname{sinc} \alpha \cos^2 \beta \left(\frac{\cos \alpha}{\alpha} - \frac{\sin \alpha}{\alpha^2} \right) \frac{\delta \alpha}{\delta y}$$
$$- 2 \operatorname{sinc}^2 \alpha \cos \beta \sin \beta \frac{\delta \beta}{\delta y} \delta \zeta \tag{123}$$

$$\frac{\delta \alpha}{\delta x} = \frac{\pi a}{\lambda} \cos \phi \frac{\delta \phi}{\delta x} \tag{124}$$

$$\frac{\delta \alpha}{\delta y} = \frac{\pi a}{\lambda} \cos \phi \frac{\delta \phi}{\delta y} \tag{125}$$

$$\frac{\delta \beta}{\delta x} = \frac{\pi d}{\lambda} \cos \phi \frac{\delta \phi}{\delta x} \tag{126}$$

$$\frac{\delta \beta}{\delta y} = \frac{\pi d}{\lambda} \cos \phi \frac{\delta \phi}{\delta y} \tag{127}$$

$$\frac{\delta \phi}{\delta x} = \frac{-y}{r^2} \tag{128}$$

$$\frac{\delta \phi}{\delta y} = \frac{x}{r^2} \tag{129}$$

Note that because the function is piecewise linear, the derivative is not well defined at the integrator peak. To resolve this, we assume that at the peak, the integrator value $I_{\varDelta t=0}$ is instantaneously a constant,

$$I_{\varDelta t=0} = N_{sx} v_0 \sqrt{G_D} . \tag{130}$$

From this we calculate partial derivatives as follows,

$$\frac{\delta I_{\Delta t=0}}{\delta v_0} = N_{sx}\sqrt{G_D} \tag{131}$$

$$\frac{\delta I_{\Delta t=0}}{\delta x} = \frac{N_{sx}v_0}{2\sqrt{G_D}}\frac{\delta G_D}{\delta x} \tag{132}$$

$$\frac{\delta I_{\Delta t=0}}{\delta y} = \frac{N_{sx}v_0}{2\sqrt{G_D}}\frac{\delta G_D}{\delta y} \tag{133}$$

$$\frac{\delta I_{\Delta t=0}}{\delta z} = 0 \tag{134}$$

From this, the linearized measurement model for the ith corrlator tap is

$$\delta I_i = \mathbf{h}_{I_i,\mathbf{x}_0}\delta\mathbf{x} \tag{135}$$

where δI_i is the differential correlator measurement (the difference between the measured correlator value and the modeled measurement for the nominal state vector $\mathbf{x}_0$), $\delta\mathbf{x}$ is the differential state vector (the difference between the states $\mathbf{x}$ and the nominal states $\mathbf{x}_0$), and $\mathbf{h}_{I_i,\mathbf{x}_0} \in \mathbb{R}^{1\times 7}$ is

$$\mathbf{h}_{I_i,\mathbf{x}_0} = \left[\left.\frac{\partial I_i}{\partial x}\right|_{\mathbf{x}_0} \quad \left.\frac{\partial I_i}{\partial y}\right|_{\mathbf{x}_0} \quad \left.\frac{\partial I_i}{\partial z}\right|_{\mathbf{x}_0} \quad \left.\frac{\partial I_i}{\partial \dot{x}}\right|_{\mathbf{x}_0} \quad \left.\frac{\partial I_i}{\partial \dot{y}}\right|_{\mathbf{x}_0} \quad \left.\frac{\partial I_i}{\partial \dot{z}}\right|_{\mathbf{x}_0} \quad \left.\frac{\partial I_i}{\partial v_0}\right|_{\mathbf{x}_0}\right]. \tag{136}$$

The altitude and altitude rate measurements given in equations (109) and (110) are also linearized,

$$\delta h = \mathbf{h}_h\delta\mathbf{x}, \tag{137}$$

$$\delta \dot{h} = \mathbf{h}_{\dot{h}} \delta \mathbf{x}, \tag{138}$$

where δh is the differential altitude measurement (the difference between the measured altitude and the altitude corresponding to the nominal state vector $\mathbf{x}_0$) and $\mathbf{h}_h \in \mathbb{R}^{1 \times 7}$ is

$$\mathbf{h}_h = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \tag{139}$$

$$\mathbf{h}_{\dot{h}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{140}$$

Combining all of these measurement models gives equation (112), where

$$\mathbf{H} = \begin{bmatrix} \vdots \\ \mathbf{h}_{l_i,\mathbf{x}_0} \\ \vdots \\ \mathbf{h}_h \\ \mathbf{h}_{\dot{h}} \end{bmatrix}. \tag{141}$$

Chapter 7

Conclusion

7.1   Summary

This dissertation proposes a conceptual design for a THz-based relative positioning system as a compelling alternative to current technology for formation flight applications. Because of their unique propagation properties, THz signals provide stealth and anti-jamming benefits for the precision airdrop application.

A novel system is proposed that fuses GPS-like ranging with a novel bearing angle sensor and communicated altitude measurements to determine aircraft relative positions. This design uses only the minimum number of hardware components, managing the overall cost of the system, and supports a wide variety of formation geometries, including the fundamental two-aircraft formation.

Five key technical challenges are identified to enable THz relative positioning: (1) a method for making time-of-flight range measurements and modeling the error needs to be developed; (2) the power losses of a novel angle sensor using a moving diffraction grating need to be quantified and mitigated; (3) an algorithm is required to solve the inverse problem and map noisy interference pattern measurements into bearing angle estimates; (4) a strategy must be developed to enable effective communications through the diffraction grating; and (5) it needs

to be demonstrated that this system is capable of meeting the requirements for precision airdrop.

7.2   Contributions

This dissertation presents five main contributions, each addressing one of the key challenges identified above

7.2.1   Leverage a hardware demonstration of THz ranging to identify differences from GPS and establish a baseline error model for algorithm development

Building on a ground-level demonstration of THz ranging performed by partners at PM&AM, an error model that highlights the unique role of multipath interference in THz systems is developed, including a signal model in equation (10), error model in equation (13), link budget in equation (15), and defining of key parameters in Table I. This error model is used to demonstrate that high precision range measurements are possible for distances of a couple kilometers at altitude, and serves as the basis for error analysis throughout the rest of the dissertation. To the author's knowledge, this is the first analysis and demonstration of GPS-like ranging in the THz domain.

7.2.2   Quantify power loss from introducing a moving diffraction grating in the THz receiver, and present a method to mitigate the power loss using spatial aliasing

A novel THz sensor is introduced that uses a moving diffraction grating to interfere the signal in space, converting the unobservable time-varying carrier signal into a spatial pattern. This pattern can be probed by the detector to observe the bearing angle. To address the power loss, a double-slit diffraction grating design is described in Table II that utilizes spatial aliasing to reduce the span of

164

the grating scan. To the author's knowledge, this novel device is the first to use a movable diffraction grating to make angle of arrival observable in the THz domain.

7.2.3 Propose a method to solve the inverse problem of obtaining bearing angle from a series of noisy power measurements modulated by the time-varying interference pattern gain

A measurement model is introduced in equations (47), (48), and (54) to map noisy THz signal measurements to the bearing angle, and an algorithm is introduced to solve the difficult inverse problem of bearing angle estimation. A WLS algorithm is developed in equation (59) and used to demonstrate the capabilities of the method and provide an initial estimate of bearing angle error.

7.2.4 Propose a technique for matching the data-word repetition frequency to diffraction-grating scan frequency to enable reliable communication through intermittent periods of low gain

A method for matching the diffraction grating scan frequency to the data bit repetition frequency is described in Figure 44 and Figure 45 that enables reliable communication. The matched frequencies make it such that if a particular data bit arrives during a period of low-power destructive interference, its repetition will arrive during a time of high-power constructive interference, as defined in equation (78). Analysis shows that with this strategy only one in 2500 data bits are lost at 2 km separation.

7.2.5 Demonstrate that the system meets the requirements for precision airdrop by incorporating component analyses (hardware, signal propagation, and algorithms) into a three-dimensional simulation of the integrated THz system

Finally, a unified simulation is used to integrate all of the elements described in this dissertation, and demonstrate the capabilities of the THz relative positioning system. Measurements of range, bearing angle, and relative altitude are integrated together in an EKF algorithm, described in Figure 52 and equations (102), (103), (112), and (113) to estimate the three-dimensional aircraft position. The cross-track position errors are found to be 11 m, two sigma, at 2 km, well below the 50 m requirement for the precision airdrop application.

7.3 Future Work

This dissertation describes a THz relative positioning system and demonstrates in simulation its capabilities. Physical tests are now needed to verify the results, first with ground-level testing using the PM&AM hardware, and eventually in the air.

Although mitigation strategies were discussed above in Chapter 2, power losses due to the diffraction grating still impose limits on the accuracy and range of the current THz system. One method to address this problem, which is discussed further in the appendix, is to explore mid-field patterns from multi-slit diffraction gratings. The double-slit design described above assumes that the detector is in the far field of the diffraction grating. It may be possible, however, to use patterns closer to the diffraction grating in the mid-field. This could allow more slits to be used and address one of the primary challenges of the device, a lack of power. See the appendix for more detail.

7.4 Broader Impact

This dissertation described the design of a THz relative positioning system for formation flight applications, and presented solutions to five key technical challenges. Simulations show that the system is capable of easily meeting the positioning requirements for precision airdrop, providing a compelling alternative to current technology. With these key challenges addressed, the system is now ready to begin prototyping, and physical testing, moving one step closer to implementation.

In addition, the high precision of position estimates in simulation suggests that this system may be extended into other applications. The high precision may be sufficient for more demanding applications like drag reduction and aerial refueling. Also, as the technology matures, the cost of hardware will likely go down, potentially making the system viable for commercial applications like civilian formation flight or for short-range terrestrial applications like autonomous vehicles.

Appendix 1

Preliminary Mid-field Diffraction Pattern Analysis for Multi-Slit Gratings

A1.1   Introduction and Motivation

Power is a major limiting factor in the accuracy of the THz system described in this dissertation. In addition to blocking a significant portion of the signal power, the proposed diffraction grating spreads the remaining power over a wide area. Chapter 3 discussed methods of mitigating the power loss with a double-slit configuration. While aliasing provides some mitigation of power loss from grating, power is still limited, particularly if the system is to be robust to environmental uncertainty or if the system is desired to support ranges longer than 2 km. The work in the body of this dissertation assumed a five-fold increase in transmitter power over current hardware. As the technology matures, this may be possible, but it is a technical risk. To address this loss, this chapter considers using mid-field patterns from multi-slit diffraction gratings.

One option to boost power further is to increase the number of slits in the diffraction grating. More slits means more power getting through. In addition, an increase in the number of slits serves to concentrate the power into narrower peaks in the far field, resulting in small regions of high power density. An

168

example is shown in Figure 58, where the pattern is calculated assuming that the back wall is far away from the diffraction grating $(D \gg (N - 1)d + a)$. The plot on the top shows the pattern for a double-slit grating $(N = 2)$, as given by equation (37) from Chapter 3. The plot on the bottom shows the pattern for a multi-slit grating with 5 slits $(N = 5)$, as given by equations (34), (35), and (36) from Chapter 3.
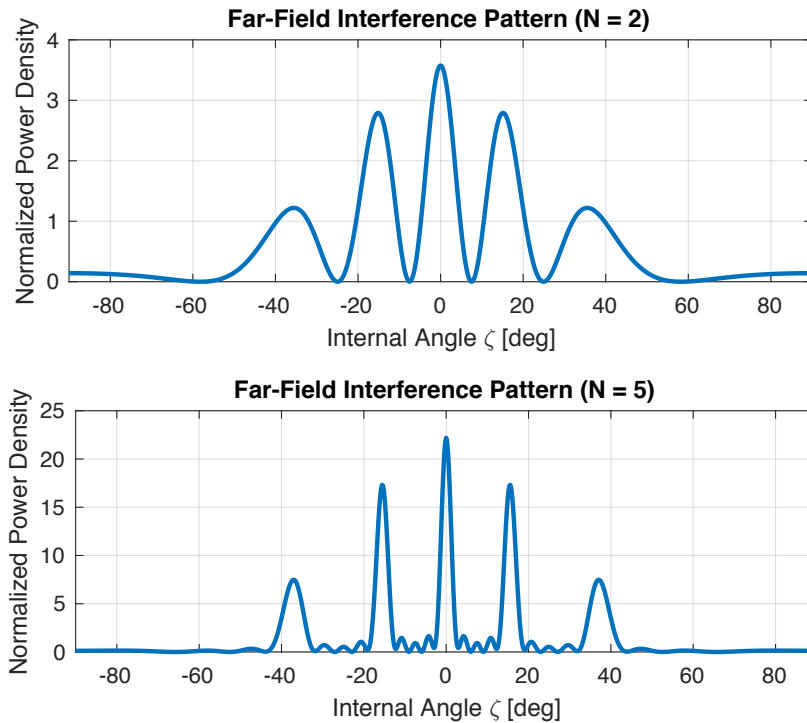
Figure 58 – Far field interference patterns for $N = 2$ (top) and $N = 5$ (bottom)

First, notice that the peaks are much sharper in the multi-slit pattern than in the double-slit pattern. In the far field, the more slits there are in the diffraction grating, the sharper the peaks and the flatter the space between the peaks. Second, notice the difference in the peak power density, measured in power per radian (note that power is normalized here to the power though one slit, so the area under the double-slit pattern is 2 and the area under the quintuple-slit pattern is 5). In arbitrary units, the normalized peak power density of the double-slit pattern is roughly 3.5, and the quintuple-slit pattern is more than six-times higher at roughly

23. The significant increase in power density is two fold in the multi-slit case: (1) there is more power getting through because there are more slits and (2) the power is being concentrated into sharper and sharper peaks.

While the far field pattern for the multi-slit grating does contain small regions of high power, it is not suitable for formation flight applications, because, as described in Chapter 3, the dead zones in the pattern present a problem for reliable communications and tracking. In addition, to maintain the far-field assumption, the device size must scale with the number of slits, resulting in large devices.

To address these challenges, this appendix explores interference patterns in the diffraction grating mid-field. This region is beyond the electromagnetic near field, but still relatively close to the diffraction grating [86]. It is loosely defined here as greater than 10 slit widths away, but less than 10 diffracting areas away,

$$10a < D < 10\big((N-1)d + a\big). \tag{142}$$

This region is marked by complicated diffraction patterns that change rapidly with distance from the diffraction grating, especially for gratings with a large number of slits. Some preliminary investigations of the patterns in this region and their potential uses for THz interferometry are discussed here.

A1.2 Physics Review

Interference is the result of slight differences in the distance traveled by signals taking different paths to the same point. Because the signals travel different distances as shown in Figure 24, Figure 26, and Figure 28 from Chapter 3, they reach the point with differing phases, resulting in interference. This process is described by the theory of Quantum Electrodynamics (QED) [85], which uses ray

tracing to calculate distance differences and phasors to track the carrier phase and sum the interference.

For the far-field patterns discussed in the main text of this dissertation, it is assumed that the back wall is far away such that the rays coming from each slit can be assumed to be essentially parallel. This makes the distance difference calculation simple, resulting in the analytical equations described in Chapter 3.

In the mid-field, it is assumed that the wall is still far enough away that the parallel ray assumption can be applied over the width a of a single slit, but close enough that the rays from different slits are no longer considered parallel. As a result, equation (36) is no longer valid, and instead the distance ℓ has to be calculated for each individual slit based on geometry before being plugged into equations (34) and (35).

A1.3   Problem Statement

The goal of this appendix is to identify usable patterns that boost the power at the receiver relative to the double-slit configuration. One of the primary factors limiting the performance of the THz interferometer is the power loss due to the diffraction grating.

To do this, the chapter will examine the mid-field of a number of multi-slit patterns and identify patterns that could be useful for THz interferometry. Recall from Chapter 3 that the criteria for a good pattern are: (1) spatial aliasing that generates multiple peaks that are relatively evenly spaced over a wide span of angles and (2) minimal dead space between peaks in the pattern.

For example, consider first the double-slit pattern on the top of Figure 58. In this pattern, 5 peaks are visible, they are roughly evenly spaced over a span of almost 90°, and there is very little dead space between the peaks, making it a good

candidate for THz interferometry. Now consider the quintuple-slit pattern shown of the bottom of the figure. In this case, the same 5 peaks are visible in the same locations, but there is now significant dead space between the peaks in the pattern, making it a poor candidate for THz interferometry.

A1.4   Diffraction Pattern Analysis

This section explores the interference pattern space for 4 different multi-slit configurations $(N = 2, 3, 4, 5)$ in an attempt to identify useful patterns. The interference is calculated using phasor analysis in equations (34) and (35) from Chapter 3. The slit width a and slit spacing d are assumed to be those used throughout the dissertation and defined in Table II.

A1.4.1   Double-Slit $N = 2$

Consider first the double-slit configuration seen throughout this dissertation. Figure 59 shows the interference in Cartesian coordinates for a variety of different lateral and longitudinal distances. The x-axis is the lateral position and y-axis is the longitudinal distance. The diffraction grating is located just out of frame at the bottom of the figure, at the origin [0 0]. Yellow indicates areas of high power density (i.e. constructive interference) and blue represents areas of low power density (i.e. destructive interference). The red line represents a slice of the mid-field pattern, which will be further discussed below.
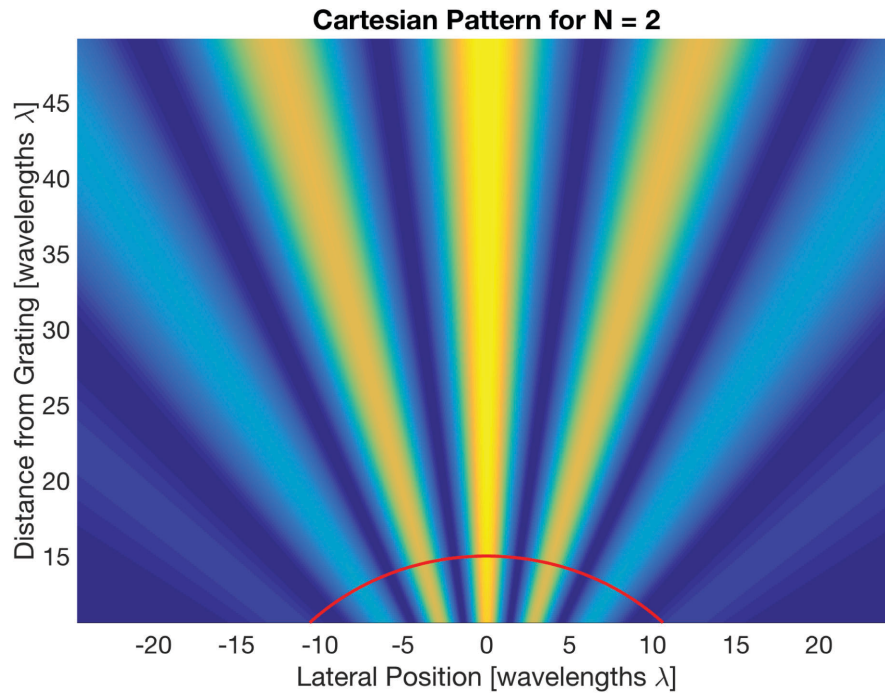
Figure 59 – Double-slit ($N = 2$) patterns in Cartesian coordinates

For the following analysis, it will be more convenient to represent the pattern in polar coordinates than Cartesian. Figure 60 shows the same double-slit pattern in polar coordinates. The x-axis is the internal angle ζ and the y-axis is the distance from the diffraction grating. Again, yellow is high power and blue is low power.
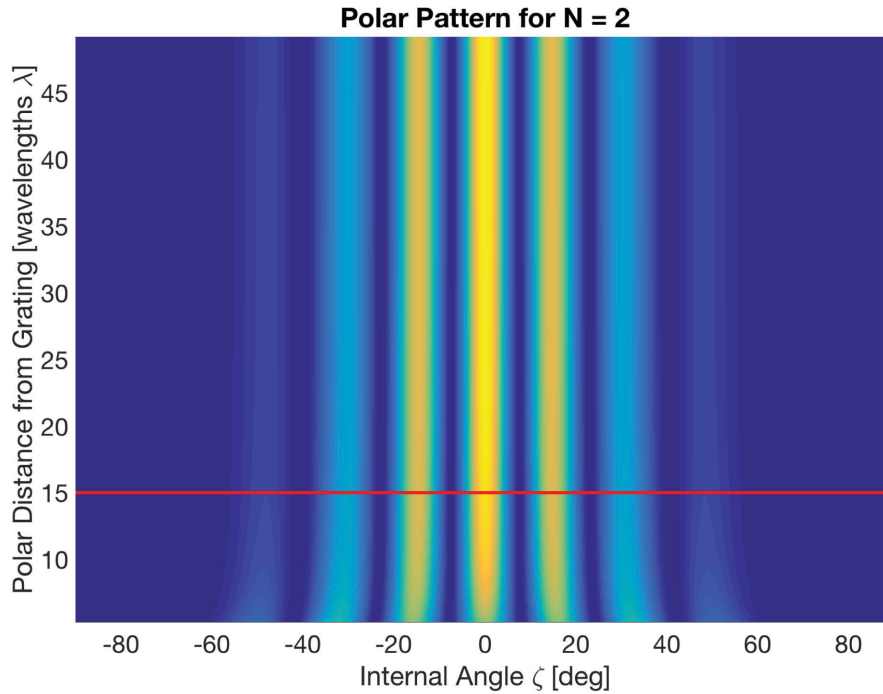
Figure 60 – Double-slit ($N = 2$) patterns in polar coordinates

The double-slit interference pattern is a strong function of angle and a weak function of radial distance. It shows very little change as the distance from the grating increases. This is an interesting result, because it suggests that the far field assumption, where $D = 10(a + d)$ is somewhat unnecessary in this case, as the mid-field patterns are almost exactly the same. As an example, the mid-field pattern at $D = 15\lambda$ is shown in Figure 61 (which represents the slice of Figure 59 and Figure 60 marked by the red line). This pattern is almost identical to the far-field pattern shown in Figure 58. The peak power is just a little bit lower in the mid-field pattern. This suggests that the far field assumption is unnecessary in the double-slit case, and that a mid-field pattern would perform just as well and also significantly shrink the size of the device.
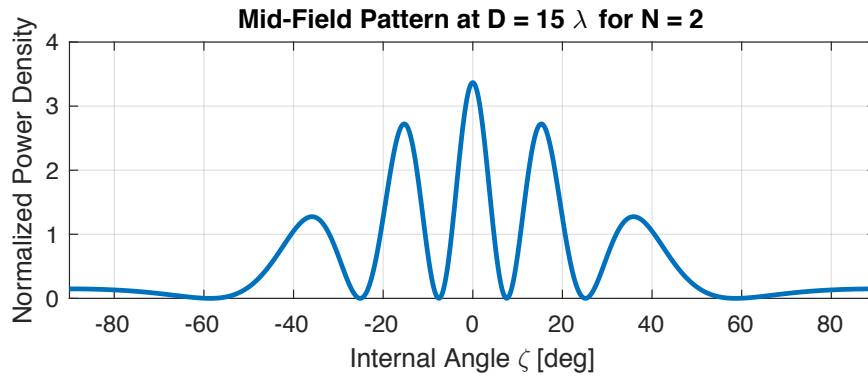
Figure 61 – Mid-field slice for double-slit ($N = 2$) configuration

A1.4.2 Triple-Slit $N = 3$

Figure 62 shows the interference for the triple-slit configuration in polar coordinates. Again, the x-axis is the internal angle ζ and the y-axis is the distance from the diffraction grating, and yellow represents high power and blue represents low power, with the red line marking the location of the slice.

