UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

AN APPROACH TO SIMULTANEOUS WIRELESS SYNCHRONIZATION AND NAVIGATION FOR MOBILE DISTRIBUTED NETWORKS OF RADAR SYSTEMS

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

Degree of

DOCTOR OF PHILOSOPHY

By

RUSSELL KENNEY

Norman, Oklahoma

2024

AN APPROACH TO SIMULTANEOUS WIRELESS SYNCHRONIZATION AND NAVIGATION FOR MOBILE DISTRIBUTED NETWORKS OF RADAR SYSTEMS

A DISSERTATION APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Jay McDaniel, Chair

Dr. Hjalti Sigmarsson

Dr. Justin Metcalf

Dr. Nathan Goodman

Dr. Amy McGovern

This dissertation is dedicated to my wife and the love of my life, Sarah, and to my Lord and Savior, Jesus Christ.

Acknowledgment

I want to begin by expressing my deepest gratitude for my doctoral advisor and close friend, Dr. Jay McDaniel. It is quite difficult to articulate how much of an impact Dr. McDaniel has had on my life, both professionally and personally. Indeed, I would likely never have considered pursuing my Ph.D. without his suggestion and encouragement. Dr. McDaniel has been instrumental in boosting my confidence in myself and my work, instilling passion for our discipline, encouraging me when research is not going as planned, and overall helping me to have a very positive graduate school experience. Dr. McDaniel is enthusiastic about our field and our work, and his passion is infectious. I cannot count the number of times we were on the phone or messaging each other to discuss new ideas and research directions or working on proposals after midnight, and I look back on all those times (mostly) with fondness. Although I wish him and his wife the best of luck as they move back to Kansas to start their family, I do so with sadness as they have been an irreplaceable part of my life that I will miss greatly.

I am similarly indebted to my committee member and close friend, Dr. Hjalti Sigmarsson. Dr. Sigmarsson hired me as a junior in computer engineering to work on the digital back-end for a radar system which formed the basis for my Master's thesis and provided me so many valuable educational opportunities in the field of radar. Dr. Sigmarsson is possibly one of the most available people I know, probably to a fault – I have spent many hours in his office talking about research and getting advice (about work and about car troubles, among other things), and possibly even

more hours talking about anything not related to work. He is a dear friend of mine and I am grateful for everything he has done for me in these past several years.

Another of my committee members and friends, Dr. Nathan Goodman, deserves all my gratitude. Virtually everything I know about signal processing, SAR imaging, and estimation theory, I have learned directly from Dr. Goodman through the four classes I have taken with him since the spring of 2017. He has taught me so much about our discipline and about academic rigor in engineering science; as I was writing out the various analyses and proofs in this dissertation, I often found myself thinking, "will this be convincing enough for Dr. Goodman?" Dr. Goodman has also provided some simulated radar data which I used during my general exam as well as to construct the images in Fig. 1.1. I am also very thankful for Dr. Goodman's support as I taught my first class in the fall semester of 2022. He provided all his notes and assignments for me to build on, and he gave me guidance, advice, and support as I learned to be an effective teacher. I hope I will get to continue to collaborate with him on many projects in the future.

I also want to thank my good friend and committee member, Dr. Justin Metcalf. Dr. Metcalf has greatly enriched my graduate studies by providing feedback in line with his large body of experience both at OU and in his research pursuits before becoming a faculty member. His radar signal processing course introduced me to the fundamentals of radar, and everything I know about signal processing that I didn't learn from Dr. Goodman, I learned from Dr. Metcalf. He has been available to me to bounce research ideas off of as well as provide personal and professional advice, and I am a much better researcher because of his contributions and mentorship. I am excited to have opportunities to continue to collaborate with him in the future.

My graduate college representative, Dr. Amy McGovern, also deserves my thanks. I am grateful for her making the time to serve on my committee, and I have enjoyed getting to know her through the few doctoral committee meetings we've had over the past few years.

I want to thank my parents, Dwaine and Diane Kenney, and my brother, Dwaine, along with the rest of my immediate family. They have all been great examples to me of what it means to work hard and be committed while still prioritizing family. I am so grateful for the sacrifices they have made for me throughout the years. I am also very thankful for my in-laws, Dan and Kelly Crooks, for all the support they have provided to Sarah and me over the past several years.

I have had the great privilege of working with some of the most intelligent and kind people you could ever meet at the ARRC. I am so grateful for my friends Rylee Mattingly, Jon Knowles, Rachel Jarvis, Cal Schone, Shane Flandermeyer, Cora DeFrancesco, Grant Karber, Rosalind Agasti, Clayton Blosser, Brian Sun, Andrew Gonzales, Eric Rackelin, Benton Smith, Alex Pham, Gokhan Ariturk, Gus Azevedo, Randy Summers, Jake Fusselman, Josh Short, Jorge Alva, Nim Ccoillo Ramos, and Kurt Konyalioglu, along with all my other colleagues at the ARRC. You all have made graduate school enjoyable and have made me look forward to coming to work every day. Thank you. I am also so thankful for the innumerable friends outside of work who have supported me over the past several years of school. A particular thanks goes to Luke Barnes, Taylor Humphrey, Jimmy Glock, Josh Stewart, Andrew Hawcroft, and Nathan Lydick. You have been fast friends to me and I am so grateful for you.

I want to thank my church family at Hope Community Church for all their love and support these past four years. You welcomed Sarah and me, a newlywed couple still learning to be married while also learning how to do school well in the middle of a global pandemic, and made us part of your family. I am so thankful for your support and all the growth you have inspired for me personally, in my marriage, and in my relationship with God. You all have my sincerest love and gratitude.

I would also like to thank Cody Piersall for his time and effort in developing the MicroZed radar platform and adapting it to my needs. Without him, the hardware demonstration in this dissertation would not have been possible.

A large number of institutions outside of OU and the ARRC have been essential in

my doctoral pursuit, from a mentorship perspective as well as from a funding perspective. At different points throughout my Ph.D., my work has been funded by Sandia National Laboratories, the Kansas City National Security Campus, and Systems Planning and Analysis, Inc. In particular, the work in this dissertation has been primarily funded by the Office of Naval Research. Additionally, I have been supported directly in my doctoral studies by the National Defense Science and Engineering Graduate (NDSEG) Fellowship. This program has been a huge blessing to me in helping me to pursue my doctoral degree without worrying about financing it.

Finally, I am forever grateful for my wife, Sarah Kenney. Sarah and I got married the summer I started my Ph.D., and she has stood by my side through every difficult moment. I have found that failure is so much more bearable, and success is so much sweeter, when I experience them with my wife. She has encouraged me, celebrated me, and propped me up in so many ways that I never could have imagined when we first said our vows. I do not think that I would be where I am today without her, and I most certainly would not be as joyful. She has made me so proud in her own vigorous pursuit of a medical degree, and I am inspired by her hard work and tenacity. This achievement belongs to her as much as it does to me.

IEEE Copyright Notice

Table of Contents

List of Tables

List of Figures

Abstract

Recent developments in all-digital phased arrays have probed the upper bound of performance for single monostatic radar system performance. In order for radar system performance to continue to improve beyond the current state-of-the-art, it is imperative that research into the implementation of distributed radar systems be performed as such systems will enable significant performance enhancements in comparison to traditional monostatic radars. For these systems to be implemented, particularly in mobile scenarios, both the accurate navigation and synchronization of the systems must be performed. These processes must be performed at the carrier wavelength accuracy which poses a strict requirement on the performance of the navigation and synchronization algorithms. These components have received significant attention in the literature. However, although they are closely related problems for implementing mobile distributed radar networks, the potential for implementing algorithms for simultaneous navigation and synchronization has largely been unexplored. Therefore, the research proposed in this dissertation aims to implement algorithms for simultaneous navigation and synchronization for distributed radar networks by leveraging time-of-flight (TOF) ranging signals and associated Doppler measurements.

This dissertation provides a comprehensive literature review of current techniques for achieving navigation and synchronization solutions. A background is provided, describing linear and nonlinear Kalman filtering for time-series state estimation, relevant

propagation effects on radio signals, a mathematical framework for inertial navigation, and the radio frequency (RF) synchronization signal model. Current research results in cooperative navigation for radar motion compensation are presented. A novel algorithm for time, phase, and frequency synchronization is described which is capable of achieving synchronization exclusively in software, enabling decentralized implementation in existing radar systems. This algorithm is provided for a single iteration of synchronization and is also adapted for the Kalman filtering of the synchronization states. A framework for combining the cooperative navigation and synchronization routines is provided along with simulated results and analysis on performance. A preliminary hardware demonstration of the synchronization algorithm is also given. Finally, a summary of future research direction and goals is provided along with a conclusion to the dissertation.

Chapter 1

Introduction

Distributed radar systems are networks of radar sensors with significant spatial separation between the individual sensors. The large separation between the sensors in these systems prohibits a physical connection between the nodes in the network but provides multiple benefits in comparison to traditional monostatic radar sensors. Because each individual node is capable of transmitting unique signals and receiving radar returns induced by the transmitted signals of other nodes, they are often referred to as distributed multiple-input multiple-output (MIMO) radar systems. Distributed MIMO systems are superior in terms of system survivability due to the lack of a single point of failure, similar to the concept of graceful degradation in phased array systems. Moreover, properly implemented distributed radar systems can achieve similar coverage areas to traditional monostatic systems but with lower transmit powers, reducing the probability of detection [6]. Additionally, implementation costs can be reduced since a network of many low-cost radar systems can match or exceed the performance of a single high-performance radar system [7].

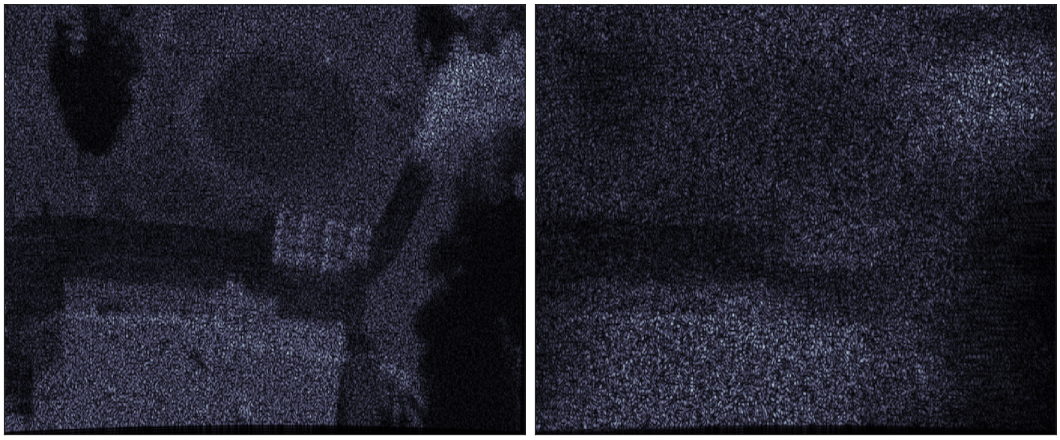
The performance enhancements provided by distributed radar networks are numerous. Distributed radar networks can localize targets with higher accuracy than monostatic systems [8–11], produce target velocity estimates above the monostatic

blind velocity caused by Doppler aliasing [12], provide three-dimensional velocity field extraction [13], improve the power gain and directivity of beamforming missions [14–16], and increase detection performance in cluttered environments [17–19]. Distributed MIMO radar systems also facilitate a dramatic improvement to the performance of synthetic aperture radar (SAR) imaging missions by providing reflectivity measurements from multiple angles [20], increasing synthetic aperture size [21], enabling faster update rates for single-pass interferometric SAR [22], and enabling single-pass holographic SAR for three-dimensional imaging [23].

1.1 Motivation

For all the capabilities promised by distributed MIMO radar systems, implementing them in practical systems poses an extraordinary challenge. For such systems to be useful, two key processes must be implemented with high accuracy: *localization* or *navigation*, and *synchronization*. Localization and navigation refer to the determination of the radar systems' position in space relative to one another or in an absolute sense relative to the Earth. In this context, localization refers to this position identification in networks of stationary nodes while navigation refers to the tracking of time-evolving positions of moving platforms. Synchronization refers to the alignment of the radio frequency (RF) electrical states of radar sensors; namely, their clock times, carrier phases, and carrier frequencies.

In order for radar systems to provide useful information, their positions must be accurately established. This is especially important for moving systems since radar data must be accurately motion-compensated for the returns collected over a single coherent processing interval (CPI) to be coherently integrated. Fig. 1.1 shows a single-platform SAR image formed by a platform with perfectly compensated motion and



(a) (b)

Fig. 1.1: Original SAR image (a) and a SAR image defocused due to navigation errors (b).

compares it to an image with improperly compensated motion, which leads to azimuth spreading phase errors and blurs the image beyond recognition. The coherent integration of data captured across multiple platforms poses an even stricter requirement on motion compensation as all platforms must each be localized; otherwise, phase errors from one platform may corrupt the coherency of the data captured by other platforms. Moreover, the navigation of such sensors must be accurate on the order of fractions of a single carrier wavelength. Therefore, the navigation requirement reduces to centimeters or millimeters as the frequencies of modern radar systems scale to Ku-band [24], Ka-band [25], and even W-band as researchers explore applications of commercial automotive radar systems to SAR imaging [26–28]. Thus, it is crucial that accurate navigation techniques continue to be developed to enable applications of these high-frequency systems.

Moreover, any form of distributed radar system requires precise synchronization of the constituent nodes in time, carrier phase, and carrier frequency [21, 29–32]. Radar systems require strict timing precision to accurately extract range values [24, 33, 34];

even one nanosecond of time error between a transmitter and receiver will result in 15 cm of range error, rendering high-resolution applications infeasible. To eliminate range errors across distributed observations of a target, the time offsets between radar systems in the network must be compensated to below nanosecond accuracy. Furthermore, phase errors between platforms will render coherent operation impossible; instead of coherent integration of data across platforms, data will destructively interfere from platform to platform and ultimately destroy any cooperative information that could otherwise be obtained. Finally, frequency errors between platforms will result in the incomplete removal of the carrier frequency from received signals generated by other platforms, which will modulate the received signal and lead to errors.

To overcome these challenges, many systems make use of the Global Positioning System (GPS), which provides position information for navigation and distributes a frequency reference for frequency synchronization of platforms through the pulse-per-second (PPS) service. For navigation purposes, inertial measurement units (IMUs) can be fused with GPS and other sensor information to provide high-accuracy estimates of position. However, GPS is an imperfect system – its timing distribution has a standard deviation of 15 ns, which is not sufficient to achieve the required time and phase synchronization performance between platforms and will also lead to positioning errors on the order of several meters. Some augmented GPS systems such as differential GPS (DGPS) or real-time kinematic (RTK) GPS can reduce positioning errors to centimeter-level accuracy by compensating for atmospheric propagation effects, but these require additional hardware or infrastructure to operate and they do not assist in fully synchronizing phase across platforms. Furthermore, some operating environments, including indoors [35], outer-space [36], and harsh terrestrial environments [37] block GPS signals such that they cannot be used at all. In certain

applications, GPS may be blocked, jammed, or spoofed by hostile agents, rendering it useless. In these cases, navigation must be performed exclusively using IMUs, which tend to diverge quickly due to the integration step required to obtain position information from the measured acceleration and angular velocity. This navigation with IMUs alone is often referred to as dead-reckoning. Moreover, GPS-denial completely eliminates the external synchronization reference to the distributed system. Therefore, to enable distributed systems that are capable of robust operation in these types of environments, new techniques must be developed to achieve accurate navigation and synchronization.

1.2 Prior Work

Much work has been done in the literature to address the navigation and synchronization problems, though they are usually considered separate problems, and oftentimes the impact on radar is not considered. In the following subsections, a summary of prior work addressing the problems of navigation and synchronization is provided.

1.2.1 Navigation

The problem of accurately determining one's location is ubiquitous across almost all civil, industrial, and military applications [38–43]. In all navigation algorithms, platforms attempt to use information from various sensors to compute their position in three-dimensional space. Oftentimes this position is given relative to some earth-centered frame of reference, but depending on the application a more localized navigation solution can be used as well. Common navigation sensors include cameras [44], light detection and ranging (LiDAR) systems [45], magnetometers [46], among others.

However, the most common sensors for navigation are satellite navigation systems such as GPS, which provide absolute position references, and IMU sensors. IMUs generate measurements of acceleration and angular velocity, which can be integrated to give estimates of velocity, position, and orientation. Usually, information from the GPS and IMU sensors is fused together to form an inertial navigation system (INS) [47], which is often done using some information fusion algorithm like the Kalman filter. The INS continues to find consistent and ubiquitous use in outdoor navigation applications [48–50]. The fusion of these two sensors continues to find so much use because of the way that they complement each other. GPS provides a stable position reference over the long term, but each individual measurement tends to be less accurate in the short term, and measurements are typically obtained at a slower rate of 1–10 Hz. On the other hand, IMU measurements are produced at a fast rate (100–1000 Hz) and tend to produce accurate position estimates in the short term. However, because the IMU measurements must be integrated to achieve position and orientation, errors due to noise in the sensors will accumulate over time and the navigation solution will diverge [51]. By fusing the sensors together, their advantages may be combined and their disadvantages may be compensated for.

1.2.1.1 Kalman Filtering

Generally, navigation algorithms utilize an information filter to fuse a known motion model with the navigating platform's pose measurements. The Kalman filter, first proposed in [52], is a time-series estimation technique that has found application in a stunning variety of fields including economics [53], target tracking [54–56], and computer vision [57], and it has been a staple of navigation algorithms for decades [58–63]. The Kalman filter works iteratively, performing two stages on each iteration: predict-

ing the current system state (e.g., position, velocity, and orientation) using the previous system state information and correcting the prediction using measurements such as a GPS measurement. The filter tracks an estimate of the system state and an associated covariance under the modeled assumption that both the prediction model and measurements are noisy.

The standard Kalman filter requires that the state and measurement models be linear. However, because the general navigation equations for a moving platform are nonlinear, variations of the Kalman filter exist, such as the extended Kalman filter (EKF) [64], to enable nonlinear models to be included in the Kalman filtering framework by approximating the nonlinear functions' local linear behavior. The EKF works by computing the first-order derivative of the nonlinear system state function in the form of the Jacobian matrix and uses the resulting linear approximation in the same way as the ordinary linear Kalman filter. The EKF works well in some circumstances but fails when the model is strongly nonlinear and cannot be approximated as linear. Moreover, it requires the complicated derivation of the Jacobian of the state model function.

More recently, a Kalman filtering technique called the unscented Kalman filter (UKF) has been introduced [65–67]. Rather than approximating the system model as linear, the UKF approximates the prior probability density function (PDF) at each iteration by a small set of sample points called sigma points. These points are propagated through the system and measurement models are used to form an output state and covariance estimate. The UKF generally achieves much better performance than the EKF while still meeting similar computation complexity, and it is far more computationally feasible in comparison to Monte Carlo-based nonlinear filters such as the particle filter (PF) [68]. Because of the robustness of the UKF, it has found ap-

plication in a wide variety of nonlinear estimation applications [69–71], including navigation [72–74].

1.2.1.2 Navigation for Radar Applications

Navigation systems with mobile radar applications generally operate like any other navigation system, and will also typically make use of an INS as the core sensing mechanism. However, as described above, one main challenge in navigation for radar is that the accuracy of the navigation solution needs to be on the order of the carrier wavelength. In particular, the navigation errors in the line-of-sight (LOS) direction should be $1/16$ of a wavelength for low-frequency errors (errors that vary by less than a wavelength over the CPI) and less than $15/1000$ of a wavelength for higher frequency sinusoidal errors [75]. Any additional navigation errors would need to be compensated for by a phase-error extraction technique such as autofocus [76].

One unique challenge to navigation for mobile radar systems is that the navigation solution must produce position estimates at the same rate as the radar pulse repetition frequency (PRF), which is often on the order of kHz or tens of kHz and therefore much faster than the update rate of an INS. To solve this problem, recent iterative up-sampled versions of the Kalman filter [62, 77] and particle filter [78, 79] have been proposed. These modifications of the traditional filtering algorithms tend to achieve higher accuracy while also producing a navigation solution at the same rate as the PRF such that each individual radar pulse can be motion-compensated.

A benefit of performing SAR with a mobile radar system is that the radar data itself can be used for the purposes of motion compensation and therefore navigation. For example, techniques have been proposed to enable motion compensation directly from raw radar data for unmanned aerial vehicle (UAV) applications [80], and one type of

autofocus algorithm called prominent point processing enables the direct extraction of the full radar flight path during the CPI assuming that a sufficient number of strong scatterers are present in the SAR scene [75, 81, 82].

Generally, traditional navigation techniques with an INS work well for most radar applications at moderately low frequencies. However, very little exists in the literature for high-fidelity radar applications in GPS-denied circumstances in which the navigation solution must be formed by dead-reckoning. While prominent point processing could in theory be used for GPS-denied navigation, it requires that strong scatterers be available for motion compensation, which is not a guarantee. Moreover, it assumes that the radar system on board the moving platform is used for SAR which may not be the case. Therefore, there is a great need for the development of navigation techniques that can provide high-accuracy positioning for radar applications in GPS-denied scenarios.

1.2.1.3 Cooperative Navigation and Localization

One approach for mitigating the lack of GPS access in swarm navigation is the cooperative navigation approach. This approach leverages information shared among sensors along with inter-node measurements to achieve higher levels of accuracy than the platforms could obtain independently [83–85]. One common inter-node measurement in a cooperative network is a time-of-arrival (TOA) or time-of-flight (TOF) based range measurement [86]. Such measurements have been shown to be useful in localization of stationary wireless sensor networks [87, 88] as well as in the navigation of mobile networks of sensors [89–91]. However, the application of these navigation algorithms has yet to be demonstrated in a radar application, and the evaluation of the impact of these cooperative localization and navigation techniques on radar system

performance is still an open area of research.

The notion of cooperative navigation is similar to the operating principle of GPS. Consider Fig. 1.2, which shows a diagram illustrating the operating principle of GPS. When a GPS receiver computes its position, it receives ranging signals from GPS satellites orbiting the Earth. These satellites are equipped with highly precise timing hardware and their positions are known with high precision at all times due to their well-defined orbits. The receiver can compute the range values between itself and all the satellites whose ranging signals it can detect, and then triangulate its own position by deducing the best possible location at which the computed ranging values would be logical. Cooperative navigation of platforms uses a similar principle leveraging range values between platforms. However, in this case, the locations of all the platforms are not known but must be estimated from the range values in conjunction with other sensors, such as IMUs. In theory, this procedure can enable highly accurate navigation even when GPS is not available.

1.2.2 Wireless Synchronization

Because synchronization of RF electrical states is such a critical component of distributed RF networks in radar and communications applications, it has received significant attention in the literature. Many of the synchronization approaches detailed in the literature are proscribed for specific applications (e.g., radar, communications, distributed phased arrays, etc.), but for the broad majority the techniques can be adapted and applied in any distributed RF application.

For the synchronization of radar systems to be complete, precise compensation of offsets in time, carrier phase, and carrier frequency must be performed. Often, the approaches described in the literature are capable of achieving one or two of these but

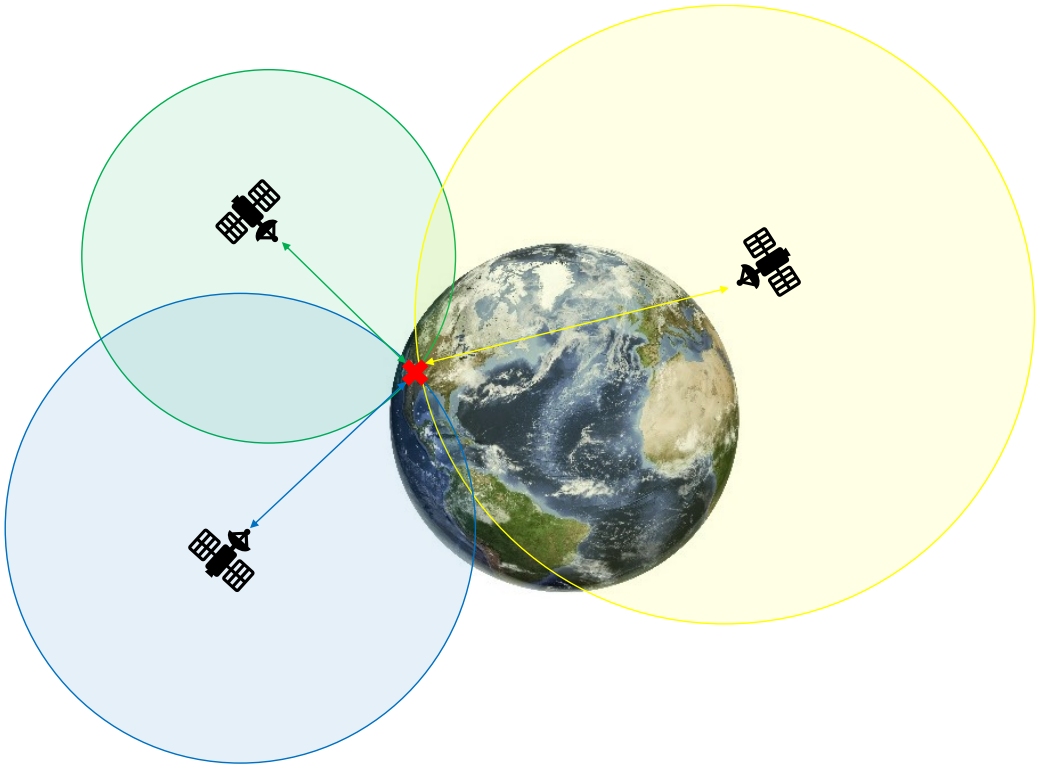


Fig. 1.2: An example of the operation of GPS through the extraction of ranging signals between GPS satellites and the receiver.

do not address the third, or they make the assumption that the third can be achieved a priori by an external source. For instance, the technique in [92] achieves time and phase synchronization but assumes that frequency synchronization is achieved by an external reference.

For radar applications, synchronization techniques can generally be broken up into two broad categories: hardware and software synchronization. In hardware synchronization techniques, RF synchronization is achieved through the inclusion of specific types of hardware in the RF system. In software synchronization techniques, estimates of the RF electrical states are produced in processing and compensated digitally.

One very common hardware-based synchronization technique that has been recently developed is the self-mixing circuit [93]. This circuit assumes that a primary

platform transmits a modulated two-tone waveform in which the spacing between the two frequencies in the signal corresponds to the desired reference clock frequency. When the secondary platform receives the two-tone waveform, it splits the waveform into two copies, with one copy being amplified and used as an input to the local oscillator (LO) port of a mixer, with the other acting as the RF input. The output of the mixing circuit therefore contains the sum and difference of the two tones, with a low-pass filter (LPF) removing the sum and leaving only the reference signal difference. This circuit has been proven to enable synchronization of frequency [94], phase [95], and time [96]. Moreover, the two-tone waveform has been demonstrated to enable high-accuracy ranging between platforms [97] which is important for the compensation of the propagation phase during the signal transmission from the primary to the secondary node. However, because the technique requires the introduction of additional hardware, it cannot be implemented on existing systems without possibly infeasible system modifications. Moreover, the technique is inherently centralized as it requires a single primary node to distribute the two-tone reference signal to the secondary nodes. Finally, the technique is dependent on the transmission of a specific signal and is therefore not flexible for the use of waveform design and optimization [98].

Software-based synchronization techniques utilize the exchange of signals from which the RF synchronization variables are extracted and compensated for in post-processing. A technique for carrier frequency synchronization is proposed in [99] which is tolerant to Doppler frequencies due to relative motion between platforms. An approach to phase synchronization for bistatic radar is developed in [100], and a summary of existing approaches to phase synchronization among MIMO radar systems is provided in [101]. A procedure is proposed in [29] for synchronizing time and phase among platforms, but it is dependent upon accurate navigation from an INS.

A technique for estimating time and phase and compensating for them in processing is proposed in [92]. This technique leverages the notion of two-way time transfer (TWTT) to exchange time offset information as well as computing the TOF range values between each pair of platforms. This technique is highly robust as it does not require high bandwidths or sophisticated RF hardware. Moreover, it is a decentralized technique with the amount of time required to complete the synchronization scaling linearly with the number of platforms in the network. However, it also assumes that frequency synchronization has been achieved by some external system such as the GPS PPS.

1.3 Research Objective

As outlined above, the operation of mobile networks of radar systems is highly dependent on accurate estimates of platform position as well as precise synchronization of the platforms. Generally, in MIMO radar applications, these two crucial components are not addressed simultaneously. Furthermore, when procedures are established for positioning, they often assume that synchronization is achieved *a priori* through some external reference such as GPS PPS; similarly, when procedures are established for synchronization, they often assume that positioning is achieved *a priori* such as in [23] and [29]. This approach potentially misses out on significant gains in information since the two procedures are ultimately related in a mobile distributed network. As established in Section 1.2.1.3, navigation performance is improved substantially by the collection of inter-node range values between platforms. Furthermore, as described in Section 1.2.2, multiple synchronization techniques require the creation of range values for the purpose of propagation phase compensation. Therefore, there is shared information between the two operations which has as of yet not been exploited

jointly for radar applications.

To address this gap in the state-of-the-art, the objective of this research is to produce techniques for simultaneously estimating navigation and synchronization system states leveraging shared information. This shared information includes range estimates produced during the synchronization procedure as described in [97] and [92] as well as Doppler shift measurements which can be used to produce relative velocity estimates. A graphical description of this proposed technique is shown in Fig. 1.3. Additionally, the synthesis of these techniques will be addressed from a GPS-denied standpoint to provide some assurance that the proposed technique will be robust to scenarios in which absolute references for position and frequency are not available. As a result, the developed synchronization techniques must be capable of producing time, phase, and frequency estimates simultaneously. Finally, the proposed synchronization technique should be software-based such that it can be implemented on existing software-defined radar systems.

This research begins in a similar manner to other approaches by addressing the cooperative navigation and distributed synchronization problems separately. Once suitable algorithms have been developed for performing the two procedures independently, work will be done to combine the two operations into a single algorithm. The initial goal will be to demonstrate a performance enhancement to radar systems in simulated data with a future goal being to implement a simple proof-of-concept system in real hardware to demonstrate the efficacy of the proposed techniques in real radar systems.

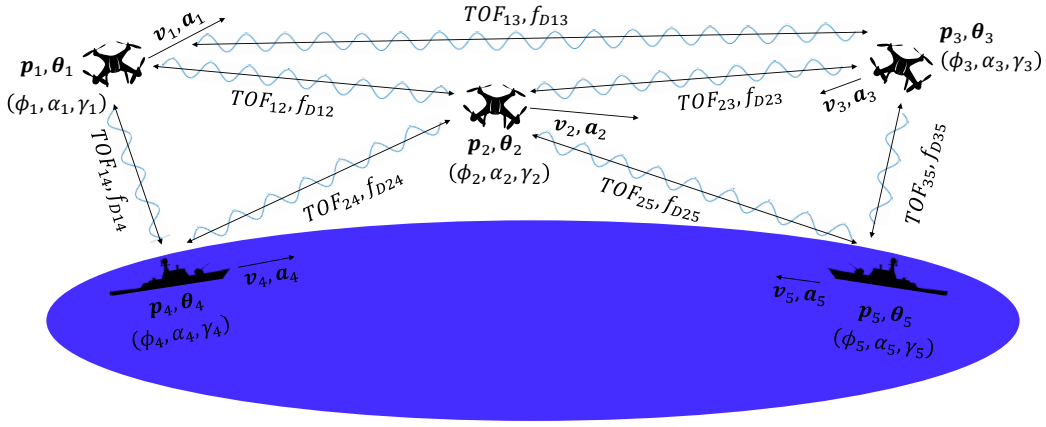


Fig. 1.3: A graphical representation of simultaneous navigation and synchronization for a mobile network of heterogeneous platforms with embedded radar sensors.

1.4 Outline

This dissertation is devoted to the development of simultaneous techniques for navigation and synchronization for networks of radar systems, with a heavy emphasis on developing digital techniques for synchronization suitable for implementation on all-digital systems that can easily be integrated with existing navigation algorithms. It is organized as follows. Following the introduction, Chapter 2 provides a mathematical background for important concepts related to this work. This chapter will introduce the framework for the Kalman filter and its nonlinear derivative, the UKF. It will also introduce important concepts in wave propagation which will have a strong impact on synchronization performance. Furthermore, nonlinear kinematics and navigation of mobile platforms will be described along with an overview of navigation error impact on radar sensing. Finally, a signal model describing the impact of time, phase, and frequency errors on inter-platform signal transmissions will also be provided.

In Chapter 3, a description of the current research in cooperative navigation for radar systems is provided. This includes a discussion of how the UKF is leveraged to

fuse IMU sensor measurements with TOF range values between platforms and GPS signals if they are available. Performance results are given to show the efficacy of the algorithm in scenarios with varying levels of GPS access and IMU quality. The chapter concludes with preliminary simulations of SAR imaging scenarios relying on cooperative navigation for motion compensation.

Chapter 4 provides a novel decentralized synchronization algorithm for estimating and compensating time, phase, and frequency in distributed radar networks. The algorithm is based on the work in [92] which is extended to enable frequency synchronization in GPS-denied environments. Building on the signal model in Chapter 2, this chapter outlines several similar techniques for synchronization which have application in various scenarios. Simulated results of these algorithms' performance are then shown. Additionally, a theoretical analysis of the algorithm performance, including a comparison to the Cramér-Rao lower bound, is provided. The chapter concludes with radar simulation results of distributed transmit beamforming and MIMO SAR using the proposed synchronization technique.

Chapter 5 extends the synchronization procedure proposed in Chapter 4 to track the synchronization states using the UKF. Two different approaches to implementing the UKF measurement model are derived and compared in simulation. It is shown that one of the approaches in particular has the capability of accurately tracking synchronization states over time and reducing estimation error in all manner of scenarios, including with relative motion and Doppler shifts, low update rates, and low SNR.

Chapter 6 derives the minor modifications required to integrate the UKF synchronization approach in Chapter 5 with the cooperative navigation approach described in Chapter 3. Results are provided demonstrating the efficacy of the simultaneous navigation and synchronization approach for efficiently producing estimates of platform

positions and clock parameters simultaneously.

Chapter 7 provides a hardware setup description for a small network of two software-defined radar units synchronized using the proposed synchronization technique described in Chapter 4. This hardware setup is demonstrated and it is shown that the carrier frequency offset between the two platforms may be entirely compensated in software with digital corrections.

Chapter 8 concludes the dissertation with a summary of the work done. It provides a list of significant contributions made over the course of this dissertation research. Finally, a list is provided of future research questions and tasks.

Chapter 2

Background

This chapter aims to provide background information and a mathematical framework for understanding the important components outlined in this dissertation. First, the mathematics of the Kalman filter are described with specific attention paid to the traditional linear Kalman filter and the nonlinear unscented Kalman filter. Next, a description of various relevant effects on wireless signals due to propagation effects is provided. After this, an outline of inertial navigation equations is provided for a general navigating platform, along with a signal model for understanding the performance of mobile radars (specifically SAR) as a function of navigation errors. Finally, a signal model describing the signals observed at one receiver transmitted from another radar before synchronization is provided to give an understanding of how unsynchronized systems impact RF performance.

2.1 Kalman Filtering

The goal of the Kalman filter is to accurately predict an unknown random vector (i.e., the *system state*) using a dynamic model of how the system evolves with time and measurements of the system state (or variables related to the system state). For example, in a navigation system where the estimated system state is the platform position

and velocity, the kinematic equations of motion may describe the dynamic model, and the measurements may be geodetic coordinates extracted from a GPS receiver. The dynamic model and the measurements are assumed to be corrupted with random noise, called *process noise* and *observation noise*, respectively. The process and observation noise introduce difficulty in accurately estimating the system state. The Kalman filtering algorithm provides a framework for recursively estimating the system state from the noisy dynamic model prediction and measurements by producing a weighted combination of the two with the weight, or *Kalman gain*, calculated based on the computed accuracy of the prediction and measurement. The Kalman filter also estimates the system state's error covariance so that the accuracy of the recorded system state may be tracked as the algorithm is iterated. When the system and measurement models are linear, and the process and measurement noises are white and Gaussian, the Kalman filter is the optimal system state estimator from a minimum mean squared error (MMSE) sense assuming that the covariances of the process and observation noise are known.

The algorithm works in two stages, typically referred to as *prediction* and *update*. The prediction stage uses the dynamic model with the system state from the previous iteration as an input to predict the system state for the current iteration. The dynamic model also transforms the covariance of the previous estimate to produce the prediction estimate covariance. In the update stage, the predicted state estimate and its covariance are corrected with the available measurements by generating a posterior state estimate and covariance. The algorithm can then be iterated indefinitely.

2.1.1 Linear Kalman Filtering

The simplest implementation of the Kalman filter assumes that the dynamic system model and measurement model are linear. Let $\mathbf{x}_k$ be an N_K-length vector representing

the true system state at iteration k of the Kalman filter. The system state between iterations is modeled by

$$\mathbf{x}_k = \mathbf{F}_k \mathbf{x}_{k-1} + \mathbf{w}_k, \tag{2.1}$$

where $\mathbf{F}_k$ is an $N_K \times N_K$ matrix called the *state transition matrix*. The matrix is often constant with k, but this is not required. This matrix maps the linear relationship between the state at iteration $k-1$ and the next iteration k. The vector $\mathbf{w}_k$ is called the process noise, which is modeled as zero-mean additive white Gaussian noise (AWGN) distributed with covariance $\mathbf{Q}_k$, which is called the process covariance. As with the state transition matrix, although $\mathbf{Q}_k$ is often constant with k, this is not required so long as the change in $\mathbf{Q}_k$ from iteration to iteration is known. The process noise can be thought of as errors or omissions in the dynamic model. For instance, higher-order derivatives of motion will introduce errors in a navigation Kalman filter whose dynamic model records only position and velocity. The magnitude of these unmodeled effects contributing to the state prediction errors is modeled in the selection of $\mathbf{Q}_k$.

Let $\mathbf{z}_k$ be an M_K-length vector representing the measurements of the system. The measurements need not directly record values of the system state, though in many cases, they will. The measurements are related to the system state by

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k, \tag{2.2}$$

where $\mathbf{H}_k$ is an $M_K \times N_K$ matrix called the *observation matrix*. This matrix maps the linear relationship between the state value on iteration k and the measurements associated with that state. The vector $\mathbf{v}_k$ is the observation noise, modeled as zero-mean AWGN distributed with observation covariance $\mathbf{R}_k$. The observation noise is typically manifested as noise on practical sensors; therefore, $\mathbf{R}_k$ can often be determined from a

datasheet or by observing the sensor measurement variance while the system is static. As with $\mathbf{F}_k$ and $\mathbf{Q}_k$, $\mathbf{H}_k$ and $\mathbf{R}_k$ are often constant with k, but they do not need to be; in fact, M_K can change between iterations depending on when measurements are available.

Because both the system state and measurements of the state are corrupted by noise, it is impossible to obtain a perfect, direct estimate of the state $\mathbf{x}_k$. The state vector must therefore be estimated imperfectly. The estimate of the system state from iteration k is denoted $\hat{\mathbf{x}}_k$. An estimate of the error covariance of this estimate is denoted $\hat{\mathbf{C}}_k$. The initial values $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{C}}_0$ must be selected based on the application.