Figure 62 – Triple-slit ($N = 3$) patterns in polar coordinates

With the triple-slit grating, some interesting behavior begins to emerge. Transient peaks appear and disappear in the pattern as the distance from the diffraction grating increases. These transient peaks create some interesting patterns at different distances from the grating. A mid-field pattern is taken at 30 wavelengths, as shown in Figure 63, in an attempt to meet the criteria described above. This pattern has some intriguing characteristics. First, it has high frequency spatial aliasing of roughly equally spaced peaks, the central 5 of which are all almost equal power. Second, the low power regions are limited, with minimal dead space between the peaks. The peak power, however, is only marginally higher than that of the double-slit pattern, resulting in only modest gains in the primary objective.
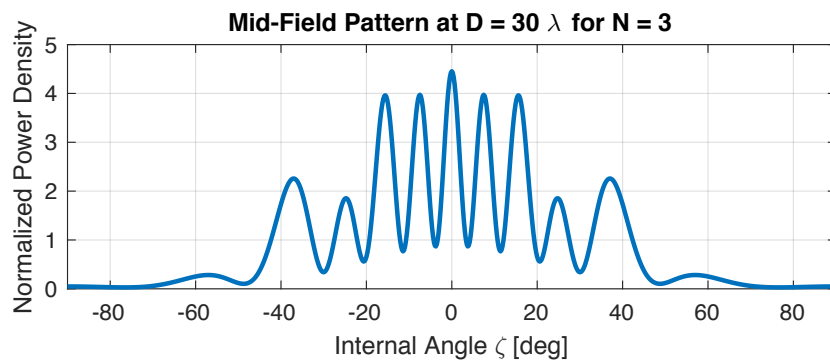
Figure 63 – Mid-field pattern for triple-slit ($N = 3$) configuration

A1.4.3 Quadruple-Slit $N = 4$

Figure 64 shows the interference for the quadruple-slit configuration. Again the polar angle ζ and distance from the grating are shown on x and y-axes, and yellow is high power and blue is low power, with the red line marking the location of the slice.
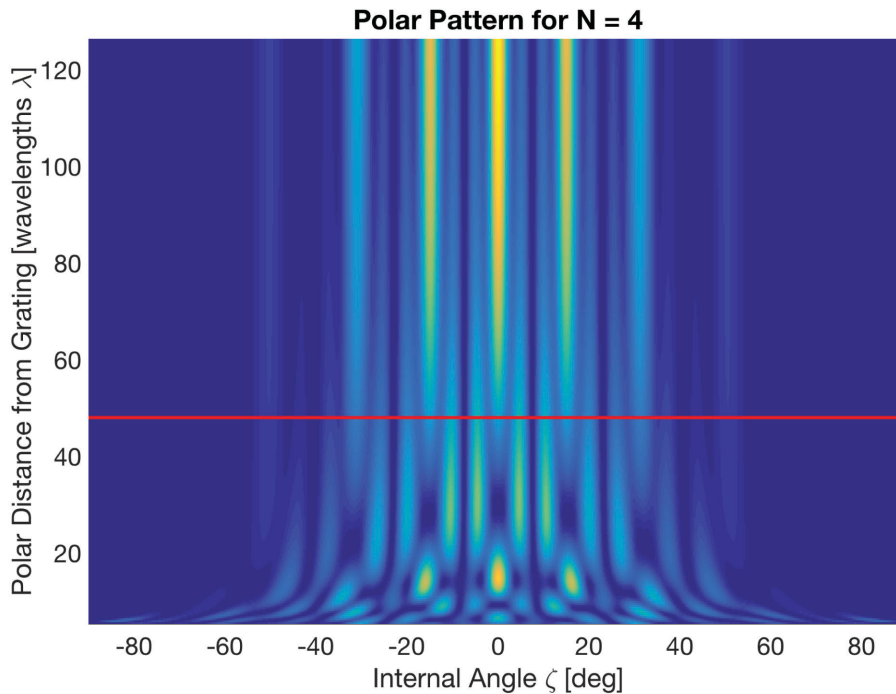
Figure 64 – Quadruple-slit ($N = 4$) patterns in polar coordinates

Again for the larger number of slits, transient peaks and valleys come and go from the pattern as the distance from the grating increases. Interestingly, this pattern has a set of persistent deep valleys at fixed angles that extend all the way through the mid-field. Taking a cross-section at 48 wavelengths, an intriguing mid-field pattern is found, as shown in Figure 65. This pattern again contains multiple peaks of roughly equal power and roughly even spacing. In addition, the power density is decently higher than in the double-slit case. One thing to note is that the valleys have differing depths. The may help with resolving the peak ambiguity if necessary, but the shallow valleys may also present a challenge in identifying the

peaks in the pattern. Once the smoothing effect of the detector width is applied, this may result in very shallow valleys that are not easily observed over noise.
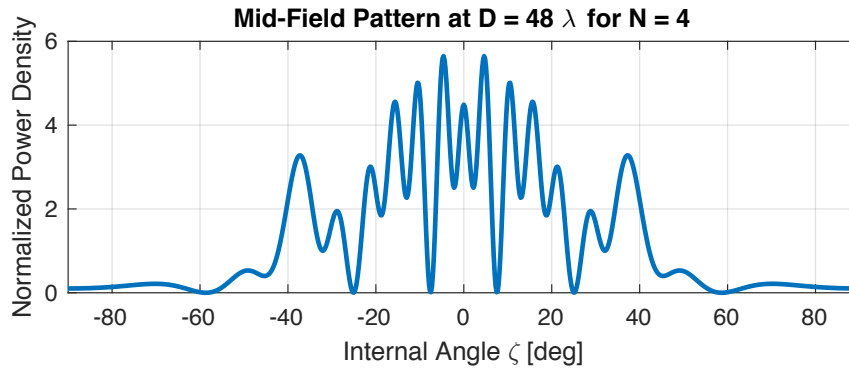
Figure 65 – Mid-field pattern for quadruple-slit ($N = 4$) configuration

It should be noted that another candidate pattern was identified for this configuration at around 20 wavelengths, however, this pattern suffers from the same deficits as the one above, and in addition, the peaks are not as evenly spaced.

A1.4.4 Quintuple-Slit $N = 5$

Figure 64 shows the interference for the quintuple-slit configuration. Again, the x-axis is the internal angle ζ and the y-axis is the distance from the diffraction grating, and yellow represents high power and blue represents low power, with the red line marking the location of the slice.
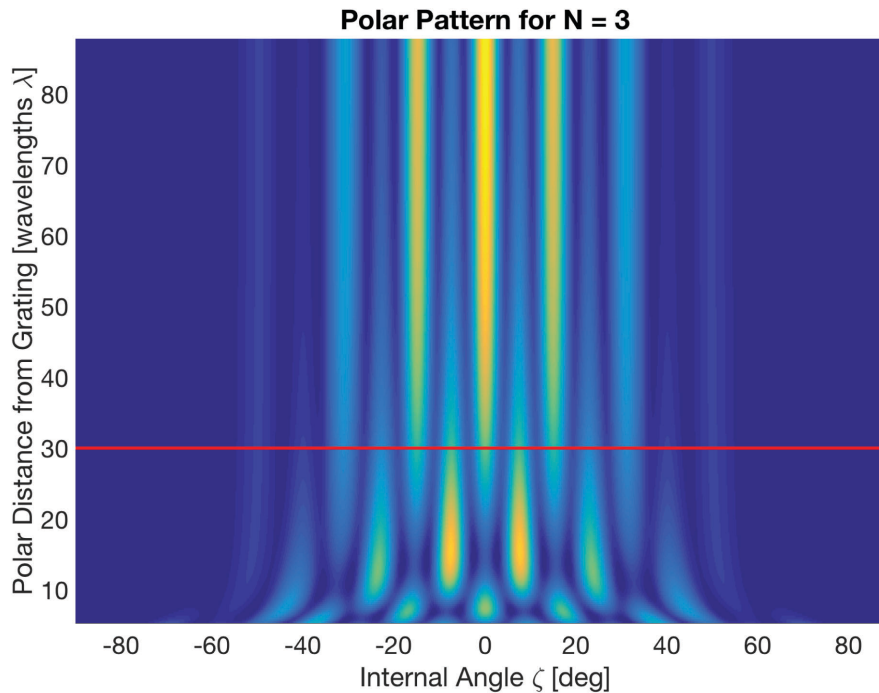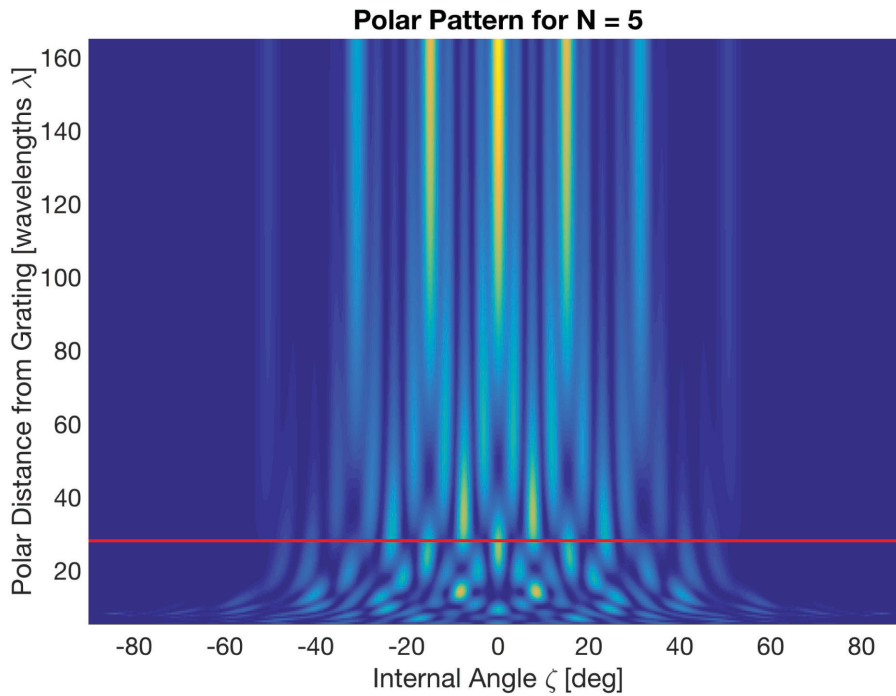

Figure 66 – Quintuple-slit ($N = 5$) patterns in polar coordinates

With the frequent coming and going of transient peaks in this case, a number of candidate patterns exist. One particular slice at 28 wavelengths that seems promising is shown in Figure 67. The pattern is composed of 7 roughly equally spaced peaks spanning a wide field of view and with minimal dead space between them. In addition, the peak power density is more than two times greater than the double-slit configuration, making this a very promising candidate pattern.
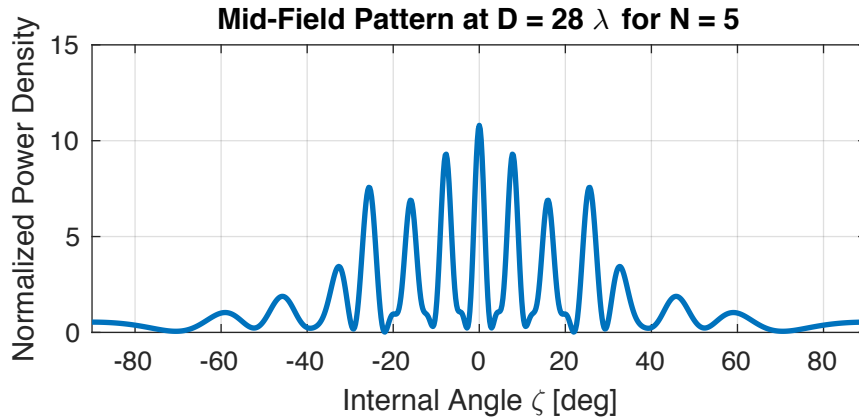
Figure 67 – Mid-field pattern for quintuple-slit ($N = 5$) configuration

An additional pattern in the quintuple case worth pointing out is the slice at 78 wavelengths, shown in Figure 68. At first glance this pattern seems an odd candidate as the peaks are of varying magnitudes and many of the valleys are shallow and therefore hard to distinguish, but this pattern is in fact quite promising.
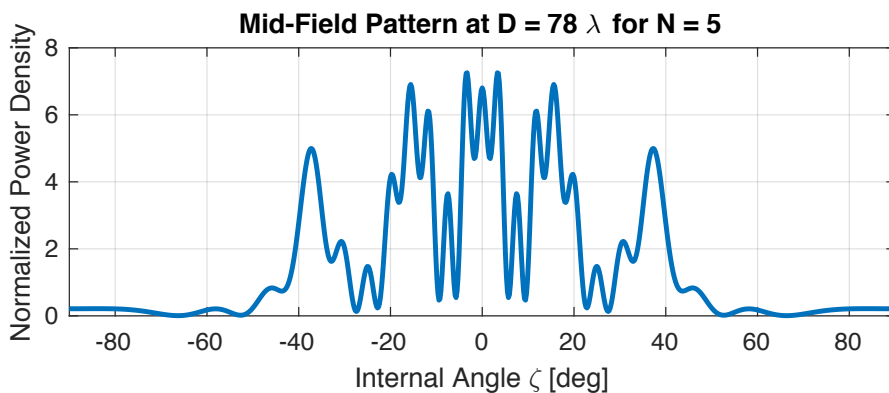
Figure 68 – Alternative mid-field pattern for quintuple-slit ($N = 5$) configuration

Consider the effect a wide detector in smoothing the pattern shown in Figure 68. In this case, the minor peaks are smoothed out leaving only the major peaks, as shown in Figure 69. In this case, a detector width w of 10 wavelengths is assumed (a roughly 40% increase from the current equipment, but readily achievable with a lens). The minor peaks are all smoothed out by the integration across the width

of the detector, leaving only the main peaks. Importantly, this pattern is similar to the double slit pattern in Figure 58, which is known to work. The primary difference is that the power is increased by over 50%.
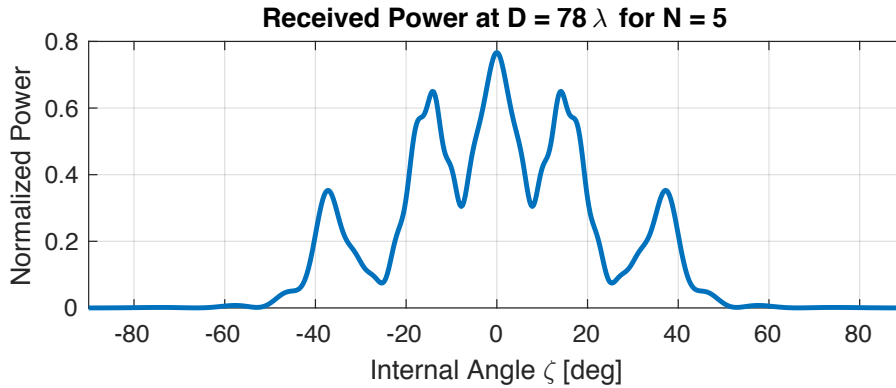
Figure 69 – Detector smoothed alternative mid-field pattern for quintuple-slit ($N = 5$) configuration

A1.5 Discussion

This preliminary work has shown that patterns exist in the mid-field of the diffraction grating that may be useful for THz interferometry. The transient peaks in the mid-field interference region of multi-slit diffraction gratings provide a number of opportunities for interesting patterns.

A number of candidate patterns are identified here, with two quintuple-slit ($N = 5$) patterns appearing most promising. The first pattern at $D = 28\lambda$ provides a number of evenly spaced peaks with high power density and minimal dead space between them. An added benefit here is that the separation distance D is almost half that of the far-field double-slit configuration, significantly reducing the size of the receiver package. The second pattern at $D = 78\lambda$ does not at first appear promising, but after smoothing via a wide detector it produces a pattern very similar to the double-slit pattern used above, but with a 50% increase in power. Because the pattern so closely matches the double-slit pattern, minimal changes would likely need to be made to the algorithm. The separation distance, however,

is a roughly 50% increase on the current hardware, and therefore would require a larger receiver package. Considering that the package is already quite small, this increase may be tolerable given the benefits for power and ease of implementation.

It should be noted that while these patterns offer potential benefits in power, there is a computational cost as the equations that describe them are significantly complicated. It may be possible to identify simplifications for specific patterns, but in general this will make the inverse problem more challenging.

A1.6 Continuing Work

The work presented in this appendix is a proof of concept. It shows that potentially useful patterns exist in the mid-field, and identifies a few promising candidates. This, however, is by no means an exhaustive search of the potential diffraction patterns, and work continues to further explore the space. At the moment, promising patterns are identified by eye. It may be possible as research continues to develop analytical rules that can be used to identify the patterns.

Additionally, candidate patterns need to be incorporated into the simulations above to test their effectiveness. Diffraction grating gains G_D will be found for the new configurations and then incorporated into the measurement model. The simulations can then test that the increased power produces has the predicted effect on position estimation.

A1.7 Conclusion

A preliminary set of promising patterns has been identified in the mid-field of multi-slit diffraction gratings. These patterns meet the spatial aliasing criteria of roughly evenly space peaks of equal magnitude with minimal dead space. More importantly, these multi-slit configurations offer the opportunity to increase the

power at the receiver, which is one of the greatest limitations of the current double-slit design.