The prediction step uses the state transition matrix to form *a priori* estimates of the state $\hat{\mathbf{x}}_k^-$ and covariance $\mathbf{C}_k^-$ by

$$\begin{aligned} \hat{\mathbf{x}}_k^- &= \mathbf{F}_k \hat{\mathbf{x}}_{k-1} \\ \mathbf{C}_k^- &= \mathbf{F}_k \hat{\mathbf{C}}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k, \end{aligned} \tag{2.3}$$

where the superscript ^{T} denotes the matrix transpose. Once the *a priori* estimates are formed, they are corrected by measurements in the update step. First, an *a priori* estimate of the measurement vector $\hat{\mathbf{z}}_k$ is formed by

$$\hat{\mathbf{z}}_k = \mathbf{H}_k \hat{\mathbf{x}}_k^-. \tag{2.4}$$

A new term $\tilde{\mathbf{z}}_k$, called the *innovation*, is formed by

$$\tilde{\mathbf{z}}_k = \mathbf{z}_k - \hat{\mathbf{z}}_k \tag{2.5}$$

and the covariance of the innovation ($\mathbf{S}_k$) is computed by

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{C}_k^- \mathbf{H}_k^T + \mathbf{R}_k. \tag{2.6}$$

The optimal Kalman gain ($\mathbf{K}_k$) is calculated by

$$\mathbf{K}_k = \mathbf{C}_{k,xz} \mathbf{S}_k^{-1}, \tag{2.7}$$

where $\mathbf{C}_{k,xz}$ is the cross-covariance of $\hat{\mathbf{x}}_k$ and $\hat{\mathbf{z}}_k$ computed by

$$\mathbf{C}_{k,xz} = \mathbf{C}_k^- \mathbf{H}_k^T. \tag{2.8}$$

Finally, the output state estimate and associated covariance are computed by

$$\begin{aligned} \hat{\mathbf{x}}_k &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k \tilde{\mathbf{z}}_k \\ \hat{\mathbf{C}}_k &= \mathbf{C}_k^- - \mathbf{K}_k \mathbf{S}_k \mathbf{K}_k^T. \end{aligned} \tag{2.9}$$

This procedure is iterated, where generally, the state is assumed to change with time and therefore must be estimated continuously.

2.1.2 Extended Kalman Filtering

The traditional formulation of the Kalman filter requires that the state and measurement models be linear (i.e., that (2.1) and (2.2) may be expressed in terms of matrices $\mathbf{F}_k$ and $\mathbf{H}_k$). However, most estimation problems in the real world introduce nonlinearities to the state transition or measurement function, rendering this approach impossible. A technique called the extended Kalman filter (EKF) can be used to enable Kalman filtering with nonlinear functions. The EKF operates very similarly to the

linear Kalman filter, where the matrices $\mathbf{F}_k$ and $\mathbf{H}_k$ are evaluated as first-order Jacobians of the true nonlinear state transition function, f_k, and the nonlinear measurement function, h_k. First, the linear state and measurement models in (2.1) and (2.2) must be rewritten to include arbitrary nonlinear functions $f_k(\mathbf{x})$ and $h_k(\mathbf{x})$ in place of $\mathbf{F}_k$ and $\mathbf{H}_k$ as

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}) + \mathbf{w}_k \tag{2.10}$$

and

$$\mathbf{z}_k = h_k(\mathbf{x}_k^-) + \mathbf{v}_k. \tag{2.11}$$

The *a priori* estimates of $\hat{\mathbf{x}}_k^-$ and $\hat{\mathbf{z}}_k$ are computed using f_k and h_k by

$$\hat{\mathbf{x}}_k^- = f(\hat{\mathbf{x}}_{k-1}) \tag{2.12}$$

and

$$\hat{\mathbf{z}}_k = h_k(\hat{\mathbf{x}}_k^-). \tag{2.13}$$

The matrices $\mathbf{F}_k$ and $\mathbf{H}_k$ must be approximated by

$$\begin{aligned} \mathbf{F}_k &= \left.\frac{\partial f_k}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k-1}} \\ \mathbf{H}_k &= \left.\frac{\partial h_k}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_k^-} \end{aligned} \tag{2.14}$$

and may be used in the original expressions containing $\mathbf{F}_k$ and $\mathbf{H}_k$. With the exception of these changes, the remaining filtering equations do not change.

The EKF has found substantial use in many practical recursive estimation problems. However, it suffers from some significant drawbacks. From a performance perspective, the EKF is not an optimal estimator in any sense. Moreover, since it uses

a linearization centered around the state estimate, it may suffer significant inaccuracy if the state estimate is not correct. The filter will also diverge if a function is strongly nonlinear around the state estimate (for instance, if a discontinuity is present). Moreover, the EKF is challenging to implement because of the complex derivation required to compute the Jacobians in (2.14). Another problem with both the linear Kalman filter and the EKF that is generally not pertinent in most applications is that the state and measurement models must be known analytically; that is, the actual mathematical form of $\mathbf{F}_k$ and $\mathbf{H}_k$ (or f_k and h_k) must be known, although this will not pose a restriction in the problems presented in this dissertation.

2.1.3 Unscented Kalman Filtering

The UKF operates with the same framework as the traditional linear Kalman filter, using prediction and measurement models in tandem to produce accurate state estimates. At the output of each iteration k, estimates of the current system state $\hat{\mathbf{x}}_k$ and associated covariance $\hat{\mathbf{C}}_k$ are produced. Because the UKF is a nonlinear filter, the linear formulations of the state and measurement models in (2.1) and (2.2) must be replaced by nonlinear formulations given in (2.10) and (2.11). Contrary to the analytic computation of the state and covariance in the linear Kalman filter, the UKF produces these estimates via the unscented transform, which seeks to approximate the transformed mean and covariance with a small set of discrete points called sigma points. A pictorial description of unscented transform is shown in Fig. 2.1.

In general, there are many ways to compute sigma points. A common method is

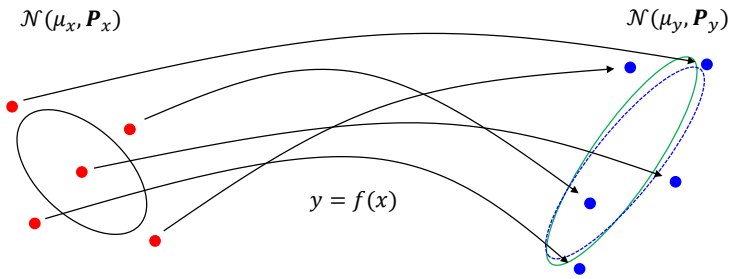


Fig. 2.1: A simple example of the unscented transform. The initial covariance ring and the input sigma points are shown on the left, while the transformed sigma points are shown on the right with an estimate of the output covariance ring in blue and the true transformed covariance ring shown in green. From [2] ©2024 IEEE.

to compute $2N_K + 1$ sigma points subject to

$$\begin{aligned}
\chi_{k,0} &= \hat{\mathbf{x}}_{k-1} \\
\chi_{k,i} &= \hat{\mathbf{x}}_{k-1} + \left(\sqrt{(N_K + \lambda_u)\hat{\mathbf{C}}_{k-1}}\right)_i \quad 1 \leq i \leq N_K \\
\chi_{k,i} &= \hat{\mathbf{x}}_{k-1} - \left(\sqrt{(N_K + \lambda_u)\hat{\mathbf{C}}_{k-1}}\right)_i \quad N_K + 1 \leq i \leq 2N_K
\end{aligned} \tag{2.15}$$

where $\chi_{k,i}$ is the i^{th} sigma point for the current iteration and λ_u is a tuning parameter computed by

$$\lambda_u = \alpha_u^2(N_K + \kappa_u) - N_K \tag{2.16}$$

where $\kappa_u \geq 0$ and is often set to $\kappa_u = 1$. The value of α_u is a scaling parameter in the range $0 < \alpha_u \leq 1$ and is often set to a very small value (e.g. $\alpha_u = 10^{-3}$ [102]). In general, the "radius" of the sigma points around the mean scales with the value of N_K. Thus, for large state vectors, the sigma points are spread out substantially from the mean, potentially straddling nonlinear portions of the modeling functions and therefore capturing non-local behavior, leading to inaccurate transformed statistics and inflating the impact of higher-order moments subject to the orientation of the sigma points. The scaled sigma points based on the proper selection of α_u enable accurate

approximation of the prior mean and covariance while ensuring that the sigma point transformation captures the local behavior of the state and measurement functions. This is particularly important for this application since the centralized tracking of the entire network state may lead to huge system states for even moderately large networks (10+ platforms).

Although the original, and most common, formulation of the UKF computes sigma points in the manner given in (2.15), more recent research has developed additional methods for computing a set of sigma points that will properly propagate the prior statistics through the nonlinear state and measurement functions. Notably, the cubature Kalman filter (CKF) has recently been proposed [103]. This filter operates very similarly to the UKF by approximating a PDF with a discrete set of weighted points. The primary distinction between the sigma-point set in the UKF and the cubature-point set in the CKF is that the cubature-point set does not include the estimated mean value, with all the weights placed on points around the mean. This approach has some advantages over the traditional UKF, primarily greater numerical stability; however, one downside is the assumption that the prior distribution of $\mathbf{x}_k$ is Gaussian. Because numerical stability has not been noted to degrade the performance of the filter, the standard sigma point set is sufficient for this application.

The square root operation in (2.15) refers to the matrix square root operation, and the subscript i refers to the i^{th} column (or row, depending on the form of the matrix square root) of the resulting matrix. The matrix square root is not unique, and multiple definitions of the matrix square root exist. For instance, a matrix $\mathbf{A}$ may be considered a square root of $\mathbf{B}$ if $\mathbf{AA} = \mathbf{B}$, $\mathbf{A}^T\mathbf{A} = \mathbf{B}$, or $\mathbf{AA}^T = \mathbf{B}$. One common technique for producing a matrix square root is the Cholesky decomposition [104], which produces an upper-triangular matrix square root $\mathbf{A}$ in the form $\mathbf{A}^T\mathbf{A} = \mathbf{B}$. The Cholesky

decomposition is popular because it is fast and numerically stable. While this works in the algorithm described here, simulations have demonstrated that a matrix square root routine that produces a matrix square root in the form $\mathbf{AA} = \mathbf{B}$, such as the Schur method described in [105, 106], performs better in this application. Because this technique is potentially less numerically stable, the resulting matrix square root may be complex with very small imaginary parts. These small imaginary parts may be discarded without adverse consequences.

When the sigma points are produced, they are first propagated through the state model f_k, yielding transformed sigma points

$$\chi^f_{k,i} = f(\chi_{k,i}). \tag{2.17}$$

The state update mean and covariance are then computed from the transformed sigma points via a weighted sum given by

$$\begin{aligned} \mathbf{x}^-_k &= \sum_{i=0}^{2N_K} w_{i,m}\chi^f_{k,i} \\ \mathbf{C}^-_k &= \sum_{i=0}^{2N_K} w_{i,c}\left[\chi^f_{k,i} - \mathbf{x}^-_k\right]\left[\chi^f_{k,i} - \mathbf{x}^-_k\right]^T + \mathbf{Q}_k \end{aligned} \tag{2.18}$$

where $w_{i,m}$ and $w_{i,c}$ are mean and covariance weights and, as with the linear Kalman filter, $\mathbf{Q}_k$ is the process covariance for iteration k. In general, the weights should be selected such that

$$\sum_i w_i = 1 \tag{2.19}$$

although a slight deviation from this rule is permitted in the scaled UKF for the co-

variance weights. In the scaled unscented transform, the weights are computed by

$$\begin{aligned} w_{0,m} &= \frac{\lambda_u}{N_K + \lambda_u} \\ w_{i,m} &= \frac{1}{2(N_K + \lambda_u)} \quad 1 \leq i \leq 2N_K \\ w_{0,c} &= \frac{\lambda_u}{N_K + \lambda_u} + 1 - \alpha_u^2 + \beta_u \\ w_{i,c} &= \frac{1}{2(N_K + \lambda_u)} \quad 1 \leq i \leq 2N_K \end{aligned} \tag{2.20}$$

where β_u is a distribution-based parameter; for Gaussian distributions, $\beta_u = 2$.

In the update step of the UKF, the measurement mean and covariance are computed. First, the sigma points are transformed through h_k by

$$\zeta_{k,i} = h_k(\chi_{k,i}^f). \tag{2.21}$$

The measurement expectation and its covariance are then computed by the weighted sums

$$\begin{aligned} \hat{\mathbf{z}}_k &= \sum_{i=0}^{2N_K} w_{i,m} \zeta_{k,i} \\ \mathbf{S}_k &= \sum_{i=0}^{2N_K} w_{i,c} \left[\zeta_{k,i} - \hat{\mathbf{z}}_k\right] \left[\zeta_{k,i} - \hat{\mathbf{z}}_k\right]^T + \mathbf{R}_k. \end{aligned} \tag{2.22}$$

Next, the cross-covariance between the measurement expectation and the state update prediction is computed by

$$\mathbf{C}_{k,xz} = \sum_{i=0}^{2N_K} w_{i,c} \left[\chi_{k,i}^f - \mathbf{x}_k^-\right] \left[\zeta_{k,i} - \hat{\mathbf{z}}_k\right]^T. \tag{2.23}$$

The Kalman gain may be computed by

$$\mathbf{K}_k = \mathbf{C}_{k,xz}\mathbf{S}_k^{-1}. \tag{2.24}$$

Finally, the output state from iteration k of the UKF is computed by

$$\hat{\mathbf{x}}_k = \mathbf{x}_k^- + \mathbf{K}_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_k\right) \tag{2.25}$$

and the updated covariance estimate $\hat{\mathbf{C}}_k$ is computed by

$$\hat{\mathbf{C}}_k = \mathbf{C}_k^- - \mathbf{K}_k\mathbf{S}_k\mathbf{K}_k^T. \tag{2.26}$$

2.2 Physical Effects on Transmitted and Received Signals

There are a number of practical effects that operate on a transmitted RF signal as it travels from the transmitter to a receiver (or in a radar, to a target and back to the radar system). These practical effects are important for radar operation, and having a precise understanding of the phenomenology and how to model it correctly will also be crucial for the derivation of the synchronization procedure in Chapter 4, which relies heavily on correctly transferring signals with precisely defined time, phase, and frequency – all of which are impacted by propagation effects. Therefore, this section provides a description of these propagation effects and how to model them properly.

2.2.1 Propagation Delay and Phase

The most fundamental propagation effect is the introduction of delay due to finite propagation speed. RF signals propagate through free space at the speed of light,

$c = 299792458$ m/s (frequently approximated as $c = 3 \times 10^8$ m/s), and assuming that there is a perfect time alignment between a transmitter and receiver, there will be an observed shift in time based on the distance between them. Suppose that a single-tone sinusoid at some carrier frequency f^c, given by $w_t(t)$, is transmitted from the transmitter, given by

$$w_t(t) = \exp\left(\mathrm{j}\, 2\pi f^c t\right), \tag{2.27}$$

where t is the time variable and the roman character j is the imaginary unit. Suppose that a receiver is placed R meters away from the transmitter. The propagation time will be given by the propagation range divided by the propagation velocity, R/c. The received signal $w_r(t)$ will therefore be given by

$$w_r(t) = \exp\left(\mathrm{j}\, 2\pi f^c \left(t - \frac{R}{c}\right)\right) = \exp\left(\mathrm{j}\, 2\pi f^c t\right) \exp\left(-\mathrm{j}\, 2\pi f^c \frac{R}{c}\right). \tag{2.28}$$

Therefore, the received signal is the same carrier frequency signal but with a phase shift proportional to the distance of signal propagation.

Suppose that the RF signal is an up-converted baseband signal, $s(t)$, given by

$$w_t(t) = s(t) \exp\left(\mathrm{j}\, 2\pi f^c\right). \tag{2.29}$$

Again, assuming time alignment between transmitter and receiver, the received signal will be a delayed version of the transmitted signal, given by

$$w_r(t) = s\left(t - \frac{R}{c}\right) \exp\left(\mathrm{j}\, 2\pi f^c t\right) \exp\left(-\mathrm{j}\, 2\pi f^c \frac{R}{c}\right). \tag{2.30}$$

Further, suppose that a down-conversion step is applied at the receiver to remove the

carrier term. The baseband received signal, $s_r(t)$, will then be given by

$$s_r(t) = s\left(t - \frac{R}{c}\right)\exp\left(-\mathrm{j}\,2\pi f^c \frac{R}{c}\right). \tag{2.31}$$

The resulting baseband signal is a phase-shifted and delayed version of the original baseband signal, with both the phase shift and delay being proportional to the propagation time.

Note that the delay and phase are for the direct propagation between a transmitter and a separate receiver. In a monostatic radar system, the transmitted signal must propagate the range to the target and then back, and as such the propagation time will be doubled, giving a received baseband signal at the radar of

$$s_r(t) = s\left(t - \frac{2R}{c}\right)\exp\left(-\mathrm{j}\,2\pi f^c \frac{2R}{c}\right). \tag{2.32}$$

2.2.2 Doppler Phase and Doppler Shift

Suppose that there is a relative movement between the transmitter and receiver such that at two different points in time, there is a separation between them of R_1 and R_2. The RF signal observed by the receiver at the first point in time will be

$$w_r(t) = s\left(t - \frac{R_1}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c t\right)\exp\left(-\mathrm{j}\,2\pi f^c \frac{R_1}{c}\right), \tag{2.33}$$

and at the second point it will be

$$w_r(t) = s\left(t - \frac{R_2}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c t\right)\exp\left(-\mathrm{j}\,2\pi f^c \frac{R_2}{c}\right). \tag{2.34}$$

Suppose that the difference between R_1 and R_2 is small enough that the delay to the baseband signal is insignificant. The carrier frequency f^c will typically be large enough that the change in phase between the two received signals is observable. This change in phase is sometimes referred to as the *Doppler phase*. Suppose that the range between platforms changes as a function of time based on the function $R(t)$. The received signal in this case will be given by

$$w_r(t) = s\left(t - \frac{R(t)}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c t\right)\exp\left(-\mathrm{j}\,2\pi f^c \frac{R(t)}{c}\right), \tag{2.35}$$

where again, the range change will be insignificant on the order of the length of the baseband signal envelope but will potentially be observable in some way in the phase term since f^c is large. Finally, suppose that $R(t)$ increases linearly due to a constant velocity v. In this case, the range may be expressed by

$$R(t) = R_1 + vt. \tag{2.36}$$

Thus, the received signal in (2.35) will reduce to

$$w_r(t) = s\left(t - \frac{R_1 + vt}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c t\right)\exp\left(-\mathrm{j}\,2\pi v\frac{f^c}{c}t\right)\exp\left(-\mathrm{j}\,2\pi f^c \frac{R_1}{c}\right). \tag{2.37}$$

The phase term thus is converted from a constant phase shift to a sinusoid with a constant frequency, f^d, given by

$$f^d = -v\frac{f^c}{c}. \tag{2.38}$$

This single frequency is referred to as the *Doppler shift* frequency. This may also be expressed in terms of the carrier wavelength, $\lambda_c = c/f^c$, by

$$f^d = -\frac{v}{\lambda_c}. \tag{2.39}$$

From the expression, it can be seen that for a positive velocity (that is, when the transmitter and receiver move away from one another), the Doppler shift is negative, while for a negative velocity (that is, when the transmitter and receiver move toward each other), the Doppler shift is positive. Generally, for a carrier frequency in the GHz range, the Doppler shift will be in the kHz range for moderate relative velocities. This shift in frequency will induce a small shift in the carrier frequency that will not be removed by the down-conversion process. This residual modulation of the baseband waveform most importantly will cause a pulse-compression peak during time-delay estimation to be shifted for many types of waveforms. This shift will be directly related to the magnitude of the Doppler shift frequency by a function called the ambiguity function.

As with the propagation phase, the expression above is for the observed Doppler shift at the signal destination from some source. In a monostatic radar system, the signal must travel to the target and back to the radar, which doubles the phase as in (2.32). This will also lead to a doubling of the observed Doppler shift given by

$$f^d = -2v\frac{f^c}{c}. \tag{2.40}$$

2.2.3 Signal Power

The above signal derivations make the assumption that the amplitude of the received signal is identical to the amplitude of the transmitted signal, which will never be the case in a real scenario. In a practical scenario, the energy of a transmitted signal will spread out in space such that the amount of the signal energy received at the destination will be significantly reduced. Suppose that the transmitter transmits a signal at a power of P_t, and that the transmitter has an antenna with a gain of G_t (in linear units). Further, assume that the receiver has an antenna with a gain of G_r. The Friis transmission formula describes the expected power of the signal received by the receiver and is given by

$$P_r = \frac{P_t G_t G_r \lambda_c^2}{(4\pi R)^2}, \tag{2.41}$$

where R is the range between the transmitter and receiver as above. This expression only describes the power received at the antenna of the receiver – typically, there will also be an RF front-end with its own gain, G_{RF}, such that the actual signal power at the receiver's analog-to-digital converter (ADC), P_D, will be given by

$$P_D = G_{\mathrm{RF}} P_r. \tag{2.42}$$

2.2.4 Noise Power and Signal-to-Noise Ratio

The input to the receiver will also generate thermal noise. This thermal noise will manifest as white noise (no correlation between the noise sequence values at different points in time), with the power in the noise related to the ambient temperature and the observed bandwidth by

$$P_N = k_b T_N B, \tag{2.43}$$

where k_b is Boltzmann's constant $k_b \approx 1.38\times10^{-23}$ J/K, T_N is the ambient temperature, and B is the bandwidth. The bandwidth will typically be set by the filters in the RF front-end. The ratio of the received power, P_r, to the noise power, P_N, is referred to as the input signal-to-noise ratio (SNR), given by

$$\mathrm{SNR}_{\mathrm{IN}} = \frac{P_r}{P_N}. \tag{2.44}$$

The SNR is a critical component of signal processing as it will directly impact the ability of useful information to be extracted. The higher the SNR, the better, and below a certain point, the noise will overpower the signal and no useful information can be extracted.

The noise will also be amplified by the gain of the RF front-end G_{RF}. Additionally, the RF front-end will add more noise to the signal such that the signal at the ADC will have a lower SNR than at the input. Define the noise figure F_N as the ratio of the input SNR to the ADC SNR by

$$F_N = \frac{\mathrm{SNR}_{\mathrm{IN}}}{\mathrm{SNR}_{\mathrm{ADC}}}. \tag{2.45}$$

The noise figure is a characteristic of the RF front-end, and more details on its computation may be found in [107]. By rearranging (2.45), the SNR at the ADC will be given by

$$\mathrm{SNR}_{\mathrm{ADC}} = F_N \mathrm{SNR}_{\mathrm{IN}}. \tag{2.46}$$

This SNR value (usually expressed in dB) will be crucial in characterizing the performance of the algorithms described in this dissertation. Usually, the SNR value alone in place of the other variables (temperature, antenna gain, etc.), since these will vary from system to system while the algorithm performance will always be constant for a given SNR.

2.3 Navigation for Radar

Localization is an important component of radar applications, particularly modalities that require motion compensation to operate, such as SAR. In the following section, basic principles of terrestrial navigation, and a description of navigation error on SAR performance, are provided.

2.3.1 Nonlinear Kinematic Equations

The following subsection describes the kinematic equations used for integrating IMU measurements into position and orientation; these equations provide the core of a dead-reckoning navigation solution and thus do not include GPS or other measurements. Let $\mathbf{x}_{i,k}^{\text{nav}}$ be the navigational system state vector of the i^{th} radar node at the k^{th} time series point composed of

$$\mathbf{x}_{i,k}^{\text{nav}} = \begin{bmatrix} \bar{p}_{i,k} \\ \bar{v}_{i,k} \\ \bar{\theta}_{i,k} \end{bmatrix}, \tag{2.47}$$

where $\bar{p}_{i,k}$, $\bar{v}_{i,k}$, and $\bar{\theta}_{i,k}$ denote the 3-dimensional position (x, y, and z), velocity (x, y, and z), and Euler angle attitude (ϕ, θ, and ψ, denoted "roll", "pitch," and "yaw," respectively) of the i^{th} radar node at the k^{th} point in a time series, respectively. The state transition function, f_k, is defined as follows. Let $\mathbf{\Omega}_{i,k}$ be the rotation matrix associated with $\bar{\theta}_{i,k}$ and let the input 3-dimensional body-frame accelerometer and gyroscope readings be denoted by $\bar{a}_{i,k}$ and $\bar{\omega}_{i,k}$, respectively. Additionally, let $\mathbf{\Omega}_e$ be the angular-rate rotation matrix for the Earth-rotation vector, let $\mathbf{g}$ be the acceleration due to gravity, and let Δt be the update time between the time steps $k-1$ and k. The state transition equations in the earth-centered, earth-fixed (ECEF) frame are then

given by [47]

$$\begin{aligned}
\mathbf{\Omega}_{i,k} &= \mathbf{\Omega}_{i,k-1}\mathbf{\Omega}^{\delta}_{i,k} - \mathbf{\Omega}_e\mathbf{\Omega}_{i,k-1}\Delta t \\
\bar{a}^r_{i,k} &= \left(\mathbf{\Omega}_{i,k-1}\mathbf{\Omega}^{a}_{i,k-1} - \frac{1}{2}\mathbf{\Omega}_e\mathbf{\Omega}_{i,k-1}\Delta t\right)\bar{a}_{p,k} \\
\bar{v}_{i,k} &= \bar{v}_{i,k-1} + \left(\bar{a}^r_{i,k} + \mathbf{g} - 2\mathbf{\Omega}_e\bar{v}_{i,k-1}\right)\Delta t \\
\bar{p}_{i,k} &= \bar{p}_{i,k-1} + \frac{1}{2}\Delta t\left(\bar{v}_{i,k-1} + \bar{v}_{i,k}\right),
\end{aligned} \tag{2.48}$$

where $\bar{\theta}_{i,k}$ is recovered from the rotation matrix $\mathbf{\Omega}_{i,k}$. The transitory matrices $\mathbf{\Omega}^{\delta}_{i,k}$ and $\mathbf{\Omega}^{a}_{i,k}$ are given by

$$\mathbf{\Omega}^{\delta}_{i,k} = \mathbf{I} + \frac{\sin\left|\bar{\alpha}_{i,k}\right|}{\left|\bar{\alpha}_{i,k}\right|}\left[\bar{\alpha}_{i,k}\right]_\times + \frac{1-\cos\left|\bar{\alpha}_{i,k}\right|}{\left|\bar{\alpha}_{i,k}\right|^2}\left[\bar{\alpha}_{i,k}\Delta t\right]^2_\times, \tag{2.49}$$

and

$$\mathbf{\Omega}^{a}_{i,k} = \mathbf{I} + \frac{1-\cos\left|\bar{\alpha}_{i,k}\right|}{\left|\bar{\alpha}_{i,k}\right|^2}\left[\bar{\alpha}_{i,k}\right]_\times + \frac{1}{\left|\bar{\alpha}_{i,k}\right|^2}\left(1 - \frac{\sin\left|\bar{\alpha}_{i,k}\right|}{\left|\bar{\alpha}_{i,k}\right|}\right)\left[\bar{\alpha}_{i,k}\right]^2_\times, \tag{2.50}$$

where $\left[\bar{\alpha}_{i,k}\right]_\times$ is the vector skew operation and

$$\bar{\alpha}_{i,k} = \bar{\omega}_{i,k}\Delta t. \tag{2.51}$$

In this formulation, $\mathbf{\Omega}^{\delta}_{i,k}$ describes the transformation of the body frame at the end of the update step to the beginning, while $\mathbf{\Omega}^{a}_{i,k}$ describes the transformation of the specific force measured into the body frame to that in the inertial frame.

The reference frame for the navigation formulation in (2.48) is ECEF, though other reference frames exist, notably earth-centered inertial (ECI), and local navigation frames such as north-east-down (NED). These other frames may also be used so

long as the state transition function is correctly implemented. Because a local frame such as NED is constructed relative to the position of a single body, it may be more difficult to implement for a distributed network of platforms such as the approach proposed here. Moreover, a Cartesian reference frame such as ECEF or ECI is convenient for the computation of the inter-node range measurements as described later. One challenge with the earth-centered formulations is that the acceleration due to gravity ($\mathbf{g}$) will generally always be rotated to point down in the local frame such that it will be rotated in the earth-centered frame as a function of position. Let $\mathbf{g}_{\text{ned}}$ be defined by

$$\mathbf{g}_{\text{ned}} = \begin{bmatrix} 0 & 0 & g_0 \end{bmatrix}, \tag{2.52}$$

where g_0 is the scalar acceleration due to gravity which varies as a function of latitude and height. More details on the precise computation of this value are given in [47]. The NED vector may then be rotated into the ECEF frame by

$$\mathbf{g} = \mathbf{\Omega}_{\text{ned}}\ \mathbf{g}_{\text{ned}}, \tag{2.53}$$

where

$$\mathbf{\Omega}_{\text{ned}} = \begin{bmatrix} -\sin(L_b)\cos(\lambda_b) & -\sin(\lambda_b) & -\cos(L_b)\cos(\lambda_b) \\ -\sin(L_b)\sin(\lambda_b) & \cos(\lambda_b) & -\cos(L_b)\sin(\lambda_b) \\ \cos(L_b) & 0 & -\sin(L_b) \end{bmatrix}, \tag{2.54}$$

where L_b is the latitude of the platform in radians and λ_b is the longitude in radians. Details about the computation of L_b and λ_b from the ECEF $\bar{p}_{i,k}$ may be found in [47].

2.3.2 Impact of Navigation Errors on SAR Imaging

Suppose that a radar transmits a series of pulses of a particular signal and on receive performs range compression of the received signal. Let t be the fast-time variable related to the signal variation within a single pulse and τ be the slow-time variable related to the signal variation from pulse to pulse. Let $\tilde{x}(t, \tau)$ describe the ideal range-compressed baseband received signal due to a single point-scatterer at 0 meters from the radar. Suppose that a single point-scatterer exists at a time-varying range $R(\tau)$ meters from the radar – this variation in range is modeled as significant only on the slow-time scale and is assumed to be primarily due to the motion of the radar mounted on a moving platform. The expected baseband receive signal due to the target reflection $\tilde{x}_R(t, \tau)$ will be a copy of $\tilde{x}(t, \tau)$ scaled in amplitude by some factor $A_x(\tau)$ and time-delayed by the two-way propagation time $2R(\tau)/c$ where c is the speed of light. In addition to the time delay, the signal will be phase-shifted as a result of the carrier phase rotation. The phase shift is given by

$$\Delta\gamma = -\frac{4\pi R(\tau)}{\lambda_c}, \tag{2.55}$$

where λ_c is the carrier wavelength. The expected receive signal is therefore given by

$$\tilde{x}_R(t, \tau) = A_x(\tau)\tilde{x}\left(t - \frac{2R(\tau)}{c}, \tau\right)\exp\left(-\frac{\mathrm{j}\, 4\pi R(\tau)}{\lambda_c}\right). \tag{2.56}$$

Note that the roman character j in (2.56) is used to refer to the imaginary unit to distinguish it from the index j later in this dissertation. The goal of motion compensation for SAR is to estimate and correct for the range-dependent delay and phase shift such that the received signal may be azimuth-compressed to form a two-dimensional image.

Let $x(t,\tau)$ be the actual range-compressed baseband receive signal for a particular SAR scene. Suppose that the receive signal is to be motion-compensated to a scene center position of $\mathbf{p}_0$ such that the range to be compensated $R_c(\tau)$ is given by

$$R_c(\tau) = \|\mathbf{p}_0 - \bar{p}_i(\tau)\|, \tag{2.57}$$

where $\bar{p}_i(\tau)$ is the true position vector of the radar platform i from (2.47) resampled to match the PRF of the radar such as in [77, 78]. Note that while some SAR algorithms will motion compensate an entire data collection based on a single reference point, others, such as the backprojection algorithm, will perform motion compensation separately for each pixel in the desired SAR image.

Let $\tilde{g}(t,\tau)$ be the signal obtained from $x(t,\tau)$ after ideal motion compensation, computed by

$$\tilde{g}(t,\tau) = x\left(t + \frac{2R_c(\tau)}{c}, \tau\right)\exp\left(\frac{\mathrm{j}\,4\pi R_c(\tau)}{\lambda_c}\right). \tag{2.58}$$

The motion compensation formulation in (2.58) assumes that $R_c(\tau)$ is known exactly, which in turn implies that a perfect estimate of $\bar{p}_i(\tau)$ is available. In practice, navigation inaccuracies will lead to an erroneous estimate $\hat{R}_c(\tau)$ of $R_c(\tau)$. Defining a range estimate error $R_e(\tau) = \hat{R}_c(\tau) - R_c(\tau)$, the signal obtained from $x(t,\tau)$ after non-ideal motion compensation $g(t,\tau)$ using the erroneous estimate $\hat{R}_c(\tau)$ is expressed by

$$\begin{aligned} g(t,\tau) &= x\left(t + \frac{2\hat{R}_c(\tau)}{c}, \tau\right)\exp\left(\frac{\mathrm{j}\,4\pi\hat{R}_c(\tau)}{\lambda_c}\right) \\ &= \tilde{g}\left(t + \frac{2R_e(\tau)}{c}, \tau\right)\exp\left(\frac{\mathrm{j}\,4\pi R_e(\tau)}{\lambda_c}\right). \end{aligned} \tag{2.59}$$

The residual range errors in the argument of $\tilde{g}(t,\tau)$ in (2.59) will result in an erroneous translation of the SAR image, while the residual range errors in the phase term

in (2.59) will result either in a translation of the SAR image (for linear range/phase errors) or a smearing of the PSR in the azimuth dimension (for nonlinear range/phase errors), or both. Generally, large but constant range errors will lead to translations of the image, while time-varying errors will lead to a blurring of the image due to azimuth smearing of the PSR. Note that because the severity of azimuth spreading is due to the magnitude of the phase error which varies proportionally to the range error and inversely proportionally to the carrier wavelength, the range error must be small on the order of a wavelength ($\lambda_c/16$ for slow-varying errors and $0.015\lambda_c$ for higher-frequency errors [75]) to avoid significant defocusing of the SAR image.

2.4 Signal Model and Synchronization for Radar

The signal model derived here is modeled after, and makes similar assumptions to, the signal model presented in [92]. Moreover, the resulting model is derived in similar fashion in [3] and [4]. First, it is assumed that there is a network of N_p arbitrarily distributed radar sensors with independent oscillators. For simplicity, it is also assumed that the characteristics of each radar system are identical. Thus, the nominal frequency of the main oscillator (MO) on each platform is identical, as are the nominal carrier frequencies and sampling clocks. It is also assumed that for each platform, a single MO is used as the frequency source for the digital sampling clocks and the local oscillator (LO). As a result, if there is a frequency error present in the MO of a platform, an identical offset will be present in the derived frequency sources.

Each MO operates with a nominal frequency of f^{MO} (e.g. 10 MHz). The MO on platform i is assumed to have a true frequency of f_i^{MO} which deviates somewhat from the nominal frequency. Define the drift of the MO for platform i (α_i) as the ratio of the

true frequency to the nominal frequency expressed by

$$\alpha_i \triangleq \frac{f_i^{\mathrm{MO}}}{f^{\mathrm{MO}}}. \tag{2.60}$$

Furthermore, it is assumed that in addition to the clock drift, each clock has an independent constant random bias (ϕ_i). Therefore, the local notion of time for platform i (τ_i) may be expressed in terms of the global time (t) by

$$\tau_i = \alpha_i t + \phi_i. \tag{2.61}$$

Equivalently, the global time may also be expressed in terms of the local time of platform i by

$$t = \frac{\tau_i - \phi_i}{\alpha_i} \tag{2.62}$$

It is assumed that each platform transmits with a nominal carrier frequency of f^c and has a nominal sampling frequency (both digital-to-analog and analog-to-digital) of f^s. Thus, the true carrier frequency for platform i is $f_i^c = \alpha_i f^c$ and the true sampling frequency for platform i is $f_i^s = \alpha_i f^s$. We may assume that the transmitter and receiver, though frequency locked to the main oscillator, have independent and random phase offset terms given by γ_i^{tx} and γ_i^{rx}, respectively.

2.4.1 Stationary Platform Case

Initially, the assumption may be made that all platforms remain stationary for throughout the synchronization procedure. Suppose that platform i transmits an arbitrary complex baseband waveform $s_j(t)$. Let $w_j(t)$ be the transmitted signal after RF up-conversion. Because the signal is generated in the clock domain of platform j, it is

expressed as

$$w_j(\tau_j) = s_j(\tau_j)\exp\left(\mathrm{j}\,2\pi f^c \tau_j\right)\exp\left(\mathrm{j}\,\gamma_j^{\mathrm{tx}}\right). \tag{2.63}$$

By substituting local time for an expression of global time as in (2.61), (2.63) may be expressed in the global time frame by

$$w_j(t) = s_j\left(\alpha_j t + \phi_j\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\alpha_j t + \phi_j\right)\right)\exp\left(\mathrm{j}\,\gamma_j^{\mathrm{tx}}\right). \tag{2.64}$$

Next, it is assumed that at the time $w_j(t)$ is transmitted, platform j is separated from another platform i by a distance $R_{i,j}$. For the stationary platform case, it may be assumed that $R_{i,j} = R_{j,i}$. The signal time-of-flight (TOF) is related to the separation distance by the speed of light (c) by

$$\mathrm{TOF}_{i,j} = \frac{R_{i,j}}{c}. \tag{2.65}$$

When the transmitted $w_j(t)$ is received by platform i, it has undergone a time delay equal to the TOF between the platforms. When down-converting, the receiving platform down-converts using the LO generated in its own local time frame. The signal received by platform i from the transmitting platform j ($r_{i,j}(t)$) is given by

$$r_{i,j}(t) = w_j\left(t - \mathrm{TOF}_{i,j}\right)\exp\left(-\mathrm{j}\,2\pi f^c \tau_i\right)\exp\left(-\mathrm{j}\,\gamma_i^{\mathrm{rx}}\right), \tag{2.66}$$

which may be expressed purely in terms of the global time frame using (2.61) by

$$r_{i,j}(t) = w_j\left(t - \mathrm{TOF}_{i,j}\right)\exp\left(-\mathrm{j}\,2\pi f^c(\alpha_i t + \phi_i)\right)\exp\left(-\mathrm{j}\,\gamma_i^{\mathrm{rx}}\right). \tag{2.67}$$

Substituting (2.64) into (2.67) above yields

$$\begin{aligned} r_{i,j}(t) &= s_j\left(\alpha_j\left(t-\frac{R_{i,j}}{c}\right)+\phi_j\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\alpha_j\left(t-\frac{R_{i,j}}{c}\right)+\phi_j\right)\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^c(\alpha_i t+\phi_i)\right)\exp\left(\mathrm{j}\left(\gamma_j^{\mathrm{tx}}-\gamma_i^{\mathrm{rx}}\right)\right) \\ &= s_j\left(\alpha_j t+\phi_j-\alpha_j\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c\left((\alpha_j-\alpha_i)t+\phi_j-\phi_i-\alpha_j\frac{R_{i,j}}{c}\right)\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right), \end{aligned} \tag{2.68}$$

where

$$\gamma_{i,j}^{\mathrm{err}}=\gamma_j^{\mathrm{tx}}-\gamma_i^{\mathrm{rx}} \tag{2.69}$$

is the difference between the random transmit and receive carrier phases for platforms j and i, respectively. Finally, substituting (2.62) into (2.68) above yields

$$\begin{aligned} r_{i,j}(\tau_i) &= s_j\left(\frac{\alpha_j}{\alpha_i}(\tau_i-\phi_i)+\phi_j-\alpha_j\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right) \\ &\quad\cdot\exp\left(\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j-\alpha_i}{\alpha_i}(\tau_i-\phi_i)+\phi_j-\phi_i-\alpha_j\frac{R_{i,j}}{c}\right)\right) \\ &= s_j\left(\frac{\alpha_j}{\alpha_i}\tau_i-\rho\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\left(\frac{\alpha_j}{\alpha_i}-1\right)\tau_i\right)\right)\exp\left(-\mathrm{j}\,2\pi f^c\rho\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right), \end{aligned} \tag{2.70}$$

where ρ is an auxiliary variable defined by

$$\rho=\frac{\alpha_j}{\alpha_i}\phi_i-\phi_j+\alpha_j\frac{R_{i,j}}{c}. \tag{2.71}$$

The resulting expression in (2.70) describes an arbitrary signal received by platform i after being transmitted by platform j as it will be observed in the local clock frame by platform i and is valid for any arbitrary waveform $s_j(t)$ assuming that platforms i and j have no relative motion.

2.4.2 Moving Platform Case

For the case of moving platforms, some modifications to the signal model assumptions are required. First, it must now be assumed that $R_{i,j} \neq R_{j,i}$ unless $w_i(t)$ and $w_j(t)$ are transmitted simultaneously. Second, the Doppler shift must be accounted for based on the radial velocity between the platforms. Recall that the Doppler shift (f^d) induced by relative motion between the source and observer of an electromagnetic wave is approximated by

$$f^d = -v\frac{f^c}{c}, \tag{2.72}$$

where v is the relative velocity between the source and observer along the line-of-sight (LOS) between them. The negative sign in (2.72) is included by convention such that a positive value of v indicates platforms moving away from one another and thus leading to a growing value of the distance between them. Suppose that there is a relative velocity of $v_{i,j}$ between platforms i and j during the transmission of the signal by platform j. Define the Doppler shift between them as $f^d_{i,j}$ which is given by

$$f^d_{i,j} = -v_{i,j}\frac{f^c_j}{c} = -v_{i,j}\frac{\alpha_j f^c}{c}. \tag{2.73}$$

As in typical pulse-Doppler processing applications, the simplifying assumption is made that because the carrier frequency is significantly larger than the bandwidth of the baseband waveform, the Doppler shift frequency is approximately constant for the full RF bandwidth of the system and is dependent almost exclusively on the carrier frequency. While the transmitted signal $w_j(t)$ given in (2.64) will not change relative to the stationary platform case, the received signal $r_{i,j}(t)$ will be subjected to a Doppler

shift, producing a new signal ($r^d_{i,j}(t)$) given by

$$\begin{aligned} r^d_{i,j}(t) &= r_{i,j}(t)\exp\left(\mathrm{j}\,2\pi f^d_{i,j}t\right) \\ &= s_j\left(\alpha_j t + \phi_j - \alpha_j\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi f^d_{i,j}t\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right) \\ &\quad\cdot\exp\left(\mathrm{j}\,2\pi f^c\left((\alpha_j-\alpha_i)t + \phi_j - \phi_i - \alpha_j\frac{R_{i,j}}{c}\right)\right). \end{aligned} \tag{2.74}$$

Finally, substituting (2.62) into (2.74) and simplifying in similar fashion to (2.70) produces the expression in terms of τ_i given by

$$\begin{aligned} r^d_{i,j}(\tau_i) &= s_j\left(\frac{\alpha_j}{\alpha_i}\tau_i - \rho\right)\exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j}{\alpha_i}-1\right)+\frac{f^d_{i,j}}{\alpha_i}\right)\tau_i\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^c\rho\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right). \end{aligned} \tag{2.75}$$

The resulting expression describes the received signal as it is observed by platform i for any arbitrary $s_j(t)$ with the inclusion of relative motion between the platforms inducing a Doppler shift.

This chapter provides an overview of relevant mathematical and physical considerations forming the basis for the rest of this dissertation, including Kalman filtering, wireless signal propagation, terrestrial navigation, and signal modeling. These concepts will be combined in relevant ways to achieve this dissertation's research goals, beginning with the UKF-based cooperative navigation algorithm described in the next chapter.

Chapter 3

Cooperative Navigation

This chapter describes the work that has been done on a cooperative navigation algorithm. It provides the mathematics of the algorithm, in particular how the algorithm builds on and integrates into the UKF. Results are shown for different scenarios with varying IMU qualities and levels of GPS access. Finally, SAR simulations are run and results are demonstrated showing the effectiveness of cooperative navigation over traditional non-cooperative navigation for motion compensation in SAR imaging.

3.1 Proposed Algorithm

The algorithm is formulated to fit directly into the UKF. Thus, the primary decisions to be made in formulating the algorithm are procuring the form of the measurement vector and measurement function, and producing estimates of the process and measurement covariances.

3.1.1 State Model

Suppose that the distributed radar network comprises N_p radar nodes. While the accelerometer and gyroscope biases might typically be included as state variables, they

are omitted for simplicity since they are not strictly utilized in the proposed algorithm. The state vector for the entire system, $\mathbf{x}_k$, is then given by

$$\mathbf{x}_k^{\text{nav}} = \begin{bmatrix} \mathbf{x}_{1,k}^{\text{nav}} \\ \mathbf{x}_{2,k}^{\text{nav}} \\ \vdots \\ \mathbf{x}_{N_p,k}^{\text{nav}} \end{bmatrix}. \tag{3.1}$$

Thus, the state vector length is $N_K = 9N_p$ (or $N = 15N_p$ if accelerometer and gyroscope biases are included). The state update function f_k is described by the navigation equations in (2.48).

3.1.2 Measurement Vector and Measurement Function

The measurement vector is composed of whatever GPS position measurements are available (if any) and whatever range measurements are available (if any). Therefore, the length M_K of the measurement vector may be subject to change at different iterations. The ordering of the available measurements within the state vector may be selected arbitrarily if a convention is well-defined. The author uses the following ordering convention where the GPS measurements (after conversion to the ECEF frame) are listed before the range measurements by

$$\mathbf{z}_k^{\text{nav}} = \begin{bmatrix} \mathbf{z}_k^{\text{GPS}} \\ \mathbf{z}_k^{R} \end{bmatrix}. \tag{3.2}$$

The most straightforward ordering of the GPS measurement vector is in the order of platform number such that the GPS measurement vector, when GPS measurements

are available for every platform, will be given by

$$\mathbf{z}_k^{\text{GPS}} = \begin{bmatrix} \mathbf{z}_{1,k}^{\text{GPS}} \\ \mathbf{z}_{2,k}^{\text{GPS}} \\ \vdots \\ \mathbf{z}_{N_p,k}^{\text{GPS}} \end{bmatrix}, \tag{3.3}$$

where $\mathbf{z}_{i,k}^{\text{GPS}}$ is a direct 3-dimensional measurement of $\bar{p}_{i,k}$. Therefore, the component of h_k for each GPS measurement is the identity function of position. The ordering of range measurements may be done based on the first platform index followed by the second platform index such that the range measurement vector composed of all possible range measurements will be given by

$$\mathbf{z}_k^R = \begin{bmatrix} R_{1,2}^k & \dots & R_{1,N_p}^k & R_{2,3}^k & \dots & R_{N_p-1,N_p}^k, \end{bmatrix}^T \tag{3.4}$$

where the range measurement component of h_k is given by

$$R_{i,j}^k = \|\bar{p}_{i,k} - \bar{p}_{j,k}\|. \tag{3.5}$$

In the work done to date, only complete network topologies are considered. As such, $\mathbf{z}_k^R$ will either contain the complete set of $L = N_p(N_p - 1)/2$ range measurements as in (3.4), where L is the number of range measurement links in the network, or it will be empty ($L = 0$).

3.1.3 Computation of Process and Measurement Covariance

Perhaps the most crucial and challenging design consideration in creating a functional Kalman filter is the proper selection of the process and measurement covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$. In this subsection, accurate approximations of the measurement and process covariance matrices are provided.

3.1.3.1 Measurement Covariance

Generally, the measurement covariance is more straightforward to estimate than the process covariance. The measurement covariance $\mathbf{R}_k$ is significantly easier to estimate if it is assumed that the measurements are independent since the matrix will be diagonal with elements equal to the variance of each measurement. Since the variance of each measurement, or sensor, can be deduced through basic experiments or a datasheet, the computation of $\mathbf{R}_k$ is trivial in this case. Let $\mathbf{R}_k^{\text{nav}}$ be the measurement covariance for the navigation UKF (to differentiate it from other instances of the UKF later on) For the measurement vector ordered in (3.2), $\mathbf{R}_k^{\text{nav}}$ is given by

$$\mathbf{R}_k^{\text{nav}} = \begin{bmatrix} \mathbf{R}_k^{\text{GPS}} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_k^{R} \end{bmatrix}. \tag{3.6}$$

The matrix blocks in (3.6) are given by

$$\mathbf{R}_k^{\text{GPS}} = \begin{bmatrix} \Sigma_1^{\text{GPS}} & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \Sigma_{N_p}^{\text{GPS}} \end{bmatrix}, \tag{3.7}$$

and

$$\mathbf{R}_k^R = \begin{bmatrix} \sigma_{R,1}^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_{R,L}^2 \end{bmatrix}, \tag{3.8}$$

where $\sigma_{R,l}^2$ is the range measurement variance of the l^{th} link. Generally, GPS will have errors with unique variances in the latitude, longitude, and height dimensions, which must be translated to a covariance matrix described in the ECEF frame. Assuming the GPS errors in latitude, longitude, and height are independent, we may define a covariance Σ_i^{LLH} as

$$\Sigma_i^{\text{LLH}} = \begin{bmatrix} \sigma_{L,i}^2 & 0 & 0 \\ 0 & \sigma_{\lambda,i}^2 & 0 \\ 0 & 0 & \sigma_{h,i}^2 \end{bmatrix}, \tag{3.9}$$

where $\sigma_{L,i}^2$, $\sigma_{\lambda,i}^2$, and $\sigma_{h,i}^2$ are the latitude, longitude, and height variance of the GPS measurement for platform p, respectively. Note that the standard deviations $\sigma_{L,i}$ and $\sigma_{\lambda,i}$ are given in meters, not in radians or degrees. The covariance may be rotated into the ECEF frame to produce

$$\Sigma_i^{\text{GPS}} = \mathbf{\Omega}_{\text{ned}} \Sigma_i^{\text{LLH}} \mathbf{\Omega}_{\text{ned}}^T. \tag{3.10}$$

While the above equations are generalized, if each platform uses identical GPS and ranging hardware, then Σ_i^{LLH} and $\sigma_{R,l}^2$ will be equal for all l and i while Σ_i^{GPS} will still vary from platform to platform as a function of position. It is assumed here that the GPS and range measurement errors are zero-mean additive white Gaussian noise (AWGN).

3.1.3.2 Process Covariance

The process covariance is often more difficult to select than the measurement covariance, particularly when there exist strong nonlinear relationships between the state variables. While the navigation equations contain nonlinear relationships between the orientation and position variables, they are not strong enough to justify consideration. Let $\mathbf{Q}_k^{\text{nav}}$ the process covariance for the navigation UKF to differentiate it from the process covariance matrices in other instances of the UKF later on. First, by making the assumption that the process noise for each platform will be independent, the process covariance may be structured by

$$\mathbf{Q}_k^{\text{nav}} = \begin{bmatrix} \mathbf{Q}_{1,k}^{\text{nav}} & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & \mathbf{Q}_{N_p,k}^{\text{nav}} \end{bmatrix}, \tag{3.11}$$

where the block $\mathbf{Q}_{i,k}^{\text{nav}}$ describes the process covariance of the i^{th} platform. Suppose that the IMU on platform i has accelerometer noise standard deviation $\sigma_{a,i}$ and gyroscope noise standard deviation $\sigma_{\omega,i}$. It is assumed that the accelerometer and gyroscope noise can both be modeled as zero-mean AWGN. Each platform's process covariance may be structured by

$$\mathbf{Q}_{k,i}^{\text{nav}} = \begin{bmatrix} \mathbf{C}_{pp,i} & \mathbf{C}_{pv,i} & \mathbf{0} \\ \mathbf{C}_{vp,i} & \mathbf{C}_{vv,i} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_{\omega\omega,i} \end{bmatrix}, \tag{3.12}$$

where the constituent matrix $\mathbf{C}_{pp,i}$ is computed by

$$\mathbf{C}_{pp,i} = \begin{bmatrix} \sigma_{a,i}^2 \Delta t^4/4 & 0 & 0 \\ 0 & \sigma_{a,i}^2 \Delta t^4/4 & 0 \\ 0 & 0 & \sigma_{a,i}^2 \Delta t^4/4 \end{bmatrix}, \tag{3.13}$$

$\mathbf{C}_{vp,i}$ and $\mathbf{C}_{pv,i}$ are computed by

$$\begin{aligned} \mathbf{C}_{pv,i} &= \mathbf{C}_{vp,i} \\ &= \begin{bmatrix} \sigma_{a,i}^2 \Delta t^3/2 & 0 & 0 \\ 0 & \sigma_{a,i}^2 \Delta t^3/2 & 0 \\ 0 & 0 & \sigma_{a,i}^2 \Delta t^3/2 \end{bmatrix}, \end{aligned} \tag{3.14}$$

$\mathbf{C}_{vv,i}$ is computed by

$$\mathbf{C}_{vv,i} = \begin{bmatrix} \sigma_{a,i}^2 \Delta t^2 & 0 & 0 \\ 0 & \sigma_{a,i}^2 \Delta t^2 & 0 \\ 0 & 0 & \sigma_{a,i}^2 \Delta t^2 \end{bmatrix}, \tag{3.15}$$

and $\mathbf{C}_{\omega\omega,i}$ is computed by

$$\mathbf{C}_{\omega\omega,p} = \begin{bmatrix} \sigma_{\omega,i}^2 \Delta t^2 & 0 & 0 \\ 0 & \sigma_{\omega,i}^2 \Delta t^2 & 0 \\ 0 & 0 & \sigma_{\omega,i}^2 \Delta t^2 \end{bmatrix}. \tag{3.16}$$

Although the above definition for $\mathbf{Q}_k^{\text{nav}}$ does not consider the slight nonlinear relationships in the state variables, it will suffice for most practical applications.

3.2 Results

In this section, the proposed algorithm is demonstrated in simulation for navigation accuracy. A set of 8 random platform flight paths are generated over a simulation time-space of 5 minutes, and noisy sensor measurements are generated to match the flight paths. The flight paths are simulated by generating a random set of maneuvers consisting of random acceleration (or deceleration) in the forward-facing x-dimension of the body frame along with random banks in attitude. The maximum acceleration during these maneuvers is capped at 8 m/s^2 while the maximum angular velocity is capped at 5°/s. The flight paths are shown in Fig. 3.1. The navigation algorithm is run using noisy measurements, with the severity of the noise varied for different simulated qualities of IMUs. The IMU update rate for each simulation is set at 100 Hz, while the update rate for the inter-node range measurements is set at 1 Hz. For the simulated case where GPS is available, the update rate of GPS is also set at 1 Hz. Different cases of the algorithm's performance are shown in this section. First, a low-quality IMU is used in cases of networks with 4 and 8 platforms. Next, the case in which a high-quality IMU is available is demonstrated. In both cases, GPS is simulated as unavailable, requiring the navigation to be performed exclusively using IMU outputs and inter-node range measurements. In another case, one platform is given access to GPS while all others act as GPS-denied platforms. It is then shown that through the inter-node measurements, the network may still produce navigation solutions that are on average as accurate as if each platform were navigating using GPS. Finally, a case is presented in which all platforms have access to GPS to demonstrate the utility of cooperative navigation even under circumstances when traditional GPS-aided navigation is possible.

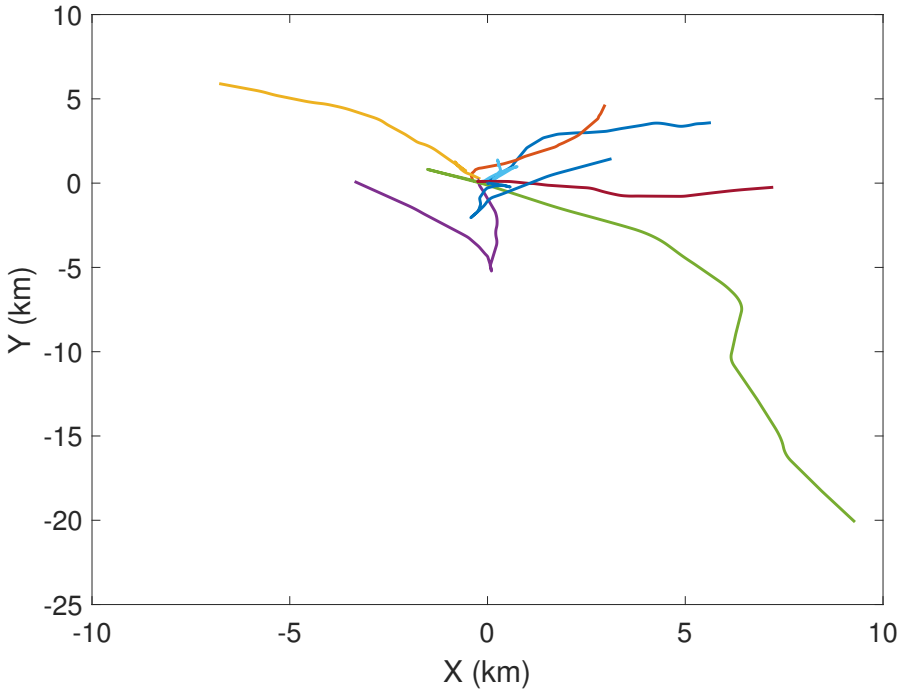


Fig. 3.1: The 8 simulated motion paths. From [1] ©2023 IEEE.

3.2.1 Low-Accuracy IMU, No GPS

The first simulations are run in a case where no platforms have access to GPS, and the IMU quality is low with parameters $\sigma_a = 0.05$ m/s/$\sqrt{\text{hr}}$ and $\sigma_\omega = 0.75$ °/$\sqrt{\text{hr}}$. The ranging accuracy is set at $\sigma_{R,l} = 0.1$ m for each link l. The accuracy results for the individual and cooperative navigation, when only four platforms are included, are shown in Fig. 3.2, while the same results with eight platforms are shown in Fig. 3.3. As expected, the cooperative navigation accuracy increases with the number of platforms in the network because more inter-node range measurements may be exploited in the update step.

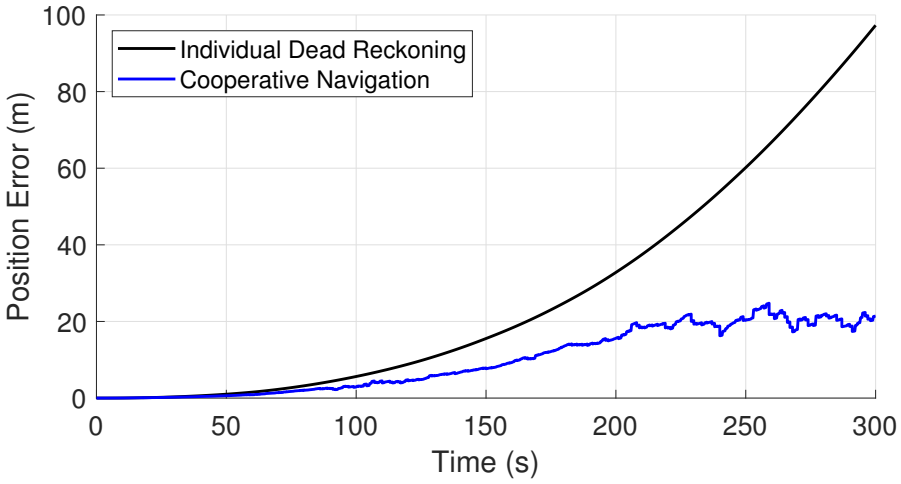


Fig. 3.2: Average platform navigation error for 4 platforms with no GPS and a low-quality IMU with individual dead-reckoning (black) and using cooperative navigation (blue). From [1] ©2023 IEEE.

3.2.2 High-Accuracy IMU, No GPS

The second simulation is run in a case where, again, no platforms have access to GPS, but the IMU quality is high with parameters $\sigma_a = 2.5 \times 10^{-5}$ m/s/$\sqrt{\text{hr}}$ and $\sigma_\omega = 2 \times 10^{-3}$ °/$\sqrt{\text{hr}}$. The accuracy results for the individual and cooperative navigation with eight platforms are shown in Fig. 3.4. As can be seen, the navigation accuracy does increase, but not on the same scale as the navigation improvement for the low-accuracy IMU.

3.2.3 Low-Accuracy IMU, One Platform GPS

The third simulation is run for a case where the IMU quality is low with $\sigma_a = 0.05$ m/s/$\sqrt{\text{hr}}$ and $\sigma_\omega = 0.75$ °/$\sqrt{\text{hr}}$. The ranging accuracy remains at $\sigma_{R,l} = 0.1$ m for each link l. In this simulation, however, one platform has access to a low-quality GPS with $\sigma_{L,1} = \sigma_{\lambda,1} = 5$ m and $\sigma_{h,1} = 7$ m. The other platforms continue to have no access to GPS. The accuracy results for this scenario are shown in Fig. 3.5. These

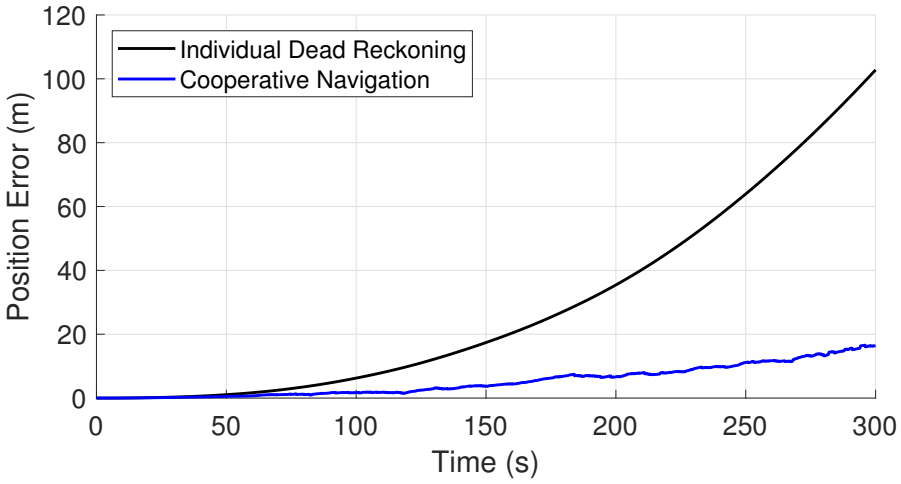


Fig. 3.3: Average platform navigation error for 8 platforms with no GPS and a low-quality IMU with individual dead-reckoning (black) and using cooperative navigation (blue). From [1] ©2023 IEEE.

results demonstrate that so long as one platform has access to GPS, it may distribute its own increase in accuracy to the remaining platforms, enabling GPS-level navigation accuracy for the entire network, even when the majority of the navigating platforms have no access to GPS.

3.2.4 Low-Accuracy IMU, All Platforms GPS

The final simulation is run for a case in which all platforms have access to GPS. In this case, the IMU qualities and GPS properties are identical to the above case. The accuracy results for this scenario are given in Fig. 3.6. In this instance, it can be seen that even in situations when GPS is widely available and traditional navigation is possible, the cooperative navigation scheme will outperform it, in this case by halving the average navigation error.

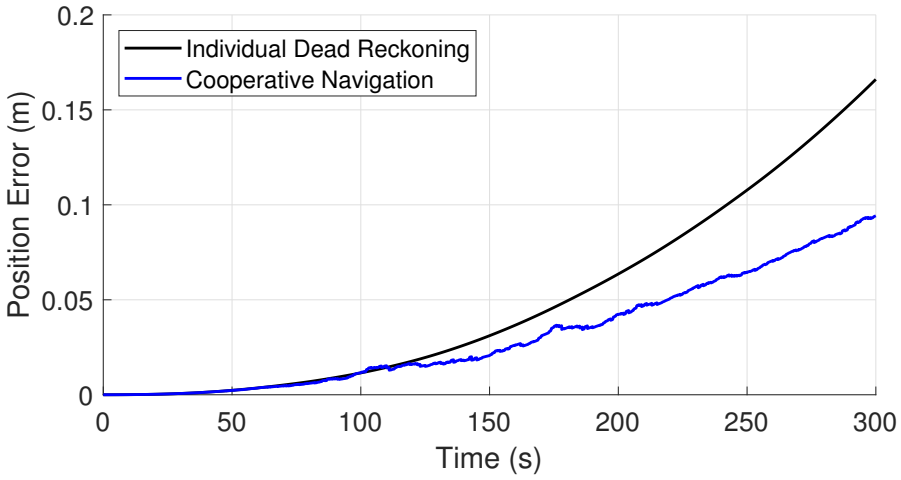


Fig. 3.4: Average platform navigation error for 8 platforms with no GPS and a high-quality IMU with individual dead-reckoning (black) and using cooperative navigation (blue). From [1] ©2023 IEEE.

3.3 SAR Results

To facilitate a demonstration of the proposed cooperative navigation principles in the context of radar remote sensing, simulations of SAR data captures are produced for several situations, highlighting the utility of cooperative navigation for radar applications. The navigation algorithms are evaluated with respect to the system point spread response (PSR) as compared to the ideal PSR. This section provides simulated a PSR for single-platform and distributed-platform SAR systems using traditional and cooperative navigation schemes. It is worth noting that the simulations conducted here do not include noise since the focus is on the shape of the PSR, which is defined with respect to the radar system properties and the navigation quality but is not dependent on the signal-to-noise ratio (SNR). It has been documented in other works [108] that there is a significant SNR gain in coherent distributed aperture radar systems, and as such, this analysis is excluded here.

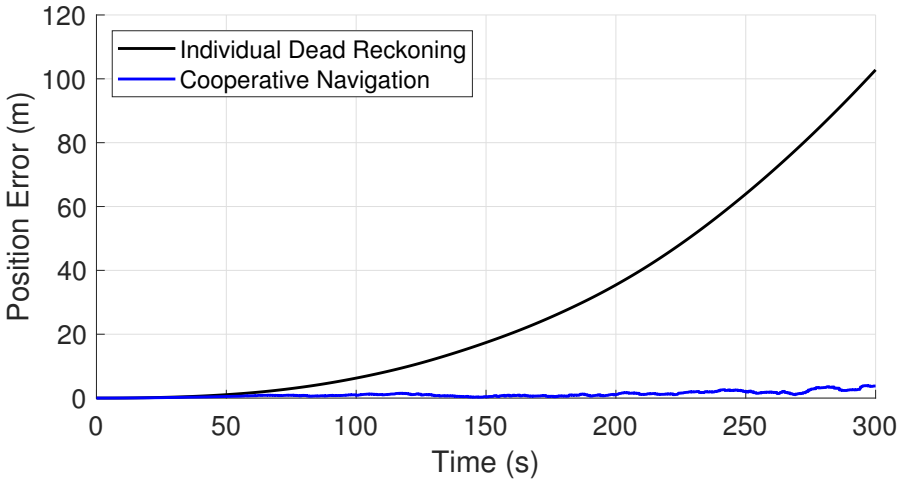


Fig. 3.5: Average platform navigation error for 8 platforms with only one platform having access to GPS and a low-quality IMU with individual dead-reckoning (black) and using cooperative navigation (blue). From [1] ©2023 IEEE.

3.3.1 Single Platform PSR

A scenario is simulated in which 8 platforms are flying in a formation, primarily in parallel straight lines. The navigation equipment is simulated with identical quality to the low-accuracy IMU and inter-node ranging accuracy described in Section 3.2.1. Similarly, in this case, no GPS signal is available on any platform. A target, placed 1000 m away, is observed by a radar system on a single platform which, at the time of the radar capture, is traversing a 100 m synthetic aperture at a velocity of 100 m/s over a capture time of 1 s. The range vector to the target is perpendicular to the velocity vector of the radar platform. It is assumed that the platforms have a precise knowledge of their locations 180 s before the radar capture occurs, but that for the time between then and the radar capture their positions must be calculated using the described cooperative technique. Because of the tendency for the estimated position to make rapid jumps each time the inter-node range measurements are used to correct the paths, the radar capture is configured to occur between two ranging updates such

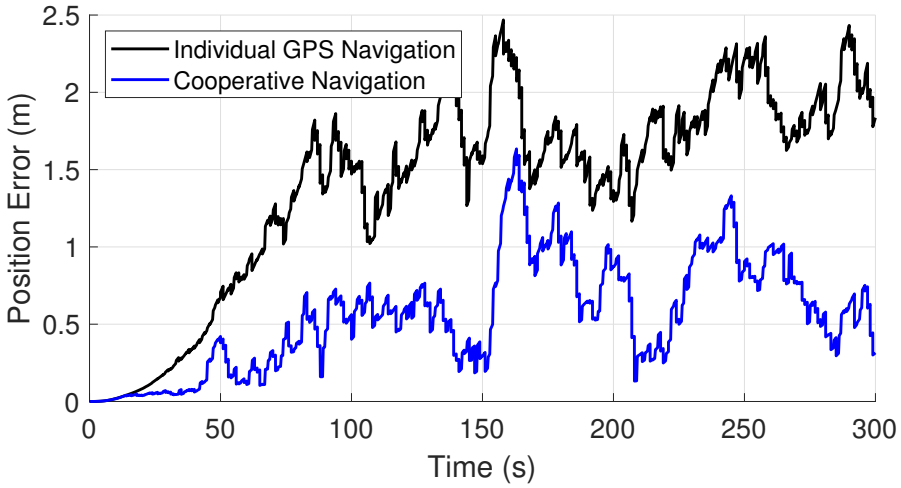


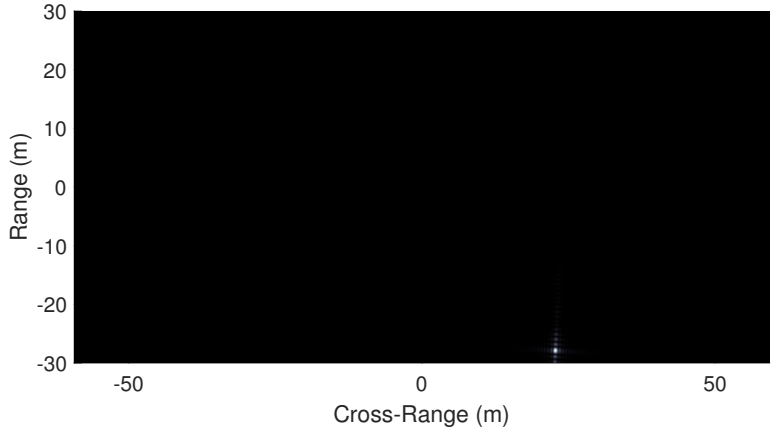
Fig. 3.6: Average platform navigation error for 8 platforms, each with access to GPS and a low-quality IMU with individual GPS-aided navigation (black) and using cooperative navigation (blue). From [1] ©2023 IEEE.

that any range error function $R_e(\tau)$ will be less jagged (to facilitate this, the inter-node ranging update rate is reduced to 0.8 Hz – this slightly reduced update rate has only a very minor impact on the overall navigation performance). Although the radar platform navigates cooperatively with the other moving platforms in the network, it is the only platform performing a radar function.

The radar system is simulated to operate with a carrier frequency of 3 GHz (λ_c = 0.1 m) with a signal bandwidth of 300 MHz. The bandwidth, aperture length, and target range are all selected interdependently such that the ideal range and azimuth resolution are approximately 0.5 m. The radar data collections are motion-compensated and images are formed using the backprojection algorithm.

Because only a single ideal point target is simulated, the resulting SAR image represents the PSR of the SAR system given the navigation solution used for motion compensation. Therefore, when motion-compensated using the true motion profile, the optimal PSR is formed. When motion compensation is performed using a different non-ideal navigation profile, deformations and translations of the PSR provide a useful

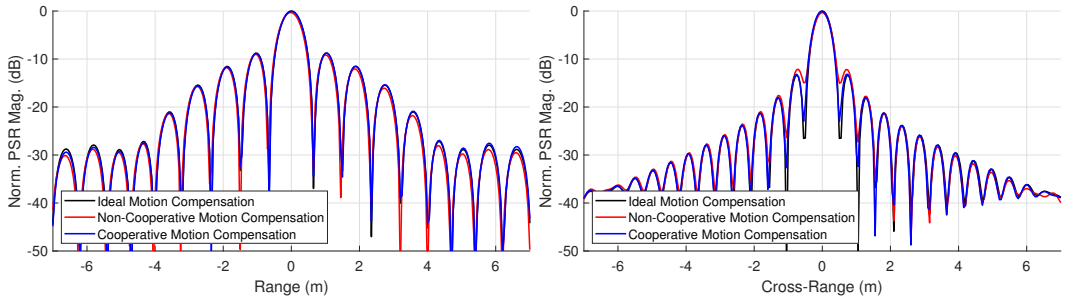
(a)



(b)

(c)

Fig. 3.7: Simulated single platform PSR for different navigation solutions: ideal navigation (a), non-cooperative navigation (b), and cooperative navigation (c). From [1] ©2023 IEEE.



(a) (b)

Fig. 3.8: One-dimensional cuts of the simulated single platform PSR for ideal, non-cooperative, and cooperative motion compensations. The cross-track cut is shown in (a) and the along-track cut is shown in (b). From [1] ©2023 IEEE.

insight into the deleterious impacts of the navigation solution on the SAR imaging capability. The simulated PSR for the given scenario is given in Fig. 3.7 for the ideal, non-cooperative, and cooperative navigation cases. The most obvious defect of the non-ideal PSRs in Fig. 3.7 is the erroneous translation of the target away from the scene center at the origin. However, the translation of the PSR formed using the non-cooperative navigation is much more substantial, moving the target's apparent location 36 m from the origin compared to a much tamer 10 m for the cooperative navigation case.

Closer inspection of the non-cooperative PSR in Fig. 3.7b reveals not only a substantial translation from the origin but also azimuth smearing of the PSR which will ultimately lead to blurring of the image. Fig. 3.8 shows plots of one-dimensional cross-track and along-track cuts of the PSR for the different navigation cases. From these plots, it can be seen that in the non-cooperative navigation case, the along-track (azimuth) PSR has a lowered peak level and a raised sidelobe level compared to the ideal case, while the cooperative navigation case has no such limitation. The defects in the non-cooperative navigation case are likely the result of quadratic phase errors (QPE) which are possibly produced due to an along-track velocity estimation error,

LOS acceleration measurement errors, or both [75].

3.3.2 Distributed Platform PSR

Another scenario is simulated similar to the single-platform case in which 8 platforms are flying in a formation. In this case, 4 platforms are flying co-linearly while 4 others are flying parallel but offset in the perpendicular dimensions such that they surround the other 4, providing inter-node range measurements in all spatial dimensions. An image depicting this formation is shown in Fig. 3.9. Unlike the previous case, this scenario considers multiple platforms in the network as transmitting and receiving radar signals. The 4 co-linear platforms (shown as colored lines in Fig. 3.9) are acting as radar systems while the 4 other systems (shown as black lines in Fig. 3.9) are acting as auxiliarly platforms whose only function is to contribute to the accuracy of the cooperative navigation technique. In the geometry of Fig. 3.9, the single point target is located at the origin, orienting its range vector perpendicular to the full aperture composed of the four subapertures. In order to achieve the required accuracy such that the returns from each radar platform coherently integrate into a focused image, low-quality GPS is enabled for each platform in this simulation. The noise properties of the IMUs, GPS, and inter-node range signals are identical to those described in Section 3.2.4. A single point target is placed 3000 m away and is observed by all 4 platforms as they each traverse a 100 m path at a velocity of 100 m/s over a capture time of 1 s. The platforms are 100 m apart along their paths and thus form a total aperture length of 400 m. It is assumed that the platforms have a precise knowledge of their locations 30 s before the radar capture occurs, but that for the time between then and the radar capture their positions must be calculated using GPS in conjunction with the described cooperative radar technique. For similar reasons as with the single plat-

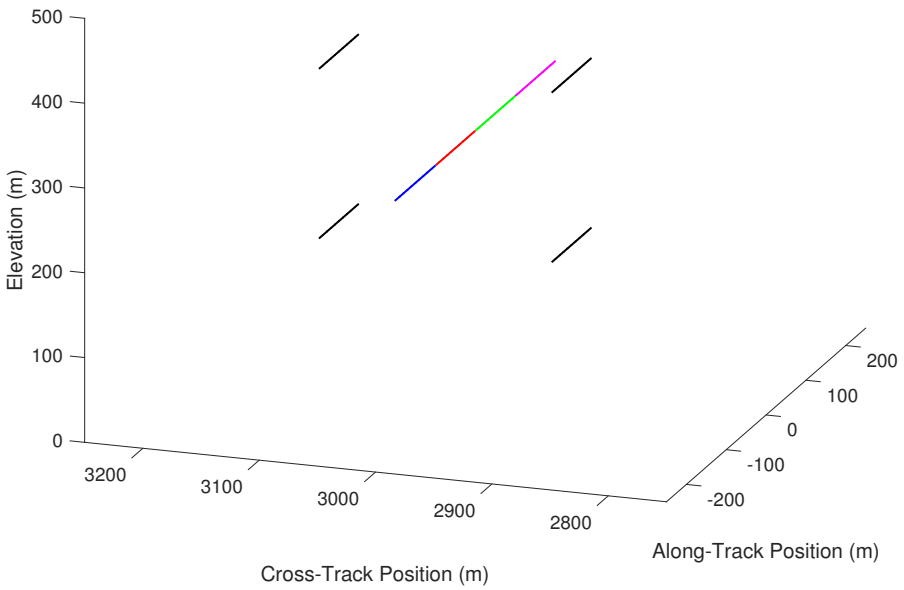
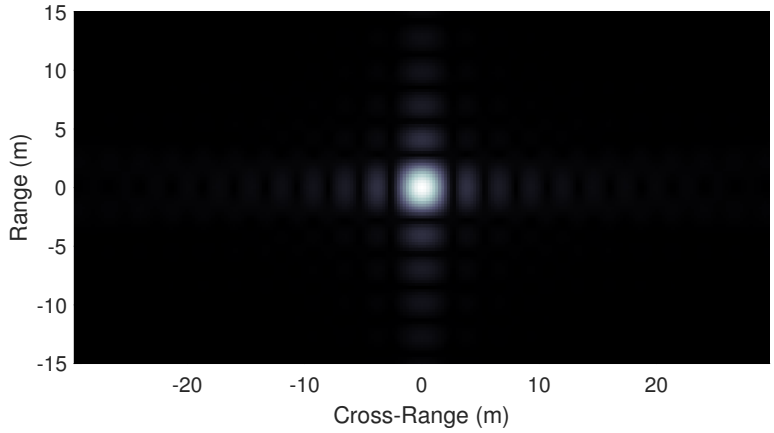


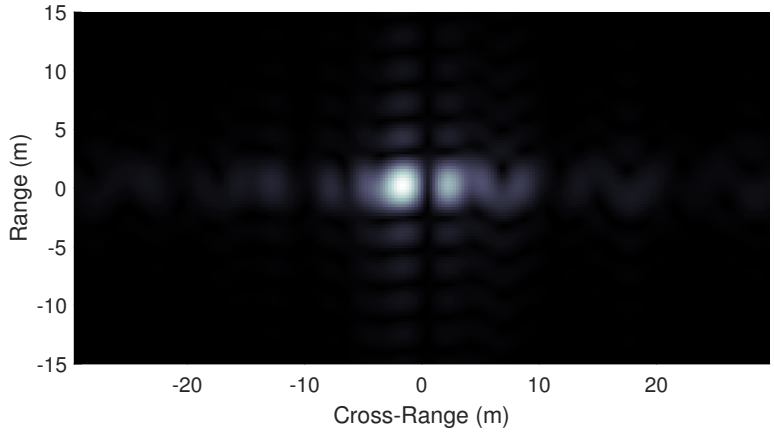
Fig. 3.9: The motion paths of the 8 distributed platforms during the radar capture. The colored paths indicate the 4 co-linear platforms that are transmitting and receiving radar signals, while the black paths indicate the 4 auxiliary platforms that are present for navigation purposes only. In the given coordinate reference frame, the point target is located at the origin. From [1] ©2023 IEEE.

form PSR simulation, the inter-node ranging updates and GPS updates are performed at a rate of 0.8 Hz.