Appendix 2

MATLAB Code

A2.1 Chapter 2 Simulations

A2.1.1 make_figs3.m

```matlab
% J. Scott Parker
% 1-26-2015

% This file plots the PM&AM ranging data and develops the error model

clear

open('figure76.fig') % open figure file
D=get(gca,'Children'); %get the handle of the line object
XData=get(D,'XData'); %get the x data
YData=get(D,'YData'); %get the y data
% this figure shows the average of the static measurements
static_truth=XData{1,1};
static_THz_avg=YData{1,1};

open('figure77.fig') % open figure
D=get(gca,'Children'); %get the handle of the line object
XData=get(D,'XData'); %get the x data
YData=get(D,'YData'); %get the y data
% This figure shows the error in each of the static measurements
static_THz_error_vec=YData{1,1};
static_THz_error=reshape(static_THz_error_vec,[5,length(static_THz_avg)]);

open('figure79.fig') % open the figure file
D=get(gca,'Children'); %get the handle of the line object
XData=get(D,'XData'); %get the x data
YData=get(D,'YData'); %get the y data
% from this file I need to extract the time series data for both the ground
% truth measurement and the THz measurement.
dynamic_truth_time=XData{2,1};
dynamic_truth_distance=YData{2,1};
dynamic_THz_time=XData{3,1};
dynamic_THz_distance=YData{3,1};

% replicate the static truth measurement for plotting.
static_truth_vec=reshape(repmat(static_truth,5,1),1,55);
% recreate each individual measurement for plotting later
static_THz_vec=static_truth_vec+static_THz_error_vec;
```

```matlab
% detrend the data
bias=mean(static_THz_error_vec(1:4*5))
detrend_static_THz_error_vec=static_THz_error_vec-bias;

% piecewise linear function for ground truth position.
linear_truth_model=spapi(2,[dynamic_truth_time 225],...
    [dynamic_truth_distance dynamic_truth_distance(end)]);

% evaluate ground truth at THz times
dynamic_truth_model=fnval(linear_truth_model,dynamic_THz_time);

calculate the error
dynamic_THz_error=dynamic_THz_distance-dynamic_truth_model;

% remove the bias
bias=mean(dynamic_THz_error(5:22))
detrend_dynamic_THz_error=dynamic_THz_error-bias;

% start with my model of thermal noise
PT=30e-3; % power of transmitter
Alens=2*pi*(2.8e-3)^2; % area of the lens at the receiver
alphat=2.5/180*pi; % spreading angle of transmitter
alphar=4/180*pi; % spreading angle for receiver
phi=10/180*pi; % range of reflection angles from ground
%Pref=0.5; % percentage of the power incident on the ground that reflects
h=1; % height of equipment off the ground
d=linspace(1,120,160); % range of separations
at=h/tan(alphat); % distance to first ground contact for transmitter cone
ar=h/tan(alphar); % distance to first ground contact for receiver cone
alpha_atm=3/1000; % attenuation coefficient
c=3e8; % speed of light
f_chip=10e6; % chipping frequency
omega=2*pi*f_chip;

rho_processing=0.33;

% calculating the thermal noise from the data set
CN_hat=5.2e7/110; %c/n estimate = model N0*B/K
C=PT*rho_processing*10.^(-alpha_atm*4/10)*...
    Alens./(2*pi*4.^2*(1-cos(alphat)));
B=1e9; % bandwidth
K=450000; % length of integrator
N0=C*K/(B*CN_hat);

% thermal noise
Nth=C/CN_hat;

% signal strength
% recieved signal power
PR=PT*rho_processing*10.^(-alpha_atm*d/10)*Alens./...
    (2*pi*d.^2*(1-cos(alphat)));

% signal strength for bounce signal
dbounce=sqrt(d.^2+2^2);
reflectivity=.9;
PRbounce=PT*reflectivity*rho_processing...
    *10.^(-alpha_atm*dbounce/10)*Alens./...
    (2*pi*dbounce.^2*(1-cos(alphat))); % max received multipath power

% convert power to amplitude
Asignal=sqrt(PR);
Abounce=sqrt(PRbounce);

A_min_received=Asignal-Abounce;
P_min_received=A_min_received.^2;

A_max_received=Asignal+Abounce;
P_max_received=A_max_received.^2;

% 2-8-2014
% add in a model for timing error on the dynamic tests
sig_timing=0.5e-9*c;
```

```matlab
% signal to noise ratio
C2N=PR./Nth; % with no interference
C2N2=P_min_received./Nth; % destrictive interference
C2N3=P_max_received./Nth; % constructive interference

sig_r_ni=c/omega*sqrt((.5)./C2N); % error with no interference
sig_r_di=c/omega*sqrt((.5)./C2N2); % destrictive interference
sig_r_ci=c/omega*sqrt((.5)./C2N3); % constructive interference

% add in the timing error
sig_r_ni=sqrt(sig_r_ni.^2+sig_timing^2);
sig_r_di=sqrt(sig_r_di.^2+sig_timing^2);
sig_r_ci=sqrt(sig_r_ci.^2+sig_timing^2);

sig_r_max=zeros(1,length(sig_r_ni));
sig_r_min=zeros(1,length(sig_r_ni));
for i=1:length(sig_r_ni)
    if d(i)<=at+ar
        sig_r_max(i)=sig_r_ni(i);
        sig_r_min(i)=sig_r_ni(i);
    else
        sig_r_max(i)=sig_r_di(i);
        sig_r_min(i)=sig_r_ci(i);
    end
end

transition_index=1;
while d(transition_index)<=at+ar
    transition_index=transition_index+1;
end

fig=20;
fs=15;

fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')
scatter(static_truth_vec,static_THz_vec)
hold on
plot([0 100],[0 100])
xlabel('Ground Truth Measurement [m]')
ylabel('THz Measurement [m]')
title('Static THz v. Ground Truth Measurements')
grid on

fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')
scatter(static_truth_vec,detrend_static_THz_error_vec)
xlabel('Ground Truth Measurement [m]')
ylabel('THz Measurement Error [m]')
title('Static THz Error (Detrended)')
grid on


fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')
scatter(static_truth_vec,static_THz_error_vec)
xlabel('Ground Truth Measurement [m]')
ylabel('THz Measurement Error [m]')
title('Static THz Error')
grid on

box_color=.3*[1 1 1];
dot_color='b';
line_color='k';
```

```matlab
% plot the de-trended data
fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')
hold on
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(sig_r_min(transition_index:end)) ...
    sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)

plot(d,sig_r_ni,line_color)
scatter(static_truth_vec,detrend_static_THz_error_vec,'filled',dot_color)

% plot the negative side of the error
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(-sig_r_min(transition_index:end)) ...
    -sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)
plot(d,-sig_r_ni,line_color)
scatter(static_truth_vec,detrend_static_THz_error_vec,'filled',dot_color)

%plot(d,sig_r_max)
%h=fill([105 0 d],[0 0 sig_r_max],'m');
%set(h,'EdgeColor','none','FaceAlpha',0.3)
%h=fill([105 0 d],[0 0 sig_r_min],'r');
%set(h,'EdgeColor','none','FaceAlpha',0.3)
%alpha(h,.5);
%plot(d,sig_r)
xlabel('Ground Truth Measurement [m]','FontSize',fs)
ylabel('Ranging Error [m]','FontSize',fs)
%title('Static THz Measurement Error Relative to Ground Truth'...
    ,'FontSize',fs)
h=legend('Predicted w/ Multipath','Predicted w/o Multipath'...
    ,'Absolute Error');
set(h,'FontSize',fs)
grid on
ylim([-4 4])

% plot the non-de-trended data
fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')
hold on
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(sig_r_min(transition_index:end)) ...
    sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)
plot(d,sig_r_ni,line_color)
scatter(static_truth_vec,abs(static_THz_error_vec),'filled',dot_color)
xlabel('Ground Truth Measurement [m]','FontSize',fs)
ylabel('Ranging Error [m]','FontSize',fs)
title('Static THz Measurement Error Relative to Ground Truth (non-
detrend)','FontSize',fs)
h=legend('Predicted w/ Multipath','Predicted w/o Multipath', ...
    'THz Ranging Error');
set(h,'FontSize',fs)
grid on
ylim([0 4])

fig=fig+1; % iterate the figure index
% plot each individual dynamic THz measurement
figure(fig)
clf
set(fig,'color','w')
scatter(dynamic_THz_time,dynamic_THz_distance,'filled')
hold on
line(dynamic_THz_time,dynamic_truth_model,'Color','r')
ylabel('Distance Measurement [m]','FontSize',fs)
```

```matlab
xlabel('Time [s]','FontSize',fs)
title('Dynamic THz and Ground Truth Measurements Time Series', ...
    'FontSize',fs+1)
h=legend('THz Range Measurements','Ground Truth Model', ...
    'Location','SouthEast');
set(h,'FontSize',fs)
grid on

fig=fig+1; % iterate the figure index
% plot the error on each individual THz measurement
figure(fig)
clf
set(fig,'color','w')
scatter(dynamic_THz_time,detrend_dynamic_THz_error)
xlabel('Time [s]')
ylabel('THz Measurement Error [m]')
title('Dynamic THz Measurement Error Relative to Ground Truth (Detrended)')
grid on

% the detrend plot
fig=fig+1; % iterate the figure index
% plot the error on each individual THz measurement
figure(fig)
clf
set(fig,'color','w')
hold on
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(sig_r_min(transition_index:end))
sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)
plot(d,sig_r_ni,line_color)
scatter(dynamic_truth_model,detrend_dynamic_THz_error, ...
    'filled',dot_color)
%plot(d,sig_r_max)

% the negative side of the plot
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(-sig_r_min(transition_index:end)) -
sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)
plot(d,-sig_r_ni,line_color)
scatter(dynamic_truth_model,detrend_dynamic_THz_error,'filled', ...
    dot_color)

xlabel('Ground Truth Measurement [m]','FontSize',fs)
ylabel('Ranging Error [m]','FontSize',fs)
%title('Dynamic THz Measurement Error Relative to Ground Truth'...
    ,'FontSize',fs)
h=legend('Predicted w/ Multipath','Predicted w/o Multipath', ...
    'Absolute Error');
set(h,'FontSize',fs)
grid on
%ylim([0 20])
ylim([-20 20])

% I want to see the non-detrend plot
fig=fig+1; % iterate the figure index
% plot the error on each individual THz measurement
figure(fig)
clf
set(fig,'color','w')
hold on
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(sig_r_min(transition_index:end))
sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)
plot(d,sig_r_ni,line_color)
scatter(dynamic_truth_model,abs(dynamic_THz_error),'filled', ...
    dot_color)
%plot(d,sig_r_max)
xlabel('Ground Truth Measurement [m]','FontSize',fs)
ylabel('Ranging Error [m]','FontSize',fs)
```

```matlab
title('Dynamic THz Measurement Error Relative to Ground Truth (non-
detrend)','FontSize',fs)
h=legend('Predicted w/ Multipath','Predicted w/o Multipath', ...
    'Absolute Error');
set(h,'FontSize',fs)
grid on
ylim([0 20])

% I want to see a plot of just the error model, nothing else
fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')
plot(d,sig_r_ni,line_color)
hold on
h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(sig_r_min(transition_index:end))
sig_r_max(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)
plot(d,sig_r_ni,line_color)
xlabel('Equipment Separation Distance [m]','FontSize',fs)
ylabel('Ranging Error [m]','FontSize',fs)
title('One-Sigma Ranging Error Model','FontSize',fs)
h=legend('Predicted One-Sigma Error with no Interference', ...
    'Predicted One-Sigma Errors with Interference');
set(h,'FontSize',fs)
grid on
ylim([0 4])

% plot of the signal to noise ratios
fig=fig+1; % iterate the figure index
% plot each individual static THz measurement
figure(fig)
clf
set(fig,'color','w')

semilogy(d,C2N,line_color)
hold on
%semilogy(d(transition_index:end),C2N2(transition_index:end),'r')
%semilogy(d(transition_index:end),C2N3(transition_index:end),'r')

h=fill([fliplr(d(transition_index:end)) d(transition_index:end)]...
    ,[fliplr(C2N2(transition_index:end)) ...
    C2N3(transition_index:end)],box_color);
set(h,'EdgeColor','none','FaceAlpha',0.5)

semilogy(d,C2N,line_color)

xlabel('Equipment Separation Distance [m]','FontSize',fs)
ylabel('Code-to-Noise Ratio C/N','FontSize',fs)
title('Code-to-Noise Ratio','FontSize',fs)
h=legend('w/o Multipath','w/ Multipath');
set(h,'FontSize',fs)
grid on
```

A2.2   Chapter 3 Simulations

A2.2.1   cos_sinc_plots.m

```matlab
% J. Scott Parker
% 12-28-2015

% plots for diffraction chapter

clear
```

```
% plot an arbitrary sinc function
theta=linspace(-90,90,501)/180*pi;
aolam=[.5 1 2 4];
for it=1:length(aolam)
  P(it,:)=sinc(aolam(it)*sin(theta)).^2;
end

fig=0; % figure number
fig=fig+1;
figure(fig)
clf
plot(theta/pi*180,P(1,:),'-','Color',[0    0.4470    0.7410],...
    'LineWidth',2)
hold on
plot(theta/pi*180,P(2,:),'--','Color',[0.8500    0.3250    0.0980],...
    'LineWidth',2)
plot(theta/pi*180,P(3,:),'-.','Color','g','LineWidth',2)
plot(theta/pi*180,P(4,:),':','Color',[0.4940    0.1840    0.5560],...
    'LineWidth',2)
grid on
xlabel('Angle \zeta [deg]')
ylabel('Power Density [normalized]')
title('Single-Slit Interference Patterns')
%legendCell = cellstr(num2str(aolam', '= %-d'));
%legendCell=strcat({'a/\lambda';'a/\lambda';'a/\lambda'},legendCell);
legend('a/\lambda = 0.5','a/\lambda =  1','a/\lambda =  2',...
    'a/\lambda =  4')
xlim([-90 90])

% plot cosine and sinc both shown
aolam=1; % a over lambda
dolam=2; % d over lambda
Psinc=sinc(aolam*sin(theta)).^2;
Pcos=cos(pi*dolam*sin(theta)).^2;
P=sinc(aolam*sin(theta)).^(2).*cos(pi*dolam*sin(theta)).^2;

fig=fig+1;
figure(fig)
clf
plot(theta/pi*180,Psinc,'-.','Color','g','LineWidth',2)
hold on
plot(theta/pi*180,Pcos,'--','Color',[0.8500    0.3250    0.0980],...
    'LineWidth',2)
plot(theta/pi*180,P,'-','Color',[0    0.4470    0.7410],'LineWidth',2)
grid on
xlabel('Angle \zeta [deg]')
ylabel('Power Density [normalized]')
title('Double-Slit Interference Pattern')
legend('Single-Slit Term','Double-Slit Term','Complete Pattern')
xlim([-90 90])
```

A2.2.2 pattern_plots.m

```
% J. Scott Parker
% 1-22-2016

% simulate the diffraction grating patterns

clear
tic % start timing clock
% parameters
f=300e9; % carrier frequency
c=3e8; % speed of light
lambda=c/f; % wavelength of signal
% Adjusted equipment parameters
PT=30e-3; % transmitter power 30 mW
alpha=.003e-3; % attenuation
```

```matlab
phi_t=2.5/180*pi; % transmitter spreading angle
rho_processing=32/pi^4; % processing losses from Jason's calculations
D_detector=5.6e-3; % diameter of detector
PN=1.1e-11; % noise power from paper
% parameters describing the movement of the grating
%sweep_angle=45/180*pi;
f_meas=1e9; % 1 GHz sample rate
reduction_factor=1e4;
f_meas=f_meas/reduction_factor; % this is to make the simulation tractable
f_sweep=5.555555555555555555555555; % frequency of the sweep
time=1/(2*f_sweep); % length of time simulated
t=linspace(0,time,time*f_meas+1); % time vector (discretizing time)
K=450000/reduction_factor;
% baseline design parameters
th_fov=70/180*pi; % field of view angle
th_amb=15/180*pi; % ambigutiy angle
% calculate the corresponding slit width and slit spacing
a=lambda/sin(th_fov); % slit width for desired field of view
d=lambda/sin(th_amb); % slit spacing for desired ambiguity

% display some information about these parameters
disp('-----------------------------------------------------------')
disp(['Field of view of ' num2str(2*th_fov/pi*180) '∞ ' ...
    'gives slit width: ' num2str(a/lambda,3) ' wavelengths'])
disp(['Ambiguity of ' num2str(th_amb/pi*180) '∞ ' ...
    'gives slit spacing: ' num2str(d/lambda,3) ' wavelengths'])

% Define the different scenarios that are being simulated
%       N        a              d
Nad=[   1        d+a            0;  % single slit (note d used to calc D)
        2        a              d;      % double slit
        5        a              d];     % multi slit

% transmitter position
r=1000; % distance to transmitter
phi=-0/180*pi; % angle to transmitter
%transmitter_pos=[r*cos(phi) r*sin(phi)]'; % position of transmitter
% power with no grating
P0=PT*rho_processing*10^(-alpha*r/10)*pi*(D_detector/2)^2 ...
    /(2*pi*r^2*(1-cos(phi_t)))*cos(phi);
disp(['The power recieved without a grating: ' num2str(P0) ' W'])
calc_time=toc; % store the time that has elapsed so far in the calculation
disp(['Time for setup ' num2str(calc_time) ' s'])

fig=20; % figure window counter
% loop
for i=1:length(Nad(:,1))
  % First extract the N, a, and d from the Nad matrix
  N=Nad(i,1);
  a=Nad(i,2);
  d=Nad(i,3);
  % calculate the power through each slit
  Aslit=D_detector*a; % "area" of the slit
  % power incident on each slit
  Pslot=PT*10^(-alpha*r/10)*Aslit/(2*pi*r^2*(1-cos(phi_t)))*cos(phi);
  % integrate the power
  th2=linspace(-90,90,5001)/180*pi; % integration variable
  dphi=2*pi/lambda*d*sin(th2); % phase difference between slits
  phasor=zeros(2,length(th2)); % initialize phasor
  for n=1:N
    phasor=phasor+1/N*[cos((n-1)*dphi); sin((n-1)*dphi)]; % sum the phasors
  end
  cos_factor=sqrt(diag(phasor'*phasor)); % cosine factor for the slits
  Pnorm=cos_factor'.^(2).*sinc(a/lambda*sin(th2)).^2; % normalized power
  % integrate and find the max power factor
  Pint=sum(Pnorm)*(th2(2)-th2(1)); % integral of Pdtheta
  Pm=N*Pslot/Pint; % this finds the max power [W/rad]
  % set the distance to the back wall
  if N==1
    D=10*a;
    sweep_angle=45/180*pi;
  else
```

```matlab
        D=10*(N-1)*d+10*a;
        sweep_angle=9/180*pi;
    end
    % calculate the grating position
    sweep_pos=D*tan(sweep_angle); % maximum lateral distance swept by grating
    y_grating=sweep_pos*linspace(-1,1,length(t)); % lateral grating position
    % calculate the power at the detector
    P_det=zeros(size(t));
    for k=1:length(t)
        y_min=y_grating(k)-D_detector/2; % bottom edge of detector
        y_max=y_grating(k)+D_detector/2; % top edge of detector
        theta_min=atan2(y_min,D); % angle bottom
        theta_max=atan2(y_max,D); % angle top
        theta_int=linspace(theta_min,theta_max,201); % integration variable
        dphi=2*pi/lambda*d*(sin(theta_int)+sin(phi)); % slit phase diff
        phasor=zeros(2,length(theta_int)); % initialize phasor
        for n=1:N
            phasor=phasor+1/N*[cos((n-1)*dphi); sin((n-1)*dphi)]; % sum phasors
        end
        cos_factor=sqrt(diag(phasor'*phasor)); % cosine factor for the slits
        dtheta=theta_int(2)-theta_int(1);
        P_det(k)=rho_processing*sum(Pm*cos_factor'.^2 ...
            .*sinc(a/lambda*(sin(theta_int)+sin(phi))).^2)*dtheta;
    end

    tn=zeros(1,floor(length(t)/K)); % plotting variable for time
    y_gratingn=zeros(1,floor(length(t)/K)); % plotting variable
    P_detn=zeros(1,floor(length(t)/K));
    n=1;
    for k=1:K:length(t)-K
        tn(n)=t(k);
        y_gratingn(n)=y_grating(k);
        P_detn(n)=1/K*sum(P_det(((n-1)*K+1):(n*K)));
        n=n+1;
    end

    calc_time=toc-calc_time;
    disp(['Iteration N = ' num2str(N) ', a = ' num2str(a) ...
        ', d = ' num2str(d) ', D = ' num2str(D)...
        ' took ' num2str(calc_time) ' s'])

    P_st(i,:)=P_detn;
    Pm_st(i)=Pm;
    theta_st(i,:)=atan(y_gratingn/D);
    ygrating_st(i,:)=y_gratingn;
    D_st(i)=D;

    theta_pat=linspace(-90,90,1001)/180*pi;
    dphi=2*pi/lambda*d*sin(theta_pat)+2*pi/lambda*d*sin(phi); % phase diff
    phasor=zeros(2,length(theta_pat)); % initialize phasor
    for n=1:N
        phasor=phasor+1/N*[cos((n-1)*dphi); sin((n-1)*dphi)]; % sum the phasors
    end
    cos_factor=sqrt(diag(phasor'*phasor)); % cosine factor for the slits
    P_pat=Pm*cos_factor'.^2.*sinc(a/lambda*(sin(theta_pat)+sin(phi))).^2;
    P_pat_st(i,:)=P_pat;
end

% fix to account for the case where the number of power
% measurements is not divisible by 5
fix=mod(length(P_st(2,:)),5);
P_double_adj=P_st(2,1:5:end-fix)+P_st(2,2:5:end-fix)...
    +P_st(2,3:5:end-fix)+P_st(2,4:5:end-fix)+P_st(2,5:5:end-fix);
% plot
fig=fig+1;
figure(fig)
clf
plot(theta_st(1,:)'/pi*180,P_st(1,:)','-','Color',...
    [0    0.4470    0.7410],'LineWidth',2)
hold on
plot(theta_st(2,1:5:end-fix)'/pi*180,P_double_adj','-.','Color',...
    [0.8500    0.3250    0.0980],'LineWidth',2)
grid on
```

```matlab
title('Power Incident on Receiver')
legend('Single-Slit','Double-Slit','Location','NorthWest')
%xlabel('Grating Position [deg]')
xlabel('Internal Angle \zeta [deg]')
ylabel('Power [W]')
xlim([-45,45])

% power percentages
Ppct=[max(P_st(1,:)) max(P_double_adj)]/P0;

% generic plot of the patterns
fig=fig+1;
figure(fig)
clf
plot(theta_pat'/pi*180,P_pat_st(1,:)'*pi/180,'-','Color',...
    [0    0.4470    0.7410],'LineWidth',2)
hold on
plot(theta_pat'/pi*180,P_pat_st(2,:)'*pi/180,'-.','Color',...
    [0.8500    0.3250    0.0980],'LineWidth',2)
grid on
title('Interference Pattern')
legend('Single-Slit','Double-Slit','Location','NorthWest')
%xlabel('Internal Angle \theta [deg]')
xlabel('Internal Angle \zeta [deg]')
%ylabel('Normalized Power Distribution')
ylabel('Power Density [W/rad]')
xlim([-90,90])

% Normalized double slit power distribution
fig=fig+1;
figure(fig)
clf
%plot(theta_pat'/pi*180,P_pat_st(1,:)'*pi/180,'-','Color',...
%    [0    0.4470    0.7410],'LineWidth',2)
%hold on
%plot(theta_pat'/pi*180,P_pat_st(2,:)'*pi/180,'-.','Color',...
%    [0.8500    0.3250    0.0980],'LineWidth',2)
plot(theta_pat'/pi*180,P_pat_st(2,:)'/max(P_pat_st(2,:)),'-.',...
    'Color',[0.8500    0.3250    0.0980],'LineWidth',2)
grid on
%title('Interference Pattern Power Distribution')
title('Double-Slit Interference Pattern')
%legend('Single-Slit','Double-Slit')
xlabel('Internal Angle \theta [deg]')
ylabel('Normalized Power')
xlim([-90,90])

% Normalized single slit power distribution
fig=fig+1;
figure(fig)
clf
%plot(theta_pat'/pi*180,P_pat_st(1,:)'*pi/180,'-','Color',...
%    [0    0.4470    0.7410],'LineWidth',2)
plot(theta_pat'/pi*180,P_pat_st(1,:)'/max(P_pat_st(1,:)),'-','Color',...
    [0    0.4470    0.7410],'LineWidth',2)
%hold on
%plot(theta_pat'/pi*180,P_pat_st(2,:)'*pi/180,'-.','Color',...
%    [0.8500    0.3250    0.0980],'LineWidth',2)
%plot(theta_pat'/pi*180,P_pat_st(2,:)'/max(P_pat_st(2,:)),'-.','Color',...
%    [0.8500    0.3250    0.0980],'LineWidth',2)
grid on
%title('Interference Pattern Power Distribution')
title('Single-Slit Interference Pattern')
%legend('Single-Slit','Double-Slit')
xlabel('Internal Angle \theta [deg]')
ylabel('Normalized Power')
xlim([-90,90])

fig=fig+1;
figure(fig)
clf
plot(theta_pat'/pi*180,P_pat_st(1,:)'*pi/180,'-','Color',...
    [0    0.4470    0.7410],'LineWidth',2)
```

```
hold on
plot(theta_pat'/pi*180,P_pat_st(2,:)'*pi/180,'-.','Color',...
    [0.8500    0.3250    0.0980],'LineWidth',2)
plot(theta_pat'/pi*180,P_pat_st(3,:)'*pi/180,':','Color','g','LineWidth',2)
grid on
title('Interference Pattern Power Distribution')
legend('Single-Slit','Double-Slit','Multi-Slit')
%legend('Multi-Slit')
xlabel('Internal Angle \theta [deg]')
ylabel('Power Distribution [W/deg]')
xlim([-90,90])
```