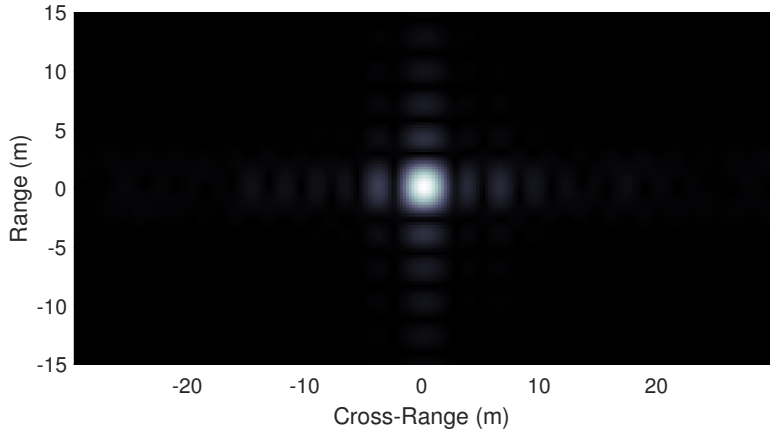
In this new scenario, the radar systems are simulated to operate with a carrier frequency of 425 MHz ($\lambda_c \approx 0.706$ m) with a signal bandwidth of 60 MHz – these parameters are taken from [23] in which a distributed SAR capture is implemented with UAVs. However, in [23] the navigation is implemented with differential global navigation satellite system (DGNSS) technology which is capable of achieving 0.02 m accuracy, in contrast to the low-quality GPS system simulated here. The aperture length and target range are selected based on the center frequency and bandwidth such that the ideal range and azimuth resolution are approximately 2.5 m. As with the



(a)



(b)

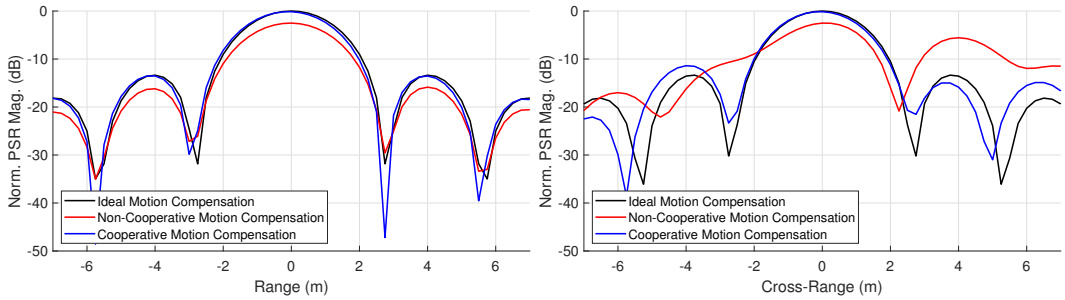


(c)

Fig. 3.10: Simulated distributed platform PSR for different navigation solutions: ideal navigation (a), non-cooperative navigation (b), and cooperative navigation (c). From [1] ©2023 IEEE.

single platform case, the radar data collections are motion-compensated and images are formed using the backprojection algorithm. For these simulations, it is assumed that the distributed radar platforms are synchronized in time and RF carrier phase and frequency. Moreover, to simplify the problem, it is assumed that each platform transmits the same signal with signal orthogonality achieved via time-division multiplexing leading to a reduction in effective pulse repetition time by a factor of 4. Finally, the simulation considers only the monostatic case for each platform, so signal reflections between bistatic pairs of transceivers are not included in the processing. To generate the PSRs, a sub-image is generated for each platform after motion compensation. Then, the sub-images are added together. Because the simulation is configured such that phase coherence between the radar systems is achieved a priori, any deformation of the ideal PSR may only be caused by erroneous motion compensation due to inaccurate navigation. One interpretation of the results is that the final image formed using all 4 platforms' data collections is functionally equivalent to the image formed by a single platform traversing the 4 motion paths in sequence but collected in a quarter of the time.

As with the single platform image, because a single ideal point target is imaged at the origin, the resulting image represents the PSR of the SAR system using each navigation solution for motion compensation. The PSRs for each case using the 4 distributed platforms for imaging are given in Fig. 3.10. In this case, because each platform uses GPS in conjunction with the cooperative navigation scheme, both the non-cooperative and cooperative navigation PSRs are not translated significantly from the origin. However, it is clear from Fig. 3.10b that the phase correction from motion compensation is not coherent, and therefore the PSR is heavily degraded in azimuth. Moreover, as can be seen from one-dimensional cuts of the PSRs in Fig. 3.11, the



(a) (b)

Fig. 3.11: One-dimensional cuts of the simulated distributed platform PSR for ideal, non-cooperative, and cooperative motion compensations. The cross-track cut is shown in (a) and the along-track cut is shown in (b). From [1] ©2023 IEEE.

PSR in the non-ideal case has its peak value reduced by 2 dB from ideal as a result of energy spreading into the degraded sidelobes. In contrast, the cooperative navigation PSR very closely resembles the ideal, implying that the phase correction from motion compensation in this case enables the distributed SAR image to be formed coherently from the sub-images.

It should be noted that the dramatically improved PSR for the cooperative navigation case does not result from perfect correction of navigation error. Per the error plot in Fig. 3.6, the navigation error for the cooperative case when GPS is available to all platforms and using the same navigation equipment quality will still be ~0.5 m on average, which is large relative to the wavelength of $\lambda_c \approx 0.706$ m. However, an advantage of the cooperative navigation solution in this case is that the inter-node range measurements enable the navigation solutions for the distributed network to often maintain roughly correct relative positions and orientations of the platforms even if their absolute positions and orientations are slightly incorrect. In addition, even in cases where the relative navigation is not conducted well, it will usually be the case that the cooperative navigation solution has lower velocity errors and therefore, as in the case of the single-platform PSR, this will result in each individual platform con-

tributing a more ideal sub-image with lower sidelobes. Therefore, although the sub-image from each radar platform may contain phase errors after motion compensation, the sub-images from the other platforms will contain very similar phase errors due to their approximately equal motion compensation errors, and therefore the sub-images will still add coherently to give the correct PSR.

3.4 Cooperative Navigation with Relative Velocities

Although the above work considers only the range between platforms and direct GPS readings as measurements for the UKF, in practice relative velocities may be available as measurements too – this will be the case when the synchronization is implemented with navigation in Chapter 6. Therefore, it will be valuable to have some indication of how cooperative navigation is impacted by the presence of these measurements. First, assuming these measurements are available, the full measurement vector must be restructured to give

$$\mathbf{z}_k^{\text{nav}} = \begin{bmatrix} \mathbf{z}_k^{\text{GPS}} \\ \mathbf{z}_k^{R} \\ \mathbf{z}_k^{v} \end{bmatrix}, \tag{3.17}$$

where $\mathbf{z}_k^v$ describes a vector of relative velocities

$$\mathbf{z}_k^v = \begin{bmatrix} v_{1,2}^k & \dots & v_{1,N_p}^k & v_{2,3}^k & \dots & v_{N_p-1,N_p}^k. \end{bmatrix}^T \tag{3.18}$$

The scalar relative velocity may be computed by

$$v_{i,j}^k = \|\bar{v}_{i,j}^k\| \operatorname{sgn}\left(\left\langle \bar{p}_{i,k} - \bar{p}_{j,k}, \bar{v}_{i,j}^k \right\rangle\right), \tag{3.19}$$

where sgn(·) is the signum (or sign) operator used to ensure the relative velocity is correctly positive or negative, $\langle \cdot, \cdot \rangle$ indicates the dot product of two vectors, and $\bar{v}_{i,j}^{k}$ is the relative velocity vector given by

$$\bar{v}_{i,j}^{k} = \frac{\left\langle \bar{p}_{i,k} - \bar{p}_{j,k}, \bar{v}_{i,k} - \bar{v}_{j,k} \right\rangle}{\|\bar{p}_{i,k} - \bar{p}_{j,k}\|^2} \left(\bar{p}_{i,k} - \bar{p}_{j,k} \right). \tag{3.20}$$

The above expressions serve as the relative velocity component of the measurement function h_k.

This chapter explores a UKF-based approach to achieving cooperative navigation in a network of platforms using range measurements between the platforms. However, it does not provided a method for computing the range measurements from raw TOF signals, and does not consider the radar case in which the systems must be synchronized. In the next chapter, a technique is described for synchronization distributed networks of radar systems in time, phase, and frequency, which is accomplished in a manner such that range measurements for navigation are produced as a byproduct for cooperative navigation.

Chapter 4

Decentralized Digital Synchronization

This chapter presents the recent advances in digital decentralized synchronization of distributed radar networks. The proposed algorithm is an extension of [92] to enable frequency exchange and synchronization without an external frequency reference. The first component to be addressed is the procedure for estimating clock drift values using one of three different frequency exchange techniques. This is followed by a description of how to produce estimates of range values between platforms and estimate and compensate for clock biases and carrier phases. Simulated results are also provided to demonstrate the performance of the proposed technique. Next, the synchronization approach is analyzed in terms of its statistical performance in comparison to the derived CRLB. Practical considerations in solving for the different clock parameters using least-squares estimation are also provided. These analyses are verified through Monte Carlo simulations. Finally, two different radar scenarios are simulated to demonstrate the proposed technique in a practical radar network.

4.1 Estimation of Drift Values

The crux of the proposed synchronization algorithm is the ability of each platform to estimate its clock drift α_i and compensate for it both on transmit and receive. To

do so effectively, some synchronization waveform must be selected as $s_j(t)$. With a synchronization signal selected and transmitted between platforms, a mathematical framework for computing estimates of α_i for each platform must be established such that each platform i knows and can compensate for an accurate value of α_i. In this section, multiple synchronization waveforms and drift computation techniques are proposed with utility in different network scenarios. The output of the estimation for each technique includes estimates $\hat{\alpha}_i$ of α_i which may be used to compensate for clock drift in future signal transmissions; a discussion of how this is accomplished is provided in Section 4.2.1. The advantages and disadvantages of each technique and the suggested usage scenarios of each are discussed at the end of this section.

4.1.1 Single-Tone Synchronization with No Platform Motion

In the first synchronization technique, a single-tone waveform is selected as the baseband transmit waveform for each platform. Suppose that platform j broadcasts a single tone at a baseband frequency of f_j^α and a pulse length of T_p, which is received by all other platforms i. Define the single-tone baseband transmit signal ($s_j^\alpha(\tau_j)$) as

$$s_j^\alpha(\tau_j) = \exp\left(\mathrm{j}\,2\pi f_j^\alpha \tau_j\right) \mathrm{rect}\left(\frac{\tau_j - \frac{T_p}{2}}{T_p}\right), \tag{4.1}$$

where

$$\mathrm{rect}(t) \triangleq \begin{cases} 1 & |t| < \frac{1}{2} \\ 0 & \text{else} \end{cases} \tag{4.2}$$

is the rectangle function centered at $t = 0$ with a width of one. Inserting (4.1) into (2.70) yields the Doppler-free receive waveform ($r_{i,j}^{\alpha}(\tau_i)$) at platform i given by

$$\begin{aligned} r_{i,j}^{\alpha}(\tau_i) &= \exp\left(\mathrm{j}\,2\pi f_j^{\alpha}\left(\frac{\alpha_j}{\alpha_i}\tau_i - \rho\right)\right)\exp\left(-\mathrm{j}\,2\pi f^c\rho\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right) \\ &\quad\cdot\exp\left(\mathrm{j}\,2\pi f^c\left(\left(\frac{\alpha_j}{\alpha_i}-1\right)\tau_i\right)\right)\mathrm{rect}\left(\frac{\frac{\alpha_j}{\alpha_i}\tau_i - \rho - \frac{T_p}{2}}{T_p}\right) \\ &= \exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j}{\alpha_i}-1\right)+\frac{\alpha_j}{\alpha_i}f_j^{\alpha}\right)\tau_i\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi\left(f^c+f_j^{\alpha}\right)\rho\right)\mathrm{rect}\left(\frac{\frac{\alpha_j}{\alpha_i}\tau_i - \rho - \frac{T_p}{2}}{T_p}\right). \end{aligned} \tag{4.3}$$

The functional form of $r_{i,j}^{\alpha}(\tau_i)$ remains a windowed single-tone sinusoid, with the frequency scaled away from the ideal transmitted frequency f_j^{α} due to mismatch between α_i and α_j (note that in the case that $\alpha_i = \alpha_j$, the frequency of the tone remains f_j^{α}). Additional time and phase shifts are also applied as a result of the clock drift mismatch, clock biases, and propagation distance $R_{i,j}$. To estimate the frequency of the single tone as observed by platform i, the Fourier transform may be computed. Mathematically, the Fourier transform of (4.3) ($\mathcal{R}_{i,j}^{\alpha}(f)$) is given by

$$\begin{aligned} \mathcal{R}_{i,j}^{\alpha}(f) &= \frac{\alpha_i}{\alpha_j}T_p\,\mathrm{sinc}\left(\frac{\alpha_i}{\alpha_j}T_p\left(f - f_{i,j}^{\alpha}\right)\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi\left(f - f_{i,j}^{\alpha}\right)\left(\frac{\alpha_i}{\alpha_j}\rho + \frac{\alpha_i}{\alpha_j}\frac{T_p}{2}\right)\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi\left(f^c+f_j^{\alpha}\right)\rho\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right), \end{aligned} \tag{4.4}$$

where

$$\mathrm{sinc}(x) \triangleq \frac{\sin(\pi x)}{\pi x} \tag{4.5}$$

is the normalized sinc function and where $f_{i,j}^{\alpha}$ is defined as the true frequency of the single tone observed by platform i which is computed by

$$f_{i,j}^{\alpha} = \left(\frac{\alpha_j}{\alpha_i} - 1\right) f^c + \frac{\alpha_j}{\alpha_i} f_j^{\alpha}. \tag{4.6}$$

The maximum magnitude of the Fourier transform occurs at the peak of the sinc function in (4.4), which is located at the value of $f_{i,j}^{\alpha}$ observed by platform i. Therefore, an estimate $\hat{f}_{i,j}^{\alpha}$ of $f_{i,j}^{\alpha}$ may be computed by determining the value of f which maximizes $|\mathcal{R}_{i,j}^{\alpha}(f)|$. The method for determining the location of the maximum value is discussed in more detail in Section 4.1.4.

After all platforms have broadcast their synchronization signals, platform i may produce an estimate $\hat{f}_{i,j}^{\alpha}$ of $f_{i,j}^{\alpha}$ for all transmitting platforms $j \neq i$. By rearranging (4.6), a linear equation in terms of α_i and α_j can be constructed as

$$\left(\hat{f}_{i,j}^{\alpha} + f^c\right) \alpha_i - \left(f_j^{\alpha} + f^c\right) \alpha_j = 0. \tag{4.7}$$

Furthermore, it will be impossible to determine the absolute truth values of α_i since no measurements can be made which are not biased by these unknown values. However, the values of α_i may be determined relative to one another, which will be sufficient to ensure coherent operation among the platforms. It is sensible, therefore, to select one drift value to be set to unity. For example,

$$\alpha_1 = 1. \tag{4.8}$$

Since $j \neq i$ in (4.7), there are $N_p - 1$ linearly independent equations based on information available to platform i. When combined with (4.8), N_p linearly independent

equations are available estimating the N_p values of α_j. Therefore, the estimates $\hat{\alpha}_j$ of α_j for all j may be solved independently on each platform using only its own measurements of the other platforms' transmitted single-tone signals by solving the system of N_p equations via a matrix inversion. The resulting estimates $\hat{\alpha}_j$ may be compensated for in future transmitted and received signals. This simple technique will be referred to as *single-tone without exchange*. In a practical implementation, assuming a large order of magnitude for f^c, (4.8) may need to be scaled by f^c to eliminate numerical errors, such as

$$f^c \alpha_1 = f^c. \tag{4.9}$$

At the cost of increased complexity and additional signal transmissions compared to the single-tone without exchange, a more robust drift estimation may be implemented. After the first round of single-tone signals have been broadcast and received by all other platforms, the estimated values $\hat{f}^{\alpha}_{i,j}$ should be exchanged among all other platforms using an arbitrary communication scheme such that all platforms i have knowledge of $\hat{f}^{\alpha}_{k,j}$ for all values of k and j with $k \neq j$. As a result, in addition to (4.8), each platform may use a linear least squares estimation technique to solve the system of $N_p^2 - N_p + 1$ equations and obtain values of $\hat{\alpha}_j$. A thorough description and analysis of the procedure for solving this system of equations is provided in Section 4.4. This more complex and robust procedure will be referred to as *single-tone with exchange*. Although this method will be more complex and require additional time and communications overhead to complete, it will be more robust against noise and Doppler shifts in the moving platform case.

4.1.2 Single-Tone Synchronization with Platform Motion

As with the case of the single-tone waveform with no platform motion, the baseband transmitted waveform $s_j^{\alpha}(\tau_j)$ is given by (4.1). To determine the received signal ($r_{i,j}^{\alpha,d}(\tau_i)$) at platform i assuming a non-zero LOS relative velocity between the platforms, (4.1) is substituted into (2.75) in similar fashion to (4.3) to obtain

$$\begin{aligned} r_{i,j}^{\alpha,d}(\tau_i) = \exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j}{\alpha_i}-1\right)+\frac{\alpha_j}{\alpha_i}f_j^{\alpha}+\frac{f_{i,j}^d}{\alpha_i}\right)\tau_i\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right) \\ \cdot\exp\left(-\mathrm{j}\,2\pi\left(\left(f^c+f_j^{\alpha}\right)\rho+f_{i,j}^d\frac{\phi_i}{\alpha_i}\right)\right)\mathrm{rect}\left(\frac{\frac{\alpha_j}{\alpha_i}\tau_i-\rho-\frac{T_p}{2}}{T_p}\right). \end{aligned} \tag{4.10}$$

As with the case with no Doppler, the functional form of the received signal remains a single-tone waveform but with the frequency shifted based on clock drift-induced mismatches as well as a shift in frequency related to the relative motion-induced Doppler shift. In the same fashion as the case with no Doppler shift, the Fourier transform can be computed to determine the frequency of the single tone observed by platform i. The Fourier transform ($\mathcal{R}_{i,j}^{\alpha,d}(f)$) is computed by

$$\begin{aligned} \mathcal{R}_{i,j}^{\alpha,d}(f) = \frac{\alpha_i}{\alpha_j}T_p\,\mathrm{sinc}\left(\frac{\alpha_i}{\alpha_j}T_p\left(f-f_{i,j}^{\alpha,d}\right)\right) \\ \cdot\exp\left(-\mathrm{j}\,2\pi\left(f-f_{i,j}^{\alpha,d}\right)\left(\frac{\alpha_i}{\alpha_j}\rho+\frac{\alpha_i}{\alpha_j}\frac{T_p}{2}\right)\right) \\ \cdot\exp\left(-\mathrm{j}\,2\pi\left(\left(f^c+f_j^{\alpha}\right)\rho+f_{i,j}^d\frac{\phi_i}{\alpha_i}\right)\right)\exp\left(\mathrm{j}\,\gamma_{i,j}^{\mathrm{err}}\right), \end{aligned} \tag{4.11}$$

where the variable $f_{i,j}^{\alpha,d}$ describes the true single-tone frequency observed by platform i, similar to (4.6), but including the Doppler shift frequency, and is computed by

$$f_{i,j}^{\alpha,d} = \left(\frac{\alpha_j}{\alpha_i} - 1\right) f^c + \frac{\alpha_j}{\alpha_i} f_j^{\alpha} + \frac{f_{i,j}^d}{\alpha_i}. \tag{4.12}$$

As with the stationary platform case, after all platforms $j \neq i$ have broadcast their synchronization signals, platform i can produce an estimate $\hat{f}_{i,j}^{\alpha,d}$ of $f_{i,j}^{\alpha,d}$ for all transmitting platforms $j \neq i$. Because the estimate of the peak value of $|\mathcal{R}_{i,j}^{\alpha,d}(f)|$ does not depend on $R_{i,j}$, the fact that in general $R_{i,j} \neq R_{i,j}$ will not impede this estimation. As with (4.7), (4.12) may be rearranged to form a linear equation in terms of α_i, α_j, and $f_{i,j}^d$ by

$$\left(\hat{f}_{i,j}^{\alpha,d} + f^c\right)\alpha_i - \left(f_j^{\alpha} + f^c\right)\alpha_j - f_{i,j}^d = 0. \tag{4.13}$$

Additionally, one value of α_i should be set to unity as in (4.8). The result is that platform i again has information to form N_p linearly independent equations for performing the single-tone without exchange estimation. However, the addition of the unknown values of $f_{i,j}^d$ with $j \neq i$ introduces $N_p - 1$ unknowns, giving a total of $2N_p - 1$ values to be estimated. As a result, platform i cannot independently estimate the drift values and Doppler shifts. Furthermore, attempting to estimate the drifts without accounting for the Doppler shift frequencies will result in large estimation errors.

Next, consider the case in which each platform broadcasts all of its estimates of $\hat{f}_{i,j}^{\alpha,d}$ for all $j \neq i$ to all other platforms such that each platform i has full knowledge of all values of $\hat{f}_{k,j}^{\alpha,d}$ for all $k \neq j$, which is the single-tone with exchange estimation. As above, in this case, each platform now has enough knowledge to construct a system of $N_p^2 - N_p + 1$ equations. If the assumption is made that the relative velocities between platforms i and j are equivalent when each platform is transmitting (i.e. $v_{i,j} = v_{j,i}$ and

$f^d_{i,j} = f^d_{j,i}$), then between the drift values and the Doppler shift values there are a total of $(N_p^2 + N_p)/2$ unknowns to be estimated. So long as $N_p \geq 2$, the system is fully determined or overdetermined and thus linear least squares estimation may be used by each platform independently to solve for the values of α_j for all j and $f^d_{k,j}$ for all k and j with $j \neq k$. Thus, each platform has an estimate of its own drift value $\hat{\alpha}_i$ which may be compensated for both on transmit and receive. Additionally, each platform will have estimates $\hat{f}^d_{i,j}$ of the Doppler frequencies, which in turn provide relative velocity information ($\hat{v}_{i,j}$) which may be extracted by

$$\hat{v}_{i,j} = -\frac{c\hat{f}^d_{i,j}}{\hat{\alpha}_j f^c}. \tag{4.14}$$

4.1.3 Two-Tone Synchronization

As demonstrated in the previous subsections, a major disadvantage of the single-tone waveform is that in order for drifts to be satisfactorily estimated when Doppler shifts are expected, a two-way transfer of information is required which increases the algorithm's complexity and the communication overhead. To address the case in which it is desirable for each platform to estimate drift after only a single round of synchronization broadcasts even in the presence of relative motion, a two-tone synchronization waveform may be used.

Recall the definition of the single-tone waveform $s^\alpha_j(\tau_j)$ given in (4.1). Suppose that another single-tone waveform is defined in a similar fashion but with a new frequency of f^β_j; let this waveform be called $s^\beta_j(\tau_j)$. The two-tone waveform ($s^{\text{tt}}_j(\tau_j)$) is then given by the sum of the single-tone waveforms

$$s^{\text{tt}}_j(\tau_j) = s^\alpha_j(\tau_j) + s^\beta_j(\tau_j). \tag{4.15}$$

Let $r_{i,j}^{\beta,d}(\tau_i)$ be defined similarly to $r_{i,j}^{\alpha,d}(\tau_i)$ in (4.10) for the second tone frequency f_j^{β}. Because the channel is linear, the two-tone signal received by platform i ($r_{i,j}^{\mathrm{tt},d}(\tau_i)$) will simply be the sum of the received signals of the individual tones

$$r_{i,j}^{\mathrm{tt},d}(\tau_i) = r_{i,j}^{\alpha,d}(\tau_i) + r_{i,j}^{\beta,d}(\tau_i). \tag{4.16}$$

Denote the Fourier transform of $r_{i,j}^{\beta,d}(\tau_i)$ as $\mathcal{R}_{i,j}^{\beta,d}(f)$, the computation of which will be identical to (4.11) but with the second tone frequency f_j^{β} replacing f_j^{α}. Thus, the Fourier transform of the two-tone waveform as observed by platform i ($\mathcal{R}_{i,j}^{\mathrm{tt},d}(f)$) is given by the sum of the two individual single-tone Fourier transforms

$$\mathcal{R}_{i,j}^{\mathrm{tt},d}(f) = \mathcal{R}_{i,j}^{\alpha,d}(f) + \mathcal{R}_{i,j}^{\beta,d}(f). \tag{4.17}$$

Based on the sinc function form of the single-tone Fourier transform in (4.11), the two-tone Fourier transform here will be the sum of two sinc functions of nominally equivalent power and main-lobe width. Assuming the two-tone frequencies are selected appropriately such that the main-lobes of the sinc functions do not overlap, the magnitude $|\mathcal{R}_{i,j}^{\mathrm{tt},d}(f)|$ will have two primary maximum values at $f = f_{i,j}^{\alpha,d}$ and $f = f_{i,j}^{\beta,d}$, where $f_{i,j}^{\beta,d}$ is defined similarly to (4.12) as

$$f_{i,j}^{\beta,d} = \left(\frac{\alpha_j}{\alpha_i} - 1\right) f^c + \frac{\alpha_j}{\alpha_i} f_j^{\beta} + \frac{f_{i,j}^d}{\alpha_i}. \tag{4.18}$$

In similar fashion to the single-tone waveforms, estimates $\hat{f}_{i,j}^{\alpha,d}$ and $\hat{f}_{i,j}^{\beta,d}$ of $f_{i,j}^{\alpha,d}$ and $f_{i,j}^{\beta,d}$, respectively, can be produced by locating the two largest peaks in the magnitude of $\mathcal{R}_{i,j}^{\mathrm{tt},d}(f)$. As a result, for each value of j with $j \neq i$, two linearly independent

equations are produced, given by

$$\begin{aligned}\left(\hat{f}_{i,j}^{\alpha,d} + f^c\right)\alpha_i - \left(f_j^{\alpha} + f^c\right)\alpha_j - f_{i,j}^{d} = 0 \\ \left(\hat{f}_{i,j}^{\beta,d} + f^c\right)\alpha_i - \left(f_j^{\beta} + f^c\right)\alpha_j - f_{i,j}^{d} = 0.\end{aligned} \tag{4.19}$$

For all j with $j \neq i$, this produces $2N_p - 2$ independent equations; when one drift value is set to unity as in (4.8), there are $2N_p - 1$ equations, which exactly matches the number of unknowns between the drift values and the Doppler shifts between platform i and all other platforms j with $j \neq i$. Thus, estimates $\hat{\alpha}_j$ may be produced for all j including $j = i$, and estimates $\hat{f}_{i,j}^{d}$ may be produced for all j with $j \neq i$, through a simple inversion of the matrix describing the full linear system of equations. This procedure will be referred to as the *two-tone* estimation procedure.

In the case that the two tones are far enough apart that each tone may experience a noticeably different Doppler shift (e.g. ultra-wideband systems), (4.19) may be formulated using a modified version of (2.73) to directly estimate the velocities by the approximation

$$\begin{aligned}\left(\hat{f}_{i,j}^{\alpha,d} + f^c\right)\alpha_i - \left(f_j^{\alpha} + f^c\right)\alpha_j + \left(\frac{f_j^{\alpha} + f^c}{c}\right)v_{i,j} = 0 \\ \left(\hat{f}_{i,j}^{\beta,d} + f^c\right)\alpha_i - \left(f_j^{\beta} + f^c\right)\alpha_j + \left(\frac{f_j^{\beta} + f^c}{c}\right)v_{i,j} = 0.\end{aligned} \tag{4.20}$$

Note that in order to maintain linearity, the above pair of linear equations approximates the velocity term by ignoring the impact of α_j on the carrier term in (2.73).

4.1.4 Estimation of the Tone Frequency Using Sinc NL-LS

The sinc NL-LS peak estimation algorithm described in [92] utilizes the Gauss-Newton nonlinear least squares optimization based on a sinc kernel. The idea is to estimate the true location of the peak of the sinc function (along with the peak value and the time-scaling parameter) from a set of three equally-spaced samples located around the true peak value. The following is a description of the sinc NL-LS algorithm with some minor modifications made to adapt it to the frequency-domain application.

The sinc NL-LS algorithm is designed to operate on data forming a real-valued sinc function. As such, the magnitude of the Fourier transform must be taken, which for the stationary platform case yields

$$\left|\mathcal{R}_{i,j}^{\alpha}(f)\right| = \left|\frac{\alpha_i}{\alpha_j} T_p \operatorname{sinc}\left(\frac{\alpha_i}{\alpha_j} T_p \left(f - f_{i,j}^{\alpha}\right)\right)\right|. \tag{4.21}$$

Practically, this Fourier transform will be computed by a discrete Fourier transform (DFT) of a discretely-sampled sequence of time-domain data after analog-to-digital conversion. This is usually done through the fast Fourier transform (FFT) algorithm. The DFT will have the discrete form

$$\left|\mathcal{R}_{i,j}^{\alpha}[\mathrm{k}]\right| = \left|\frac{\alpha_i}{\alpha_j} T_p \operatorname{sinc}\left(\frac{\alpha_i}{\alpha_j} \frac{T_p}{T_c} \left(\mathrm{k} - T_c f_{i,j}^{\alpha}\right)\right)\right|, \tag{4.22}$$

where T_c is the capture length over which samples are taken, and the roman character k is used as the frequency index of the DFT to differentiate it from the Kalman filter index k. A similar result will be obtained for the moving platform case with $\mathcal{R}_{i,j}^{\alpha,d}(f)$ and $\mathcal{R}_{i,j}^{\alpha,d}[\mathrm{k}]$. For the two-tone case, the single-tone frequency values must each be measured separately by two independent executions of the sinc NL-LS algorithm. For

each execution, a band-pass filter should be applied centered around the tone whose frequency is to be estimated.

First, find the index $\mathrm{k_{pk}}$ associated with the peak value in the sequence for $\left|\mathcal{R}_{i,j}[\mathrm{k}]\right|$

$$\mathrm{k_{pk}} = \arg\max_{\mathrm{k}} \left|\mathcal{R}_{i,j}[\mathrm{k}]\right|. \tag{4.23}$$

The sinc function kernel $f_{\mathrm{sinc}}(\mathbf{x};\boldsymbol{\lambda})$ is parameterized by

$$f_{\mathrm{sinc}}(\mathbf{x};\boldsymbol{\lambda}) = \lambda_0 \mathrm{sinc}\left((\mathbf{x}-\lambda_1)\,\lambda_2\right) \tag{4.24}$$

with

$$\begin{aligned} \mathbf{x} &= [-1 \quad 0 \quad 1]^T \\ \boldsymbol{\lambda} &= [\lambda_0 \quad \lambda_1 \quad \lambda_2]^T. \end{aligned} \tag{4.25}$$

Note that (4.22) matches this kernel form. In this parameterization, λ_0 represents the peak value of the sinc function, λ_1 represents the frequency value of the peak, and λ_2 represents the width-scaling of the sinc function. The vector $\mathbf{x}$ is normalized to unit samples. A vector $\mathbf{y}$ is defined as the three samples nearest to the peak of $\left|\mathcal{R}_{i,j}[k]\right|$

$$\mathbf{y} = \left[\left|\mathcal{R}_{i,j}[\mathrm{k_{pk}}-1]\right| \quad \left|\mathcal{R}_{i,j}[\mathrm{k_{pk}}]\right| \quad \left|\mathcal{R}_{i,j}[\mathrm{k_{pk}}+1]\right|\right]. \tag{4.26}$$

The goal of sinc NL-LS is to determine the values of $\boldsymbol{\lambda}$ that most closely match the three points in $\mathbf{y}$ by minimizing the cost function

$$S = \sum_{i=0}^{2} \left(y_i - f(x_i;\boldsymbol{\lambda})\right)^2. \tag{4.27}$$

The gradients of the sinc kernel with respect to the given parameters in $\boldsymbol{\lambda}$ are given by

$$\begin{aligned}
\frac{\partial f_{\text{sinc}}(\mathbf{x};\boldsymbol{\lambda})}{\partial \lambda_0} &= \text{sinc}(\lambda_2(\mathbf{x}-\lambda_1)) \\
\frac{\partial f_{\text{sinc}}(\mathbf{x};\boldsymbol{\lambda})}{\partial \lambda_1} &= \frac{\lambda_0\left[\text{sinc}\left(\lambda_2\left(\mathbf{x}-\lambda_1\right)\right)-\cos\left(\pi\lambda_2\left(\mathbf{x}-\lambda_1\right)\right)\right]}{\mathbf{x}-\lambda_1} \\
\frac{\partial f_{\text{sinc}}(\mathbf{x};\boldsymbol{\lambda})}{\partial \lambda_2} &= \frac{\lambda_0\left[\cos\left(\pi\lambda_2\left(\mathbf{x}-\lambda_1\right)\right)-\text{sinc}\left(\lambda_2\left(\mathbf{x}-\lambda_1\right)\right)\right]}{\lambda_2}.
\end{aligned} \tag{4.28}$$

We begin by initializing

$$\boldsymbol{\lambda}_0 = \begin{bmatrix} \left|\mathcal{R}_{i,j}[\text{k}_{\text{pk}}]\right| & 0 & \frac{T_p}{T_c} \end{bmatrix}^T. \tag{4.29}$$

This initialization reflects the idea that the best initial estimate for the sinc peak is the observed peak value, the best initial estimate for the time shift is 0 (implying that the observed peak sample is the true peak), and the best initial estimate for the sinc argument scaling factor is T_p/T_c, which, assuming the values of α_i and α_j are close to 1, is very close to the true scaling factor in (4.22). Note that for the computation of the gradient with respect to λ_1, the expression has indeterminate form when $x_i - \lambda_1 = 0$, which is guaranteed on the first iteration since $x_1 = 0$ and the initial value of $\lambda_1 = 0$. Using L'Hôpital's rule it can be shown that in this indeterminate case the gradient resolves to zero.

The algorithm is then performed iteratively. On the n^{th} iteration, the matrix Jacobian J is computed by

$$J = \begin{bmatrix} \frac{\partial f(\mathbf{x};\boldsymbol{\lambda}_n)}{\partial \lambda_0} & \frac{\partial f(\mathbf{x};\boldsymbol{\lambda}_n)}{\partial \lambda_1} & \frac{\partial f(\mathbf{x};\boldsymbol{\lambda}_n)}{\partial \lambda_2} \end{bmatrix} \tag{4.30}$$

and the residual error $\Delta\mathbf{y}$ is computed by

$$\Delta\mathbf{y} = \mathbf{y} - f(\mathbf{x};\boldsymbol{\lambda}_n). \tag{4.31}$$

Assuming that three points are used for $\mathbf{x}$ and $\mathbf{y}$, the update vector $\Delta\boldsymbol{\lambda}$ may be computed by

$$\Delta\boldsymbol{\lambda} = J^{-1}\Delta\mathbf{y}. \tag{4.32}$$

If more than three points are used, then J will not be square and $\Delta\boldsymbol{\lambda}$ must be computed by

$$\Delta\boldsymbol{\lambda} = (J^T J)^{-1} J^T \Delta\mathbf{y}. \tag{4.33}$$

With $\Delta\boldsymbol{\lambda}$ computed, the next parameter values may be computed by

$$\boldsymbol{\lambda}_{n+1} = \boldsymbol{\lambda}_n + \Delta\boldsymbol{\lambda}. \tag{4.34}$$

Once convergence is achieved, the final estimated parameters are given in $\tilde{\boldsymbol{\lambda}}$, and the true frequency peak estimate is computed by

$$\hat{f}_{i,j}^{\alpha} = f[\mathrm{k_{pk}}] + \frac{\tilde{\lambda}_1}{T_c} \tag{4.35}$$

where $f[\mathrm{k_{pk}}]$ denotes the frequency value associated with the peak sample $\mathrm{k_{pk}}$. A processing diagram describing the steps to compute the frequency estimates $\hat{f}_{i,j}^{\alpha}$ for the single-tone case is shown in Fig. 4.1, which can be extended to the two-tone case in which two sinc functions will be present in the frequency domain.

4.1.5 Selection of Frequencies Single- and Two-Tone Waveforms

The selection of the tones used in the frequency transfer will depend on the bandwidth of the systems in use and what sort of orthogonality between the broadcasts is desired – time or frequency. If a time-division multiplexing scheme is utilized (i.e. each platform has a prescribed time slot for their frequency broadcast), then it makes

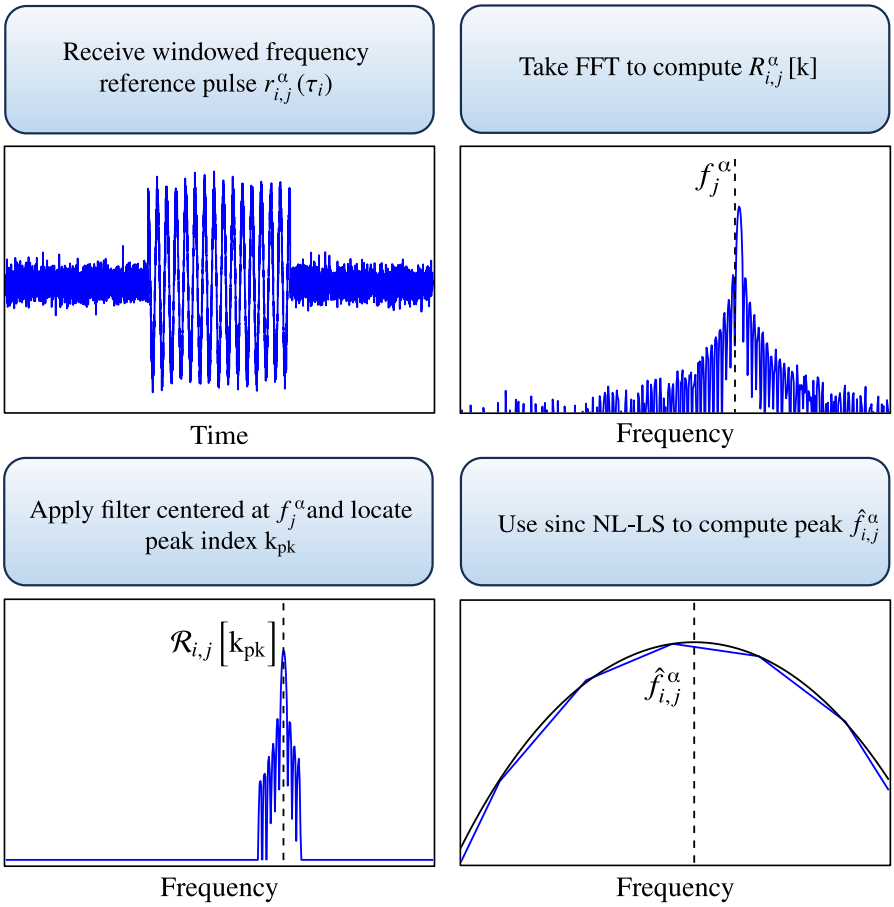


Fig. 4.1: A process diagram for computing the frequency estimates $\hat{f}_{i,j}^{\alpha}$. From [4] ©2024 IEEE.

the most sense for $f_j^{\alpha} = 0$ for all j. That is, each platform directly broadcasts at its carrier frequency.