A2.3   Chapter 4 Simulations

A2.3.1   inverse_prob_plots_v3.m

```
% J. Scott Parker
% 11-12-2016

% generate all of the plots for the inverse problem

clear

% The first plot is a simple one showing some generic diffraction patterns
% Start with parameters
f=300e9; % carrier frequency
c=3e8; % speed of light
lambda=c/f; % wavelength of signal
% baseline design parameters
th_fov=70/180*pi; % field of view angle
th_amb=15/180*pi; % ambigutiy angle
% calculate the corresponding slit width and slit spacing
a=lambda/sin(th_fov); % slit width for desired field of view
d=lambda/sin(th_amb); % slit spacing for desired ambiguity
D=10*(d+a);

phi=0; % angle of incidence
zeta_max=45/180*pi; % departure/internal angles
zeta=linspace(-zeta_max,zeta_max,501);
% and the diffration pattern
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;

fig=1;
figure(fig)
clf
subplot(3,1,1) % plot the 0∞ case
% plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[.55 .55 1])
% hold on
% plot(zeta(201:301)/pi*180,Pnorm(201:301),'LineWidth',2,'Color',[0 0 1])
% line([9 9],[0 1],'Color','k','LineStyle','--')
% line([-9 -9],[0 1],'Color','k','LineStyle','--')
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
grid on
%title(['Double-Slit Interference Pattern for \phi = '...
%    num2str(phi/pi*180) '∞'])
%title(['\phi = ' num2str(phi/pi*180) '∞'])
text(-40,.6,['\phi = ' num2str(phi/pi*180) '∞'],'FontSize',14)
xlim([-zeta_max zeta_max]/pi*180)
% plot the 10∞ case
subplot(3,1,2)
phi=-10/180*pi;
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;
% plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[.55 .55 1])
% hold on
```

194

```matlab
% plot(zeta(201:301)/pi*180,Pnorm(201:301),'LineWidth',2,'Color',[0 0 1])
% line([9 9],[0 1],'Color','k','LineStyle','--')
% line([-9 -9],[0 1],'Color','k','LineStyle','--')
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
grid on
%title(['Double-Slit Interference Pattern for \phi = '...
%    num2str(phi/pi*180) '∞'])
%title(['\phi = ' num2str(phi/pi*180) '∞'])
text(-40,.6,['\phi = ' num2str(phi/pi*180) '∞'],'FontSize',14)
%ylabel('Normalized Power Density')
ylabel('Normalized Diffraction Pattern Amplitude \it{p/p_m}')
%ylabel({'Diffraction Pattern Amplitude Normalized to One'...
%    ['\phi = ' num2str(phi/pi*180) '∞']})
xlim([-zeta_max zeta_max]/pi*180)
% plot the 20∞ case
subplot(3,1,3)
phi=-20/180*pi;
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;
% plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[.55 .55 1])
% hold on
% plot(zeta(201:301)/pi*180,Pnorm(201:301),'LineWidth',2,'Color',[0 0 1])
% line([9 9],[0 1],'Color','k','LineStyle','--')
% line([-9 -9],[0 1],'Color','k','LineStyle','--')
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
grid on
%title(['Double-Slit Interference Pattern for \phi = '...
%    num2str(phi/pi*180) '∞'])
text(-40,.6,['\phi = ' num2str(phi/pi*180) '∞'],'FontSize',14)
%ttl=title(['\phi = ' num2str(phi/pi*180) '∞']);
%set(ttl, 'horizontalAlignment', 'right')
%set(ttl, 'units', 'normalized')
%h1 = get(ttl, 'position');
%set(ttl, 'position', [0.1 h1(2) h1(3)])
xlim([-zeta_max zeta_max]/pi*180)
xlabel('Internal Angle \zeta [deg]')

% plot the same thing but with the center highlighted and the tails faded
phi=0; % incidence angle 0
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2; % normalized power
% fade color
fade=[.8 .8 1];
% call figure window
fig=fig+1;
figure(fig)
clf
subplot(3,1,1)
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',fade)
hold on
plot(zeta(201:301)/pi*180,Pnorm(201:301),'LineWidth',2,'Color',[0 0 1])
line([9 9],[0 1],'Color','k','LineStyle','--','LineWidth',1)
line([-9 -9],[0 1],'Color','k','LineStyle','--')
% plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
grid on
%title(['Double-Slit Interference Pattern for \phi = '...
%    num2str(phi/pi*180) '∞'])
text(-40,.6,['\phi = ' num2str(phi/pi*180) '∞'],'FontSize',14)
xlim([-zeta_max zeta_max]/pi*180)
subplot(3,1,2)
phi=-10/180*pi;
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',fade)
hold on
plot(zeta(201:301)/pi*180,Pnorm(201:301),'LineWidth',2,'Color',[0 0 1])
line([9 9],[0 1],'Color','k','LineStyle','--')
line([-9 -9],[0 1],'Color','k','LineStyle','--')
% plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
grid on
%title(['Double-Slit Interference Pattern for \phi = '...
```

```matlab
%     num2str(phi/pi*180) '∞'])
text(-40,.6,['\phi = ' num2str(phi/pi*180) '∞'],'FontSize',14)
ylabel('Normalized Diffraction Pattern Amplitude \it{p/p_m}')
xlim([-zeta_max zeta_max]/pi*180)
subplot(3,1,3)
phi=-20/180*pi;
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',fade)
hold on
plot(zeta(201:301)/pi*180,Pnorm(201:301),'LineWidth',2,'Color',[0 0 1])
line([9 9],[0 1],'Color','k','LineStyle','--')
line([-9 -9],[0 1],'Color','k','LineStyle','--')
% plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
grid on
%title(['Double-Slit Interference Pattern for \phi = '...
%     num2str(phi/pi*180) '∞'])
text(-40,.6,['\phi = ' num2str(phi/pi*180) '∞'],'FontSize',14)
xlim([-zeta_max zeta_max]/pi*180)
xlabel('Internal Angle \zeta [deg]')

% plot instantaneous measurement locations
% parameters
grating_sweep_angle=9/180*pi; % max grating sweep
phi=-10/180*pi; % bearing angle
A_lens=4.9e-5; % lens area in m^2
w=sqrt(A_lens); % detector width
delta_max=D*tan(grating_sweep_angle); % max position of sweep

zeta=linspace(-grating_sweep_angle,grating_sweep_angle,101);
zeta2=linspace(-grating_sweep_angle,grating_sweep_angle,40);

Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;
Pnorm2=sinc(a/lambda*(sin(zeta2)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta2)+sin(phi))).^2;


% this is the plot with zeta along the x-axis
fig=fig+1;
figure(fig)
clf
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
hold on
scatter(zeta2/pi*180,Pnorm2,'filled')
grid on
%title(['Snapshot Measurement Locations for \phi = '...
%     num2str(phi/pi*180) '∞'])
title('Snapshot Measurement Locations')
xlim([zeta(1) zeta(end)]/pi*180)
ylim([0 1])
xlabel('Grating Position \zeta [deg]')
ylabel('Normalized Pattern Amplitude \it{p/p_m}')
legend('Interference Pattern','Measurement Locations',...
    'Location','SouthWest')

% Analytical analysis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%          User Defined Parameters      %%%%%%%%%%%%%
%transmit_power=4*30e-3;
transmit_power=5*30e-3;
transmit_spreading_angle=2.5/180*pi;
K_integration=5*450000;
lambda=1e-3; % carrier wavelength
slit_a=1.06*lambda; % slit width
slit_d=3.86*lambda; % slit spacing
grating_sweep_angle=9/180*pi;
measurements_per_sweep=40;
tol=1e-3; % tolerance of newton-raphson
max_it=50; % maximum number of iterations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```matlab
% baseline equipment parameters
P_T=transmit_power; % transmitter power 30 mW
alpha_atm=.003e-3; % attenuation is .003dB/km at altitude
phi_t=transmit_spreading_angle; % transmitter spreading angle is 2.5∞
rho_processing=32/pi^4; % processing losses from Jason's calculations
D_detector=5.6e-3; % diameter of detector
PN=1.1e-11; % noise power from paper
A_lens=4.9e-5; % lens area in m^2
% frequency parameters
f_carr=300e9; % carrier frequency
c=3e8; % speed of light
%lambda=c/f_carr; % wavelength of signal
% signal modulation parameters
f_mod=10e6;
w_mod=2*pi*f_mod;
lambda_mod=c/f_mod;
% Equipment parameters
a=slit_a; % slit width
d=slit_d; % slit spacing
D=10*((2-1)*d+a); % distance to detector
w=sqrt(A_lens); % detector width
A_slit=a*w; % area of one slit
%Gr=11.9;
Gr=6.81;

% set the noise value
sig_v=.0077; % one-sigma error on voltage measurements
% IQ integrator variables
K=K_integration; % number of points integrated by IQ integrators
sig_IQ=sig_v*sqrt(K/2); % one-sigma error on IQ integrators
% grating position
delta=linspace(-delta_max,delta_max,measurements_per_sweep);
% Equipment position
r=1000; % distance
phi=linspace(-15,15,175)/180*pi; % bearing angle range

% initialize the arrays
var_v0=zeros(length(r),length(phi));
var_theta=var_v0;
var_phi=var_v0;

% Start the loop
for ir=1:length(r)
  for ip=1:length(phi)

    % first calculate the power through the slits
    P0_slits=P_T*rho_processing*exp(-alpha_atm*r(ir)/10*log(10))...
        *2*A_slit/(2*pi*r(ir)^2*(1-cos(phi_t)))*cos(phi(ip));
    % calculate normalized power density pattern over the entire pattern
    zeta=linspace(-pi/2,pi/2,801);
    Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi(ip)))).^2 ...
        .*cos(pi*d/lambda*(sin(zeta)+sin(phi(ip)))).^2;
    % integrate power density
    Pnorm_int=sum(Pnorm)*(zeta(2)-zeta(1));
    % And finally, calculate the max power density, which will scale the
    % normalized power density from above
    P_max=P0_slits/Pnorm_int;

    V0=Gr*sqrt(P_max); % signal amplitude
    theta=r(ir)/lambda_mod*2*pi; % ranging code phase shift

    % calculate the power incident on the detector
    GD=zeros(size(delta)); % initialize the grating gain vector
    for k=1:length(delta)
      % pick points between edges of detector
      zeta=linspace(atan2(delta(k)-w/2,D),atan2(delta(k)+w/2,D),501);
      dzeta=zeta(2)-zeta(1);
      GD=sum(sinc(a/lambda*(sin(zeta)+sin(phi(ip)))).^2 ...
          .*cos(pi*d/lambda*(sin(zeta)+sin(phi(ip)))).^2)*dzeta;

      % now on to caluclate the C matrix
      ba=a/lambda*(sin(zeta)+sin(phi(ip))); % shorthand for bracketed a
      bd=pi*d/lambda*(sin(zeta)+sin(phi(ip))); % shorthand for bracketed d
```

```matlab
        % calculate the derivative of GD
        dGDdphi=sum(2*sinc(ba).*cos(bd)*pi/lambda*cos(phi(ip))...
            .*(a*cos(bd).*(cos(ba)./ba-sin(ba)./ba.^2)...
            -d*sinc(ba).*sin(bd)))*dzeta;
        % the I portion of the C matrix
        CI(k,:)=2*K/pi*sqrt(GD)...
            *[cos(theta) -V0*sin(theta) 1/2*V0/GD*cos(theta)*dGDdphi];
        % the Q portion of the C matrix
        CQ(k,:)=2*K/pi*sqrt(GD)...
            *[sin(theta) V0*cos(theta) 1/2*V0/GD*sin(theta)*dGDdphi];
    end

    % caluclate the error and store its components
    error=sig_IQ^2*inv([CI;CQ].'*[CI;CQ]);
    var_v0(ir,ip)=error(1,1);
    var_theta(ir,ip)=error(2,2);
    var_phi(ir,ip)=error(3,3);
  end
end

% caluclate the one-sigma errors
sig_v0=sqrt(var_v0);
sig_theta=sqrt(var_theta);
sig_phi=sqrt(var_phi);
sig_r=sig_theta/(2*pi)*lambda_mod;

% plot
fig=fig+1;
figure(fig)
clf
plot(phi/pi*180,2*sig_phi/pi*180,'LineWidth',3)
hold on
%plot(phi/pi*180,sig_phi(2,:)/pi*180)
%plot(phi/pi*180,sig_phi(1,:)/max(sig_phi(1,:)))
%plot(phi/pi*180,sig_phi(2,:)/max(sig_phi(2,:)))
%ylim([0 inf])
grid on
xlabel('Bearing Angle \phi [deg]')
ylabel('Two-Sigma Error 2 \sigma_\phi [deg/km]')
title('Bearing Estimate Error')

fig=fig+1;
figure(fig)
clf
plot(phi/pi*180,2*sig_r,'LineWidth',3)
%ylim([0 inf])
grid on
xlabel('Bearing Angle \phi [deg]')
ylabel('Two-Sigma Error 2 \sigma_r [m/km]')
title('Range Estimate Error')

phi_st4=phi/pi*180;
% from this index 146 goes with the 10∞ one-sig I want to store
sig_phi_10=sig_phi(146);

phi=0;
zeta=linspace(-zeta_max,zeta_max,501);
Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi))).^2 ...
    .*cos(pi*d/lambda*(sin(zeta)+sin(phi))).^2;

fig=fig+1;
figure(fig)
clf
plot(zeta/pi*180,Pnorm,'LineWidth',2,'Color',[0 0 1])
xlim([-zeta_max zeta_max]/pi*180)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%               Monte Carlo Simulation        %%%%%%%%%
t=90e-3;
Nsw=measurements_per_sweep;
time=300; %[s]
N=floor(time/t); % set number of time steps based on total time
r=1000*ones(1,N); % range
```

```
phi=10/180*pi*ones(1,N); % bearing angle
theta=r/lambda_mod*2*pi;
% initialize the storage arrays
x_hat_st=zeros(3,N+1);
x_hat_st(:,1)=[0.0008 theta(1) phi(1)];
solution_check=zeros(2,N);

% loop through each simulation
for n=1:N
  % calculate the power through the slits
  P0_slits=P_T*rho_processing*exp(-alpha_atm*r(n)/10*log(10))...
      *2*A_slit/(2*pi*r(n)^2*(1-cos(phi_t)))*cos(phi(n));
  % itegrate the normaized power density pattern over the entire pattern
  zeta=linspace(-pi/2,pi/2,801);
  Pnorm=sinc(a/lambda*(sin(zeta)+sin(phi(n)))).^2 ...
      .*cos(pi*d/lambda*(sin(zeta)+sin(phi(n)))).^2;
  % integrate
  Pnorm_int=sum(Pnorm)*(zeta(2)-zeta(1));
  % And finally, calculate the max power density, which will scale the
  % normalized power density from above
  P_max=P0_slits/Pnorm_int;

  % calculate the power incident on the detector
  P=zeros(1,Nsw); % inigtialize the gain vector
  for k=1:Nsw
    % pick points between edges
    zeta=linspace(atan2(delta(k)-w/2,D),atan2(delta(k)+w/2,D),501);
    dzeta=zeta(2)-zeta(1);
    P(k)=sum(P_max*sinc(a/lambda*(sin(zeta)+sin(phi(n)))).^2 ...
        .*cos(pi*d/lambda*(sin(zeta)+sin(phi(n)))).^2)*dzeta;
  end

  % generate measurements
  I=2*K/pi*Gr*sqrt(P)*cos(theta(n))+sig_IQ*randn(1,Nsw); % I integrator
  Q=2*K/pi*Gr*sqrt(P)*sin(theta(n))+sig_IQ*randn(1,Nsw); % Q integrator

  % initial guess for the states is estimate from the last iteration
  x_hat=x_hat_st(:,n); % iterative guess from last solution
  %x_hat=x_hat_st(:,1); %same guess every time

  % enter the solver
  for it=1:max_it
    V0_hat=x_hat(1); % guess for V0
    theta_hat=x_hat(2); % guess for theta
    phi_hat=x_hat(3); % guess for phi

    for k=1:Nsw
      % define the integration variable zeta for the width of the detector
      zeta=linspace(atan2(delta(k)-w/2,D),atan2(delta(k)+w/2,D),501);
      dzeta=zeta(2)-zeta(1); % the interval of numerical integration
      ba=a/lambda*(sin(zeta)+sin(phi_hat)); % shorthand for bracketed a
      bd=pi*d/lambda*(sin(zeta)+sin(phi_hat)); % shorthand for bracketed d
      % calculate the estimated diffraction grating gain
      GD_hat=sum(sinc(ba).^(2).*cos(bd).^2)*dzeta;
      % calculate the derivative of the diffraction gain
      dGDdphi=sum(2*sinc(ba).*cos(bd)*pi/lambda*cos(phi_hat)...
          .*(a*cos(bd).*(cos(ba)./ba-sin(ba)./ba.^2)...
          -d*sinc(ba).*sin(bd)))*dzeta;
      % the I portion of the C matrix
      CI(k,:)=2*K/pi*sqrt(GD_hat)...
          *[cos(theta_hat) -V0_hat*sin(theta_hat)...
          1/2*V0_hat/GD_hat*cos(theta_hat)*dGDdphi];
      % the Q portion of the C matrix
      CQ(k,:)=2*K/pi*sqrt(GD_hat)...
          *[sin(theta_hat) V0_hat*cos(theta_hat)...
          1/2*V0_hat/GD_hat*sin(theta_hat)*dGDdphi];

      % estimate the y_hat vector
      I_hat(k)=2*K/pi*V0_hat*sqrt(GD_hat)*cos(theta_hat);
      Q_hat(k)=2*K/pi*V0_hat*sqrt(GD_hat)*sin(theta_hat);
    end

    % update the state estimate
```

```matlab
      dy=[I'; Q']-[I_hat'; Q_hat'];
      dx=[CI;CQ]\dy;
      x_hat=x_hat+dx;

      % Check tolerance to see if converged
      if norm(dx)<tol
        %disp('Solution found')
        solution_check(:,n)=[1 it]';
        break
      end
      if it==max_it
        disp(['max iterations reached n = ' num2str(n)])
        solution_check(:,n)=[0 it]';
        x_hat=x_hat_st(:,n); % this is to catch it from slipping if
        % Notify if the solver failed to converge
      end

    end

  % store the solved value
  x_hat_st(:,n+1)=x_hat;

end

% calculate angle errors
mean_phi_error=mean(x_hat_st(3,2:end)-phi);
std_phi=std(x_hat_st(3,2:end)-phi);
% calculate range errors
mean_theta_error=mean(x_hat_st(2,2:end)-theta);
std_theta=std(x_hat_st(2,2:end)-theta);
% display summary of results
disp(['range: ' num2str(r(1)) '  and bearing is '...
    num2str(phi(1)/pi*180) '∞'])
disp(['mean phi error:    ' num2str(mean_phi_error/pi*180)...
    '∞  with STD:  ' num2str(std_phi/pi*180) '∞'])
disp(['mean theta error:  ' num2str(mean_theta_error/pi*180)...
    '∞  with STD:  ' num2str(std_theta/pi*180) '∞'])
disp('------------------------------------------------')
% check how many times the solver failed to converge
failure_count=0;
for n=N:-1:1
  if solution_check(1,n)==0
    x_hat_st(:,n)=[];
    failure_count=failure_count+1;
  end
end
disp(['Failed to converge ' num2str(failure_count) ' time(s)'])
% range estimates
r_hat=x_hat_st(2,:)/(2*pi)*lambda_mod;
% x-y position estimates
x_pos_hat=r_hat.*cos(x_hat_st(3,:));
y_pos_hat=r_hat.*sin(x_hat_st(3,:));
x=r(1)*cos(phi(1));
y=r(1)*sin(phi(1));

fig=fig+1;
figure(fig)
clf
scatter(x_pos_hat,y_pos_hat,70,'x','g')
hold on
grid on
scatter([x(1) x(end)],[y(1) y(end)],40,'filled','k')
%line(x,y,'Color','k')
axis equal
title('Position Estimates from Simulation')
xlabel('x-position [m]')
ylabel('y-position [m]')
legend('Est','True','Location','SouthEast')
set(gca,'Xdir','reverse')
xlim([973 995])
ylim([130 205])
```

```
fig=fig+1;
figure(fig)
clf
plot(0:t:time,(x_hat_st(3,:)-[phi phi(end)])/pi*180)
grid on
hold on
line([0 time],[sig_phi_10 sig_phi_10]*2/pi*180,'Color','r',...
    'LineStyle','-.','LineWidth',2)
line([0 time],-[sig_phi_10 sig_phi_10]*2/pi*180,'Color','r',...
    'LineStyle','-.','LineWidth',2)
title('Bearing Estimate \phi Error from Simulation')
xlabel('Time [s]')
ylabel('Error [deg]')
legend('Error','2\sigma Bound')
ylim([-1.9 1.9])
```

A2.4   Chapter 5 Simulations

A2.4.1   plot_bit_error_6_21_2017.m

```
% J. Scott Parker
% 6-21-2017

% calculate the rate of missed data bits

clear

r=2000;
%phi=[-10 0 10]/180*pi;
phi=linspace(-10,10,113)/180*pi;
options.motion_model='constant';
options.Fg=7; % frequency of the diffraction grating sweep
options.time=4/options.Fg; % length of simulation [s]
options.Nsc=500;
options.Nxd=3;
options.GR=6.81;%11.9; %8.5;
%options.PT=150e-3;%4*30e-3;
options.PT=2*30e-3;
%options.zeta_max=7.6/180*pi;
options.data='no';
% run once just to get time
t=geometry(options);
% allot storage arrays
GD=zeros(length(phi),length(t));
v0_tr=zeros(length(phi),length(t));

for i=1:length(phi)
  options.pos_ctrl=[r*cos(phi(i)) r*sin(phi(i)) 0]'; % AC rel pos [m]
  % generate the geometry of the aircraft and THz equipment
  [t,g,p]=geometry(options);
  for k=1:length(t)
    [I,I_tr,h,h_tr,hdot,hdot_tr,v0_tr(i,k),GD(i,k)]=...
        meas(g,k,g.pos_tr(:,k),p);
  end
end

msg_st=linspace(0,p.Tg-p.Tx,111); % start times of message (approximate)
% the model of the correlators does not change and neither does the error
% so  calculate those once outside the loop
H=p.Nsc*(p.Ncx+(p.Ncx+1)/p.Tc*p.taps.*(p.taps<=0)...
    -(p.Ncx+1)/p.Tc*p.taps.*(p.taps>0));
Hcm=repmat(H,p.Nxd,1);
% now the covariance matrix
NI=length(p.taps);
RI=zeros(NI*p.Nxd);
for i=1:p.Nxd
  RI((i-1)*NI+1:i*NI,(i-1)*NI+1:i*NI)=p.Sig_I;
```

```matlab
end
% and finally the variance and STD
var_D=inv((Hcm'/RI)*Hcm);
sig_D=sqrt(var_D);
% loop
for im=1:length(msg_st)
  ims(im)=round(msg_st(im)/p.Tx); % index of message start
  imsg=(0:p.Nxd:p.Nxd*p.Nb2w-1)+ims(im)+round(p.Nxd/2); % msg center inds
  ED=v0_tr(:,imsg).*sqrt(GD(:,imsg)); % expected value of D
  p_be=normcdf(-ED/sig_D); % probability of missing a given bit
  for j=1:4 % loop through each word
    ind=(j-1)*p.Nbw+(1:p.Nbw); % indices of the jth word
    p_test=zeros(size(phi))';
    for is=ind(1):ind(end)
      p_test=p_test+p_be(:,is)./(1-p_be(:,is)).*prod(1-p_be(:,ind),2);
    end
    p_we1b(:,j)=p_test;
    p_we(:,j)=1-prod(1-p_be(:,ind),2); % probability missing 1 bit in word
    p_test=zeros(size(phi))';
    for is=ind(1):ind(end)
      for is2=ind(1):ind(end)
        if is~=is2
          p_test=p_test+p_be(:,is).*p_be(:,is2)./...
              ((1-p_be(:,is)).*(1-p_be(:,is2)))...
              .*prod(1-p_be(:,ind),2);
        end
      end
    end
    p_we2b(:,j)=p_test; % probability of 2 bit errors in word

    % probability of 3 bit errors
%     p_test=zeros(size(phi))';
%     for is=ind(1):ind(end)
%       for is2=ind(1):ind(end)
%         for is3=ind(1):ind(end)
%           if is~=is2 && is~=is3 && is2~=is3
%             p_test=p_test+p_be(:,is).*p_be(:,is2).*p_be(:,is3)./...
%                 ((1-p_be(:,is)).*(1-p_be(:,is2)).*(1-p_be(:,is3)))...
%                 .*prod(1-p_be(:,ind),2);
%           end
%         end
%       end
%     end
%     p_we3b(:,j)=p_test;

  end
  pmw1(:,im)=p_we(:,1).*p_we(:,3);
  pmw2(:,im)=p_we(:,2).*p_we(:,4);

  pw11b(:,im)=p_we1b(:,1).*p_we1b(:,3);
  pw21b(:,im)=p_we1b(:,2).*p_we1b(:,4);

  pw12b(:,im)=p_we2b(:,1).*p_we2b(:,3);
  pw22b(:,im)=p_we2b(:,2).*p_we2b(:,4);

%   pw13b(:,im)=p_we3b(:,1).*p_we3b(:,3);
%   pw23b(:,im)=p_we3b(:,2).*p_we3b(:,4);

end

% plot the results
figure(40)
clf
surf(phi/pi*180,p.Delta(ims+1)*1000,pmw1','EdgeColor','none')
view(0,90)
ylim([p.Delta(ims(1)+1) p.Delta(ims(end)+1)]*1000)
ylabel('Grating Position \Delta at Message Start [mm]')
xlabel('Bearing Angle \phi [deg]')
h=colorbar;
set(h,'ylim',[0 15]*1e-3)
title('Probability of Missing Data Word 1')

figure(42)
```

```
clf
surf(phi/pi*180,p.Delta(ims+1)*1000,pmw2','EdgeColor','none')
view(0,90)
ylim([p.Delta(ims(1)+1) p.Delta(ims(end)+1)]*1000)
ylabel('Grating Position \Delta at Message Start [mm]')
xlabel('Bearing Angle \phi [deg]')
h=colorbar;
set(h,'ylim',[0 15]*1e-3)
title('Probability of Missing Data Word 2')

figure(43)
clf
surf(phi/pi*180,p.Delta(ims+1)*1000,pmw1','EdgeColor','none')
view(0,90)
ylim([p.Delta(ims(1)+1) p.Delta(ims(end)+1)]*1000)
ylabel('Grating Position \Delta at Msg Start [mm]')
xlabel('Bearing Angle \phi [deg]')
h=colorbar;
%set(h,'ylim',[0 15]*1e-3)
%title('Probability of Missing Data Word')
%title(['Probability of Missing Word (F_g = ' num2str(p.Fg) ' Hz)'])
title(['Continuity Risk (F_g = ' num2str(p.Fg) ' Hz)'])

figure(44)
clf
surf(phi/pi*180,p.Delta(ims+1)*1000,pw11b','EdgeColor','none')
view(0,90)
ylim([p.Delta(ims(1)+1) p.Delta(ims(end)+1)]*1000)
ylabel('Grating Position \Delta at Message Start [mm]')
xlabel('Bearing Angle \phi [deg]')
h=colorbar;
%set(h,'ylim',[0 15]*1e-3)
title('Probability of Missing Data Word 1 bit error')

figure(45)
clf
surf(phi/pi*180,p.Delta(ims+1)*1000,0.12*pw12b','EdgeColor','none')
view(0,90)
ylim([p.Delta(ims(1)+1) p.Delta(ims(end)+1)]*1000)
ylabel('Grating Position \Delta at Msg Start [mm]')
xlabel('Bearing Angle \phi [deg]')
h=colorbar;
%set(h,'ylim',[0 15]*1e-3)
%title('Probability of Undetected Bit Error')
title(['Integrity Risk (F_g = ' num2str(p.Fg) ' Hz)'])
```

A2.5   Chapter 6 Simulations

A2.5.1   dissertation_sim.m

```
% J. Scott Parker
% June 2017

% This file simulates the THz relative positioning system described in
% Chapter 6 of my dissertation

clear

disp('--------------------------------------------------')
c = clock; % get current time when sim starts
disp(datestr(datenum(c(1),c(2),c(3),c(4),c(5),c(6)))); % display time
tic % start the simulation timer

% simulation parameters
options.time=1; % length of simulation [s]
options.wp_size=10; % size of simulated aircraft motion [m]
options.T_wp=6; % period of simulated aircraft motion [s]
```

```matlab
options.pos_ctrl=[2000 100 0]'; % relative position of aircraft [m]
%options.pos_ctrl=[1000 200 0]'; % relative position of aircraft [m]
options.Fg=7; % frequency of the diffraction grating sweep
options.Nsc=500; % number of samples per code chip (sets chipping rate)
options.Nxd=3; % number of codes per data bit (sets data rate)
options.sig_h=1; % altitude measurement error
options.sig_hdot=.2; % altitude rate measurement error
options.q_vel=200e-3; % velocity state process noise
options.q_v0=1e-8; % amplitude state process noise
options.GR=6.81; % receiver gain
options.PT=150e-3; % transmit power

% generate the geometry of the aircraft and THz equipment
[t,g,p]=geometry(options);

% display some info about the simulation
sim_time=toc; % store the current time elapsed in the simulation
disp(['Simulating ' num2str(p.time) 's of aircraft motion'])
disp(['Building aircraft positions took ' num2str(sim_time) 's'])
last_sim_time=sim_time; % store last time in simulation


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%           The Simulation and Estimation Loop        %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Initialize the Extended Kalman Filter loop with values for 1st iteration
% measurements at first time step
[I,I_tr,h,h_tr,hdot,hdot_tr,v0_tr]=meas(g,1,g.pos_tr(:,1),p);
sig_pos_gps=3; % error in position initialization from GPS
sig_vel_gps=.5; % error in velocity initialization from GPS
sig_v0=.2*v0_tr; % error in amplitude initialization
xHatP=[g.pos_tr(:,1); g.vel_tr(:,1); v0_tr]... % initial state guess
    +[sig_pos_gps*randn(3,1); sig_vel_gps*randn(3,1); sig_v0*randn(1)];
% initial state uncertainty covariance
PP=diag([sig_pos_gps^2*ones(1,3) sig_vel_gps^2*ones(1,3) sig_v0^2]);

% allocate memory for state estimates
xHatP_st=zeros(length(xHatP),length(t)); % initialize state estimates
xHatP_sig=zeros(length(xHatP),length(t)); % initialize state errors
xHatP_sig(:,1)=[sig_pos_gps*ones(3,1); sig_vel_gps*ones(3,1); sig_v0];
% allocate memory for wls bit estimation
y_c=I; % include first I meas in comms y vector
d_hat=[]; % initialize data estimation with 0 bits
d_st=[]; % initialize the data storage with 0 bits
% allocate memory for truth states
g.I_tr=zeros(length(p.taps),length(t)); % initialize true I measurement
g.h_tr=NaN*ones(1,p.Nmsgs); % initialize the true h measurement
g.hdot_tr=NaN*ones(1,p.Nmsgs); % initialize the true h measurement
g.v0_tr=zeros(1,length(t)); % initialize true amplitude constant
% store the first time steps
xHatP_st(:,1)=xHatP; % store first position estimate
g.I_tr(:,1)=I_tr; % store first set of true measurements
g.h_tr(1)=h_tr; % store first true altitude
g.hdot_tr(1)=hdot_tr; % store first true altitude rate
g.v0_tr(1)=v0_tr; % store first true amplitude

% display wait bar to keep track of simulation progress
hwb=waitbar(0,['Loop Progress: t = ' num2str(t(1)) 's of '...
    num2str(p.time) 's']); % create waitbar to keep track of sim progress

% Simulation and estimation loop
for k=2:length(t)

  % state prediction steps
  xHatM=p.F*xHatP;
  PM=p.F*PP*p.F'+p.G_til*p.Q*p.G_til';

  % predict the measurements
  [yHat,H]=measMod(xHatM,p,p.Delta(k));
  % calculate the actual measurements
  [I,g.I_tr(:,k),h,g.h_temp,hdot,hdot_tr_temp,g.v0_tr(k),GD(k)]=...
      meas(g,k,xHatM,p);
```

```matlab
  % run the data bit estimation and determine whether data is available
  [w1_avail,w2_avail,y_c,d_hat,d_st]=dataAlg(p,I,y_c,d_hat,d_st);

  % assemble the system model depending on data availability
  [y,yHat,H,R]=sysMod(p,I,h,hdot,yHat,xHatM,H,w1_avail,w2_avail);

  % correction/innovation steps
  L=PM*H'/(H*PM*H'+R);
  xHatP=xHatM+L*(y-yHat);
  PP=(eye(7)-L*H)*PM*(eye(7)-L*H)'+L*R*L';

  % Store stuff
  xHatP_st(:,k)=xHatP; % store state estimate
  xHatP_sig(:,k)=sqrt(diag(PP)); % store state estimate error
  I_st(:,k)=I; % store the measurements

  % display the progress on the progress bar
  if mod(k,round(length(t)/90))==0 % only at certain time steps
    pct_done=k/length(t); % calculate the percentage completed
    calc_time=toc-sim_time; % calculate time elapsed so far
    remaining_time=calc_time/(pct_done)-calc_time; % calculate time left
    waitbar(pct_done,hwb,{['Loop Progress: t = ' num2str(t(k),3) ...
        ' s of ' num2str(p.time) ' s'] ...
        [num2str(floor(calc_time/60))...
        ' min ' num2str(round(mod(calc_time,60)))...
        ' s elapsed -- roughly ' ...
        num2str(floor(remaining_time/60)) ' min '...
        num2str(round(mod(remaining_time,60))) ' s remaining']});
  end
end

close(hwb) % close the progress bar

% positioning error
xHatP_error=xHatP_st-[g.pos_tr;g.vel_tr;g.v0_tr]; % calc pos est error
xHatP_mean_error=mean(xHatP_error,2); % calc mean pos error

% display the time it took to run the simulation
sim_time=toc;
disp(['Simulation and estimation took '...
    num2str(sim_time-last_sim_time) 's'])
last_sim_time=sim_time;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%                   Plotting the Results              %%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Plot the aircraft path
fig=70;
figure(fig)
clf
scatter3(g.pos_tr(1,1),g.pos_tr(2,1),g.pos_tr(3,1),200,'x','k')
hold on
scatter3(g.pos_tr(1,end),g.pos_tr(2,end),g.pos_tr(3,end),200,'o','k')
line(g.pos_tr(1,:),g.pos_tr(2,:),g.pos_tr(3,:),'Color','k')
legend('start','end')
title('Simulated Motion Track','interpreter','latex')
xlabel('$x$ [m]','interpreter','latex')
ylabel('$y$ [m]','interpreter','latex')
zlabel('$z$ [m]','interpreter','latex')
axis equal
grid on
set(gca,'Xdir','reverse')
set(gca,'Zdir','reverse')
view(-15,13)
% plot the x position, velocity, and acceleration
fig=fig+1;
figure(fig)
clf
subplot(3,1,1)
plot(t,g.pos_tr(1,:))
title('Motion in x-Dimension','interpreter','latex')
```

```matlab
ylabel('$x$ [m]','interpreter','latex')
grid on
subplot(3,1,2)
plot(t,g.vel_tr(1,:))
ylabel('$\dot{x}$ [m/s]','interpreter','latex')
grid on
subplot(3,1,3)
plot(t,g.acc_tr(1,:))
ylabel('$\ddot{x}$ [m/s$^2$]','interpreter','latex')
grid on
xlabel('Time [s]','interpreter','latex')
% plot the y position, velocity, and acceleration
fig=fig+1;
figure(fig)
clf
subplot(3,1,1)
plot(t,g.pos_tr(2,:))
title('Y-Direction')
ylabel('Position [m]')
grid on
subplot(3,1,2)
plot(t,g.vel_tr(2,:))
ylabel('Velocity [m/s]')
grid on
subplot(3,1,3)
plot(t,g.acc_tr(2,:))
ylabel('Acceleration [m/s^2]')
grid on
xlabel('Time [s]')
% plot the z position, velocity, and acceleration
fig=fig+1;
figure(fig)
clf
subplot(3,1,1)
plot(t,g.pos_tr(3,:))
title('Z-Direction')
ylabel('Position [m]')
grid on
subplot(3,1,2)
plot(t,g.vel_tr(3,:))
ylabel('Velocity [m/s]')
grid on
subplot(3,1,3)
plot(t,g.acc_tr(3,:))
ylabel('Acceleration [m/s^2]')
grid on
xlabel('Time [s]')
% plot the true v0 parameter
fig=fig+1;
figure(fig)
clf
plot(t,g.v0_tr)
ylabel('Amplitude Parameter v_0 [V]')
xlabel('Time [s]')
% plot the true data
fig=fig+1;
figure(fig)
clf
plot(t,g.Datak)
title('true data')
% plot the measured data
fig=fig+1;
figure(fig)
clf
plot(d_st)
title('measured data')
% plot the position estimate error in the x-dimension
fig=fig+1;
figure(fig)
clf
plot(t,xHatP_error(1,:))
grid on
title(['x-Position Error (Mean: '...
```

```matlab
    num2str(xHatP_mean_error(1)*100,3) ' cm)'],'interpreter','latex')
hold on
plot(t,2*xHatP_sig(1,:),'k')
plot(t,-2*xHatP_sig(1,:),'k')
legend('error','2-\sigma bound')
xlabel('Time [s]','interpreter','latex')
ylabel('Error [m]','interpreter','latex')
% plot the position estimate error in the y-dimension
fig=fig+1;
figure(fig)
clf
plot(t,xHatP_error(2,:))
grid on
title(['y-Position Error (Mean: '...
    num2str(xHatP_mean_error(2)*100,3) ' cm)'],'interpreter','latex')
hold on
plot(t,2*xHatP_sig(2,:),'k')
plot(t,-2*xHatP_sig(2,:),'k')
legend('error','2-\sigma bound')
xlabel('Time [s]','interpreter','latex')
ylabel('Error [m]','interpreter','latex')
% plot the position estimate error in the z-dimension
fig=fig+1;
figure(fig)
clf
plot(t,xHatP_error(3,:))
grid on
title(['z-Position Error (Mean: '...
    num2str(xHatP_mean_error(3)*100) ' cm)'],'interpreter','latex')
hold on
plot(t,2*xHatP_sig(3,:),'k')
plot(t,-2*xHatP_sig(3,:),'k')
legend('error','2-\sigma bound')
xlabel('Time [s]','interpreter','latex')
ylabel('Error [m]','interpreter','latex')

% plot the position estimate error in the all three dimensions
fig=fig+1;
figure(fig)
clf
% x
subplot(3,1,1)
plot(t,xHatP_error(1,:))
grid on
hold on
plot(t,2*xHatP_sig(1,:),'k')
plot(t,-2*xHatP_sig(1,:),'k')
%legend({'error','2-\sigma bound'},'Position',[0.86 0.95 0.01 0.01])
legend({'error','2-\sigma bound'},'Position',[0.86 0.94 0.01 0.01])
ylabel('x [m]','interpreter','latex')
title('Position Estimate Error','interpreter','latex')
%ylim([-4.5 4.5])
ylim([-3 3.5])
% y
subplot(3,1,2)
plot(t,xHatP_error(2,:))
grid on
hold on
plot(t,2*xHatP_sig(2,:),'k')
plot(t,-2*xHatP_sig(2,:),'k')
ylabel('y [m]','interpreter','latex')
%ylim([-18 18])
ylim([-7 7])
% z
subplot(3,1,3)
plot(t,xHatP_error(3,:))
grid on
hold on
plot(t,2*xHatP_sig(3,:),'k')
plot(t,-2*xHatP_sig(3,:),'k')
ylabel('z [m]','interpreter','latex')
ylim([-15 15])
xlabel('Time [s]','interpreter','latex')
```

```
% calculate what the true 2-sig error is on position estimates
real_2sig_error=2*[std(xHatP_error(1,:));
    std(xHatP_error(2,:)); std(xHatP_error(3,:))];
% calculate how often the estimate lies within the 2-sig bound
pct_in_bounds=[sum(2*xHatP_sig(1,:)>abs(xHatP_error(1,:)));
    sum(2*xHatP_sig(2,:)>abs(xHatP_error(2,:)));
    sum(2*xHatP_sig(3,:)>abs(xHatP_error(3,:)));]/length(t);
```