However, it is also possible for a frequency-division multiplexing scheme to be employed, enabling all platforms to broadcast simultaneously. The caveat is that due to the unknown drift values between platforms, the selected tone frequencies must be spaced out enough such that the carrier frequency error cannot skew them enough to overlap with the tone frequency allocated to another platform. To do so requires an understanding of the quality of the MO. For instance, suppose it is known that each platform is equipped with an MO whose frequency stability is ±10 parts per million

(ppm), and it is equipped on all platforms in a network with a carrier frequency of $f^c = 10$ GHz. Further, suppose that platform 1 has no oscillator drift ($\alpha_1 = 1$), platform 2 has +10 ppm drift ($\alpha_2 = 1.00001$), and platform 3 has −10 ppm drift ($\alpha_3 = 0.99999$). Further, suppose that $f_2^\alpha = -100$ kHz and $f_3^\alpha = 100$ kHz. Using (4.6) for stationary platforms, it can be shown that due to the carrier drift, the two tones overlap at $f_{1,2}^\alpha = f_{1,3}^\alpha \approx 0$ when received by platform 1. Thus, in this example network, values of f_j^α should be spaced out by more than 200 kHz to avoid overlap. Additional guard spacing should also be included based on the expected severity of the Doppler shift.

In the two-tone waveform, the two tones should be spaced out enough such that their main lobes do not overlap. The main lobe width is set by the pulse length T_p, and the distance between the two nulls on either side of the sinc main lobe is $2/T_p$. Therefore, the two tones should be separated by at least $2/T_p$, though additional spacing may be desirable to avoid unwanted interference due to sidelobes. If frequency-domain multiplexing of the two-tone waveforms is desired for simultaneous transmit, then it is possible to interleave the two-tone waveforms from different platforms so long as the tone separation is wide enough and the spacing rules for the single-tone waveforms are obeyed.

4.1.6 Comparison of Techniques

The first proposed technique in Section 4.1.1, single-tone without exchange, is by far the simplest. It requires only the first round of broadcasts of the single-tone synchronization signals, so it eliminates the communication overhead required of transmitting the frequency estimates among the other platforms after this first round. Because only one tone is required for each platform, it is bandwidth efficient. However,

as discussed in Section 4.1.2, it has no way of accounting for Doppler shift frequencies in the presence of relative motion and will not work at all unless the network is stationary. Moreover, as can be seen in Section 4.3, this technique will be more susceptible to SNR, and if high synchronization performance is required with lower SNR then it will possibly be better to implement the more complex sharing of frequency estimates among the full network.

The second proposed technique in Section 4.1.1 and explored in more detail in Section 4.1.2, single-tone with exchange, is more complex but more robust than without exchange. It is more robust to noise due to the overdetermined nature of the system of equations and may be modified to enable direct estimation and removal of the Doppler shift frequencies, which not only nullifies their impact on the estimation of the drift values but also allows them to be used in relative navigation applications. Furthermore, because only single tones are required, it will more efficiently utilize the available bandwidth. However, the additional round of communications required to share the frequency estimates doubles the communication overhead required for frequency synchronization. This technique is recommended in applications where the system must be robust to low SNR and certainly in applications in which large Doppler shifts are expected. It will still hold an advantage over the two-tone procedure when the bandwidth of the RF subsystems is limited and frequency multiplexing is desired. However, if the additional communications overhead is unacceptable, then the two-tone procedure provides a workable alternative.

Finally, the two-tone estimation procedure proposed in Section 4.1.3 provides the ability to compensate for Doppler without requiring an additional round of communication broadcasts. However, it has the drawback of requiring double or more than double the bandwidth per broadcast. Moreover, as shown in Section 4.3, the two-tone

procedure has significantly worse performance than the single-tone with exchange technique and requires large tone separation and a long pulse length to produce sufficiently high-quality estimates of clock drifts and Doppler shifts, even with moderate SNR. This technique is recommended in applications where Doppler shifts are expected but when an additional round of communications for exchanging the frequency estimates is not acceptable.

4.2 Compensating for Clock Drifts and Biases

In the previous section, multiple techniques for producing estimates $\hat{\alpha}_i$ of the clock drifts on each platform in the network. While previous work has addressed how to compensate for the drifts once estimates are available [3], it is presented here as well for completeness. Moreover, methods described in [92] for compensating clock bias and carrier phase in a priori frequency synchronized networks are adapted here for post-synchronization estimation and compensation of the residual clock biases and carrier phases.

4.2.1 Compensating for Clock Drift

After the drift estimation procedure from Section 4.1 is completed, each platform i will have an estimate of its own clock drift $\hat{\alpha}_i$. Using the estimates, each platform can modify its transmitted and received signals and properly scale the time axes of each to bring its signals in line with those of all other platforms and enabling frequency-synchronized operation.

It is assumed here that while the Doppler shift from moving platforms will impact frequency synchronization, these shifts will not be large enough to impact the pulse

compression waveforms required for time and phase synchronization and as such they can be ignored here. Beginning with the global time signal transmitted by platform j in (2.64), it can be seen that the transmitted signal has its baseband waveform time axis dilated based on the value of α_j and the carrier frequency is shifted in a similar way. With a more accurate understanding of the digital-to-analog sampling frequency, the baseband waveform may be recomputed by dividing the time axis by $\hat{\alpha}_j$. Moreover, the baseband waveform should be multiplied by a corrective sinusoid to ensure the transmit waveform is truly at the carrier frequency. Define this new baseband waveform ($s'_j(\tau_j)$) by

$$s'_j(\tau_j) = s_j\left(\frac{\tau_j}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\frac{1}{\hat{\alpha}_j}-1\right)\tau_j\right). \tag{4.36}$$

This new baseband waveform is simply the result of stretching or compressing the time axis of the original baseband waveform and applying a modest frequency shift. The transmitted signal after up-conversion ($w'_j(\tau_j)$) is given by

$$\begin{aligned} w'_j(\tau_j) &= s'_j(\tau_j)\exp\left(\mathrm{j}\,2\pi f^c\tau_j\right)\exp\left(\mathrm{j}\,\gamma_j^{\mathrm{tx}}\right) \\ &= s_j\left(\frac{\tau_j}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\frac{1}{\hat{\alpha}_j}\right)\tau_j\right)\exp\left(\mathrm{j}\,\gamma_j^{\mathrm{tx}}\right). \end{aligned} \tag{4.37}$$

Substituting (2.61) into (4.37) above produces the global time transmit waveform

$$w'_j(t) = s_j\left(\frac{\alpha_j}{\hat{\alpha}_j}t+\frac{\phi_j}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi f^c\frac{\alpha_j}{\hat{\alpha}_j}t\right)\exp\left(\mathrm{j}\,2\pi f^c\frac{\phi_i}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,\gamma_j^{\mathrm{tx}}\right). \tag{4.38}$$

If $\hat{\alpha}_j = \alpha_j$, (4.38) can be simplified to

$$w'_j(t) = s_j\left(t+\frac{\phi_j}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi f^c t\right)\exp\left(\mathrm{j}\,2\pi f^c\frac{\phi_i}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,\gamma_j^{\mathrm{tx}}\right), \tag{4.39}$$

whereby the baseband waveform is correctly scaled in global time and the actual carrier frequency is equal to the nominal carrier frequency. In reality, because the drift estimates cannot be computed absolutely in the global time frame but are instead estimated relative to one another, this simplification cannot be exactly justified. However, this exercise demonstrates that this method of compensating $\hat{\alpha}_j$ will ensure transmit coherence in some relative frame with the other platforms. Save some relative estimation errors, the platforms will be frequency synchronized with one another even though they may not be perfectly synchronized in the global frame.

To compute the transmit compensated signal received by platform i in its local clock domain ($r'_{i,j}(\tau_i)$), (4.36) is inserted into (2.70) to produce

$$\begin{aligned} r'_{i,j}(\tau_i) &= s'_j\left(\frac{\alpha_j}{\alpha_i}\tau_i - \rho\right)\exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j}{\alpha_i}-1\right)+\frac{f^d_{i,j}}{\alpha_i}\right)\tau_i\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^c\rho\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right) \\ &= s_j\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\tau_i - \frac{\rho}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}-1\right)+\frac{f^d_{i,j}}{\alpha_i}\right)\tau_i\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^c\rho\frac{1}{\hat{\alpha}_j}\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right). \end{aligned} \tag{4.40}$$

Recall that ideally, the received waveform will be a copy of the original transmitted waveform $s_j(t)$ with no time dilation or carrier frequency distortion. However, both effects are clearly present in (4.40). To remedy this, the fact that the sampling clock on the receiver is multiplied by α_i may be addressed by defining a new time axis (τ_i) for the signal defined by

$$\tau'_i = \frac{\tau_i}{\hat{\alpha}_i}, \tag{4.41}$$

which is essentially equivalent to acknowledging the true sampling frequency as be-

ing different from the nominal sampling frequency. Substituting this new time axis into (4.40) yields

$$\begin{aligned} r'_{i,j}(\tau'_i) =& s_j\left(\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i}\tau'_i - \frac{\rho}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i} - \hat{\alpha}_i\right) + \frac{\hat{\alpha}_i}{\alpha_i}f^d_{i,j}\right)\tau'_i\right) \\ &\cdot \exp\left(-\mathrm{j}\,2\pi f^c\rho\frac{1}{\hat{\alpha}_j}\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right). \end{aligned} \tag{4.42}$$

As mentioned above, the estimated values of $\hat{\alpha}_i$ will likely never be equivalent to the true value of α_i. This is due to the assumption that one of the drift values (e.g. α_1) is equal to unity, which is likely untrue in the global time frame. However, assuming perfect noiseless estimation of $\hat{\alpha}_i$ for all i, the ratio of $\hat{\alpha}_i$ to α_i will be equivalent to the ratio of $\hat{\alpha}_1$ to α_1. That is,

$$\frac{\hat{\alpha}_i}{\alpha_i} = \frac{\hat{\alpha}_1}{\alpha_1} \Longrightarrow \frac{\alpha_i}{\hat{\alpha}_i} = \frac{\alpha_j}{\hat{\alpha}_j} \tag{4.43}$$

which results in

$$\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i} = 1 \tag{4.44}$$

and

$$\frac{\alpha_j}{\hat{\alpha}_j\alpha_i} = \frac{1}{\hat{\alpha}_i}. \tag{4.45}$$

By substituting in (4.44), the compensated received signal in (4.42) may be simplified to

$$\begin{aligned} r'_{i,j}(\tau'_i) =& s_j\left(\tau'_i - \frac{\rho}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi\left(f^c\left(1-\hat{\alpha}_i\right) + \frac{\hat{\alpha}_i}{\alpha_i}f^d_{i,j}\right)\tau'_i\right) \\ &\cdot \exp\left(-\mathrm{j}\,2\pi f^c\rho\frac{1}{\hat{\alpha}_j}\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right). \end{aligned} \tag{4.46}$$

The resulting signal has the time dilation effect removed, with only a time and phase shift remaining as well as a residual carrier frequency shift. Define a carrier compen-

sation signal ($b_i(\tau_i')$) where

$$b_i(\tau_i') = \exp\left(\mathrm{j}\,2\pi f^c\left(\hat{\alpha}_i - 1\right)\tau_i'\right). \tag{4.47}$$

In digital processing, the received signal should be multiplied by $b_i(\tau_i')$ to yield the final compensated signal ($r^b_{i,j}(\tau_i')$) given by

$$\begin{aligned} r^b_{i,j}(\tau_i') &= r'_{i,j}(\tau_i')b(\tau_i') \\ &= s_j\left(\tau_i' - \frac{\rho}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi\frac{\hat{\alpha}_i}{\alpha_i}f^d_{i,j}\tau_i'\right) \\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^c\rho\frac{1}{\hat{\alpha}_j}\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\,\gamma^{\mathrm{err}}_{i,j}\right). \end{aligned} \tag{4.48}$$

This final correction yields the desired signal transmitted by platform j with only a time delay and phase shift and all adverse time-scaling and frequency offsets eliminated. Note that for cases with non-zero relative platform motion, the impacts of the Doppler shift frequency and a small Doppler phase due to receiver clock bias are still present in the signal. Further, it should be noted that the tacit assumption is made in (4.48) that $R_{i,j}$ is the range between platforms i and j at $\tau_i' = 0$. Therefore, the *initial* phase of the signal will be set based on $R_{i,j}$ and the other components of ρ, while the Doppler shift will alter the signal phase over time. Alternately, the Doppler shift term could be replaced by reformulating the expression to include a time-varying range term, which is fundamentally the source of the Doppler shift. In later formulations in which $R_{i,j} \neq R_{j,i}$, it will be assumed that the Doppler shift phase due to the change in time is accounted for by the inequality of these ranges, and thus, the Doppler shift term will be ignored for the purpose of computing phase and range values.

4.2.2 Compensating for Clock Bias and Carrier Phases

To achieve full synchronization in time, phase, and frequency, the remaining unknown phase and time shifts resulting from the clock biases and RF transmitter and receiver phase shifts must be corrected. A procedure for doing this is presented in [92]. In this work, it is assumed that the systems have already been synchronized in frequency by locking to some global frequency reference (i.e. that $\alpha_i = 1$). Therefore, it can be applied to the estimation of the remaining biases and carrier phases after drift compensation with some modifications (detailed below) to account for the globally incorrect drift estimates and the potential for $R_{i,j} \neq R_{j,i}$ due to relative motion between signal broadcasts which is not considered in [92].

As described in [92], this process begins with each platform broadcasting some pulse-compression waveform (such as a linear frequency-modulated (LFM) pulse) which can be used for time-delay estimation. Each receiving platform i computes the cross-correlation of the received signal $r^b_{i,j}(\tau'_i)$ with the known baseband pulse-compression waveform $s_j(\tau'_i)$, yielding the cross-correlation signal $d_{i,j}(\tau'_i)$, which will have a sinc-function form assuming the spectrum of $s_j(\tau'_i)$ is approximately rectangular. Note that for severe relative platform motion, it will be beneficial to apply a Doppler shift correction to $r^b_{i,j}(\tau'_i)$ based on the estimate $\hat{f}^d_{i,j}$ to avoid a perceived delay shift as dictated by the ambiguity function of $s_j(\tau'_i)$. Using the time-domain sinc NL-LS described in [92], the peak value of the cross-correlation $m_{i,j}$ can be computed, which mathematically is given by

$$m_{i,j} = \frac{\rho}{\hat{\alpha}_j} = \frac{1}{\hat{\alpha}_i}\phi_i - \frac{1}{\hat{\alpha}_j}\phi_j + \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}. \tag{4.49}$$

The delay values $m_{i,j}$ determined by platform i for all platforms j with $j \neq i$ should

then be broadcast in a final round of communications to all other platforms such that every platform has shared knowledge of $m_{i,j}$ for all (i, j) pairs with $j \neq i$. Before this broadcast is carried out, however, additional information should be included to assist in computing carrier phases. Platform i will have full knowledge of $m_{i,j}$ and $d_{i,j}(\tau_i')$ for all j with $j \neq i$ independently of other platforms. It can be deduced from (4.48) that the phase of the cross-correlation signal ($\angle d_{i,j}(\tau_i')$) at the peak value (delayed to $m_{i,j}$) will be equal to

$$\angle d_{i,j}(m_{i,j}) = \gamma_{i,j}^{\text{err}} - 2\pi\left(f^c m_{i,j} + f_{i,j}^d \frac{\phi_i}{\alpha_i}\right), \tag{4.50}$$

where $d_{i,j}(m_{i,j})$ refers to the value of the cross-correlation signal evaluated at $\tau_i' = m_{i,j}$. Each receiving platform i should then compute the phase values $\angle d_{i,j}(m_{i,j})$ for all $j \neq i$. Once all these values have been collected, platform i should broadcast a message containing all its computed values of $m_{i,j}$ and $\angle d_{i,j}(m_{i,j})$ for all j with $j \neq i$. When all platforms have carried out this broadcast, every platform will share common knowledge of all values of $m_{i,j}$ and $\angle d_{i,j}(m_{i,j})$.

4.2.2.1 Computing Clock Biases with Symmetric Range

Computation of the clock biases is more straightforward in the case where $R_{i,j} = R_{j,i}$, which is true if there is no relative motion during the synchronization process or if the ranging signals are broadcast simultaneously using some orthogonality scheme other than time-division, such as code-division multiplexing. In this case, noting the equivalence of the drift to drift estimate ratios in (4.43), an estimate of the range between platforms $\hat{R}_{i,j}$ may be computed by

$$\hat{R}_{i,j} = \frac{\alpha_j}{\hat{\alpha}_j} R_{i,j} = \frac{c\left(m_{i,j} + m_{j,i}\right)}{2} \tag{4.51}$$

and an estimate of the clock bias difference, with each clock bias being scaled by the associated clock drift estimate, is computed by

$$\hat{\phi}_{i,j} = \frac{1}{\hat{\alpha}_i}\phi_i - \frac{1}{\hat{\alpha}_j}\phi_j = \frac{m_{i,j} - m_{j,i}}{2}. \tag{4.52}$$

These estimates are computed from $m_{i,j}$ once all values are shared commonly among all platforms. Note that it must be assumed that $\alpha_j/\hat{\alpha}_j$ is sufficiently close to 1 that the range estimate in (4.51) is not significantly affected by the scaling.

As in [92], define $\tilde{\phi}_i$ by

$$\tilde{\phi}_i = \frac{1}{N_p}\sum_j \hat{\phi}_{i,j}, \tag{4.53}$$

which ultimately describes the adjusted clock bias for platform i toward the network mean. On transmit, the clock bias of the platform j may be corrected by applying a fractional delay of $\tilde{\phi}_j$ and a phase correction of $\exp\left(-\mathrm{j}\,2\pi f^c \tilde{\phi}_j\right)$, while the clock bias of platform i on receive may be corrected by applying a fractional delay of $-\tilde{\phi}_i$ and a phase correction of $\exp\left(\mathrm{j}\,2\pi f^c \tilde{\phi}_i\right)$.

Instead of adjusting the clock bias of each platform by correcting to the network mean, each clock bias value may also be computed directly since the equations described in (4.52) form a system of linear equations with $(N_p^2 - N_p)/2 + 1$ equations with N_p unknowns assuming there is an equation included setting a value to zero (e.g., $\phi_1 = 0$) to ensure that the system is full rank. While there are technically $N_p^2 - N_p + 1$ equations, they are not linearly independent because $\hat{\phi}_{i,j} = \hat{\phi}_{j,i}$. If this approach to bias compensation is used, the correction as above but by setting the correction amount to $\hat{\phi}_i/\hat{\alpha}_i$ rather than just $\hat{\phi}_i$. A detailed description of how this linear system of equations may be formulated and solved is provided in Section 4.4.

4.2.2.2 Computing Clock Biases with Asymmetric Range

Computing the clock biases under the assumption that $R_{i,j} \neq R_{j,i}$ is less straightforward. This will generally only be the case if the synchronization broadcasts from each platform are allocated using a time-division multiple-access (TDMA) scheme, and the impacts will be more significant if the relative velocities of the platforms are large in comparison to the TDMA time-slot size (Δ_{TDMA}).

This derivation uses similar notation to [92] as the technique described there makes the inherent assumption that TDMA is used to achieve orthogonality of the transmitted synchronization signals. This derivation also assumes that the drift compensation procedure described in Section 4.1 has been completed. Suppose that a TDMA time slot of length Δ_{TDMA} is used and that the platform TDMA broadcast slots are assigned in ascending order of platform number j. According to its own clock time, platform j will transmit its signal at τ_j^{tx} where

$$\tau_j^{\text{tx}} = (j-1)\Delta_{\text{TDMA}}, \tag{4.54}$$

which is expressed in the global frame as t_j^{tx} by

$$t_j^{\text{tx}} = (j-1)\Delta_{\text{TDMA}} - \frac{\phi_j}{\hat{\alpha}_j}. \tag{4.55}$$

The signal is received by sensor i relative to its own clock at time $\tau_{i,j}^{\text{rx}}$ where

$$\tau_{i,j}^{\text{rx}} = (j-1)\Delta_{\text{TDMA}} + \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c} + \frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}, \tag{4.56}$$

which is expressed in the global frame as $t_{i,j}^{\text{rx}}$ by

$$t_{i,j}^{\text{rx}} = (j-1)\Delta_{\text{TDMA}} + \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c} - \frac{\phi_j}{\hat{\alpha}_j}. \tag{4.57}$$

Note that

$$\tau_{i,j}^{\text{rx}} = (j-1)\Delta_{\text{TDMA}} + m_{i,j}. \tag{4.58}$$

For the purposes of estimating Doppler shift, the assumption is made in Section 4.1 that $v_{i,j} = v_{j,i}$, which is reasonable if it can be assumed that platforms undergo negligible acceleration during the synchronization procedure. Therefore, this assumption is repeated here. Under this assumption, $R_{i,j}$ and $R_{j,i}$ will have a difference between them equal to the constant relative velocity multiplied by the time between their respective signal receipts. Define $\Delta t_{i,j}$ to be

$$\Delta t_{i,j} = t_{i,j}^{\text{rx}} - t_{j,i}^{\text{rx}} = (j-i)\Delta_{\text{TDMA}} + \frac{R_{i,j} - R_{j,i}}{c} + \frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}. \tag{4.59}$$

The relationship between the range values is given by

$$\begin{aligned} R_{i,j} &= R_{j,i} + v_{i,j}\Delta t_{i,j} \\ &= R_{j,i} + v_{i,j}\left((j-i)\Delta_{\text{TDMA}} + \frac{R_{i,j} - R_{j,i}}{c} + \frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}\right). \end{aligned} \tag{4.60}$$

Substituting (4.60) into (4.56) obtains an alternate expression for $\tau^{\text{rx}}_{i,j}$ given by

$$\begin{aligned}\tau^{\text{rx}}_{i,j} &= (j-1)\Delta_{\text{TDMA}} + \frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j} \\ &\quad + \frac{\alpha_j}{\hat{\alpha}_j}\frac{1}{c}\left(R_{j,i} + v_{i,j}\left((j-i)\Delta_{\text{TDMA}} + \frac{R_{i,j}-R_{j,i}}{c} + \frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}\right)\right) \\ &= (j-1)\Delta_{\text{TDMA}} + \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{j,i}}{c} + v_{i,j}\frac{\alpha_j}{\hat{\alpha}_j}\frac{(j-i)\Delta_{\text{TDMA}}}{c} \\ &\quad + v_{i,j}\frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}-R_{j,i}}{c^2} + \frac{\alpha_j}{\hat{\alpha}_j}\frac{v_{i,j}}{c}\left(\frac{\phi_i}{\hat{\alpha}_i} + \frac{\phi_j}{\hat{\alpha}_j}\right) + \frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}.\end{aligned} \tag{4.61}$$

Of the eight terms in the final expression in (4.61), the first can be directly subtracted since it is known exactly, which converts $\tau^{\text{rx}}_{i,j}$ to $m_{i,j}$. Using the estimate $\hat{v}_{i,j}$, the third can also be approximated and removed to give a new value of $m_{i,j}$ ($m'_{i,j}$) expressed by

$$m'_{i,j} = m_{i,j} - \hat{v}_{i,j}\frac{(j-i)\Delta_{\text{TDMA}}}{c}, \tag{4.62}$$

which, similar to the assumption for the range estimate in (4.51), requires the assumption that $\alpha_j/\hat{\alpha}_j$ is very close to 1. The fourth term may be considered negligibly small due to the c^2 in the denominator and will be disregarded in this derivation. Assuming an accurate value of $\hat{v}_{i,j}$, (4.61) may be simplified to

$$m'_{i,j} = \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{j,i}}{c} + \left(1 + \frac{\alpha_j}{\hat{\alpha}_j}\frac{v_{i,j}}{c}\right)\left(\frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}\right). \tag{4.63}$$

Finally, the delay $m_{j,i}$ may be subtracted from the velocity-compensated delay $m'_{i,j}$ to give

$$m'_{i,j} - m_{j,i} = \left(2 + \frac{v_{i,j}}{c}\right)\left(\frac{\phi_i}{\hat{\alpha}_i} - \frac{\phi_j}{\hat{\alpha}_j}\right). \tag{4.64}$$

Therefore, a velocity-compensated value of $\hat{\phi}_{i,j}$ may be computed by

$$\hat{\phi}_{i,j} = \frac{m'_{i,j} - m_{j,i}}{2 + \frac{\hat{v}_{i,j}}{c}}. \tag{4.65}$$

The remaining clock drift compensation may be performed using the result of (4.53) identically to the stationary case.

Should the range values be needed (for instance, for use in relative navigation), an estimate $\hat{R}_{j,i}$ may be computed by

$$\hat{R}_{j,i} = c\left(m_{j,i} + \hat{\phi}_{i,j}\right) \tag{4.66}$$

and by rearranging (4.59) and (4.60) and assuming $\alpha_j/\hat{\alpha}_j \approx 1$, an estimate $\hat{R}_{i,j}$ may be computed by

$$\hat{R}_{i,j} = \frac{\hat{R}_{j,i}\left(1 - \frac{\hat{v}_{i,j}}{c}\right) + \hat{v}_{i,j}(j-i)\Delta_{\text{TDMA}} + \hat{v}_{i,j}\hat{\phi}_{i,j}}{1 - \frac{\hat{v}_{i,j}}{c}}. \tag{4.67}$$

The key differences here compared to the stationary estimate in (4.52) involve the compensation of the relative platform velocity occurring in the removal of the additional TDMA slots in (4.65) and the additional compensation in (4.65). This procedure is reliant on an accurate estimate of the relative velocities, whose estimation is highly dependent on SNR as seen in Section 4.3. Therefore, the TDMA approach should only be used if moderately high SNR can be guaranteed or if additional navigation equipment such as inertial sensors can be employed to assist in computing accurate relative velocity estimates. Otherwise, implementing another orthogonality scheme and avoiding the required estimate of $\hat{v}_{i,j}$ will be better.

4.2.2.3 Computing Carrier Phase Values

For the computation of phase, once all platforms have broadcast and received broadcasts containing all values of $m_{i,j}$ and $\angle d_{i,j}(m_{i,j})$ and after each platform has computed estimates of $\tilde{\phi}_i$, the carrier phase error estimates for all pairs (i, j) with $i \neq j$ may be produced by

$$\hat{\gamma}_{i,j}^{\text{err}} = \angle d_{i,j}(m_{i,j}) + 2\pi\left(f^c m_{i,j} + \hat{f}_{i,j}^d \tilde{\phi}_i\right). \tag{4.68}$$

Next, a system of $N_p^2 - N_p + 1$ (along with an equation such as $\gamma_1^{\text{tx}} = 0$, see below) linear equations described by (2.69) may be solved for the $2N_p$ unknowns to produce estimated values of γ_i^{tx} and γ_i^{rx}, which each platform can independently solve for using the commonly known values of $\hat{\gamma}_{i,j}^{\text{err}}$. These must be solved for with the understanding that one of the phase values must be selected as a known constant phase reference point (e.g. $\gamma_1^{\text{tx}} = 0$). With all transmit and receive phases known, each platform i must apply a phase shift of $-\gamma_i^{\text{tx}}$ to all transmitted waveforms and a phase shift of γ_i^{rx} to all received signals. Note that in the edge case where $N_p = 2$, both systems must have a phase value with a reference phase of zero (e.g., $\gamma_1^{\text{tx}} = \gamma_2^{\text{tx}} = 0$). Note that although the carrier phases may in theory be computed by linear least-squares, there is a large likelihood that phase difference values contain wrapped phases from the true phase difference values expected to be computed from (2.69), which will introduce systematic errors in the solution. A discussion of how to properly set up and solve the system of linear equations, as well as how to handle phase wrapping during the estimation process, is given in Section 4.4.

4.2.3 Fully Synchronized Signal Model

After all synchronization clock corrections are applied, the resulting synchronized model is derived here for the general Doppler shift case. Begin with the drift-compensated transmit waveform as described in (4.36). After computing the clock bias estimate (either $\hat{\phi}_j$ directly or $\tilde{\phi}_j$), it is compensated for on transmit through a fractional delay and phase shift. Similarly, the transmit carrier phase estimate $\hat{\gamma}_j^{\text{tx}}$ is compensated for. Define $\tilde{s}'(\tau_j)$ as the clock drift, clock bias, and carrier phase compensated baseband transmit waveform, which is given by

$$\tilde{s}'(\tau_j) = s_j\left(\frac{\tau_j - \hat{\phi}_i}{\hat{\alpha}_j}\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\frac{1}{\hat{\alpha}_j} - 1\right)\tau_j\right)\exp\left(-\mathrm{j}\,2\pi f^c\frac{\hat{\phi}_j}{\hat{\alpha}_j}\right)\exp\left(-\mathrm{j}\,\hat{\gamma}_j^{\text{tx}}\right). \quad (4.69)$$

While the above expression assumes that the clock bias correction is applied using a direct estimate of the clock bias $\hat{\phi}_j$, such that the correction $\hat{\phi}_j/\hat{\alpha}_j$ is applied, this correction can be replaced with the network mean correction $\tilde{\phi}_j$ without any additional changes to the expression. A similar substitution will need to be made in future steps related to clock bias correction. From (2.75), the received signal, $\tilde{r}_{i,j}(\tau_i)$, will be given

by

$$\begin{aligned}\tilde{r}_{i,j}(\tau_i) &= \tilde{s}_j\left(\frac{\alpha_j}{\alpha_i}\tau_i - \frac{\alpha_j}{\alpha_i}\phi_i + \phi_j - \alpha_j\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi\left(f^c\left(\frac{\alpha_j}{\alpha_i}-1\right)+\frac{f^d_{i,j}}{\alpha_i}\right)\tau_i\right)\\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j}{\alpha_i}\phi_i - \phi_j + \alpha_j\frac{R_{i,j}}{c}\right)\right)\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\left(\gamma_j^{\mathrm{tx}}-\gamma_i^{\mathrm{rx}}\right)\right)\\ &= s_j\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\tau_i - \frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\phi_i + \frac{1}{\hat{\alpha}_j}\left(\phi_j-\hat{\phi}_j\right) - \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}-1\right)\tau_i\right)\\ &\quad\cdot\exp\left(\mathrm{j}\,2\pi\frac{f^d_{i,j}}{\alpha_i}\tau_i\right)\exp\left(-\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\phi_i - \frac{1}{\hat{\alpha}_j}\left(\phi_j-\hat{\phi}_j\right)+\frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\right)\\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\left(\left(\gamma_j^{\mathrm{tx}}-\hat{\gamma}_j^{\mathrm{tx}}\right)-\gamma_i^{\mathrm{rx}}\right)\right).\end{aligned} \tag{4.70}$$

Next, the local receiver time axis, τ_i, is dilated by the estimated clock drift $\hat{\alpha}_i$ as in (4.41). This results in

$$\begin{aligned}\tilde{r}_{i,j}(\tau_i') &= s_j\left(\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i}\tau_i' - \frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\phi_i + \frac{1}{\hat{\alpha}_j}\left(\phi_j-\hat{\phi}_j\right) - \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i}-\hat{\alpha}_i\right)\tau_i'\right)\\ &\quad\cdot\exp\left(\mathrm{j}\,2\pi\frac{\hat{\alpha}_i}{\alpha_i}f^d_{i,j}\tau_i'\right)\exp\left(-\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\phi_i - \frac{1}{\hat{\alpha}_j}\left(\phi_j-\hat{\phi}_j\right)+\frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\right)\\ &\quad\cdot\exp\left(-\mathrm{j}\,2\pi f^d_{i,j}\frac{\phi_i}{\alpha_i}\right)\exp\left(\mathrm{j}\left(\left(\gamma_j^{\mathrm{tx}}-\hat{\gamma}_j^{\mathrm{tx}}\right)-\gamma_i^{\mathrm{rx}}\right)\right).\end{aligned} \tag{4.71}$$

Multiplying by the carrier correction term $b_i(\tau_i')$ yields the fully receive drift corrected signal, $\tilde{r}^b_{i,j}(\tau_i')$, given by

$$\begin{aligned}
\tilde{r}^b_{i,j}(\tau_i') &= \tilde{r}'_{i,j}(\tau_i') b_i(\tau_i') \\
&= s_j \left(\frac{\alpha_j \hat{\alpha}_i}{\hat{\alpha}_j \alpha_i} \tau_i' - \frac{\alpha_j}{\hat{\alpha}_j \alpha_i} \phi_i + \frac{1}{\hat{\alpha}_j} \left(\phi_j - \hat{\phi}_j \right) - \frac{\alpha_j}{\hat{\alpha}_j} \frac{R_{i,j}}{c} \right) \exp\left(\mathrm{j}\, 2\pi f^c \left(\frac{\alpha_j \hat{\alpha}_i}{\hat{\alpha}_j \alpha_i} - 1 \right) \tau_i' \right) \\
&\quad \cdot \exp\left(\mathrm{j}\, 2\pi \frac{\hat{\alpha}_i}{\alpha_i} f^d_{i,j} \tau_i' \right) \exp\left(-\mathrm{j}\, 2\pi f^c \left(\frac{\alpha_j}{\hat{\alpha}_j \alpha_i} \phi_i - \frac{1}{\hat{\alpha}_j} \left(\phi_j - \hat{\phi}_j \right) + \frac{\alpha_j}{\hat{\alpha}_j} \frac{R_{i,j}}{c} \right) \right) \\
&\quad \cdot \exp\left(-\mathrm{j}\, 2\pi f^d_{i,j} \frac{\phi_i}{\alpha_i} \right) \exp\left(\mathrm{j} \left(\left(\gamma_j^{\mathrm{tx}} - \hat{\gamma}_j^{\mathrm{tx}} \right) - \gamma_i^{\mathrm{rx}} \right) \right).
\end{aligned} \tag{4.72}$$

The receiver then partially compensates for clock bias by applying a fractional delay of $-\hat{\phi}_i / \hat{\alpha}_i$, which produces the signal $\tilde{r}^c_{i,j}(\tau_i')$, given by

$$\begin{aligned}
\tilde{r}^c_{i,j}(\tau_i') &= \tilde{r}^b_{i,j} \left(\tau_i' + \frac{\hat{\phi}_i}{\hat{\alpha}_i} \right) \\
&= s_j \left(\frac{\alpha_j \hat{\alpha}_i}{\hat{\alpha}_j \alpha_i} \tau_i' - \frac{\alpha_j}{\hat{\alpha}_j \alpha_i} \left(\phi_i - \hat{\phi}_i \right) + \frac{1}{\hat{\alpha}_j} \left(\phi_j - \hat{\phi}_j \right) - \frac{\alpha_j}{\hat{\alpha}_j} \frac{R_{i,j}}{c} \right) \\
&\quad \cdot \exp\left(-\mathrm{j}\, 2\pi f^c \left(\frac{\alpha_j}{\hat{\alpha}_j \alpha_i} \left(\phi_i - \hat{\phi}_i \right) - \frac{1}{\hat{\alpha}_j} \left(\phi_j - \hat{\phi}_j \right) + \frac{\alpha_j}{\hat{\alpha}_j} \frac{R_{i,j}}{c} \right) \right) \\
&\quad \cdot \exp\left(\mathrm{j}\, 2\pi f^c \left(\frac{\alpha_j \hat{\alpha}_i}{\hat{\alpha}_j \alpha_i} - 1 \right) \tau_i' \right) \exp\left(\mathrm{j}\, 2\pi \frac{\hat{\alpha}_i}{\alpha_i} f^d_{i,j} \tau_i' \right) \\
&\quad \cdot \exp\left(-\mathrm{j}\, 2\pi \frac{f^d_{i,j}}{\alpha_i} \left(\phi_i - \hat{\phi}_i \right) \right) \exp\left(-\mathrm{j}\, 2\pi f^c \frac{\hat{\phi}_i}{\hat{\alpha}_i} \right) \exp\left(\mathrm{j} \left(\left(\gamma_j^{\mathrm{tx}} - \hat{\gamma}_j^{\mathrm{tx}} \right) - \gamma_i^{\mathrm{rx}} \right) \right).
\end{aligned} \tag{4.73}$$

Finally, to yield the fully corrected receive signal, $\tilde{r}^{f}_{i,j}(\tau'_i)$, the phase shifts of $\exp\left(\mathrm{j}\,2\pi f^c \hat{\phi}_i/\hat{\alpha}_i\right)$ and $\exp\left(\mathrm{j}\,\hat{\gamma}^{\mathrm{rx}}_i\right)$ are applied, yielding

$$\begin{aligned}\tilde{r}^{f}_{i,j}(\tau'_i) = s_j\left(\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i}\tau'_i - \left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\left(\phi_i-\hat{\phi}_i\right) - \frac{1}{\hat{\alpha}_j}\left(\phi_j-\hat{\phi}_j\right) + \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\right)\\ \cdot \exp\left(-\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j}{\hat{\alpha}_j\alpha_i}\left(\phi_i-\hat{\phi}_i\right) - \frac{1}{\hat{\alpha}_j}\left(\phi_j-\hat{\phi}_j\right) + \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\right)\\ \cdot \exp\left(\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j\hat{\alpha}_i}{\hat{\alpha}_j\alpha_i}-1\right)\tau'_i\right)\exp\left(\mathrm{j}\,2\pi\frac{\hat{\alpha}_i}{\alpha_i}f^d_{i,j}\tau'_i\right)\\ \cdot \exp\left(-\mathrm{j}\,2\pi\frac{f^d_{i,j}}{\alpha_i}\left(\phi_i-\hat{\phi}_i\right)\right)\exp\left(\mathrm{j}\,\left(\left(\gamma^{\mathrm{tx}}_j-\hat{\gamma}^{\mathrm{tx}}_j\right)-\left(\gamma^{\mathrm{rx}}_i-\hat{\gamma}^{\mathrm{rx}}_i\right)\right)\right).\end{aligned} \tag{4.74}$$

Under the assumption of ideal estimation of clock bias and carrier phase (i.e., $\hat{\phi}_i = \phi_i$, $\hat{\gamma}^{\mathrm{tx}}_j = \gamma^{\mathrm{tx}}_j$, and $\hat{\gamma}^{\mathrm{rx}}_i = \gamma^{\mathrm{rx}}_i$) and ideal relative estimation of clock drift (i.e., $\alpha_i/\hat{\alpha}_i = \alpha_j/\hat{\alpha}_j$), this expression will reduce to

$$\tilde{r}^{f}_{i,j}(\tau_i) = s_j\left(\tau'_i - \frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi\frac{\hat{\alpha}_i}{\alpha_i}f^d_{i,j}\tau'_i\right)\exp\left(-\mathrm{j}\,2\pi f^c\frac{\alpha_j}{\hat{\alpha}_j}\frac{R_{i,j}}{c}\right), \tag{4.75}$$

which shows the ideal receive signal but where the propagation delay, the propagation phase, and Doppler shift frequency are scaled by the erroneous drift ratio. Making the further assumption that the clock drifts are estimated perfectly (i.e., $\hat{\alpha}_i = \alpha_i$, which would only be possible if $\alpha_1 = 1$), (4.74) further reduces to

$$\tilde{r}^{f}_{i,j}(\tau_i) = s_j\left(\tau'_i - \frac{R_{i,j}}{c}\right)\exp\left(\mathrm{j}\,2\pi f^d_{i,j}\tau'_i\right)\exp\left(-\mathrm{j}\,2\pi f^c\frac{R_{i,j}}{c}\right), \tag{4.76}$$

which is the ideal received signal with the correct propagation delay, propagation phase, and Doppler shift frequency.