A2.5.2 geometry.m

```
function [t,g,p]=geometry(options)

% Check if any options were given, if not, set the options to the default
if nargin<1 % optional input handling
  options.default='yes'; % if no options, then default
else
  options.default='no'; % otherwise state that not default
end

p=options;

% Define parameters from options structure

%%%%%%%%%%%%%%%%%%%%%%%%%   numbers   %%%%%%%%%%%%%%%%%%%%%%%%%%
if isfield(options,'Nsc')==0
  p.Nsc=100; % number of samples per chip
end
if isfield(options,'Ncx')==0
  p.Ncx=1023; % number of chips per code
end
if isfield(options,'Nxd')==0
  p.Nxd=4; % number of codes per data bit
end
p.Nsx=p.Nsc*p.Ncx; % number of samples per code
p.Nsd=p.Nsx*p.Nxd; % number of samples per data bit
p.Ncd=p.Ncx*p.Nxd; % number of chips per data bit

%%%%%%%%%%%%%%%%%%%%%  signal frequencies  %%%%%%%%%%%%%%%%%%%%
if isfield(options,'Fs')==0
  p.Fs=1e9; % sampling frequency
end
p.Fc=p.Fs/p.Nsc; % chipping frequency
p.Fx=p.Fc/p.Ncx; % code frequency
p.Fd=p.Fx/p.Nxd; % data bit frequency

%%%%%%%%%%%%%%%%%%%%%%%  signal periods  %%%%%%%%%%%%%%%%%%%%%%%
p.Ts=1/p.Fs; % sample period
p.Tc=1/p.Fc; % chipping period
p.Tx=1/p.Fx; % code period
p.Td=1/p.Fd; % data bit period

%%%%%%%%%%%%%%%%%%%%%%%%%%%%   time   %%%%%%%%%%%%%%%%%%%%%%%%%%%
if isfield(options,'time')==0
  p.time=100; % set the end time of the simulation
end
t=0:p.Tx:p.time; % time vector for the simulation

%%%%%%%%%%%%%%%%%%%  Data message signal  %%%%%%%%%%%%%%%%%%%%%
p.ND=ceil(length(t)/p.Nxd); % number of data bits in simulation
if isfield(options,'data')==0
  p.data='yes'; % is there data on the signal
end
if isfield(options,'crc')==0
  p.crc=[1 0 1 1]; % set the crc used
end
p.n_crc=length(p.crc)-1; % number of bits per crc msg
if strcmp(p.data,'no') % if no data
```

```matlab
  g.Data=ones(1,p.ND); % the data message is all ones
%else
end
if isfield(options,'Ndw')==0
  p.Ndw=17; % number of data bits per data word
end
g.word=round(rand(1,p.Ndw)); % randomly generate word
% calculate the appended CRC code for the word
g.crc_rem=crcRem(g.word,zeros(1,p.n_crc),p.crc);
g.word_crc=[g.word g.crc_rem];
p.Nbw=p.Ndw+p.n_crc;
p.Nb2w=length(g.word_crc)*4; % number of bits for 2 words to repeat
p.Nbmsg=p.Nb2w*8; % number of bits per 1Hz msg
g.Data1Hz=2*[repmat(g.word_crc,1,4) ones(1,p.Nbmsg-p.Nb2w)]-1;
Data_precursor=repmat(g.Data1Hz,1,ceil(p.ND/p.Nbmsg));
g.Data=Data_precursor(1:p.ND);
p.Nmsgs=floor((p.ND-p.Nb2w)/p.Nbmsg)+1; % number of messages in sim
p.Tword=p.Td*(p.Ndw+p.n_crc);
g.Datak=reshape(repmat(g.Data,p.Nxd,1),1,p.ND*p.Nxd);
g.Datak=g.Datak(1:length(t)); % trim to match time vector

%%%%%%%%%%%%%%  THz hardware parameters  %%%%%%%%%%%%%%%%
if isfield(options,'lambda')==0
  p.lambda=1e-3; % wavelength
end
if isfield(options,'w')==0
  p.w=7e-3; % detector width
end
if isfield(options,'PT')==0
  %p.PT=120e-3; % transmit power
  p.PT=30e-3; % transmit power
end
if isfield(options,'phi_t')==0
  p.phi_t=2.5/180*pi; % transmission spreading angle
end
if isfield(options,'alpha_atm')==0
  p.alpha_atm=3e-6; % atmospheric attenuation coefficient
end
if isfield(options,'GR')==0
  p.GR=11.9; % receiver gain
end
p.Alens=p.w^2; % collection area of the lens
p.c=3e8; % speed of light

%%%%%%%%%%%%  diffraction grating parameters  %%%%%%%%%%%%
if isfield(options,'a')==0
  p.a=1.06*p.lambda; % slit width
end
if isfield(options,'d')==0
  p.d=3.86*p.lambda; % slit spacing
end
if isfield(options,'D')==0
  p.D=49.2*p.lambda; % distance to back wall
end
if isfield(options,'zeta_max')==0
  p.zeta_max=9/180*pi; % max grating sweep angle
end
p.Delta_max=p.D*tan(p.zeta_max); % max position of sweep
if isfield(options,'Fg')==0
  p.Fg=10; % frequency of the grating sweep
end
p.Tg=1/p.Fg; % period of grating sweep
grating_linear_distance=2*p.Delta_max/p.Tg*t; % total linear distance
p.Delta=mod(grating_linear_distance,2*p.Delta_max)-p.Delta_max; % position
p.Aslit=p.a*p.w; % area of one slit

%%%%%%%%%%%%%%%  aicraft position/motion  %%%%%%%%%%%%%%%%
if isfield(options,'pos_ctrl')==0
  g.pos_ctrl=[1500 200 0]'; % control position
else
  g.pos_ctrl=p.pos_ctrl; % change ctrl pos to geometry structure
  p=rmfield(p,'pos_ctrl'); % remove from parameters
end
```

```matlab
if isfield(options,'motion_model')==0
  g.motion_model='spline'; % model of the aircraft motion
else
  g.motion_model=p.motion_model; % change ctrl pos to geometry structure
  p=rmfield(p,'motion_model'); % remove from parameters
end

%%%%%%%%%%%%%%  Build Aircraft Geometry  %%%%%%%%%%%%%%%%%%
switch g.motion_model
  case 'spline'
    if isfield(options,'wp_size')==0
      g.wp_size=10; % size of motion
    else
      g.wp_size=p.wp_size; % change ctrl pos to geometry structure
      p=rmfield(p,'wp_size'); % remove from parameters
    end
    if isfield(options,'T_wp')==0
      g.T_wp=4.0726; % waypoint time constant
    else
      g.T_wp=p.T_wp; % change ctrl pos to geometry structure
      p=rmfield(p,'T_wp'); % remove from parameters
    end
    if isfield(options,'spline_order')==0
      g.spline_order=6; % order of splin used to connect way points
    else
      g.spline_order=p.spline_order; % change ctrl pos to geometry structure
      p=rmfield(p,'spline_order'); % remove from parameters
    end

    % define waypoint times with a little extra time at the beginning and
    % end of the time to deal with some of the edge effects of the spline
    t_wp=-g.spline_order/2.1*g.T_wp:g.T_wp:p.time+g.spline_order/2*g.T_wp;
    % check if pos_ctrl is only 1 control position, or series of waypoints
    if length(g.pos_ctrl(1,:))==1 % if only one position defined
      wp_bl=repmat(g.pos_ctrl,1,length(t_wp)); % make matrix of baseline wp
    else % if series of waypoints
      g.T_wp=p.time/(length(g.pos_ctrl(1,:))-1); % time spacing of wp
      t_wp=0:g.T_wp:p.time; % waypoint time vector
      wp_bl=g.pos_ctrl; % baseline waypoints
    end
    % waypoints
    wp=wp_bl+g.wp_size*(rand(3,length(t_wp))-.5); % add randomness to wps
    xs=spapi(g.spline_order,t_wp,wp(1,:)); % these structures that contain
    ys=spapi(g.spline_order,t_wp,wp(2,:)); % the polynomial equations
    zs=spapi(g.spline_order,t_wp,wp(3,:));
    xs=fn2fm(xs,'pp'); % this converts the structures from "B-" to "pp"
    ys=fn2fm(ys,'pp'); % for x, y, and z
    zs=fn2fm(zs,'pp');
    g.pos_tr=[ppval(xs,t); ppval(ys,t); ppval(zs,t)]; % define the position
    % calculate the derivatives
    xsp=xs; % start by setting the prime values for the
    ysp=ys; % structures equal to the function values
    zsp=zs;
    xsp.order=xs.order-1; % reduce the order
    ysp.order=ys.order-1; % in the structure
    zsp.order=zs.order-1;
    xsp.coefs=zeros(xsp.pieces,xsp.order); % initialize the
    ysp.coefs=zeros(ysp.pieces,ysp.order); % coeficients to zero
    zsp.coefs=zeros(zsp.pieces,zsp.order);
    for n=1:xsp.pieces
      xsp.coefs(n,:)=polyder(xs.coefs(n,:)); % calculate the derivatives of
      ysp.coefs(n,:)=polyder(ys.coefs(n,:)); % the polynomials (new coefs)
      zsp.coefs(n,:)=polyder(zs.coefs(n,:));
    end
    g.vel_tr=[ppval(xsp,t); ppval(ysp,t); ppval(zsp,t)]; % define velocity
    % calculate the derivatives again
    xspp=xsp; % start by setting the double
    yspp=ysp; % primed value equal to the prime
    zspp=zsp;
    xspp.order=xsp.order-1; % reduce the order
    yspp.order=ysp.order-1; % of the polynomial
    zspp.order=zsp.order-1;
    xspp.coefs=zeros(xspp.pieces,xspp.order); % initialize the coeficients
```

```
    yspp.coefs=zeros(yspp.pieces,yspp.order); % of the polynomials
    zspp.coefs=zeros(zspp.pieces,zspp.order);
    for n=1:xspp.pieces
      xspp.coefs(n,:)=polyder(xsp.coefs(n,:)); % take the derivative
      yspp.coefs(n,:)=polyder(ysp.coefs(n,:)); % of the polynomials
      zspp.coefs(n,:)=polyder(zsp.coefs(n,:));
    end
    g.acc_tr=[ppval(xspp,t); ppval(yspp,t); ppval(zspp,t)]; % define accel

  case 'constant'
    g.pos_tr=repmat(g.pos_ctrl,1,length(t));
    g.vel_tr=zeros(3,length(t));
    g.acc_tr=zeros(3,length(t));

  case 'random'
    if isfield(options,'sig_vel')==0
      g.sig_vel=.001*p.Nsc/100; % sigma of velocity
    else
      g.sig_vel=p.sig_vel; % change velocity sigma to geometry structure
      p=rmfield(p,'sig_vel'); % remove from parameters
    end

    F= [1 0 0 p.Tx 0 0;
        0 1 0 0 p.Tx 0;
        0 0 1 0 0 p.Tx;
        0 0 0 1 0 0;
        0 0 0 0 1 0;
        0 0 0 0 0 1];
    G_til=[0 0 0; 0 0 0; 0 0 0; 1 0 0; 0 1 0; 0 0 1];
    vel_noise=g.sig_vel*randn(3,length(t)-1);

    x=zeros(6,length(t));
    x(:,1)=[g.pos_ctrl(:,1); 0; 0; 0];
    for k=1:length(t)-1
      x(:,k+1)=F*x(:,k)+G_til*vel_noise(:,k);
    end
    g.pos_tr=x(1:3,:);
    g.vel_tr=x(4:6,:);
    g.acc_tr=[[0 0 0]' diff(g.vel_tr,1,2)/p.Tx]; % check is this right?


  case 'random 2'
    if isfield(options,'acc_disturb')==0
      g.acc_disturb=1; % size of acceleration disturbances
    else
      g.acc_disturb=p.acc_disturb; % change to geometry structure
      p=rmfield(p,'acc_disturb'); % remove from parameters
    end
    if isfield(options,'tau_motion')==0
      g.tau_motion=2; % the time constant of the acceleration disturbances
    else
      g.tau_motion=p.tau_motion; % change to geometry structure
      p=rmfield(p,'tau_motion'); % remove from parameters
    end

    % convert the acceleration disturbance into a one-sigma value
    sig_acc=g.acc_disturb*sqrt(g.tau_motion/p.Tx);
    % system motion model
    F= [1 0 0 p.Tx 0 0;
        0 1 0 0 p.Tx 0;
        0 0 1 0 0 p.Tx;
        0 0 0 1 0 0;
        0 0 0 0 1 0;
        0 0 0 0 0 1];
    G= [p.Tx^2/2 0 0;
        0 p.Tx^2/2 0;
        0 0 p.Tx^2/2;
        p.Tx 0 0;
        0 p.Tx 0;
        0 0 p.Tx];
    acc_noise=sig_acc*randn(3,length(t)-1);
    acc=zeros(3,length(t));
    acc(:,1)=g.acc_disturb*randn(3,1);
```

```matlab
    x=zeros(6,length(t));
    x(:,1)=[g.pos_ctrl(:,1); 0; 0; 0];
    for k=1:length(t)-1
      % advance the positions in time
      x(:,k+1)=F*x(:,k)+G*acc(:,k);
      % first order filter to determine the new acceleration
      acc(:,k+1)=(1-p.Tx/g.tau_motion)*acc(:,k)...
          +p.Tx/g.tau_motion*acc_noise(:,k);
    end
    % store the positions
    g.pos_tr=x(1:3,:);
    g.vel_tr=x(4:6,:);
    g.acc_tr=acc;

end


%%%%%%%%%%%%%%%%%%  measurement parameters  %%%%%%%%%%%%%%%%%%%%%
if isfield(options,'sig_v')==0
  p.sig_v=7.7e-3; % noise on raw signal measurements [V]
end

%%%%%%%%%%%%%%%%%%%  algorithm parameters  %%%%%%%%%%%%%%%%%%%%%
if isfield(options,'num_int_const')==0
  p.num_int_const=101; % numerical integration constant
end
if isfield(options,'taps')==0
  p.taps=linspace(-0.8*p.Tc,0.8*p.Tc,15)'; % correlator tap locations
end
p.NI=length(p.taps);
% the variance/std of errors on measurements of h, hdot, and I
if isfield(options,'sig_h')==0
  p.sig_h=1; % one-sigma error in the altitude measurement
end
if isfield(options,'sig_hdot')==0
  p.sig_hdot=1; % one-sigma error in the altitude rate measurement
end
% building the process and measurement noise covariance matrices
rho=zeros(length(p.taps)); % initialize matrix
for i=1:length(p.taps) % cycle through the taps
  rho(i,:)=1-abs((p.taps(i)-p.taps)/p.Tc); % distances between taps
end
Rho=rho.*(rho>0); % eliminate the negative values
p.Sig_I=p.Nsx*p.sig_v^2*Rho; % covariance of the correlator measurements

% the process noise
if isfield(options,'q_vel')==0
  p.q_vel=100e-3; % velocity state process noise [m/s^2]
end
if isfield(options,'q_v0')==0
  p.q_v0=1e-8; % amplitude state process noise [V]
end

p.Q=diag([p.q_vel^2 p.q_vel^2 p.q_vel^2 p.q_v0^2]); % process cov matrix
p.G_til=[0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    0 0 0 1];

p.R=p.Sig_I; % measurement noise matrix

p.F=[1 0 0 p.Tx 0 0 0; % state matrix
    0 1 0 0 p.Tx 0 0;
    0 0 1 0 0 p.Tx 0;
    0 0 0 1 0 0 0;
    0 0 0 0 1 0 0;
    0 0 0 0 0 1 0;
    0 0 0 0 0 0 1];
```

```
H_wls=p.Nsc*(p.Ncx+(p.Ncx+1)/p.Tc*p.taps.*(p.taps<=0)...
        -(p.Ncx+1)/p.Tc*p.taps.*(p.taps>0));
p.H_wls=repmat(H_wls,p.Nxd,1);

p.R_wls=zeros(p.NI*p.Nxd);
for i=1:p.Nxd
  p.R_wls((i-1)*p.NI+1:i*p.NI,(i-1)*p.NI+1:i*p.NI)=p.R;
end


end
```

A2.5.3   meas.m

```
function [I,I_tr,h,h_tr,hdot,hdot_tr,v0_tr,GD,Pm] = meas(g,k,x_guess,p)

% true states
r_tr=norm(g.pos_tr(:,k)); % true range between aircraft
%phi_tr=atan2(g.pos_tr(2,k),g.pos_tr(1,k)); % true planar angle
phi_3d_tr=atan2(norm(g.pos_tr(2:3,k)),g.pos_tr(1,k)); % true 3D angle
h_tr=g.pos_tr(3,k); % true altitude difference
hdot_tr=g.vel_tr(3,k); % true altitude rate
tau_tr=r_tr/p.c; % true time-of-flight delay
% estimated states
r_hat=norm(x_guess(1:3)); % range estimate
tau_hat=r_hat/p.c; % phase delay estimate

% link budget for power incident on detector
Pslit=p.PT*10^(-r_tr*p.alpha_atm/10)... % power incident on each slit
    *p.Aslit/(2*pi*r_tr^2*(1-cos(p.phi_t)))*cos(phi_3d_tr);
Pint=GDint(-pi/2,pi/2,g.pos_tr(:,k),10*p.num_int_const,p); % norm pow dist
Pm=2*Pslit/Pint; % max power density observed at detector
% convert power to voltage amplitude in the reciever electronics
v0_tr=p.GR*sqrt(Pm); % signal amplitude at each time step [V]

% integration limits for detector
zetap=atan2((p.Delta(k)+p.w/2),p.D); % top edge of detector
zetam=atan2((p.Delta(k)-p.w/2),p.D); % bottom edge of detector
GD=GDint(zetam,zetap,g.pos_tr(:,k),p.num_int_const,p); % diffraction gain

% Now calculate the measurements at each tap location
ti=tau_hat+p.taps; % tap locations
I_tr=zeros(size(ti)); % initialize the correlator vector
% calculate the predicted measurement based on states
for i=1:length(ti) % cycle through the taps
  if p.taps(i)<=0 % check if tap is positive slope
    I_tr(i)=g.Datak(k)*p.Nsc*v0_tr*sqrt(GD)...
        *(p.Ncx+(p.Ncx+1)/p.Tc*(ti(i)-tau_tr));
  else % otherwise assume tap is negative slope
    I_tr(i)=g.Datak(k)*p.Nsc*v0_tr*sqrt(GD)...
        *(p.Ncx-(p.Ncx+1)/p.Tc*(ti(i)-tau_tr));
  end
end

% calculate the noisy integrator measurements
I=mvnrnd(I_tr,p.Sig_I)';

% calculate the noisy altitude measurements
h=h_tr+p.sig_h*randn(1);
hdot=hdot_tr+p.sig_hdot*randn(1);

end
```

A2.5.4   measMod.m

```matlab
function [y,H] = measMod(x,p,Delta,lm_flag)
% x=[x y vx vy v0]'

% Check for optional input flag, if not, set the it to the default
if nargin<4 % optional input handling
  lm_flag=1; % if linear model flag not given, set to default of yes
end


% parameters from the parameter structure p
lambda=p.lambda; % carrier wavelength
a=p.a; % slit width
d=p.d; % slit spacing
w=p.w; % width of detector
D=p.D; % distance to detector


% states
phi=atan2(x(2),x(1)); % bearing angle
r=norm(x(1:3)); % range
rho=norm(x(1:2)); % planar range
%h=x(3); % altitude
v0=x(end); % signal amplitude (power)


% calculated values
c=p.c; % speed of light
Nsc=p.Nsc;
Ncx=p.Ncx;
Tc=p.Tc;
taps=p.taps;
%tau=r/c; % time of flight delay


% intgration limits
zetap=atan2((Delta+w/2),D); % top edge of detector
zetam=atan2((Delta-w/2),D); % bottom edge of detector
% variables in measurement equation
[GD,alpha,beta,dzeta]=GDint(zetam,zetap,x(1:3),p.num_int_const,p);


% calculate the predicted measurement based on states
y=Nsc*v0*sqrt(GD)*(Ncx+(Ncx+1)/Tc*taps.*(taps<=0)...
    -(Ncx+1)/Tc*taps.*(taps>0));


% Check if the linear model flag says yes or no to calculating
if lm_flag
  % calculate the derivatives
  dphidx=-x(2)/rho^2;
  dphidy=x(1)/rho^2;
  drdx=x(1)/r;
  drdy=x(2)/r;
  drdz=x(3)/r;
  dalphadx=pi*a/lambda*cos(phi)*dphidx;
  dalphady=pi*a/lambda*cos(phi)*dphidy;
  dbetadx=pi*d/lambda*cos(phi)*dphidx;
  dbetady=pi*d/lambda*cos(phi)*dphidy;

  dGDdx=sum(2*sinc(alpha/pi).*cos(beta).^2 ... % partial derivative x
      .*(cos(alpha)./alpha-sin(alpha)./alpha.^2)*dalphadx...
      -2*sinc(alpha/pi).^(2).*cos(beta).*sin(beta)*dbetadx)*dzeta;
  dGDdy=sum(2*sinc(alpha/pi).*cos(beta).^2 ... % partial derivative y
      .*(cos(alpha)./alpha-sin(alpha)./alpha.^2)*dalphady...
      -2*sinc(alpha/pi).^(2).*cos(beta).*sin(beta)*dbetady)*dzeta;

  % Now I can calculate the partial derivatives that go into the matrix
  dIidv0=Nsc*sqrt(GD)*(Ncx...
      +(Ncx+1)/Tc*taps.*(taps<0)...
      -(Ncx+1)/Tc*taps.*(taps>0));
  dIidx=Nsc*v0*(.5/sqrt(GD)*dGDdx*(Ncx...
      +(Ncx+1)/Tc*taps.*(taps<0)...
      -(Ncx+1)/Tc*taps.*(taps>0))...
      -sqrt(GD)*(Ncx+1)/(c*Tc)*drdx*(taps<0)...
      +sqrt(GD)*(Ncx+1)/(c*Tc)*drdx*(taps>0));
  dIidy=Nsc*v0*(.5/sqrt(GD)*dGDdy*(Ncx...
      +(Ncx+1)/Tc*taps.*(taps<0)...
      -(Ncx+1)/Tc*taps.*(taps>0))...
      -sqrt(GD)*(Ncx+1)/(c*Tc)*drdy*(taps<0)...
```

```
       +sqrt(GD)*(Ncx+1)/(c*Tc)*drdy*(taps>0));
    dIidz=Nsc*v0*sqrt(GD)*(...
        -(Ncx+1)/(c*Tc)*drdz*(taps<0)...
        +(Ncx+1)/(c*Tc)*drdz*(taps>0));

    % Put it all together into the matrix
    H=[dIidx dIidy dIidz zeros(length(taps),3) dIidv0];

else
    H=0;
end

end
```

A2.5.5 sysMod.m

```
function [y,yHat,H,R]=sysMod(p,I,h,hdot,yHat,xHatM,H,w1_avail,w2_avail)
% first check if the data needs to be absolute valued. I may be able to
% play some games here, but for now I am just going to keep it simple
if strcmp(p.data,'yes') % if there is data modulated onto the signal
    I=abs(I); % then absolute value the integrator measurement
end

if w1_avail && w2_avail % if both words are available
    y=[I; h; hdot];
    yHat=[yHat; xHatM(3); xHatM(6)];
    H=[H; 0 0 1 0 0 0 0; 0 0 0 0 0 1 0];
    R=[p.R zeros(p.NI,2); zeros(2,p.NI) [p.sig_h^2 0; 0 p.sig_hdot^2]];
elseif w1_avail % if only word 1 is available
    y=[I; h];
    yHat=[yHat; xHatM(3)];
    H=[H; 0 0 1 0 0 0 0];
    R=[p.R zeros(p.NI,1); zeros(1,p.NI) p.sig_h^2];
elseif w2_avail % if only word 2 is available
    y=[I; hdot];
    yHat=[yHat; xHatM(6)];
    H=[H; 0 0 0 0 0 1 0];
    R=[p.R zeros(p.NI,1); zeros(1,p.NI) p.sig_hdot^2];
else % if none of the words are available
    y=I;
    R=p.R;
end

end
```

A2.5.6 GDint.m

```
function [GD,alpha,beta,dzeta] = GDint(low_lim,up_lim,pos,int_const,p)

phi=atan2(pos(2),pos(1)); % bearing angle
zeta=linspace(low_lim,up_lim,int_const); % angles integrated over
dzeta=zeta(2)-zeta(1); % integration step size
alpha=pi*p.a/p.lambda*(sin(zeta)+sin(phi)); % sinc argument
beta=pi*p.d/p.lambda*(sin(zeta)+sin(phi)); % cos argument
GD=sum(sinc(alpha/pi).^(2).*cos(beta).^2)*dzeta; % grating gain

end
```

A2.5.7 dataAlg.m

```matlab
function [w1_avail,w2_avail,y_c,d_hat,d_st]=dataAlg(p,I,y_c,d_hat,d_st)

y_c=[y_c; I]; % add new integrator measurements to the comm meas vector

% check if there are enough measurements to perform bit estimation, if not
% the algorithm will say the data is not available
if length(y_c)==p.Nxd*p.NI % if enough measurements
  % estimate the data value
  D_hat=(p.H_wls'/p.R_wls*p.H_wls)\p.H_wls'/p.R_wls*y_c;
  d_hat=[d_hat sign(D_hat)]; % store the estimated data bit
  d_st=[d_st d_hat(end)]; % add the new bit to the storage array
  % Now, check if an entire set of 2 words has come in
  if length(d_hat)==p.Nb2w
    % check if this segment is just blank space by looking
    % for a lot of ones. In theory if there were no errors then it should
    % sum to 80, the number of bits in the message, but this number could
    % be smaller due to bit errors, so I will settle for a large percentage
    % of that number. If the sum is below that large percentage, I will
    % assume that it is an actual data transmission, otherwise data is not
    % available
    if sum(d_hat)<0.85*p.Nb2w
      % run a CRC check to see if the words transmitted successfully
      [w1_avail,w2_avail]=msgCheck(d_hat,p); % check the crcs for the message
    else % if no data in this segment
      w1_avail=0; % data not available
      w2_avail=0; % data not available
    end
    d_hat=[]; % reset the data bit vector
  else % if not enough data bits to check for the word
    w1_avail=0; % data not available
    w2_avail=0; % data not available
  end
  y_c=[]; % reset the measurement vector
else % if not enough integrator measurements
  w1_avail=0; % data not available
  w2_avail=0; % data not available
end

end
```

A2.5.8 msgCheck.m

```matlab
function [w1c,w2c]=msgCheck(msg,p)

% extract the two words and their repetitions from the data message and
% adjust them to be between 0 and 1, not -1 and 1
word1=(msg(1:p.Nbw)+1)/2; % word 1
word2=(msg(1*p.Nbw+(1:p.Nbw))+1)/2; % word 2
word1r=(msg(2*p.Nbw+(1:p.Nbw))+1)/2; % word 1 repeat
word2r=(msg(3*p.Nbw+(1:p.Nbw))+1)/2; % word 2 repeat

% check the data to make sure there were no errors in transmission. First
% find the remainders after processing through the CRC divisor
remw1=crcRem(word1(1:p.Ndw),word1(p.Ndw+(1:p.n_crc)),p.crc); % word 1
remw2=crcRem(word2(1:p.Ndw),word2(p.Ndw+(1:p.n_crc)),p.crc); % word 2
remw1r=crcRem(word1r(1:p.Ndw),word1r(p.Ndw+(1:p.n_crc)),p.crc); % word 1 r
remw2r=crcRem(word2r(1:p.Ndw),word2r(p.Ndw+(1:p.n_crc)),p.crc); % word 2 r
% and then check that the word was correct at least one of the times it was
% transmitted.
w1c=isequal(remw1,zeros(1,p.n_crc)) || isequal(remw1r,zeros(1,p.n_crc));
w2c=isequal(remw2,zeros(1,p.n_crc)) || isequal(remw2r,zeros(1,p.n_crc));

end
```

A2.5.9 crcRem.m

```matlab
function rem=crcRem(D,app,crc)
% This function performs a cyclic redundancy check on an array of binary
% data bits D. It takes the data word D, the appended bits for the check,
% and the CRC divisor, and it returns the remainder of the CRC operation.

if min(D)<0 || max(D)>1 % check that data is between 1 and 0
  error('Warning: Data must be 0s and 1s')
end
if min(app)<0 || max(app)>1 % check that appeded bits are between 1 and 0
  error('Warning: Appended bits must be 0s and 1s')
end

n=length(crc)-1; % calculate the order of the CRC divisor
if length(app)~=n % check that the appeded bits are the correct length
  error('Warning: Appended bits must one less than length of CRC')
end

D_app=[D app]; % append the data

for i=1:length(D) % cycle through each bit in the data
  if D_app(i)==1 % if the bit is a 1, then apply the divisor
    D_app=[D_app(1:i-1) xor(D_app(i:i+n),crc) D_app(i+n+1:end)];
  end
end

rem=D_app(end-n+1:end); % return the reminder after the operation

end
```

A2.6   Appendix Simulations

A2.6.1   near_field2_2017_6_13.m

```matlab
% J. Scott Parker
% 6-13-2017

% mid-field plots for appendix of dissertation

clear

% User parameters
N=2; % number of slits
phi=0/180*pi; % angle of incidence
slice_distance=15; % pattern slice distance normalized by wavelength
% N=2 --> 15
% N=3 --> 30
% N=4 --> 20, 48
% N=5 --> 28, 19, 78
% N=6 --> 38
% N=7 --> 23


% equipment parameters
lambda=1e-3; % wavelength
a=1.06*lambda; % width of slit
d=3.86*lambda; % slit spacing
x_slit=(0:N-1)*d-(N-1)*d/2; % locations of slits
D_select=slice_distance*lambda; % selected slice distance

% define the variables that will be swept for the cartesian plot
D_max=10*((N-1)*d+a); % limit for D
```

```matlab
x_max=.5*D_max; % limit for x
D=linspace(10*a,D_max,310); % horizontal distance from grating center
x=linspace(-x_max,x_max,1250); % lateral distance from center of grating

% initialize some arrays for the loop
P=zeros(length(D),length(x)); % the power matrix
sinc_term=zeros(1,N); % the terms of the sinc function
p=zeros(2,N); % the phasors

for iD=1:length(D) % cycle through all Ds
  for ix=1:length(x) % cycle through all xs
    for n=1:N % cycle through each slit
      zeta_slit=atan2(x(ix)+x_slit(n),D(iD)); % angle of point from slit
      alpha=pi*a/lambda*(sin(zeta_slit)+sin(phi)); % slit alpha coordinate
      sinc_term(n)=1/N*sinc(alpha/pi)^2; % sinc term for the slit
      % calculate the extra distance associated with each individual slit
      ln=norm([x(ix)+x_slit(n) D(iD)]')+d*(n-1)*sin(phi); % slit distance
      p(:,n)=[cos(2*pi*ln/lambda); sin(2*pi*ln/lambda)]; % phasor for slit
    end
    P(iD,ix)=sum(sinc_term)*norm(1/N*sum(p,2))^2; % power seen at point
  end
end

y_select=sqrt(D_select^2-x.^2);
% plot the power
fig=30;
figure(fig)
clf
surf(x/lambda,D/lambda,P,'EdgeColor','none')
view(0,90)
hold on
title(['Cartesian Pattern for N = ' num2str(N)])
xlabel('Lateral Position [wavelengths \lambda]')
ylabel('Distance from Grating [wavelengths \lambda]')
xlim([x(1) x(end)]/lambda)
ylim([D(1) D(end)]/lambda)
line(x/lambda,y_select/lambda,20*ones(size(x)),'color','r','LineWidth',3)
set(gca,'fontsize',24)

% polar plot
% define variables for the polar sweep
r=linspace(5*a,D_max,310); % radial distance from center of grating
zeta=linspace(-90,90,821)/180*pi; % angular position from grating center
% convert the D_select into polar coordinates
x_select=linspace(-D_select*tan(pi/2-.1),D_select*tan(pi/2-.1),500);
zeta_select=atan2(x_select,D_select); % internal angle at selected slice
r_select=zeros(size(x_select));
for ix=1:length(x_select)
  r_select(ix)=norm([x_select(ix); D_select]);
end

% initialize the arrays
P=zeros(length(r),length(zeta)); % power matrix

for ir=1:length(r) % cycle through all radii
  for iz=1:length(zeta) % cycle through all angles
    for n=1:N % cycle through all slits
      % calculate the cartesian coordinates
      x=r(ir)*sin(zeta(iz))+x_slit(n); % lateral position
      D=r(ir)*cos(zeta(iz)); % horizontal position
      zeta_slit=atan2(x,D); % internal angle to slit
      alpha=pi*a/lambda*(sin(zeta_slit)+sin(phi)); % alpha coord for slit
      sinc_term(n)=1/N*sinc(alpha/pi)^2; % sinc term for slit
      ln=norm([x D]')+d*(n-1)*sin(phi); % additional distance due to slit
      p(:,n)=[cos(2*pi*ln/lambda); sin(2*pi*ln/lambda)]; % phasor
    end
    P(ir,iz)=sum(sinc_term)*norm(1/N*sum(p,2))^2; % power
  end
end

% plot results
fig=fig+1;
figure(fig)
```

```
clf
surf(zeta/pi*180,r/lambda,P,'EdgeColor','none')
view(0,90)
hold on
line(zeta/pi*180,D_select*ones(size(zeta))/lambda,20*ones(size(zeta)),...
    'color','r','LineWidth',3)
title(['Polar Pattern for N = ' num2str(N)])
xlabel('Internal Angle \zeta [deg]')
ylabel('Polar Distance from Grating [wavelengths \lambda]')
xlim([zeta(1) zeta(end)]/pi*180)
ylim([r(1) r(end)]/lambda)
set(gca,'fontsize',24)

% power at the slice, accounting for the power distribution
P_select=zeros(1,length(zeta)); % initialize the array

for iz=1:length(zeta) % cycle through all angles
  for n=1:N % cycle through all slits
    % calculate the cartesian coordinates
    x=D_select*sin(zeta(iz))+x_slit(n); % lateral position
    D=D_select; % horizontal position
    zeta_slit=atan2(x,D); % internal angle to slit
    alpha=pi*a/lambda*(sin(zeta_slit)+sin(phi)); % alpha coord for slit
    sinc_term(n)=1/N*sinc(alpha/pi)^2; % sinc term for slit
    ln=norm([x D]')+d*(n-1)*sin(phi); % additional distance due to slit
    p(:,n)=[cos(2*pi*ln/lambda); sin(2*pi*ln/lambda)]; % phasor
  end
  P_select(iz)=sum(sinc_term)*norm(1/N*sum(p,2))^2; % power
end

dzeta=zeta(2)-zeta(1);
P_select_int=sum(P_select)*dzeta;
P_m_select=N/P_select_int;
P_select=P_m_select*P_select;

% plot the pattern of the specific slice
fig=fig+1;
figure(fig)
clf
plot(zeta/pi*180,P_select,'LineWidth',2)
grid on
title(['Mid-Field Pattern at D = ' num2str(D_select/lambda)...
    '\lambda for N = ' num2str(N)])
xlabel('Internal Angle \zeta [deg]')
ylabel('Normalized Power Density')
xlim([-90 90])

P_far=zeros(1,length(zeta)); % initialize the array
for iz=1:length(zeta) % cycle through all angles
  for n=1:N % cycle through all slits
    % calculate the cartesian coordinates
    x=D_max*sin(zeta(iz))+x_slit(n); % lateral position
    D=D_max; % horizontal position
    zeta_slit=atan2(x,D); % internal angle to slit
    alpha=pi*a/lambda*(sin(zeta_slit)+sin(phi)); % alpha coord for slit
    sinc_term(n)=1/N*sinc(alpha/pi)^2; % sinc term for slit
    ln=norm([x D]')+d*(n-1)*sin(phi); % additional distance due to slit
    p(:,n)=[cos(2*pi*ln/lambda); sin(2*pi*ln/lambda)]; % phasor
  end
  P_far(iz)=sum(sinc_term)*norm(1/N*sum(p,2))^2; % power
end

dzeta=zeta(2)-zeta(1);
P_far_int=sum(P_far)*dzeta;
P_m_far=N/P_far_int;

P_far=P_m_far*P_far;

alpha=pi*a/lambda*(sin(zeta)+sin(phi));
beta=pi*d/lambda*(sin(zeta)+sin(phi));
P_doub=3.575*sinc(alpha/pi).^(2).*cos(beta).^2;

% plot the pattern of the far field
```

```
fig=fig+1;
figure(fig)
clf
plot(zeta/pi*180,P_far,'LineWidth',2)
hold on
%plot(zeta/pi*180,P_doub,'LineWidth',2,'LineStyle','-.')
%legend('N = 2',['N = ' num2str(N)])
grid on
title(['Far-Field Interference Pattern (N = ' num2str(N) ')'])
xlabel('Internal Angle \zeta [deg]')
ylabel('Normalized Power Density')
xlim([-90 90])
```

Works Cited

[1] D. Büchner, J. A. A. Engelbrecht, J. Adams, and C. Redelinghuys, "Towards Automatic Flight Control for Commercial Airliners in Formation Flight⋆," in *19th IFAC world congress, Cape Town*, 2014.

[2] M. Brodecki, K. Subbarao, and Q. P. Chu, "Formation Flight Control System for In-Flight Sweet Spot Estimation," in *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Dallas/Ft. Worth, TX, 2013.