4.2.4 Algorithm Summary

At the completion of the information exchange, each radar platform has produced an estimate of its own clock drift, clock bias, and transmit and receive phases. Each of these can continue to be compensated as described here to maintain coherent operation. A diagram describing the full operation of the proposed synchronization technique is given in Fig. 4.2.

4.3 Simulated Results

To evaluate and compare the utility of the proposed techniques, multiple simulations are run, sweeping over a variety of parameters to demonstrate their impact on the outcomes of the synchronization. For all simulations, each root-mean-square error (RMSE) data point is computed from 1000 independent trials. Additionally, each simulation is run with $f^c = 3$ GHz and $N_p = 3$. Unless stated otherwise, a band-pass filter with a bandwidth of 5 MHz is applied around each tone during the estimation process in both the single-tone and two-tone procedures. For the results showing the accuracy of the drift estimates $\hat{\alpha}_i$, the values are normalized by the true value of α_1 under the assumption that $\hat{\alpha}_1 = 1$.

4.3.1 Single-Tone Algorithm Performance

First, the single-tone estimation techniques are simulated. In these simulations, the pulse length T_p is set to 100 μs, the capture length T_c is set to 200 μs, and the sampling frequency f^s is set to 200 MHz. The drift values are randomly generated from a normal distribution with zero mean and a standard deviation $\sigma_\alpha = 100 \times 10^{-6}$. The clock bias values are randomly generated from a normal distribution with zero

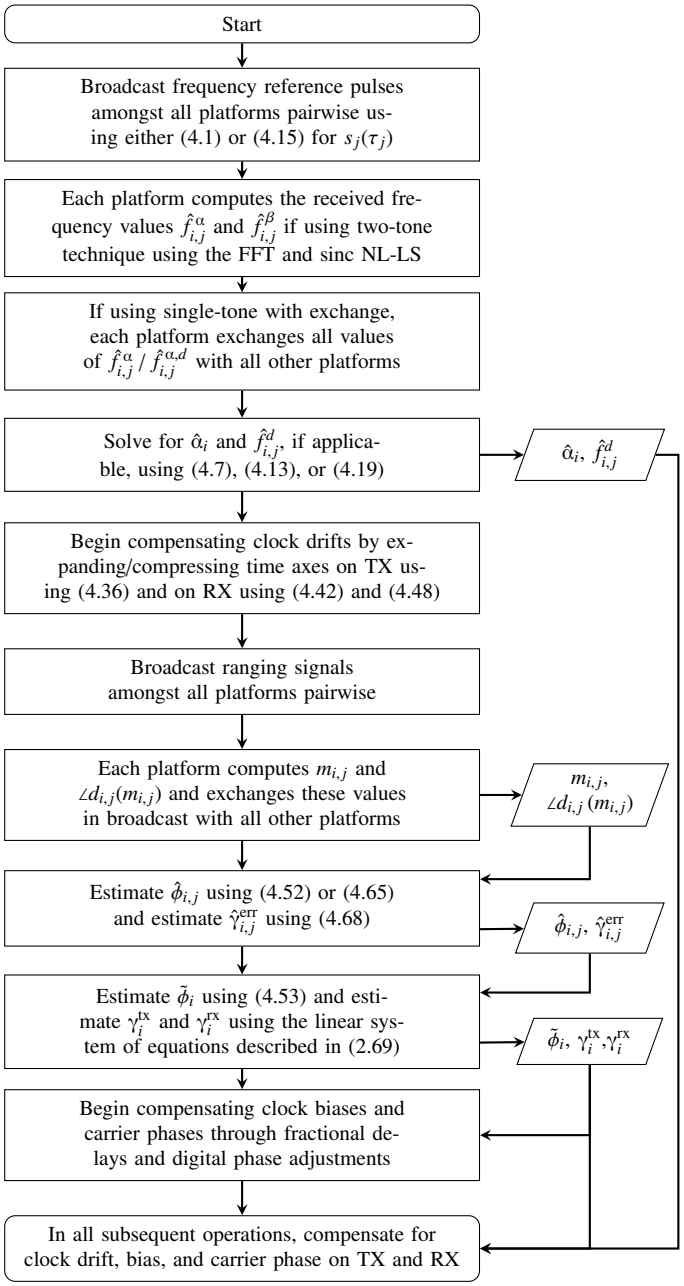


Fig. 4.2: A flowchart of the proposed synchronization algorithm. From [4] ©2024 IEEE.

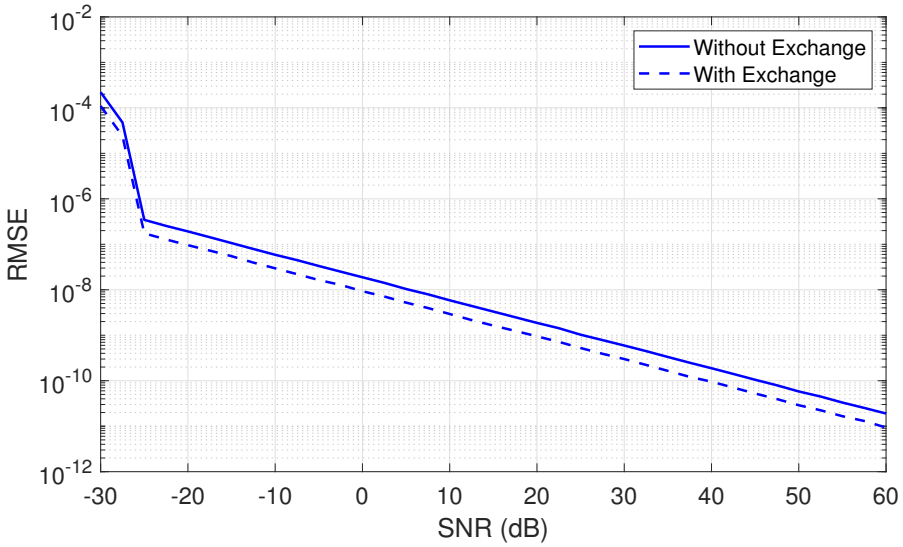


Fig. 4.3: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the SNR at the estimating platform as produced by the single-tone estimation techniques, both with and without exchange, when there is no relative motion between platforms. From [4] ©2024 IEEE.

mean and a standard deviation $\sigma_\phi = 10 \times 10^{-9}$.

Fig. 4.3 shows the RMSE of $\hat{\alpha}_i$ for the single-tone techniques, both with and without exchange, as a function of SNR. For this simulation, each platform is simulated as stationary and thus there is no error introduced by uncompensated Doppler shift frequencies. Each technique has an RMSE which decreases inversely with an increase in SNR for SNR values greater than -25 dB. As expected, the single-tone estimation with exchange slightly outperforms the estimation without exchange. For the single-tone with exchange, at an SNR of 0 dB, the drift value is estimated with an RMSE of $\sim 10^{-8}$, giving a carrier frequency misalignment of $\sim$30 Hz when $f^c = 3$ GHz. For an SNR of 30 dB, the RMSE is $\sim 3 \times 10^{-10}$, giving a carrier frequency misalignment of less than 1 Hz.

Consider the case in which the single-tone techniques are employed where the platforms experience uncompensated relative motion. Fig. 4.4 shows the RMSE of

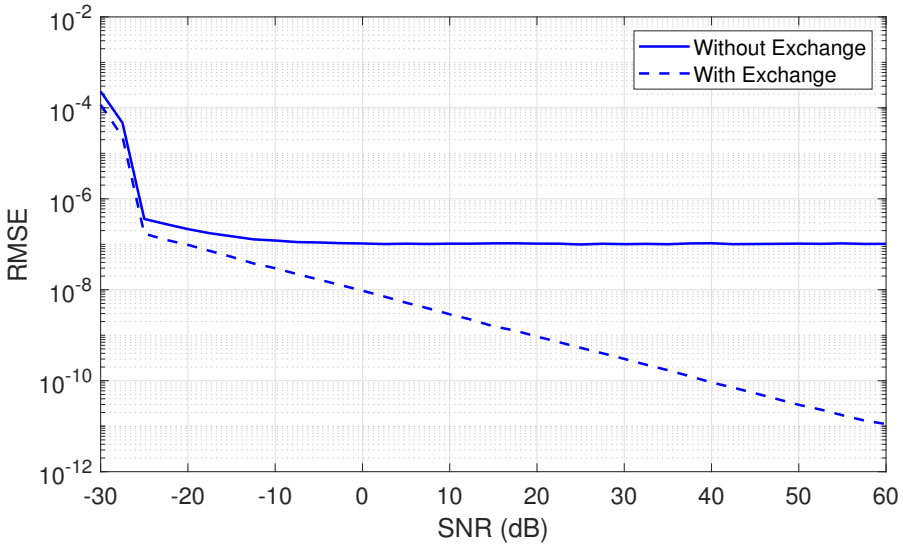


Fig. 4.4: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the SNR at the estimating platform as produced by the single-tone estimation techniques, both with and without exchange, where there is uncompensated relative motion between the platforms. From [4] ©2024 IEEE.

$\hat{\alpha}_i$ for the single-tone techniques as a function of SNR when the platforms are not stationary. In this simulation, platforms are generated with random velocities from a normal distribution with zero mean and a standard deviation of $\sigma_v = 10$ m/s. The velocities and resulting Doppler shifts are not estimated or compensated by the single-tone with exchange technique. All other parameters of the simulation are identical to those of the simulation with no velocity displayed in Fig. 4.3. In this case, the single-tone without exchange technique levels off in RMSE at 10^{-7} for SNR> -20 dB, while the single-tone with exchange technique exhibits similar performance, indicating that it is robust to small relative velocities even if the Doppler shifts are ignored.

4.3.2 Estimation Performance with Doppler Compensation

Next, the techniques which include velocity and Doppler frequency compensation are evaluated. Unless otherwise noted, the simulations here are run with the same

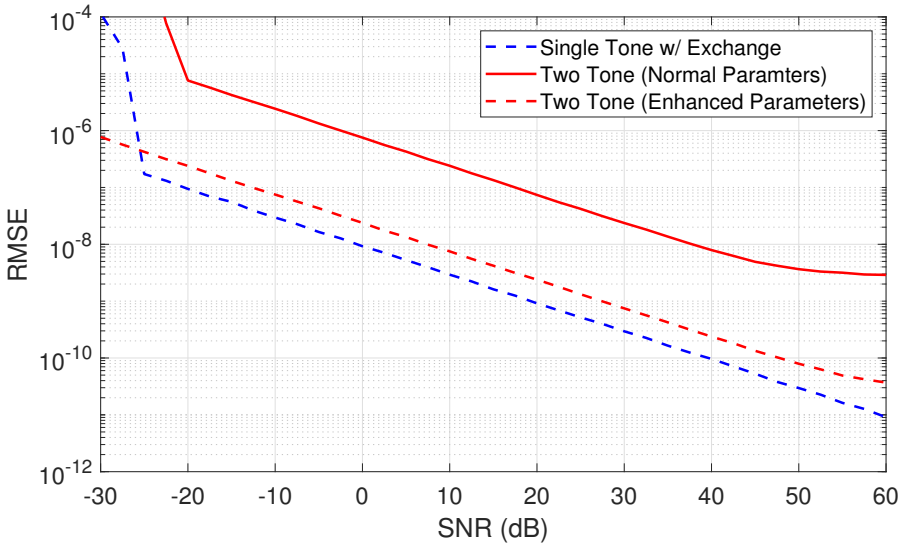


Fig. 4.5: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the SNR at the estimating platform as produced by the Doppler-compensating estimation techniques. From [4] ©2024 IEEE.

parameters as in the single-tone algorithm simulations.

Fig. 4.5 shows the RMSE of $\hat{\alpha}_i$ for the single-tone with exchange and two-tone Doppler-compensating synchronization algorithms. In one case of the two-tone technique (the solid red line), the same pulse length, capture time, and sampling frequency are used, and a tone separation of 150 MHz. However, these parameters lead to generally poor performance with nearly two orders of magnitude increase in RMSE compared to the single-tone technique. Therefore, an enhanced set of two-tone parameters is also simulated, with a pulse length T_p of 500 μs, a capture time T_c of 1 ms, a sampling frequency f^s of 400 MHz, and a tone separation of 300 MHz. The RMSE for this enhanced case is given in Fig. 4.5 as the dashed red line. As with the no-Doppler case above, the RMSE decreases proportionally to the inverse of SNR. With normal parameters, the two-tone technique performs poorly and never is able to reduce the RMSE below 10^{-9} with less than the maximum simulated 60 dB of SNR. However, the enhanced parameters nearly match the performance of the single-tone technique.

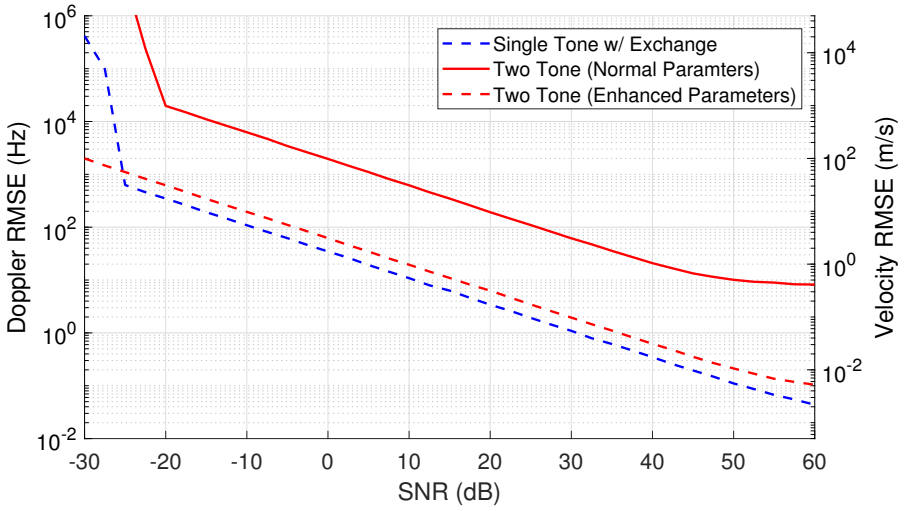


Fig. 4.6: A plot comparing the RMSE of the Doppler shift and velocity estimates $\hat{\alpha}_i$ as a function of the SNR at the estimating platform as produced by the Doppler-compensating estimation techniques. From [4] ©2024 IEEE.

Therefore, should the no-exchange advantage of the two-tone technique be required, the designer must be willing to accept the trade-off of a number of increased performance parameters to maintain similar synchronization performance compared to the single-tone with-exchange procedure.

For the same simulations, the RMSE of Doppler frequency $\hat{f}^d_{i,j}$ and $\hat{v}_{i,j}$ are given in Fig. 4.6, respectively. As with the drift estimates, the enhanced parameters are required for the two-tone technique to come close to matching the single-tone with exchange technique. For these more capable techniques, an SNR of 0 dB gives a 2–3 m/s velocity RMSE, while for an SNR of 30 dB this figure reduces to less than 0.1 m/s velocity RMSE. Therefore, even for the single-tone technique (at least, with the parameters provided here; the pulse length and sampling frequency could be increased to improve performance), a moderately high SNR will be required to give accurate relative velocity information.

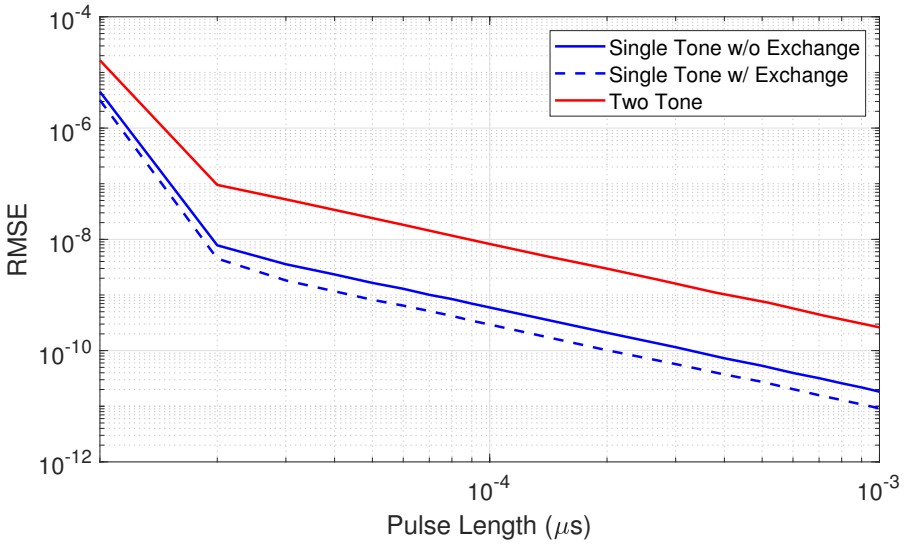


Fig. 4.7: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the pulse length (T_p) at the estimating platform as produced by the single-tone and two-tone estimation techniques. From [4] ©2024 IEEE.

4.3.3 Impact of Pulse Length

The pulse length T_p will play a large role in the performance of the synchronization techniques. Because a longer pulse has more energy, it will reduce the RMSE of the estimation procedure even as SNR remains the same. Fig. 4.7 shows the $\hat{\alpha}_i$ RMSE of the two single-tone techniques and the two-tone technique as a function of pulse length. In this simulation, SNR is fixed at 30 dB and the platforms are stationary (i.e. $v_{i,j} = 0$). This demonstrates that for the same pulse length, the two-tone procedure has more than an order of magnitude higher RMSE, highlighting the strong trade-off in performance introduced by the two-tone estimation technique.

4.3.4 Impact of Tone Separation on Two-Tone Estimation

One important consideration in the use of the two-tone synchronization procedure is the spacing of the two transmitted tones f_j^{α} and f_j^{β}. In general, larger separations

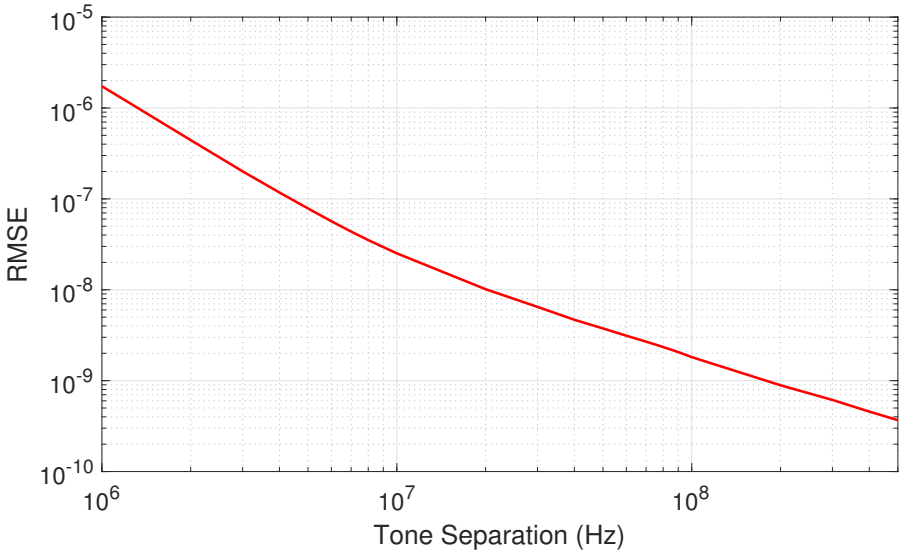


Fig. 4.8: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the tone separation ($|f_j^{\alpha} - f_j^{\beta}|$) for the two-tone estimation technique. From [4] ©2024 IEEE.

between the two tones will lead to better performance. Fig. 4.8 shows the RMSE of $\hat{\alpha}_i$ as a function of tone separation given by $|f_j^{\alpha} - f_j^{\beta}|$. In this simulation, the pulse length is set to 500 μs while the capture length is set to 1 ms. To accommodate the higher tone separation values, the sampling frequency is increased to 600 MHz. Moreover, to accommodate the lower tone separation values, the standard deviation of the randomly generated drift values is reduced to $\sigma_{\alpha} = 10 \times 10^{-6}$ and the bandwidth of the band-pass filter around the tones is reduced to 1 MHz. These results indicate that fairly large tone separations are required to produce reasonable estimates of the clock drift; as a result, this technique may only prove viable in situations where wide bandwidths are available.

4.3.5 Impact of Platform Velocity

It is of interest to determine how tolerant the different techniques are as the relative velocities between the platforms increase, leading to more significant Doppler shift

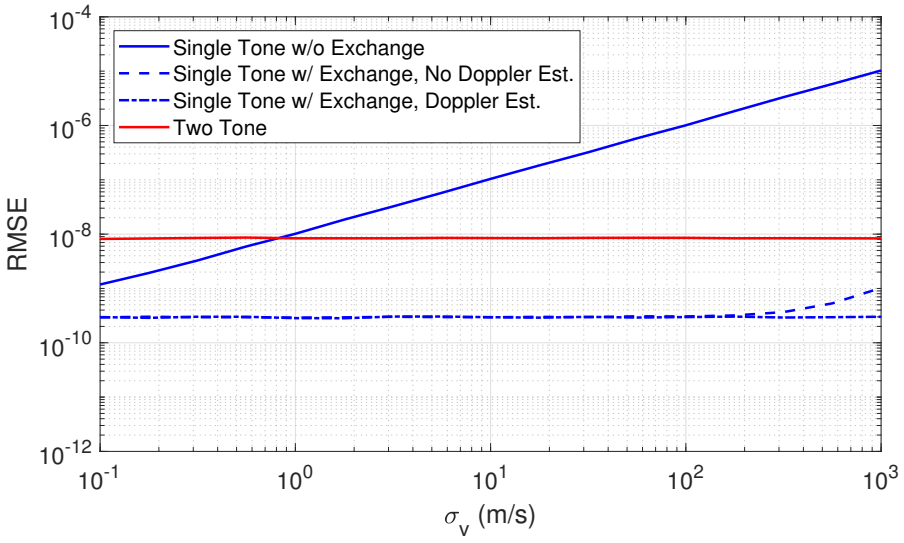


Fig. 4.9: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the random platform velocity standard deviation (σ_v) for all single-tone techniques (with and without exchange, with and without Doppler estimation) and the two-tone technique. From [4] ©2024 IEEE.

frequencies. Fig. 4.9 shows the RMSE of $\hat{\alpha}_i$ for all the estimation techniques as a function of the standard deviation of the simulated random platform velocities, σ_v. As expected, single-tone without exchange has performance that degrades linearly as σ_v increases since it makes no attempt to compensate for the Doppler shifts. The single-tone with exchange is resilient to increased velocities even when the Doppler shifts are not directly estimated; however, after the velocities increase past $\sigma_v \approx 200$ m/s, the RMSE for the case in which Doppler frequencies are not estimated begins to degrade while the RMSE for the case in which they are estimated and compensated maintains consistent RMSE performance. Similarly, the two-tone procedure maintains constant performance regardless of the magnitude of σ_v. These results indicate that with the single-tone with exchange procedure, so long as the relative velocity information is not needed and the relative velocities can be assumed to be small, it is not necessary to estimate or compensate for the Doppler shifts. Moreover, if no-exchange is a required parameter of the network, the two-tone technique will clearly outperform the single-

tone without exchange technique with similar parameters so long as $\sigma_v > 1$ m/s.

4.3.6 Impact of Network Size

In some cases, there is a performance dependence on the number of platforms, N_p, in the network. Fig. 4.10 shows the RMSE of $\hat{\alpha}_i$ as a function of network size N_p. The simulations here are performed without platform motion, noting that from Fig. 4.9 it can be inferred that the performance of the single-tone with exchange and two-tone techniques will not experience a performance degradation if velocities are non-zero while the single-tone without exchange technique will; therefore, to include a comparison with single-tone without exchange, platform velocities are set to zero. The SNR of the simulations is set to a constant 30 dB. The single-tone parameters are identical to those from previous simulations, while the two-tone parameters are identical to the normal parameters demonstrated in Section 4.3.2.

From Fig. 4.10, it can be seen that the drift estimation RMSE of the single-tone with exchange technique has an inversely proportional relationship with N_p such that larger network sizes produce higher performance. For the single-tone without exchange and the two-tone techniques (i.e., the two techniques that do not exchange information before forming estimates), the performance shows a minor degradation as N_p increases from 3 to 5, after which the performance remains constant for increasing values of N_p. These results follow logically from the amount of information available to each respective estimation algorithm: for the single-tone without exchange and two-tone techniques, there are exactly enough equations available to solve for the number of unknowns regardless of the network size. However, in the single-tone with exchange technique, the number of unknowns grows linearly while the number of independent equations grows quadratically with the number of platforms. Therefore, it

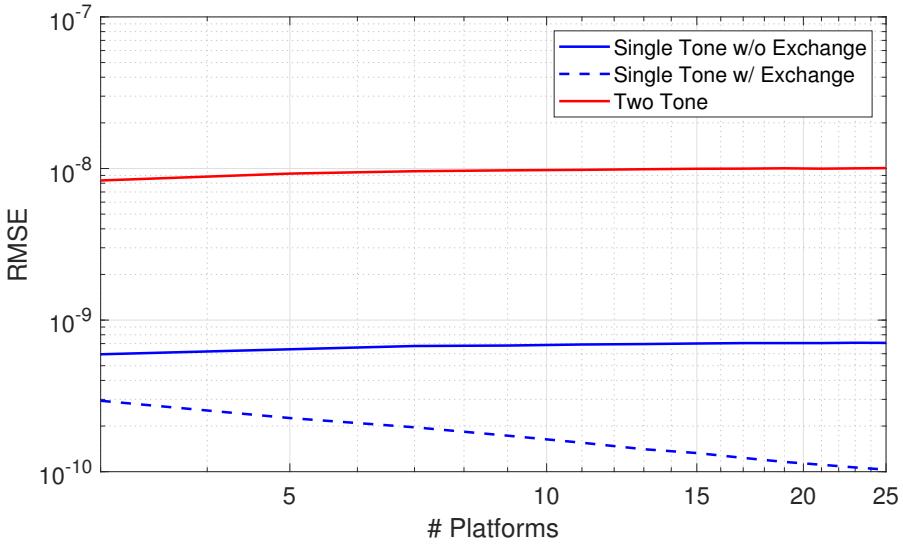


Fig. 4.10: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the number of platforms in the network (N_p) for both single-tone techniques (with and without exchange) and the two-tone technique. From [4] ©2024 IEEE.

is reasonable to expect the estimation error produced by this algorithm to be reduced as the network size increases.

4.3.7 Impact of Carrier Frequency

Because the underlying signal model has an explicit dependence on carrier frequency, it is reasonable to expect that in some cases the carrier frequency will impact the overall performance. Fig. 4.11 shows the RMSE of $\hat{\alpha}_i$ as a function of carrier frequency f^c. Noting again that the single-tone without exchange technique will have a varying performance as a function of platform velocity while the other two techniques will not, Fig. 4.11 includes results for the single-tone without exchange technique for both $\sigma_v = 0$ m/s and $\sigma_v = 10$ m/s. As with other simulations, the network size is set to $N_p = 3$. Unlike previous simulations, the filter bandwidth for isolating the tones is substantially increased to 200 MHz since for a fixed clock drift, the observed tone frequency will shift further at a higher carrier frequency. Thus, if the filter bandwidth

were kept at 5 MHz, the procedure would stop working entirely after a certain carrier frequency threshold was exceeded. All other simulation parameters are identical to those described in Section 4.3.6 above.

From Fig. 4.11 it can be deduced that for the single-tone with exchange and the single-tone without exchange with no velocity, there is a decrease in error proportional to the increase in carrier frequency. However, for the two-tone case and the single-tone without exchange with velocity, the performance is constant as a function of f^c. For the single-tone without exchange technique, this is because the lower bound of performance is set by σ_v per Fig. 4.9. For the two-tone technique, this is likely because estimation errors induced by high Doppler velocities cancel out any benefit caused by the increased carrier frequency, while for the single-tone with exchange, the additional information enables higher-quality estimation in the same way that it enables this in the presence of noise as demonstrated in Fig. 4.5.

4.4 Least-Squares Solutions and Statistical Analysis

Throughout the proposed synchronization procedure, clock parameters are estimated as the solutions to several sets of linear equations. These systems tend to be over-determined with noisy coefficients, and as such the estimation process must utilize either ordinary least-squares (OLS) or weight least-squares (WLS) estimation. Generally, OLS may be applied when the covariance matrix of the solution vector is either unknown or a scalar matrix, while WLS may be applied when the covariance matrix is known to have varying values along the diagonal or when there are non-zero off-diagonal terms in the matrix. In this section, the OLS and WLS solutions for the clock parameter estimations are provided. Furthermore, the statistical performance of the least-squares estimation process is analyzed through the Cramér-Rao

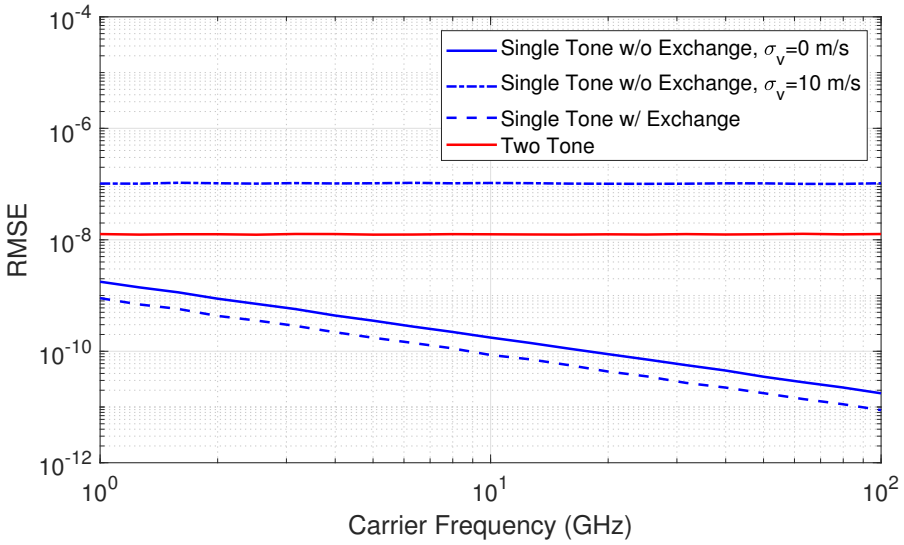


Fig. 4.11: A plot comparing the RMSE of the clock drift estimate $\hat{\alpha}_i$ as a function of the carrier frequency (f^c) for both single-tone techniques (with and without exchange) and the two-tone technique. The single-tone without exchange technique is also shown for cases with and without velocity. From [4] ©2024 IEEE.

lower bound (CRLB) and the covariance propagation through a linear system during OLS and WLS. Because the statistical performance of the synchronization technique is heavily dependent on SNR, and in particular the WLS solution requires accurate knowledge of the SNR, a simple procedure is given for estimating the SNR. Finally, simulated results are provided which demonstrate the theoretical performance bound of the proposed synchronization technique.

The analysis in this section is as follows. First, the OLS and WLS estimation techniques are described for the general case, and specific expressions for producing WLS estimates of clock drift, clock bias, and carrier phase are provided along with expressions for computing the associated estimate covariance. Next, it will be shown that the sinc NL-LS procedure meets the CRLB for frequency and time-delay estimation over a useful range of SNR values. It will also be shown that the phase extraction after pulse compression meets the CRLB for phase estimation over a particular range of useful SNR values. Expressions will then be provided for converting the CRLB variance

values into useful covariance matrices to be used during WLS estimation. A simple procedure will also be described for producing SNR estimates from the received signals such that the CRLB variance values may be accurately estimated. Finally, it will be shown through a set of case studies that the full estimation of the synchronization parameters approaches or meets the expected variance computed through the WLS covariance transformations.

4.4.1 Solving for Clock Parameters with OLS and WLS

As described in Sections 4.1 and 4.2, the clock drifts, clock biases, and carrier phases may be estimated by solving sets of linear equations. A general system of linear equations may be formulated as

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{4.77}$$

where $\mathbf{A}$ is an $M_b \times M_x$ matrix describing the coefficients of the system of equations relating $\mathbf{x}$ and $\mathbf{b}$, $\mathbf{x}$ is an M_x vector of values to be solved for, and $\mathbf{b}$ is an $M_b \times 1$ solution vector containing values describing the solutions to each equation. It is further assumed that $\mathbf{b}$ is corrupted by additive noise with covariance $\mathbf{C}_b$. Assuming the case where $M_b > M_x$, The most straightforward solution to this system of equations is OLS, which makes use of the Moore-Penrose pseudo-inverse matrix. In this formulation, the solution $\hat{\mathbf{x}}$ may be computed by

$$\hat{\mathbf{x}} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{b}. \tag{4.78}$$

However, this formulation is only valid when the covariance $\mathbf{C}_b$ is a scalar matrix – that is, when the variance of each component of $\mathbf{b}$ is equivalent and there are no

correlations between the different entries of $\mathbf{b}$. Therefore, the OLS solution should only be used in cases in which this is true or when the covariance matrix is not known or cannot be estimated.

In the case where $\mathbf{C}_b$ is known and is not scalar, the WLS solution $\hat{\mathbf{x}}$ may be computed by [109]

$$\hat{\mathbf{x}} = \left(\mathbf{A}^T \mathbf{C}_b^{-1} \mathbf{A}\right)^{-1} \mathbf{A}^T \mathbf{C}_b^{-1} \mathbf{b}. \tag{4.79}$$

Furthermore, the covariance of the solution $\hat{\mathbf{x}}$, Σ_x, may be computed by

$$\Sigma_x = \left(\mathbf{A}^T \mathbf{C}_b^{-1} \mathbf{A}\right)^{-1}, \tag{4.80}$$

which will be valid for both the OLS and WLS cases. Generally, if the covariance matrix is known (which will be the case for all the linear least-squares procedures in this chapter), using WLS is preferable, and as such the WLS estimation is assumed from this point on.

4.4.1.1 Clock Drift WLS

The system of equations for solving for clock drifts described in (4.7) (or (4.13) in the Doppler case) does not exactly follow the model in (4.77) because the noisy data in $\hat{f}_{i,j}^{\alpha}$ or $\hat{f}_{i,j}^{\alpha,d}$ are included in the coefficients of the system of equations rather than the solution vector. However, using WLS may still be justified by making a minor approximation. First, assuming $\hat{f}_{i,j}^{\alpha}$ is corrupted by additive noise, it may be rewritten as

$$\hat{f}_{i,j}^{\alpha} = f_{i,j}^{\alpha} + \eta_{i,j}^{\alpha} \tag{4.81}$$

where $\eta_{i,j}^{\alpha}$ is the noise component of $\hat{f}_{i,j}^{\alpha}$, which will crucially have the same covariance as $\hat{f}_{i,j}^{\alpha}$. Thus, (4.7) may be rewritten as

$$\left(f_{i,j}^{\alpha} + \eta_{i,j}^{\alpha} + f^{c}\right)\alpha_i - \left(f_j^{\alpha} + f^{c}\right)\alpha_j = 0. \tag{4.82}$$

Because $\alpha_i \approx 1$, $\eta_{i,j}^{\alpha}\alpha_i \approx \eta_{i,j}^{\alpha}$, and therefore (4.82) may be rearranged to give

$$\left(f_{i,j}^{\alpha} + f^{c}\right)\alpha_i - \left(f_j^{\alpha} + f^{c}\right)\alpha_j \approx -\eta_{i,j}^{\alpha}, \tag{4.83}$$

which does have the linear form required by WLS. Therefore, the system of equations in (4.7) may be solved using WLS with the covariance of the frequency measurements. A similar result can be achieved with the Doppler case in (4.13) to give

$$\left(f_{i,j}^{\alpha,d} + f^{c}\right)\alpha_i - \left(f_j^{\alpha} + f^{c}\right)\alpha_j - f_{i,j}^{d} \approx -\eta_{i,j}^{\alpha,d}, \tag{4.84}$$

where $\eta_{i,j}^{\alpha,d}$ is the additive noise component of $\hat{f}_{i,j}^{\alpha,d}$.

Let $\mathbf{A}_f$ be the $(N_p^2 - N_p + 1) \times N_p$ matrix describing the coefficients of the system of equations in (4.7) for the stationary case and let $\mathbf{A}_{f,d}$ be the $(N_p^2 - N_p + 1) \times (N_p + N_p^2/2 - N_p/2)$ matrix describing the coefficients of the system of equations in (4.13) for the Doppler case. Furthermore, let $\hat{\mathbf{x}}_\alpha$ be a vector of the clock drift estimates produced by least-squares estimation, let $\hat{\mathbf{x}}_d$ be a vector of the Doppler shift frequency estimates produced by least-squares estimation, and define a vector $\hat{\mathbf{x}}_f$ as

$$\hat{\mathbf{x}}_f = \begin{bmatrix} \hat{\mathbf{x}}_\alpha \\ \\ \hat{\mathbf{x}}_d \end{bmatrix}. \tag{4.85}$$

In future integrations of the synchronization estimates with position estimates, it will

be convenient to have direct radial velocity estimates. Assuming each platform transmits at the same carrier frequency, a vector of velocity estimates may be computed by

$$\hat{\mathbf{x}}_v = -\frac{c}{f^c}\hat{\mathbf{x}}_d \tag{4.86}$$

and another frequency vector

$$\hat{\mathbf{x}}_{f,v} = \begin{bmatrix} \hat{\mathbf{x}}_\alpha \\ \hat{\mathbf{x}}_v \end{bmatrix}. \tag{4.87}$$

Let $\mathbf{b}_f$ be a vector containing the frequency estimates $\hat{f}_{i,j}^{\alpha}$ for the stationary case, ordered such that the entries align properly with the rows of $\mathbf{A}_f$. Define the covariance of $\mathbf{b}_f$ as $\mathbf{C}_f$. Similarly, define $\mathbf{b}_{f,d}$ as a vector containing the frequency estimates $\hat{f}_{i,j}^{\alpha,d}$ for the Doppler shift case, ordered such that the entries align properly with the rows of $\mathbf{A}_{f,d}$. Note that for future applications, it will be convenient to directly compute the clock drifts rather than the relative velocity – in this case, a matrix may be defined as $\mathbf{A}_{f,v}$ in which the rows are constructed with coefficients solving for relative velocity directly rather than Doppler shift (for example, as in (4.20), but with only a single equation for one of the tones). In the same way, $\mathbf{b}_{f,d}$ has covariance $\mathbf{C}_{f,d}$. A discussion of how to compute these covariance matrices is provided in Section 4.4.3. The WLS solution in the stationary case may therefore be computed by

$$\hat{\mathbf{x}}_\alpha = \left(\mathbf{A}_f^T\mathbf{C}_f^{-1}\mathbf{A}_f\right)^{-1}\mathbf{A}_f^T\mathbf{C}_f^{-1}\mathbf{b}_f, \tag{4.88}$$

while the WLS solution in the Doppler case is given by

$$\hat{\mathbf{x}}_f = \left(\mathbf{A}_{f,d}^T\mathbf{C}_{f,d}^{-1}\mathbf{A}_{f,d}\right)^{-1}\mathbf{A}_{f,d}^T\mathbf{C}_{f,d}^{-1}\mathbf{b}_{f,d}. \tag{4.89}$$

Finally, the solution covariance matrix for the stationary case, Σ_α, is computed by

$$\Sigma_\alpha = \left(\mathbf{A}_f^T \mathbf{C}_f^{-1} \mathbf{A}_f\right)^{-1}, \tag{4.90}$$

while the solution covariance matrix for the Doppler shift case, Σ_f, is computed by

$$\Sigma_f = \left(\mathbf{A}_{f,d}^T \mathbf{C}_{f,d}^{-1} \mathbf{A}_{f,d}\right)^{-1}. \tag{4.91}$$

Similarly, for the direct velocity estimation case, the WLS solution, $\hat{\mathbf{x}}_{f,v}$ is given by

$$\hat{\mathbf{x}}_{f,v} = \left(\mathbf{A}_{f,v}^T \mathbf{C}_{f,d}^{-1} \mathbf{A}_{f,v}\right)^{-1} \mathbf{A}_{f,v}^T \mathbf{C}_{f,d}^{-1} \mathbf{b}_{f,d}, \tag{4.92}$$

while the solution covariance matrix for the direct velocity estimation case, $\Sigma_{f,v}$, is computed by

$$\Sigma_{f,v} = \left(\mathbf{A}_{f,v}^T \mathbf{C}_{f,d}^{-1} \mathbf{A}_{f,v}\right)^{-1}. \tag{4.93}$$

4.4.1.2 Clock Bias WLS

In the case where direct estimates of clock bias are desired instead of normalization to the network mean, WLS can be used to solve the system of equations in (4.52). Let $\mathbf{A}_\phi$ be the $N_p^2/2 - N_p/2 + 1) \times N_p$ matrix describing the coefficients of ϕ_i and ϕ_j in the system of equations in (4.52) (note that $m_{i,j}$ and $m_{j,i}$ are used to compute $\hat{\phi}_{i,j}$ but are not included in this system of equations). Let $\hat{\mathbf{x}}_\phi$ be a vector of the clock bias estimates produced by least-squares estimation, and let $\mathbf{b}_\phi$ be a vector containing the clock bias difference estimates $\hat{\phi}_{i,j}$, which can be computed either from (4.52) or from (4.65). Define the covariance of $\mathbf{b}_\phi$ as $\mathbf{C}_\phi$. A description of how to compute $\mathbf{C}_\phi$ is provided in

Section 4.4.3. The WLS solution will be computed by

$$\hat{\mathbf{x}}_\phi = \left(\mathbf{A}_\phi^T \mathbf{C}_\phi^{-1} \mathbf{A}_\phi\right)^{-1} \mathbf{A}_\phi^T \mathbf{C}_\phi^{-1} \mathbf{b}_\phi, \tag{4.94}$$

while the solution covariance, Σ_ϕ, is computed by

$$\Sigma_\phi = \left(\mathbf{A}_\phi^T \mathbf{C}_\phi^{-1} \mathbf{A}_\phi\right)^{-1}. \tag{4.95}$$

4.4.1.3 Carrier Phase WLS

Finally, for carrier phase estimates, WLS can be used to solve the system of equations in (2.69). As before, let $\mathbf{A}_\gamma$ be the $N_p^2 - N_p + 1) \times 2N_p$ matrix describing the coefficients of γ_j^{tx} and γ_i^{rx} in the system of equations in (2.69). Let $\hat{\mathbf{x}}_\gamma$ be a vector of the carrier phase estimates produced by least-squares estimation, and let $\mathbf{b}_\gamma$ be a vector containing the carrier phase difference estimates $\hat{\gamma}_{i,j}^{\text{err}}$. Define the covariance of $\mathbf{b}_\gamma$ as $\mathbf{C}_\gamma$. A description of how to compute $\mathbf{C}_\gamma$ is provided in Section 4.4.3. The WLS solution will be computed by

$$\hat{\mathbf{x}}_\gamma = \left(\mathbf{A}_\gamma^T \mathbf{C}_\gamma^{-1} \mathbf{A}_\gamma\right)^{-1} \mathbf{A}_\gamma^T \mathbf{C}_\gamma^{-1} \mathbf{b}_\gamma, \tag{4.96}$$

while the solution covariance, Σ_γ, is computed by

$$\Sigma_\gamma = \left(\mathbf{A}_\gamma^T \mathbf{C}_\gamma^{-1} \mathbf{A}_\gamma\right)^{-1}. \tag{4.97}$$

4.4.1.4 Unwrapping Phase Values

Because the phase of a complex number will by necessity be computed between $-\pi$ and π, it is possible that the system of equations for phase in (4.96) cannot be

solved correctly using the known phase estimates in $\mathbf{b}_\gamma$ since the difference between two wrapped phase values may fall outside the range of $-\pi$ and π. Therefore, to enable correct phase estimation, the phase values in $\mathbf{b}_\gamma$ must be unwrapped properly such that the estimation in (4.96) will produce valid phase values.

Each individual linear equation in $\mathbf{A}_\gamma$ only has two variables as dictated by (2.69). Since it is initially assumed that $\gamma_1^{\text{tx}} = 0$, the equations of the form $\hat{\gamma}_{i,1}^{\text{err}} = \gamma_1^{\text{tx}} - \gamma_i^{\text{rx}}$ may be solved for all γ_i^{rx} with $i \geq 2$ using the known estimates of $\hat{\gamma}_{i,1}$. Let $\tilde{\gamma}_i^{\text{rx}}$ describe the estimates of γ_i^{rx} calculated in this way, and similarly define $\tilde{\gamma}_j^{\text{tx}}$. From the estimates of $\tilde{\gamma}_i^{\text{rx}}$ for $i \geq 2$, all remaining estimates for $\tilde{\gamma}_j^{\text{tx}}$ may be computed, with the assumption that $\tilde{\gamma}_1^{\text{tx}} = 0$. Finally, any of the estimates $\tilde{\gamma}_j^{\text{tx}}$ with $j \neq 1$ may be used to compute $\tilde{\gamma}_1^{\text{rx}}$. Let $\tilde{\mathbf{x}}_\gamma$ be the vector containing the estimates $\tilde{\gamma}_j^{\text{tx}}$ and $\tilde{\gamma}_i^{\text{rx}}$ with an identical ordering to $\mathbf{x}_\gamma$.

The estimates in $\tilde{\mathbf{x}}_\gamma$ will be serviceable for a moderate SNR but will not achieve the minimum mean-squared error since they are not computed using the WLS solution in (4.96). To use the WLS solution, the original values of $\hat{\gamma}_{i,j}^{\text{err}}$ in $\mathbf{b}_\gamma$ must be unwrapped. First, compute a new estimate of $\mathbf{b}_\gamma$, $\tilde{\mathbf{b}}_\gamma$, given by

$$\tilde{\mathbf{b}}_\gamma = \mathbf{A}_\gamma \tilde{\mathbf{x}}_\gamma. \tag{4.98}$$

Because these values are recomputed from true estimates of the carrier phase values rather than the range delay values, they will include the required phase wrappings and are not required to fall in the range of $-\pi$ to π. A vector of the phase wrapping amounts, $\mathbf{b}_{\text{wrap}}$, can be determined by computing the difference between $\mathbf{b}_\gamma$ and $\tilde{\mathbf{b}}_\gamma$ and rounding each element of the difference vector to the nearest multiple of 2π as

$$\mathbf{b}_{\text{wrap}} = \text{round}\left(\tilde{\mathbf{y}}_\gamma - \mathbf{y}_\gamma,\, 2\pi\right). \tag{4.99}$$

Finally, the WLS solution may be computed by

$$\hat{\mathbf{x}}_\gamma = \left(\mathbf{A}_\gamma^T \mathbf{C}_\gamma^{-1} \mathbf{A}_\gamma\right)^{-1} \mathbf{A}_\gamma^T \mathbf{C}_\gamma^{-1} \left(\mathbf{b}_\gamma + \mathbf{b}_{\text{wrap}}\right). \tag{4.100}$$

4.4.2 Cramér-Rao Lower Bound Analysis

To proceed with the statistical performance analysis, it is convenient to have a closed-form expression for the variance of the frequency, time-delay, and phase measurements from which the synchronization variables are derived. Because of the nonlinear nature of the sinc NL-LS estimation, the explicit analysis of the variance of these estimators is difficult. A simpler approach is to compute the CRLB of the time-delay, frequency, and phase measurements, and then show by simulation that the estimators employed here meet it over a relevant range of SNRs.

The CRLB is a statistical bound that describes the theoretical minimum achievable variance of an estimator given the PDF of a set of data parameterized by the quantity to be estimated. For a scalar parameter φ, the variance of an unbiased estimator $\hat{\varphi}$ will satisfy [109]

$$\text{var}(\hat{\varphi}) \geq \text{CRLB}(\varphi) = \frac{1}{-E\left[\frac{\partial^2 \ln p(\mathbf{x};\varphi)}{\partial \varphi^2}\right]}, \tag{4.101}$$

where $E[\cdot]$ denotes the expected value operator, $\mathbf{x}$ denotes the data available to the estimator, and $p(\mathbf{x};\varphi)$ denotes the PDF of $\mathbf{x}$ parameterized by φ. This expression is the inverse of a quantity called the *Fisher information* of the parameter to be estimated. An estimator that meets the minimum possible variance is referred to as *efficient*. It is worth noting that the CRLB only describes the minimum variance achievable by an estimator given a particular data model, but does not usually provide a way of determining the form of the estimator or even describing whether such an estimator exists.

Furthermore, for more complex estimators, it may be intractable to analytically show that an estimator meets the CRLB, but it can sometimes be shown through simulation that the estimator meets the CRLB in some cases. This is the approach taken in this analysis.

To begin, the CRLB of the variance of the measurement of the frequency parameter $\hat{f}$, denoted σ_f^2, of a complex sinusoid in AWGN is given by [109]

$$\mathrm{var}\left(\hat{f}\right) = \sigma_f^2 \geq \frac{3}{2\pi^2 \cdot T_p^3 \cdot f^s \cdot \mathrm{SNR} \cdot \left(1 - \left(\frac{1}{T_p f^s}\right)^2\right)}. \tag{4.102}$$

Furthermore, it is shown in [109] that the efficient estimator for f is to identify the frequency value for which the periodogram is maximized – that is, for which the power spectral density (PSD) of the signal is maximized. This approach to finding the maximum frequency value in a frequency-domain representation of the data is identical to the sinc NL-LS frequency estimation approach – however, to do so using the periodogram by itself would require up-sampling the data to achieve high resolution or would ultimately degrade the estimator performance due to low frequency-resolution. Because the frequency estimation through sinc NL-LS aims to achieve the same end as maximizing the periodogram (just in a computationally more efficient way), it is intuitive to assume that the sinc NL-LS frequency estimator will be efficient.

To demonstrate this, Fig. 4.12 shows a simulation case study for frequency estimation using sinc NL-LS as proposed here, using a sampling frequency of $f^s = 100$ MHz and a pulse length of $T_p = 200\,\mu$s. Each data point is computed as the standard deviation of error over 1000 independent trials. In this simulation, the FFT is zero-padded to up-sample by a factor of 4, which has been determined through experience to increase the accuracy of the frequency estimation. Two simulation cases are shown in

Fig. 4.12: one in which the data log is not truncated and contains only noise before and after the pulse location within the data capture, as would likely be the case in the proposed synchronization technique, and one in which the data log is truncated such that the frequency may be observed throughout the full data log. It may be seen that for the truncated case, the estimator variance meets the CRLB for SNR values greater than -25 dB, while for the non-truncated case, the estimator variance closely approaches but does not quite meet the CRLB for the same values of SNR. However, because in either case the variance closely approaches the CRLB, the CRLB expression in (4.102) may be used to directly estimate σ_f^2. It should also be noted that the point at which the estimation begins to meet the CRLB will vary somewhat with parameters such as the pulse length and sampling frequency.

Next, the CRLB of the variance of the measurement of a time-delay parameter $\hat{\tau}$, denoted σ_τ^2, of a complex signal in additive white Gaussian noise (AWGN) is given by [92]

$$\text{var}\,(\hat{\tau}) = \sigma_\tau^2 \geq \frac{3}{2\pi^2 \cdot B^2 \cdot f^s \cdot T_p \cdot \text{SNR}} \tag{4.103}$$

where B is the signal bandwidth. Note that the SNR in the above equation refers to the input SNR and not the output SNR after the matched-filtering is performed. A simulation case study showing that the time-delay estimation using sinc NL-LS meets the CRLB is given in [92]. Fig. 4.13 shows a plot of the standard deviation of the time-delay estimates as a function of SNR for a receiver with $f^s = 100$ MHz, $B = 50$ MHz, and $T_p = 200\,\mu$s. As verified in this similar case study, the time-delay estimate applied here will meet the CRLB for values of SNR greater than -25 dB, and thus the measurement variance, σ_τ^2, may be computed using the CRLB expression in (4.103). Note that the SNR level at which the time-delay estimate begins to meet the CRLB will depend on signal parameters, though in general the estimate will meet

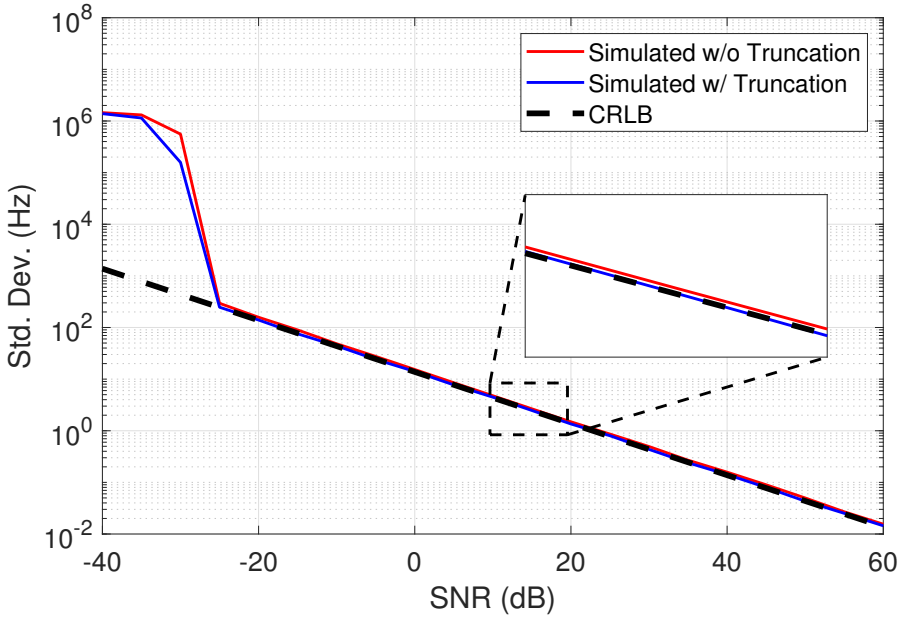


Fig. 4.12: The simulated frequency estimate standard deviation compared to the CRLB.

the CRLB so long as SNR is greater than 0 dB.

Finally, the CRLB of the variance of the measurement of a phase parameter $\hat{\gamma}$, denoted σ_γ^2, of an arbitrary complex pulsed signal in AWGN must be derived as the typical phase CRLB listed in traditional sources such as [109] typically refers to the estimation of the phase of a single-frequency sinusoid rather than an arbitrary signal such as the pulse-compression LFM waveform used in this work. To begin, let $p(t)$ be an arbitrary complex signal with pulse length T_p and bandwidth B that is non-zero only between $t = 0$ and $t = T_p$. Let the measured signal $s(t)$ be given by

$$s(t) = p(t)\, e^{j\gamma}, \tag{4.104}$$

where γ is the unknown phase to be estimated. The signal samples to be leveraged in the hypothetical estimator are sampled with a sampling time of T_s, with a sampling

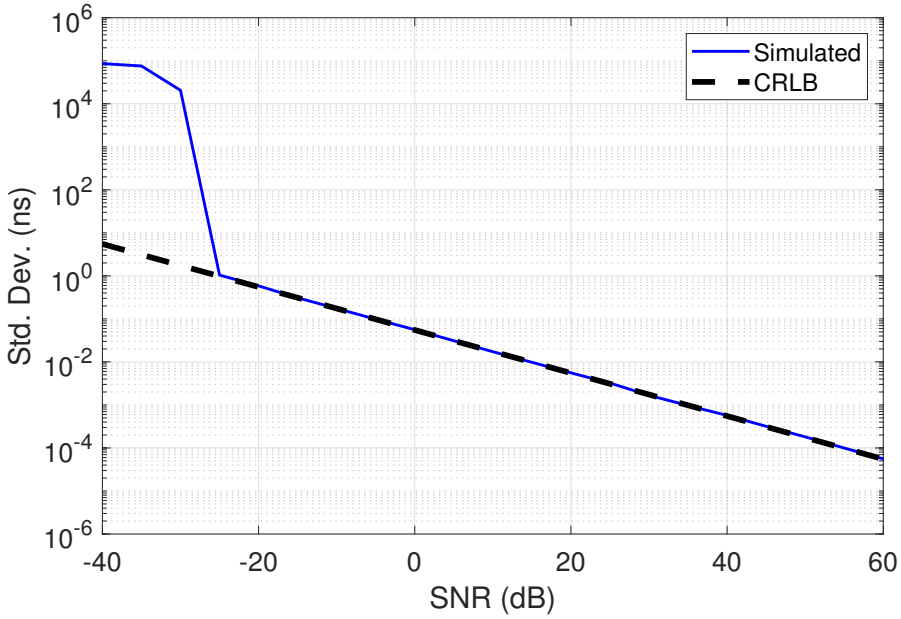


Fig. 4.13: The simulated time-delay estimate standard deviation compared to the CRLB.

frequency $f^s = 1/T_s$, giving the sampled signal

$$s[n] = p\,(nT_s)\,e^{\mathrm{j}\gamma}. \tag{4.105}$$

Assuming the signal is embedded in AWGN with noise variance σ_N^2, the Fisher information for γ is provided in a simplified form by [109]

$$I(\gamma) = \frac{2}{\sigma_N^2}\sum_{n=0}^{N_s-1}\left|\frac{\partial s[n;\,\gamma]}{\partial\gamma}\right|^2, \tag{4.106}$$

where N_s is the number of samples in the signal. The partial derivative of $s[n]$ with respect to γ is given by

$$\frac{\partial s[n;\,\gamma]}{\partial\gamma} = \mathrm{j}\,p\,(nT_s)\,e^{\mathrm{j}\gamma}, \tag{4.107}$$

such that (4.106) may be rewritten as

$$I(\gamma) = \frac{2}{\sigma_N^2} \sum_{n=0}^{N_s-1} |p(nT_s)|^2 . \tag{4.108}$$

Noting that only the non-zero samples of $p(nT_s)$ will contribute to the sum, the summation bounds may be changed to give

$$I(\gamma) = \frac{2}{\sigma_N^2} \sum_{n=0}^{M_p-1} |p(nT_s)|^2 , \tag{4.109}$$

where M_p is the number of samples within the pulse length and is given by the floor of T_p/T_s. The summation may be approximated by an integral, yielding

$$I(\gamma) \approx \frac{2}{\sigma_N^2} \frac{1}{T_s} \int_0^{T_p} |p(t)|^2 \, dt. \tag{4.110}$$

Noting that the power of the signal P_s is given by

$$P_s = \frac{1}{T_p} \int_0^{T_p} |p(t)|^2 \, dt, \tag{4.111}$$

the Fisher information can be expressed by

$$I(\gamma) = \frac{2}{\sigma_N^2} \frac{P_s T_p}{T_s}. \tag{4.112}$$

Observing that the SNR is given by P_s/σ_N^2, and noting that $T_s = 1/f^s$, the substitutions yield

$$I(\gamma) = 2 \cdot T_p \cdot f^s \cdot \text{SNR}. \tag{4.113}$$

Finally, inverting $I(\gamma)$ yields the final CRLB for γ

$$\text{var}\left(\hat{\gamma}\right) = \sigma_{\gamma}^{2} \geq \frac{1}{2 \cdot T_p \cdot f^s \cdot \text{SNR}}, \tag{4.114}$$

which is the final result for the phase CRLB.

Using the same signal parameters as the simulation demonstrated in Fig. 4.13, Fig. 4.14 shows that the phase estimate meets the CRLB for SNR values greater than -25 dB but less than 40 dB. This deviation from the CRLB after 40 dB is due to the error in phase lookup from the digitized data since the integer sample peak is used for computing phase rather than interpolating the phase in between samples during the sinc NL-LS process; however, it will usually be inconsequential given the high likelihood that a reasonable SNR will fall within the -25 to 40 dB range. Because the SNR will likely fall within this range, the measurement variance, σ_{γ}^{2}, may be computed using the CRLB expression in (4.114).

4.4.3 Covariance of Estimates

To properly perform WLS, good estimates of the covariance matrices for the frequency, time, and phase measurements are necessary. These matrices can be constructed under two key assumptions: that the estimates each meet or closely approach the CRLB, which has been shown above, and that the measurements are independent, and as such the covariance matrices may be diagonal with each entry along the diagonal being computed by the associated CRLB expression. The independence assumption follows from the fact that the error in each measurement arises from the introduction of AWGN, which by definition will not introduce correlated errors.

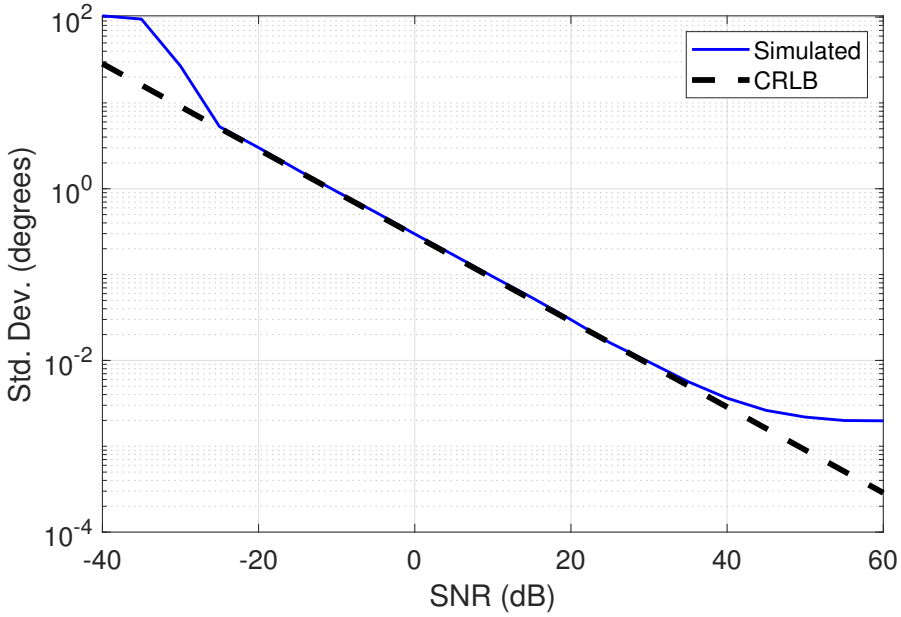


Fig. 4.14: The simulated phase estimate standard deviation compared to the CRLB.

First, for the frequency estimates, $\mathbf{C}_f$ and $\mathbf{C}_{f,d}$ may both be computed as

$$\mathbf{C}_f = \mathbf{C}_{f,d} = \text{diag}\left(\left[V_{1,2}^f \; V_{1,3}^f \; \ldots \; V_{1,N_p}^f \; V_{2,1}^f \; V_{2,3}^f \; \ldots \; V_{N_p-1,N_p}^f \; \epsilon_f\right]\right), \tag{4.115}$$

where $V_{i,j}^f$ is defined as $V_{i,j}^f = \text{var}\left(\hat{f}_{i,j}^{\alpha}\right) = \text{var}\left(\hat{f}_{i,j}^{\alpha,d}\right)$, which may be computed using the CRLB in (4.102), and where the diag $(\cdot)$ operator indicates a diagonal matrix with the enclosed vector containing the diagonal entries. The value of ϵ_f is included to describe the variance in the solution to $\alpha_1 = 1$, which is nominally 0, though in a real computation should be set to an arbitrarily low non-zero value to avoid numerical errors. This covariance matrix describes the covariance of the vector containing all the values of $\eta_{i,j}^{\alpha}$ or $\eta_{i,j}^{\alpha,d}$. Note that because in general this frequency estimation will not be done with the data truncated to the width of the single-tone pulse, the variance of the frequency measurement will likely not meet the CRLB. Assuming an up-sampling of the FFT by a factor of 4, the measurement variance should approach closely to the CRLB, and therefore (4.102) may still be applied with minimal error.

The time-delay measurements, $m_{i,j}$, are produced with variance $V^{\tau}_{i,j}$ computed from the CRLB in (4.103). Let $V^{R}_{i,j}$ be the variance of the range measurements $\hat{R}_{i,j}$. From (4.51), $V^{R}_{i,j}$ may be computed by

$$V^{R}_{i,j} = \frac{c^2}{4}\left(V^{\tau}_{i,j} + V^{\tau}_{j,i}\right). \tag{4.116}$$

The range estimates can be constructed into a covariance matrix, Σ_R, by

$$\Sigma_R = \text{diag}\left(\left[V^{R}_{1,2}\ V^{R}_{1,3}\ \ldots\ V^{R}_{1,N_p}\ V^{R}_{2,3}\ \ldots\ V^{R}_{N_p-1,N_p}\ \right]\right). \tag{4.117}$$

Similarly, define $V^{\phi}_{i,j}$ as the variance of the clock bias offset estimates $\hat{\phi}_{i,j}$. From (4.52), $V^{\phi}_{i,j}$ may be computed by

$$V^{\phi}_{i,j} = \frac{1}{4}\left(V^{\tau}_{i,j} + V^{\tau}_{j,i}\right). \tag{4.118}$$

If the direct estimates of clock bias are desired, the covariance matrix $\mathbf{C}_{\phi}$ can be constructed in a similar manner to $\mathbf{C}_f$ by

$$\mathbf{C}_{\phi} = \text{diag}\left(\left[V^{\phi}_{1,2}\ V^{\phi}_{1,3}\ \ldots\ V^{\phi}_{1,N_p}\ V^{\phi}_{2,3}\ \ldots\ V^{\phi}_{N_p-1,N_p}\ \epsilon_{\phi}\right]\right). \tag{4.119}$$

In a similar fashion to ϵ_f, the value of ϵ_{ϕ} is included to describe the variance in the solution to $\phi_1 = 0$, which is nominally 0, though in a real computation should be set to an arbitrarily low non-zero value to avoid numerical errors.

Finally, the post-pulse compression phase measurements, $\angle d_{i,j}(m_{i,j})$, are produced with variance $V^{\gamma}_{i,j}$ computed from the CRLB in (4.114). From (4.68), the variance of the carrier phase difference estimates, $V^{\text{err}}_{i,j}$, may be computed by

$$V^{\text{err}}_{i,j} = V^{\gamma}_{i,j} + (2\pi f^{c})^2\left(\frac{V^{\tau}_{i,j}}{2} + \frac{V^{\tau}_{j,i}}{2}\right). \tag{4.120}$$

From here, the covariance matrix $\mathbf{C}_\gamma$ may be defined as

$$\mathbf{C}_\gamma = \text{diag}\left(\left[V_{1,2}^{\text{err}}\ V_{1,3}^{\text{err}}\ \ldots\ V_{1,N_p}^{\text{err}}\ V_{2,1}^{\text{err}}\ V_{2,3}^{\text{err}}\ \ldots\ V_{N_p-1,N_p}^{\text{err}}\ \epsilon_{\text{err}}\right]\right), \tag{4.121}$$

where, similarly to the clock drift case, the value of ϵ_{err} is included to describe the variance in the solution to the equation $\gamma_1^{\text{tx}} = 0$, which is nominally 0, though in a real computation should be set to an arbitrarily low non-zero value to avoid numerical errors.

4.4.4 Computing SNR

During the runtime of the proposed synchronization algorithm, it will be important for an accurate estimate of the SNR of each signal to be available for the correct construction of the covariance matrices for use in WLS estimation. A large body of literature exists on this topic, though much of it is specifically devoted to SNR estimation for communications signals [110]. The approach used in this work is the matched-filter-based approach in [111], which has the advantage of being fast and requiring only one pulse of the signal to achieve high accuracy.

4.4.5 Statistical Analysis Simulated Results

The proposed technique is evaluated here in terms of RMSE as a function of several system and signal parameters. To confirm the theoretical performance, the RMSE is also compared directly to the CRLB derived previously in this section. The following normalizations are performed in this analysis. Since it is assumed that $\alpha_1 = 1$, all true clock drift values are normalized by α_1 to ensure a fair error comparison. For the clock drift and carrier phase estimates, α_1 and γ_1^{tx} are omitted from the RMSE com-

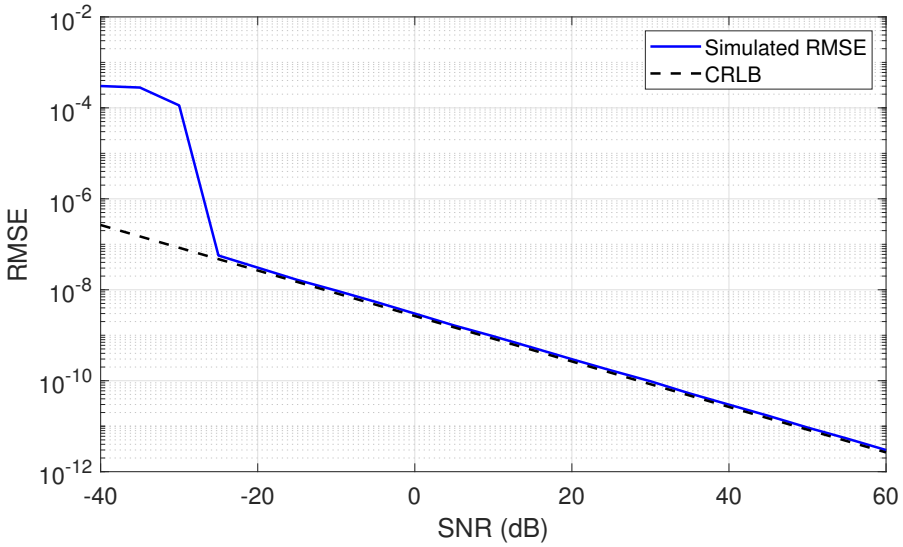


Fig. 4.15: The simulated clock drift estimate RMSE compared to the predicted variance derived from the CRLB as a function of SNR.

putation to avoid artificially reducing the RMSE since their error is zero by definition. For the computation of the CRLB for clock drift and carrier phase, which are extracted from the covariance matrices Σ_α and Σ_γ, the CRLB is computed by taking the mean of the variance values along the diagonal. In a similar fashion to the RMSE, the variance values associated with α_1 and γ_1^{tx} are omitted from this computation.

In each of the following simulations, the following errors are included. First, the clock drift values are simulated as Gaussian distributed random values with a mean of 1 and a standard deviation of $\sigma_\alpha = 25 \times 10^{-6}$. The clock biases are simulated as Gaussian distributed random values with a mean of 0 and standard deviation of $\sigma_\phi = 1 \times 10^{-8}$ s. The TX and RX carrier phase values are uniformly distributed between $-\pi$ and π. Finally, the platforms are simulated to have three-dimensional positions whose x, y, and z position values are simulated as Gaussian distributed random values with a mean of 0 and a standard deviation of $\sigma_p = 100$ m. Furthermore, all of the following simulations use the single-tone with exchange technique for clock drift estimation. A TDMA time slot of 1 ms is used for all simulations as well, which will produce a

noticeable impact when platform motion is included.

4.4.5.1 Stationary Platforms

The first set of simulations concerns the case of stationary platforms. These will have the property that Doppler shift need not be considered. Crucially, this means that time-delay errors due to incorrect Doppler estimation will not reduce range, bias, and phase estimation accuracy. The case of Doppler shift is considered in Section 4.4.5.2. First, the synchronization performance is analyzed as a function of SNR. This simulation is performed with a pulse length of $T_p = 200$ μs, a carrier frequency of $f^c = 3$ GHz, a sampling frequency of $f^s = 100$ MHz, and a signal bandwidth of $B = 50$ MHz for the time-delay estimation. Each data point is computed from 1000 independent simulated trials. Fig. 4.15 shows the RMSE for the clock drift estimates, $\hat{\alpha}_i$, in this scenario along with the variance predicted as derived from the frequency estimate CRLB. The RMSE fails to exactly converge to the CRLB, which is a direct result of the frequency estimates not reaching the CRLB due to the lack of pulse truncation, as shown in Fig. 4.12. For these waveform and system parameters, an SNR of 0 dB leads to an RMSE of 3 parts per billion (ppb), which at 3 GHz leads to a carrier frequency misalignment of 9 Hz; at 30 dB, the RMSE goes below 100 parts per trillion (ppt), leading to a carrier frequency misalignment of 0.3 Hz. As with the frequency CRLB simulation in Fig. 4.12 produced with similar parameters, the RMSE begins to approach the CRLB for SNR values above -25 dB.

For the same simulation parameters, Fig. 4.16 shows the RMSE for the clock bias difference estimate, $\hat{\phi}_{i,j}$, along with the variance predicted by the time-delay CRLB. As with the clock drift estimate, the RMSE closely approaches but does not meet the CRLB. However, this is not a result of the time-delay estimate accuracy, which is

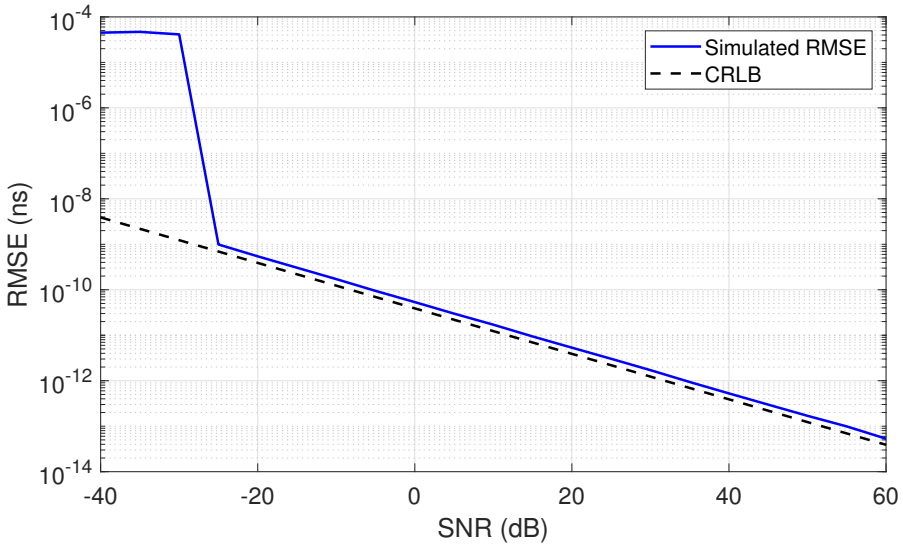


Fig. 4.16: The simulated clock bias estimate RMSE compared to the predicted variance derived from the CRLB as a function of SNR.

demonstrated in Fig. 4.13 to meet the CRLB. Instead, this is due to the clock drift estimate error, which adds a minor frequency error to the time-delay waveform due to the carrier frequency skew. This frequency error results in a small but non-zero shift in the pulse compression output as predicted by the ambiguity function of the chirp waveform. This time-delay error ultimately manifests as additional error in the clock bias estimates, forcing it to be slightly above the CRLB. Even so, the RMSE at 0 dB SNR is as low as 53.5 ps, while at the higher 30 dB of SNR the RMSE is less than 2 ps. As with the time-delay CRLB simulation in Fig. 4.13 produced with similar parameters, the RMSE begins to approach the CRLB for SNR values above -25 dB.

Similarly, Fig. 4.17 shows the RMSE for the range estimates, $\hat{R}_{i,j}$, along with the variance predicted by the time-delay CRLB. Fig. 4.17 shows two RMSE plots for range: an unnormalized plot in which the RMSE is computed using the true range value $R_{i,j}$, and a normalized plot in which the RMSE is computed using the range value scaled by the ratio of the clock drift estimate to the true clock drift (i.e., $\frac{\alpha_j}{\hat{\alpha}_j} R_{i,j}$). This is a valuable RMSE comparison since the ratio $\frac{\alpha_j}{\hat{\alpha}_j}$ will usually not equal 1 even if

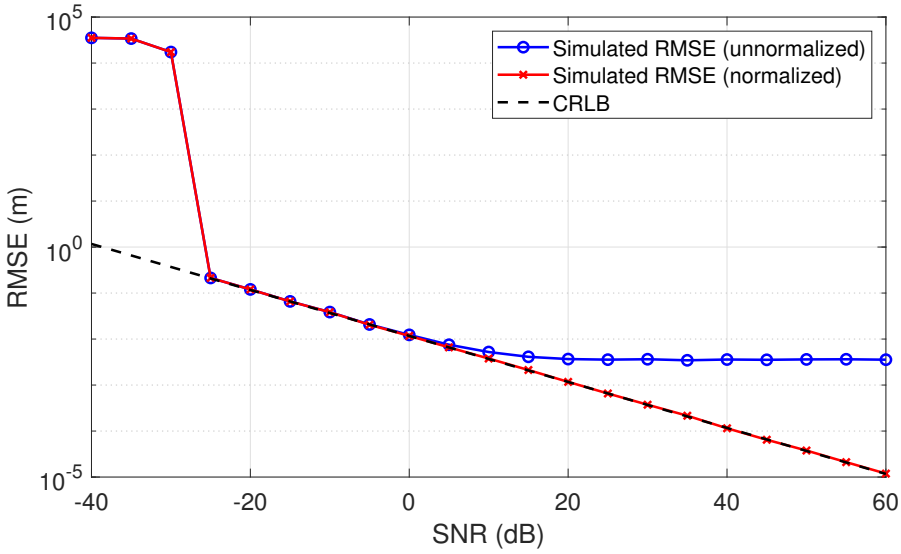


Fig. 4.17: The simulated range estimate RMSE compared to the predicted variance derived from the CRLB as a function of SNR. The blue plot shows the unnormalized RMSE computed against the true range values, while the red plot shows the normalized RMSE computed against the drift ratio scaled range values.

the clock drift values are perfectly estimated due to the assumption that $\alpha_1 = 1$. The unnormalized RMSE levels off at 3.5 mm error at 10 dB SNR, while the normalized RMSE meets the CRLB for all SNR values above -25 dB. Interestingly, unlike the clock bias estimate RMSE shown in Fig. 4.16, the normalized range estimate RMSE does meet the CRLB despite both estimates being derived from the same time-delay estimates. The reason for this is that the time-delay estimates $m_{i,j}$ and $m_{j,i}$ for a pair of platforms will have equal and opposite errors introduced by clock drift estimate errors due to the inverse clock skew between the platforms. Thus, since the range is computed in (4.51) as the average of the two time-delay measurements, these opposite errors will cancel each other out. This contrasts with the clock bias estimate, which is computed in (4.52) as the weighted difference of the time-delay estimates, and as such the opposite errors will add together.