[3] M. S. Hemati, J. D. Eldredge, and J. L. Speyer, "Wake Sensing for Aircraft Formation Flight," *Journal of Guidance, Control, and Dynamics*, vol. 37, no. 2, pp. 513–524, 2014.

[4] S. M. Khanafseh and B. Pervan, "Autonomous Airborne Refueling of Unmanned Air Vehicles Using the Global Positioning System," *Journal of Aircraft*, vol. 44, no. 5, pp. 1670–1682, 2007.

[5] W. R. Williamson, G. J. Glenn, V. T. Dang, J. L. Speyer, S. M. Stecko, and J. M. Takacs, "Sensor Fusion Applied to Autonomous Aerial Refueling," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 1, pp. 262–275, 2009.

[6] C. M. Haissig, "Military formation flight as a model for increased capacity in civilian airspace," in *Digital Avionics Systems Conference, 2004. DASC 04. The 23rd*, 2004, vol. 1, p. 1.C.4-1.1-9 Vol.1.

[7] N. Bizinos and C. Redelinghuys, "Tentative Study of Passenger Comfort During Formation Flight Within Atmospheric Turbulence," *Journal of Aircraft*, vol. 50, no. 3, pp. 886–900, 2013.

[8] G. C. Bower, T. Flanzer, and I. M. Kroo, "Formation Geometries and Route Optimization for Commercial Formation Flight," in *AIAA Applied Aerodynamics Conference, AIAA-2009-3615*, 2009.

[9] A. Ning, T. C. Flanzer, and I. M. Kroo, "Aerodynamic performance of extended formation flight," *Journal of Aircraft*, vol. 48, no. 3, pp. 855–865, Jun. 2011.

[10] J. Rife, "Collaborative Positioning for Formation Flight of Cargo Aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 1, pp. 304–307, 2012.

[11] R. Benney, J. Barber, J. McGrath, J. McHugh, G. Noetscher, and S. Tavan, "The New Military Applications of Precision Airdrop Systems," in *American Institute of Aeronautics and Astronautics*, Arlington, VA, 2005, vol. 7069.

[12] J. Ismay, "How the U.S. Pulled Off its Humanitarian Aid Missions to the Yazidis," *Los Angeles Daily News*, 18-Aug-2014.

[13] P. Ross, "US Airdrops Iraq: What It Is And How It Works," *International Business Times*, 08-Aug-2014.

[14] S. Beaubien, "Rethinking Strategic Brigade Airdrop," DTIC Document, 1997.

[15] W. B. Blake, "Development of the C-17 Formation Airdrop Element Geometry," *Journal of Aircraft*, vol. 35, no. 2, pp. 175–182, 1998.

[16] S. Pullen and G. X. Gao, "GNSS Jamming in the Name of Privacy-Potential Threat to GPS Aviation," *Inside GNSS*, vol. 7, no. 2, pp. 34–43, 2012.

[17] John A. Volpe National Transportation System Center, "Vulnerability Assessment of the Transportation Infrastructure Relying on the Global Positioning System," U.S. Department of Transportation, Aug. 2001.

[18] S. Lo, Y.-H. Chen, and P. Enge, "Flight Test of Universal Access Transceiver (UAT) Transmissions to Provide Alternative Positioning Navigation and Timing (APNT)," in *Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2015)*, Tampa, FL, 2015, pp. 1468–1477.

[19] T. Ertan and M. L. Psiaki, "Alternative Position and Navigation Based on DME Accumulated Delta Range," in *Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014)*, Tampa, FL, 2014, pp. 3055–3064.

[20] Y.-S. Lee, *Principles of Terahertz Science and Technology*, vol. 170. Springer, 2009.

[21] E. R. Brown, "Fundamentals of Terrestrial Millimeter-Wave and THz Remote Sensing," *Int. J. Hi. Spe. Ele. Syst.*, vol. 13, no. 4, pp. 995–1097, Dec. 2003.

[22] E. Bründermann, H.-W. Hübers, and M. F. Kimmitt, *Terahertz techniques*, vol. 151. Berlin: Springer, 2012.

[23] C. M. Armstrong, "The Truth About Terahertz," *IEEE Spectrum*, vol. 49, no. 9, pp. 36–41, 17-Aug-2012.

[24] M. Cooke, "Pushing Semiconductor Detectors into the Terahertz Gap," *III-Vs Review*, vol. 19, no. 8, pp. 36–38, Nov-2006.

[25] G. P. Williams, "Filling the THz Gap—High Power Sources and Applications," *Rep. Prog. Phys.*, vol. 69, no. 2, p. 301, Feb. 2006.

[26] G. A. Siles, J. M. Riera, and P. García-del-Pino, "THz Propagation Research Within the TERASENSE Project: Atmospheric Gases Attenuation," in *Antennas and Propagation (EuCAP), 2010 Proceedings of the Fourth European Conference on*, 2010, pp. 1–5.

[27] M. Tonouchi, "Cutting-Edge Terahertz Technology," *Nat Photon*, vol. 1, no. 2, pp. 97–105, Feb. 2007.

[28] X.-C. Zhang and J. Xu, *Introduction to THz Wave Photonics*. Springer Science & Business Media, 2009.

[29] R. A. Lewis, "A review of terahertz sources," *J. Phys. D: Appl. Phys.*, vol. 47, no. 37, p. 374001, 2014.

[30] T. Kleine-Ostmann and T. Nagatsuma, "A Review on Terahertz Communications Research," *J Infrared Milli Terahz Waves*, vol. 32, no. 2, pp. 143–171, Feb. 2011.

[31] X. Mei *et al.*, "First Demonstration of Amplification at 1 THz Using 25-nm InP High Electron Mobility Transistor Process," *IEEE Electron Device Letters*, vol. 36, no. 4, pp. 327–329, Apr. 2015.

[32] Virginia Diodes, Inc., "Virginia Diodes, Inc.," *Your Source for Terahertz and mm-Wave Products*, 2014. [Online]. Available: http://vadiodes.com/index.php/en/. [Accessed: 31-Oct-2014].

[33] A. J. Fitzgerald *et al.*, "Terahertz Pulsed Imaging of Human Breast Tumors," *Radiology*, vol. 239, no. 2, pp. 533–540, May 2006.

[34] S. Wang, B. Ferguson, D. Abbott, and X.-C. Zhang, "T-ray Imaging and Tomography," *Journal of Biological Physics*, vol. 29, no. 2–3, pp. 247–256, Jun. 2003.

[35] Shuting Fan, Yuezhi He, B. S. Ung, and E. Pickwell-MacPherson, "The Growth of Biomedical Terahertz Research," *Journal of Physics D: Applied Physics*, vol. 47, no. 37, p. 374009 (11 pp.), Sep. 2014.

[36] C. Kulesa, "Terahertz Spectroscopy for Astronomy: From Comets to Cosmology," *IEEE Transactions on Terahertz Science and Technology*, vol. 1, no. 1, pp. 232–240, Sep. 2011.

[37] A. Partnership *et al.*, "The 2014 ALMA Long Baseline Campaign: First Results from High Angular Resolution Observations toward the HL Tau Region," *ApJL*, vol. 808, no. 1, p. L3, 2015.

[38] T. Yasui, Y. Kabetani, Y. Ohgi, S. Yokoyama, and T. Araki, "Absolute distance measurement of optically rough objects using asynchronous-optical-sampling terahertz impulse ranging," *Appl. Opt.*, vol. 49, no. 28, pp. 5262–5270, Oct. 2010.

[39] B. Hussain *et al.*, "Simultaneous Determination of Thickness and Refractive Index Based on Time-of-Flight Measurements of Terahertz Pulse," *Appl. Opt.*, vol. 51, no. 21, pp. 5326–5330, Jul. 2012.

[40] D. Mittleman, *Sensing with Terahertz Radiation*. Springer, 2013.

[41] M. Caris *et al.*, "Very high resolution radar at 300 GHz," in *2014 11th European Radar Conference (EuRAD), 8-10 Oct. 2014*, 2014, pp. 494–6.

[42] N. Pohl, S. Stanko, M. Caris, A. Tessmann, and M. Schlechtweg, "An ultra-high resolution radar-system operating at 300 GHz," in *2015 IEEE Topical Conference on Wireless Sensors and Sensor Networks (WiSNet), 25-28 Jan. 2015*, 2015, pp. 62–4.

[43] M. Caris, S. Stanko, S. Palm, R. Sommer, A. Wahlen, and N. Pohl, "300 GHz radar for high resolution SAR and ISAR applications," in *2015 16th International Radar Symposium (IRS), 24-26 June 2015*, 2015, pp. 577–80.

[44] A. Levi, N. S. Kopeika, A. Abramovich, and D. Rozban, "Fourier Imaging and Distance Approximation Using Time of Flight Method for Terahertz Wave Imaging," *Opt. Eng*, vol. 53, no. 8, pp. 083104–083104, 2014.

[45] P. Dzwonkowski, P. Samczynski, K. Kulpa, J. Drozdowicz, D. Gromek, and P. Krysik, "Towards a mobile low-terahertz SAR imaging radar #x2014; Limitations and prospects," in *Signal Processing Symposium (SPSympo), 2015*, 2015, pp. 1–5.

[46] D. R. Vizard, M. Gashinova, E. G. Hoare, and M. Cherniakov, "Low THz automotive radar developments employing 300-600 GHz frequency extenders," in *16th International Radar Symposium, IRS 2015, June 24, 2015 - June 26, 2015*, 2015, vol. 2015–August, pp. 209–214.

[47] W. R. Tribe, D. A. Newnham, P. F. Taday, and M. C. Kemp, "Hidden Object Detection: Security Applications of Terahertz Technology," 2004, vol. 5354, pp. 168–176.

[48] K. Kawase, Y. Ogawa, Y. Watanabe, and H. Inoue, "Non-destructive terahertz imaging of illicit drugs using spectral fingerprints," *Opt. Express*, vol. 11, no. 20, pp. 2549–2554, Oct. 2003.

[49] M. Salhi, T. Kleine-Ostmann, M. Kannicht, S. Priebe, T. Kürner, and T. Schrader, "Broadband channel propagation measurements on millimeter and sub-millimeter waves in a desktop download scenario," in *Microwave Conference Proceedings (APMC), 2013 Asia-Pacific*, 2013, pp. 1109–1111.

[50] M. Salhi, T. Kleine-Ostmann, T. Schrader, M. Kannicht, S. Priebe, and T. Kurner, "Broadband Channel Measurements in a Typical Office Environment at Frequencies Between 50 GHz and 325 GHz," in *Microwave Conference (EuMC), 2013 European*, 2013, pp. 175–178.

[51] J. Federici and L. Moeller, "Review of Terahertz and Subterahertz Wireless Communications," *Journal of Applied Physics*, vol. 107, no. 11, p. 111101, Jun. 2010.

[52] C. Han and I. F. Akyildiz, "Distance-aware multi-carrier (DAMC) modulation in Terahertz Band communication," in *2014 IEEE International Conference on Communications (ICC)*, 2014, pp. 5461–5467.

[53] C. Jastrow, K. Munter, R. Piesiewicz, T. Kurner, M. Koch, and T. Kleine-Ostmann, "300 Ghz transmission system," *Electronics Letters*, vol. 44, no. 3, pp. 213–214, Jan. 2008.

[54] C. Jastrow *et al.*, "Wireless digital data transmission at 300 GHz," *Electronics Letters*, vol. 46, no. 9, pp. 661–663, Apr. 2010.

[55] S. Priebe, C. Jastrow, M. Jacob, T. Kleine-Ostmann, T. Schrader, and T. Kurner, "Channel and Propagation Measurements at 300 GHz," *IEEE*

Transactions on Antennas and Propagation, vol. 59, no. 5, pp. 1688–1698, May 2011.

[56] R. Piesiewicz, M. Jacob, M. Koch, J. Schoebel, and T. Kurner, "Performance Analysis of Future Multigigabit Wireless Communication Systems at THz Frequencies With Highly Directive Antennas in Realistic Indoor Environments," *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 14, no. 2, pp. 421–430, Mar. 2008.

[57] J. O'Callaghan, "Terahertz Radiation Breakthrough Could Lead To 100 Times Faster Wireless Networks," *IFLScience*, 14-Sep-2015. [Online]. Available: http://www.iflscience.com/technology/terahertz-radiation-breakthrough-could-lead-100-times-faster-wireless-networks. [Accessed: 22-Dec-2015].

[58] T. Nagatsuma, "300-GHz-band wireless communications with high-power photonic sources," in *General Assembly and Scientific Symposium (URSI GASS), 2014 XXXIth URSI*, 2014, pp. 1–4.

[59] T. Nagatsuma *et al.*, "30-Gbit/s Wireless Transmission over 10 meters at 300 GHz," in *Proceedings of International Conference on Infrared, Millimeter, and Terahertz Waves*, Tucson, AZ, 2014.

[60] G. A. Siles, J. M. Riera, and P. García-del-Pino, "Considerations on Cloud Attenuation at 100 and 300 GHz for Propagation Measurements Within the TeraSense Project," in *Antennas and Propagation (EUCAP), Proceedings of the 5th European Conference on*, 2011, pp. 90–94.

[61] S. M. Fox, T. G. Bailey, and W. B. Carlton, "Personnel Airdrop Simulation," *Simulation*, vol. 76, no. 1, pp. 4–21, 2001.

[62] W. Blake, "Prediction of paratroop/wake vortex encounters during formation airdrop," in *Proceedings of the AIAA Atmospheric Flight Mechanics Conference*, 1996, vol. 96, p. 3387.

[63] J. S. Parker and J. Rife, "Precise Bearing Determination for Formation Flight Using Terahertz Signals," in *Proceedings of the ION 2013 Pacific PNT Meeting*, Honolulu, Hawaii, 2013, pp. 811–821.

[64] J. S. Parker, "Terahertz-Based Relative Positioning of Aircraft Flying in Formation," M.S. Thesis, Tufts University, Medford, MA, 2013.

[65] J. S. Parker, P. Mickelson, J. Yeak, K. Kremeyer, and J. Rife, "Exploiting the Terahertz Band for Radionavigation," *J Infrared Milli Terahz Waves*, pp. 1–22, Jun. 2016.

[66] N. D. Pham, "The Economic Benefits of Commercial GPS Use in the U.S. and The Costs of Potential Disruption," NDP Consulting Group, Jun. 2011.

[67] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements and Performance Second Edition*. Lincoln, MA: Ganga-Jamuna Press, 2011.

[68] R. Piesiewicz, T. Kleine-Ostmann, N. Krumbholz, D. Mittleman, M. Koch, and T. Kurner, "Terahertz characterisation of building materials," *Electronics Letters*, vol. 41, no. 18, pp. 1002–1004, Sep. 2005.

[69] S. Loyka and A. Kouki, "Using two ray multipath model for microwave link budget analysis," *IEEE Antennas and Propagation Magazine*, vol. 43, no. 5, pp. 31–36, Oct. 2001.

[70] H. K. Lee and C. Rizos, "Position-domain Hatch filter for kinematic differential GPS/GNSS," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 30–40, Jan. 2008.

[71] J. S. Parker, "Terahertz (THz) Interferometry for Bearing Angle Measurement," in *Proceedings of the 29th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2016)*, Portland, OR, 2016.

[72] J. T. Isaacs *et al.*, "GPS-Optimal Micro Air Vehicle Navigation in Degraded Environments," in *2014 American Control Conference - ACC 2014, 4-6 June 2014*, 2014, pp. 1864–71.

[73] H. Kloeden, D. Schwarz, E. M. Biebl, and R. H. Rasshofer, "Vehicle Localization Using Cooperative RF-Based Landmarks," in *2011 IEEE Intelligent Vehicles Symposium, IV'11, June 5, 2011 - June 9, 2011*, 2011, pp. 387–392.

[74] L. Xiong, "A Selective Model to Suppress NLOS Signals in Angle-of-Arrival (AOA) Location Estimation," in *The Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 1998*, 1998, vol. 1, pp. 461–465 vol.1.

[75] J. Rife, "Design of a Distributed Localization Algorithm to Process Angle-of-Arrival Measurements," in *2015 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, Woburn, MA, 2015, pp. 1–6.

[76] G. Vukasin and J. Rife, "Decentralized Position and Attitude Estimation Using Angle-of-Arrival Measurements," in *Proceedings of the 28th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2015)*, Tampa, FL, 2015, pp. 1436–1445.

[77] K. Lee, J. Lim, and J. Ahn, "Young's Experiment With a Double Slit of Sub-Wavelength Dimensions," *Optics Express*, vol. 21, no. 16, pp. 18805–18811, 2013.

[78] A. D. Squires, E. Constable, and R. A. Lewis, "3D Printed Terahertz Diffraction Gratings And Lenses," *J Infrared Milli Terahz Waves*, vol. 36, no. 1, pp. 72–80, Jan. 2015.

[79] K. G. Williams and J. J. Kurtz, "Infrared Spectrophotometer Employing Sweep Diffraction Grating," US6528791 B1, 04-Mar-2003.

[80] C.-J. Lin, Y.-T. Li, C.-F. Hsieh, R.-P. Pan, and C.-L. Pan, "Manipulating terahertz wave by a magnetically tunable liquid crystal phase grating," *Opt. Express, OE*, vol. 16, no. 5, pp. 2995–3001, Mar. 2008.

[81] C. J. Lin, C. H. Lin, Y. T. Li, R. P. Pan, and C. L. Pan, "Electrically Controlled Liquid Crystal Phase Grating for Terahertz Waves," *IEEE Photonics Technology Letters*, vol. 21, no. 11, pp. 730–732, Jun. 2009.

[82] J. Chen, P. J. Bos, H. Vithana, and D. L. Johnson, "An electro-optically controlled liquid crystal diffraction grating," *Appl. Phys. Lett.*, vol. 67, no. 18, pp. 2588–2590, Oct. 1995.

[83] W. L. Chan, H.-T. Chen, A. J. Taylor, I. Brener, M. J. Cich, and D. M. Mittleman, "A Spatial Light Modulator for Terahertz Beams," *Applied Physics Letters*, vol. 94, no. 21, p. 213511, May 2009.

[84] Y. Monnai, K. Altmann, C. Jansen, H. Hillmer, M. Koch, and H. Shinoda, "Terahertz beam steering and variable focusing using programmable diffraction gratings," *Opt. Express, OE*, vol. 21, no. 2, pp. 2347–2354, Jan. 2013.

[85] R. P. Feynman and A. Zee, *QED: The Strange Theory of Light and Matter*. Princeton, NJ: Princeton University Press, 1988.

[86] D. Halliday, R. Resnick, and J. Walker, *Fundamentals of Physics Extended*, 6th ed. New York, NY: John Wiley & Sons, 2003.

[87] J. S. Parker and J. H. Rife, "Relative Position Estimates from Terahertz Observables," in *Proceedings of the 2017 International Technical Meeting of The Institute of Navigation*, Monterey, CA, 2017, pp. 1067–1082.

[88] J. Shi and Y. Gao, "A comparison of three PPP integer ambiguity resolution methods," *GPS Solut*, vol. 18, no. 4, pp. 519–528, Oct. 2014.

[89] S. C. Chapra, *Applied Numerical Methods*. 2004.

[90] R. F. Stengel, *Optimal Control and Estimation*. Courier Dover Publications, 2012.

[91] J. S. Parker, "Signal and Data Structure for Navigation with a Terahertz Interferometer," in *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017)*, Portland, OR, 2017.

[92] R. Gold, "Optimal binary sequences for spread spectrum multiplexing (Corresp.)," *IEEE Transactions on Information Theory*, vol. 13, no. 4, pp. 619–621, Oct. 1967.

[93] M. Schwartz, *Information Transmission, Modulation, and Noise : A Unified Approach to Communication Systems*, 3rd ed. New York, NY: McGraw-Hill, 1980.

[94] W. W. Peterson and D. T. Brown, "Cyclic Codes for Error Detection," *Proceedings of the IRE*, vol. 49, no. 1, pp. 228–235, Jan. 1961.

[95] J. Rife, "Overbounding False-Alarm Probability for a Chi-Square Monitor with Natural Bias," in *Proc. ION Global navigation Satellite Systems (ION GNSS+ 2015)*, Tampa, FL, 2015.

[96] J. H. Rife, "Comparing performance bounds for chi-square monitors with parameter uncertainty," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 3, pp. 2379–2389, Jul. 2015.

[97] J. Park, S. Mackay, and E. Wright, *Practical Data Communications for Instrumentation and Control - Knovel*. Burlington, MA: Elsevier, 2003.

[98] Microsemi Corporation, "Microsemi | Semiconductor & System Solutions | Power Matters," *Microsemi*. [Online]. Available: https://www.microsemi.com/. [Accessed: 22-Jun-2017].

[99] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. John Wiley & Sons, 2006.