Also using the same simulation parameters, Fig. 4.18 shows the RMSE for the

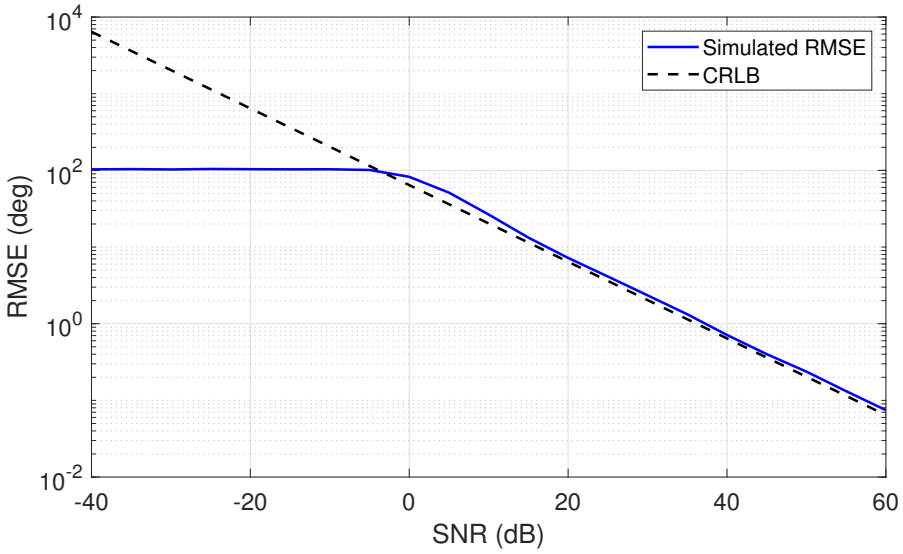


Fig. 4.18: The simulated carrier phase estimate RMSE compared to the predicted variance derived from the CRLB as a function of SNR.

carrier phase estimates, $\hat{\gamma}_i^{\text{tx}}$ and $\hat{\gamma}_i^{\text{rx}}$, along with the variance predicted by the time-delay and phase CRLB. As with the clock bias estimates, the carrier phase RMSE approaches but does not meet the CRLB exactly due to additional time-delay errors caused by the clock drift-induced frequency error. This plot highlights the difficulty in phase synchronization since, despite very nearly meeting the CRLB, it requires ~15 dB of SNR to achieve even 10° RMSE in carrier phase; to achieve 1° RMSE requires greater than 30 dB. At 0 dB SNR, the phase RMSE is at an unacceptable 100°. The primary hurdle in achieving high phase synchronization performance is achieving high time-delay estimation performance, since this component of the phase variance in (4.120) dominates the variance introduced by the phase measurement itself. Thus, increasing phase estimate accuracy in this case requires increasing the time-delay accuracy, which can be accomplished through a combination of higher bandwidths and sampling frequencies, longer pulse lengths, and greater SNR, as dictated by the CRLB in (4.103).

Fig. 4.19 shows the clock drift estimate RMSE as a function of carrier frequency,

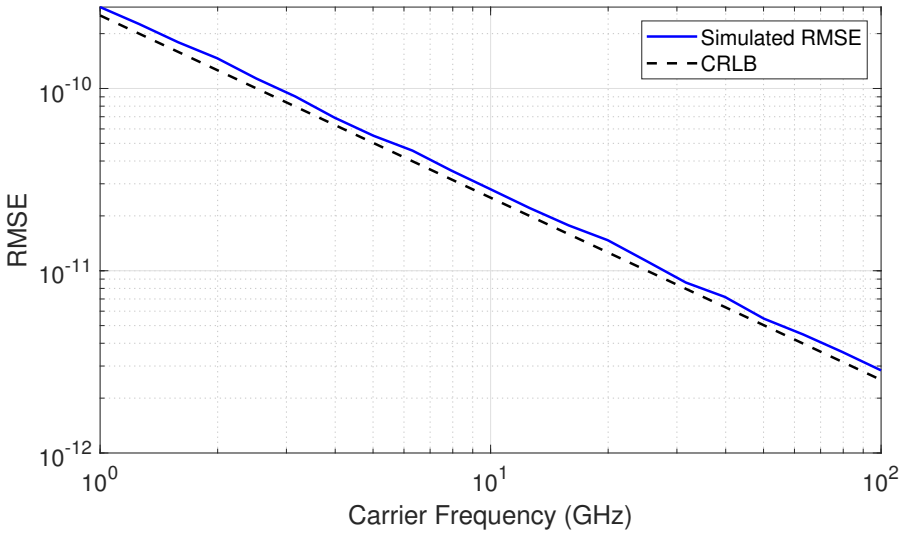


Fig. 4.19: The simulated clock drift estimate RMSE compared to the predicted variance derived from the CRLB as a function of carrier frequency f^c.

f^c, while Fig. 4.20 shows the RMSE as a function of the number of platforms in the network, N_p. In both of these, the SNR is held constant at 30 dB. First, these plots show that the clock drift will approach but not meet the CRLB due to the frequency estimate not meeting the CRLB. These plots also show that the clock drift estimate RMSE is inversely related to both the carrier frequency and network size, neither of which is included in the expression for frequency CRLB in (4.102). Instead, these parameters contribute to the predicted variance bound in the covariance computation in (4.90). For the carrier frequency case, a higher carrier frequency adds "resolution" to the drift estimation since varying the clock drift by some amount will lead to larger changes in frequency, while the frequency estimate itself retains the same accuracy. The downside to increasing carrier frequency is that the accuracy requirement for clock drift estimation increases to achieve the same level of frequency alignment between systems; as a result, this increased accuracy due to higher carrier frequency will be canceled out by the increased accuracy requirement. In the case of network size, a larger number of platforms yields a quadratically scaled number of equations and

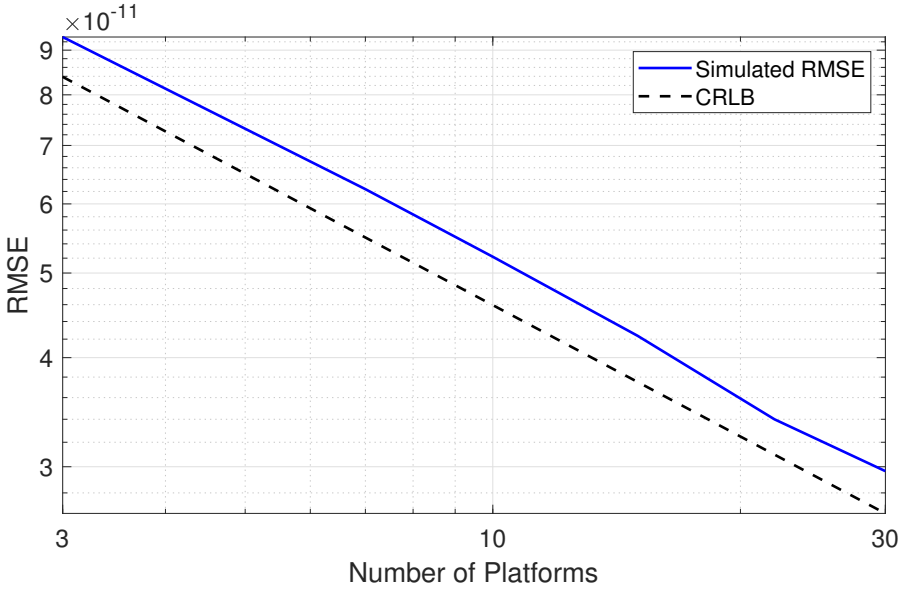


Fig. 4.20: The simulated carrier phase estimate RMSE compared to the predicted variance derived from the CRLB as a function of the network size N_p.

increases the degree to which the clock drift system of equations is overdetermined.

4.4.5.2 Moving Platforms

In the next scenario, platforms in the Monte Carlo simulation are simulated to have platform motion, with the velocity standard deviation in each dimension being set to $\sigma_v = 10$ m/s. Other than this, the properties of the simulation with the stationary platform case are unchanged. First, the clock drift RMSE along with predicted variance derived from the CRLB is shown in Fig. 4.21. When comparing to the stationary case in Fig. 4.15, the RMSE, as well as the CRLB, are unchanged, implying that there is no expected or actual impact on the clock drift estimation performance when platform motion is present. This is generally corroborated by the results in Fig. 4.9.

For the same simulation parameters, Fig. 4.22 shows the RMSE for the Doppler shift frequency estimate, $\hat{f}^d_{i,j}$, along with the variance predicted by the transformation of the frequency CRLB. As with the clock drift estimate, this closely approaches the

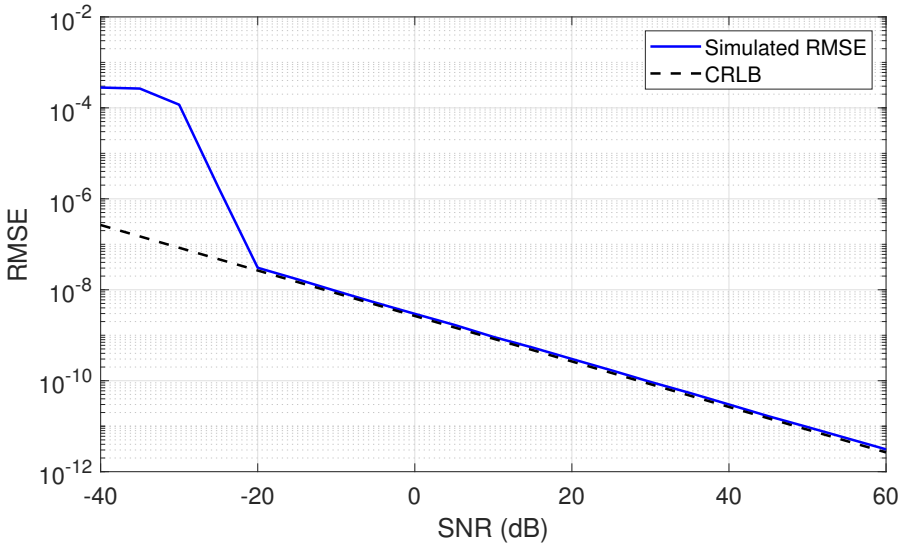


Fig. 4.21: The simulated clock drift estimate RMSE with platform motion compared to the predicted variance derived from the CRLB as a function of SNR.

performance predicted in the CRLB but does not reach it entirely. This is also due to the frequency estimates not reaching the CRLB due to the lack of pulse truncation, as shown in Fig. 4.12. It can be seen that the Doppler frequency estimation with these specific simulation parameters leads to a 10 Hz RMSE at an SNR of 0 dB, and decreases by 5 Hz per decade. Therefore, sub-Hz level accuracy is achievable for greater than 20 dB of SNR.

The RMSE for the clock bias difference estimate, $\hat{\phi}_{i,j}$, with platform motion is shown in Fig. 4.23, along with the variance predicted by the time-delay CRLB. Compared to the stationary case in Fig. 4.16, the RMSE is higher for all values of SNR. This is due to the increased frequency offset of the time-delay waveform, now caused both by carrier frequency skew present in the stationary case as well as the uncompensated Doppler shift resulting from error in the Doppler shift estimation, both of which lead to erroneous shifting as predicted by the ambiguity function. Although the performance is certainly lowered for the platform motion case, it is not lowered significantly, and moderate levels of SNR will still lead to clock bias estimation errors

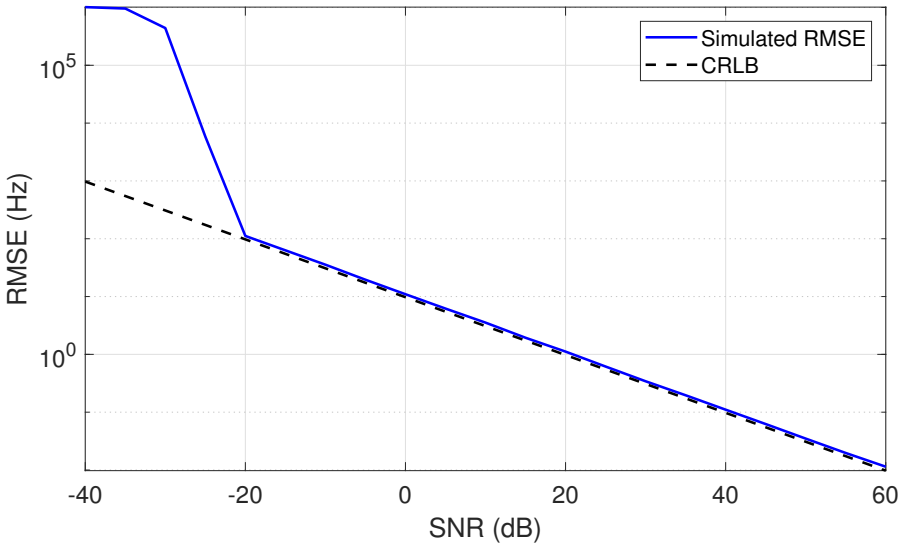


Fig. 4.22: The simulated Doppler shift estimate RMSE with platform motion compared to the predicted variance derived from the CRLB as a function of SNR.

in the low ps range.

The range RMSE is shown in Fig. 4.24. Recall from Section 4.2.2.2 that the range between platforms will not be symmetric when TDMA is used to multiplex the ranging signals during the time synchronization procedure. Therefore, the expression in (4.67) should be used in place of the expression in (4.51) to compensate for the amount the platforms move relative to one another between their respective time slots. It can be seen here that the performance of the range estimation is degraded. This will be primarily due to the imperfect estimation due to the TDMA – the motion will be compensated somewhat by (4.67), but this expression makes some assumptions and will therefore lead to error. In particular, the range estimates hit an accuracy floor after a certain level of SNR, and no additional SNR will improve the estimate RMSE. However, it can clearly be seen that both the drift normalized and unnormalized estimates are improved and have a lower SNR floor when the asymmetric range computation is used in place of the symmetric range computation. Furthermore, the unnormalized RMSE with the asymmetric range computation still reaches an RMSE of close to

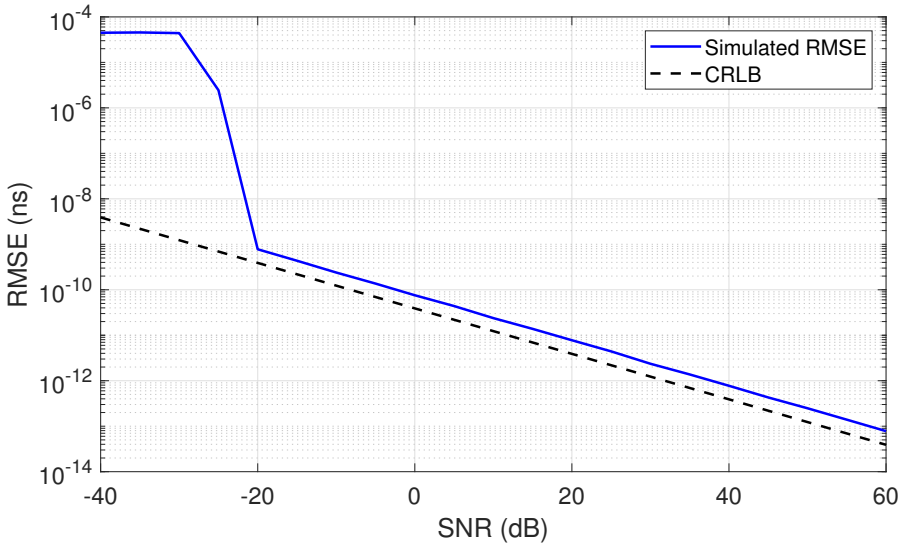


Fig. 4.23: The simulated clock bias estimate RMSE with platform motion compared to the predicted variance derived from the CRLB as a function of SNR.

5 mm at an SNR of 20 dB, which should be an acceptable error level for cooperative navigation.

Finally, Fig. 4.25 shows the RMSE for the carrier phase estimates, γ_i^{tx} and γ_i^{rx}, for the platform motion case, along with the variance predicted by the time-delay and phase CRLB. Due to the range estimate error in Fig. 4.24 at higher levels of SNR as a result of the TDMA velocity errors, the phase estimates also get lower at higher levels of SNR, leveling off at around 1° after 40 dB of SNR. This error will be difficult to overcome since, as discussed above, the carrier phase error will be dominated by the range RMSE rather than the direct phase estimation RMSE described by the CRLB shown in Fig. 4.14. Tor the lower values of SNR (0–30 dB), the carrier phase estimates will generally meet the same RMSE as the stationary case shown in Fig. 4.18 since the range RMSE will not strongly deviate from the CRLB over these values of SNR.

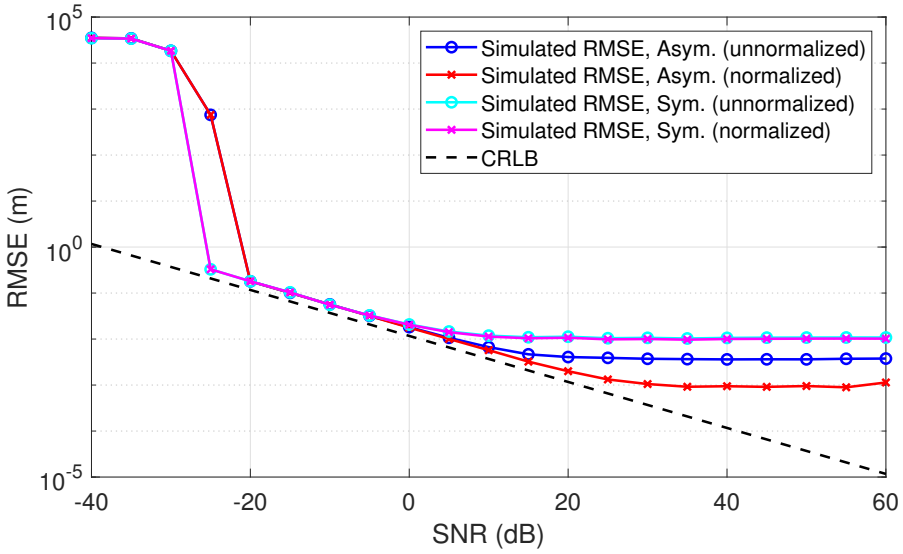


Fig. 4.24: The simulated range estimate RMSE with platform motion compared to the predicted variance derived from the CRLB as a function of SNR. The blue plot shows the unnormalized RMSE computed against the true range values, while the red plot shows the normalized RMSE computed against the drift ratio scaled range values, both computed using the asymmetric range computation in (4.67). The cyan and magenta plots show the same unnormalized and normalized range RMSE but using the symmetric range computation in (4.51).

4.5 Radar Results

In this section, the proposed synchronization technique is demonstrated in two practical radar scenario simulations. In the first, a wirelessly distributed array of stationary transmitters is used to perform transmit beamforming toward a target location. The coherent energy gain of the transmitted signals at the destination is analyzed over a short period for varying values of SNR during the synchronization procedure. In the second, three platforms are simulated in a distributed SAR scenario with relative motion between the SAR platforms. The SAR PSRs are computed for several different values of SNR to show the synchronization effectiveness in scenarios with relative motion.

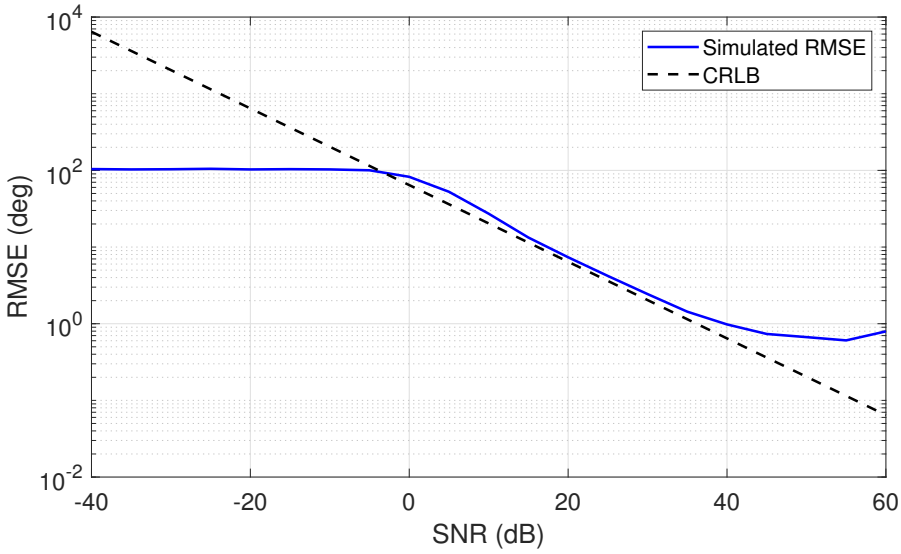


Fig. 4.25: The simulated carrier phase estimate RMSE with platform motion compared to the predicted variance derived from the CRLB as a function of SNR.

4.5.1 Distributed Beamforming Results

First, a simulated scenario is devised in which a distributed array of transmitter platforms are coordinated to transmit a single signal toward a known destination and beamform such that the transmitted signals add coherently at the destination. This scenario is simulated with and without synchronization, with synchronization performed at several different levels of SNR to demonstrate the coherent gain performance at the desired destination. The synchronization procedure used in this case is single-tone with exchange.

In the simulated scenario, the beamforming destination is placed at the origin, and 20 transmitters are placed in a line 1000 m from the destination. The transmitters are spaced 100 m apart and the line is centered along the line-of-sight vector to the destination. A diagram of the simulated scenario is provided in Fig. 4.26. In this scenario, it is assumed that each transmitter knows its own position, the positions of the other transmitters, and the position of the destination such that localization errors

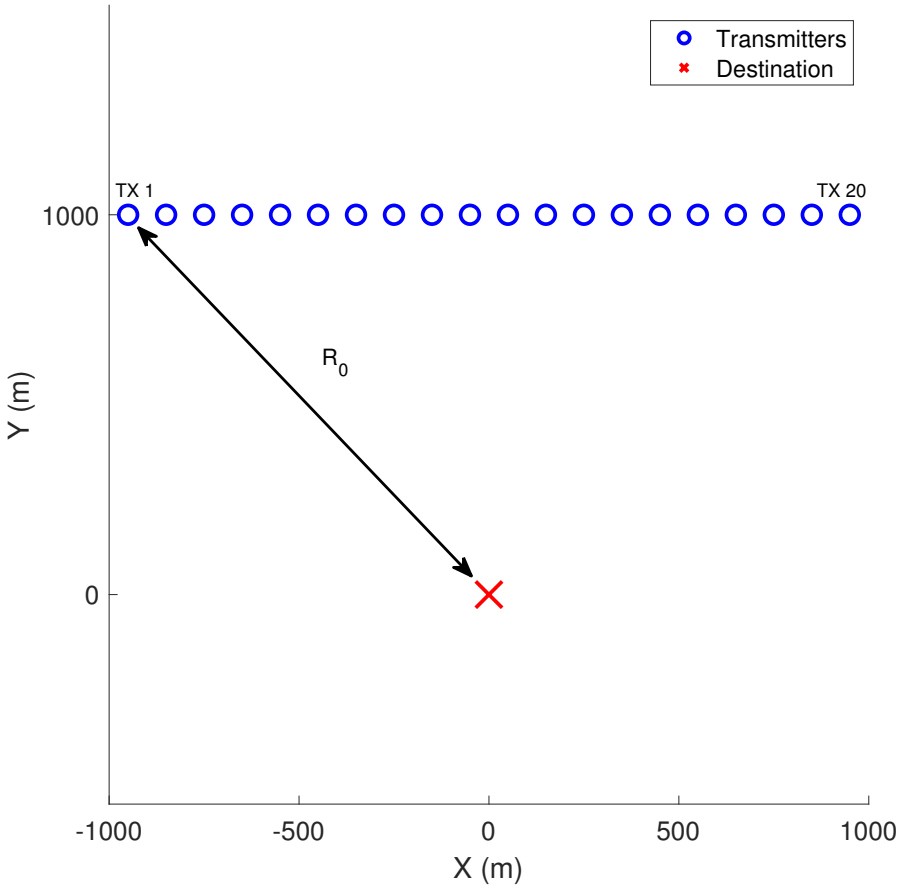


Fig. 4.26: The simulated distributed beamforming scenario.

will not contribute any loss to the coherent gain at the destination.

After synchronization, each platform transmitter begins transmitting LFM pulses toward the destination. In addition to applying the time and phase compensations required by the synchronization routine, each transmitter i also applies a fractional time delay, τ_0^i, and a phase shift, γ_0^i, relative to a reference range, R_0, such that its transmitted signal will arrive at the destination at the same time as a signal transmitted from a transmitter separated from the destination by a distance R_0. The time shift is determined by

$$\tau_0^i = \frac{R_0 - R_i}{c}, \tag{4.122}$$

where R_i is the distance between transmitter i and the destination, and the phase shift is determined by

$$\gamma_0^i = -2\pi f^c \frac{R_0 - R_i}{c}. \tag{4.123}$$

The time delay in conjunction with the phase shift will ensure that there is RF coherence as well as information coherence at the destination.

In this scenario, each transmitter transmits at a carrier frequency of $f^c = 3$ GHz. The transmitted signals are LFM waveforms with a pulse length of 5 μs and a bandwidth of 50 MHz. The reference range, R_0, is selected as the largest distance between the destination and any one transmitter, as shown in Fig. 4.26. The destination is simulated with reference to the clock drift of transmitter 1 (shown on the far left of Fig. 4.26) such that the simulated signal window will remain stationary over time.

The signal transmissions are simulated periodically over a 1 s time span to show how decoherence will occur over time due to the imperfectly corrected carrier frequencies at the transmitters. The clock biases are distributed with a low standard deviation of 10 ns such that at $t = 0$, even when unsynchronized, the transmitted signals will mostly appear within the signal window even though they will not coherently integrate due to phase and frequency offsets. Over time, however, the signals will drift out of the signal window and will not contribute to the coherent gain. For simplicity, the signals are simulated at baseband as though they were down-converted at the destination, though in reality, the destination would be a target reflecting the radar signals back to the transmitters.

Let t_d be the reference time at the destination, and let $r_{d,j}(t_d)$ be the signal transmitted by transmitter i in the reference time at the destination. The total signal observed

at the destination, $r_d(t_d)$, will be given by

$$r_d(t_d) = \sum_{i=1}^{20} r_{d,i}(t_d). \tag{4.124}$$

The energy E_r of the total signal may be computed by

$$E_r = \int |r_d(t_d)|^2 \, dt_d. \tag{4.125}$$

In this simulation, the energy computation is only performed over the time interval in which the signals are expected to appear with a 100 ns buffer on either side. As a result, signals with significant time drift due to unsynchronized clock drifts will drift outside this integration window and will not be included in the energy calculation.

Fig. 4.27 shows the result of the summation of the 20 incident signals at the destination for several different synchronization cases. For the case of no synchronization, the coherent sum has a very low amplitude at $t_d = 0$ owing to the lack of phase coherence amongst the signals. The amplitude is also modulated due to the lack of frequency synchronization amongst the transmitters. At $t_d = 1$, the individual signals have largely drifted out of the signal window and the sum has a low amplitude and does not match the LFM template at all. With synchronization at 10 dB SNR, the summation at $t_d = 0$ has an amplitude of nearly 20, indicating nearly perfect coherence. However, at $t_d = 1$, the amplitude has diminished significantly due to the loss of phase coherence owing to small but non-zero errors in carrier frequency amongst the transmitters. This problem is observed with 30 dB SNR to a much lesser degree, and the problem is nearly eliminated when synchronized with 50 dB SNR.

To compute an explicit metric for coherent gain, the ratio of the energy, E_r, in the summed signal to the energy in the ideal summed signal is computed for the different

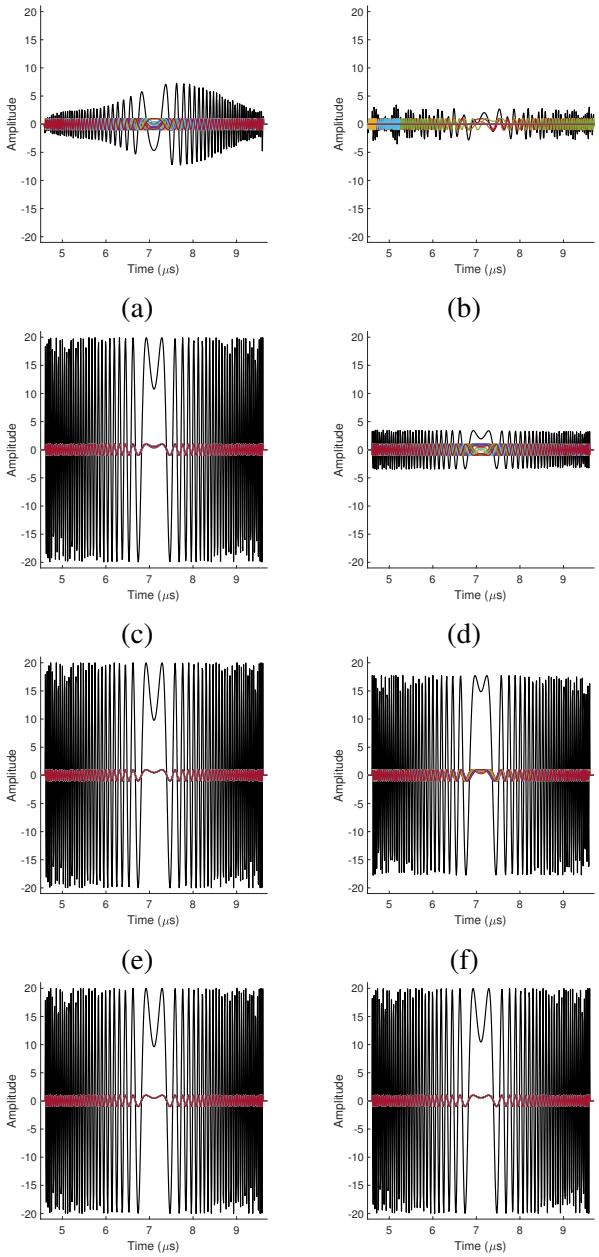


Fig. 4.27: The 20 individual signals at the destination and their sum: without synchronization, at $t_d = 0$ (a) and $t_d = 1$ (b); with synchronization at 10 dB SNR, at $t_d = 0$ (c) and $t_d = 1$ (d); with synchronization at 30 dB SNR, at $t_d = 0$ (e) and $t_d = 1$ (f); and with synchronization at 50 dB SNR, at $t_d = 0$ (g) and $t_d = 1$ (h). The coherent sum of the signals is shown in black.

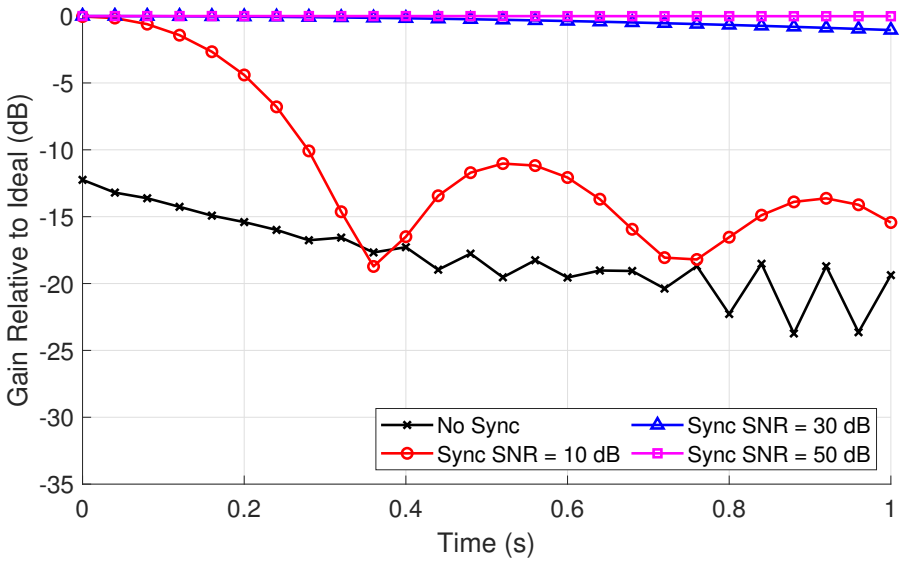


Fig. 4.28: The coherent energy gain of the beamformed signal, relative to the ideal case, as a function of time.

synchronization cases, such that 0 dB of coherent gain indicates perfect performance. This evaluation is performed over the 1 s simulation time to show how the gain performance changes with time. The results are given in Fig. 4.28. For the case of no synchronization, the gain performance starts below -10 dB at $t_d = 0$ and deteriorates further as time goes on, ending with -20 dB of gain. The synchronization case with 10 dB of SNR has nearly perfect performance at $t_d = 0$, but its performance quickly deteriorates due to the carrier frequency misalignment between transmitters. However, with 30 dB of SNR, only 1 dB of coherent gain is lost over the simulation time, and with 50 dB of SNR, the coherent gain loss is negligible. These results indicate that if the synchronization is performed in a case with lower SNR (<20 dB, for example), it will need to be performed again at regular intervals to maintain high performance. If larger values of SNR are expected, then the procedure can be performed less frequently.

4.5.2 SAR PSR Results

To demonstrate the impact of the synchronization procedure on radar performance, a distributed radar network scenario is simulated. The performance of the simulated network is compared with and without synchronization. The simulated scenario is a MIMO SAR mission in which three SAR platforms fly co-linearly at different velocities and with different sub-aperture lengths during the CPI. The platforms are simulated with random clock drifts, clock biases, and carrier phases to be compensated to enable MIMO operation.

In this scene, a single point target is located at the origin such that the generated SAR images describe the PSR of the simulated radar network, which provides a holistic description of the quality of the images formed by this particular MIMO SAR imaging configuration. Each system flies 9 km away from the target with a flight altitude of 300 m. Platform 1 flies with a velocity of 40 m/s, Platform 2 with a velocity of 50 m/s, and Platform 3 with a velocity of 60 m/s. The CPI is 1 s long and the platforms fly co-linearly but with an uneven spread between the subapertures such that the full aperture formed by the platforms is 360 m long with the gaps being filled due to the MIMO configuration equivalent phase centers [23]. Fig. 4.29 shows the SAR imaging geometry of this scene.

The radar systems operate with a carrier frequency of 3 GHz. Each system captures 1200 pulses at a PRF of 1200 Hz over the 1 s CPI. The platforms are configured to transmit in a MIMO configuration through TDMA in which each platform is assigned 1/3 of the pulse repetition time (PRT). The radar waveform used by each system is an LFM pulse with a length of 10 μs and a bandwidth of 100 MHz.

The simulated clock drifts α_i are randomly generated from a uniform distribution

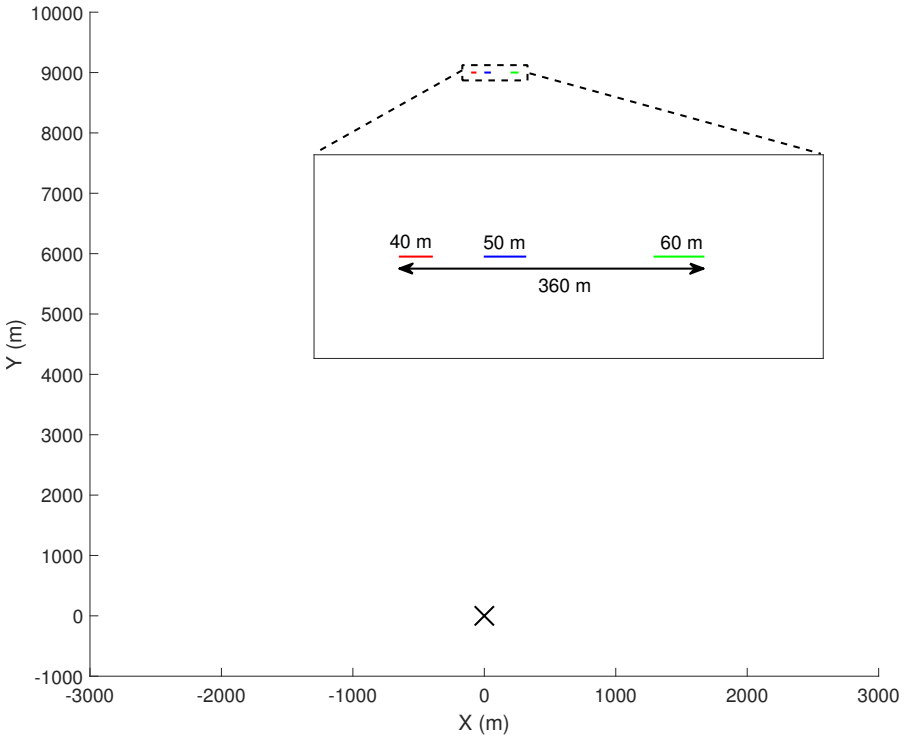


Fig. 4.29: The motion paths of the three MIMO SAR platforms. The single target location at the origin is indicated by an X. From [4] ©2024 IEEE.

bounded by $1 \pm 25 \times 10^{-6}$, modeling oscillators with a ±25 ppm accuracy. The clock biases ϕ_i are randomly generated from a normal distribution with a standard deviation of 10 ns. The carrier phases γ_i^{tx} and γ_i^{rx} are randomly generated from a uniform distribution between $-\pi$ and π rad.

Four different synchronization cases are simulated. In the first, the radar systems are simulated with perfect synchronization to each other and to the global time frame – that is, $\alpha_i = 1$, $\phi_i = 0$ s, and $\gamma_i^{\text{tx}} = \gamma_i^{\text{rx}} = 0$ rad. In the second, the randomly generated clock and phase errors are not compensated at all. In the third and fourth scenarios, the single-tone with exchange synchronization technique is performed to achieve synchronization, with 30 and 50 dB of SNR during the synchronization procedure, respectively.

The two-dimensional magnitude images of the PSRs resulting from these four

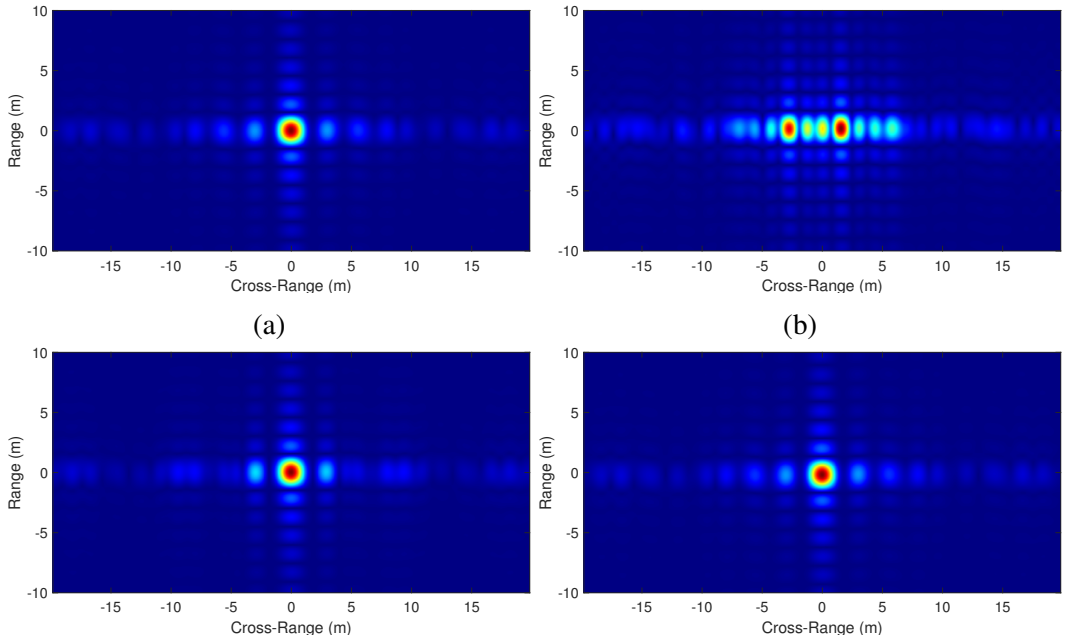
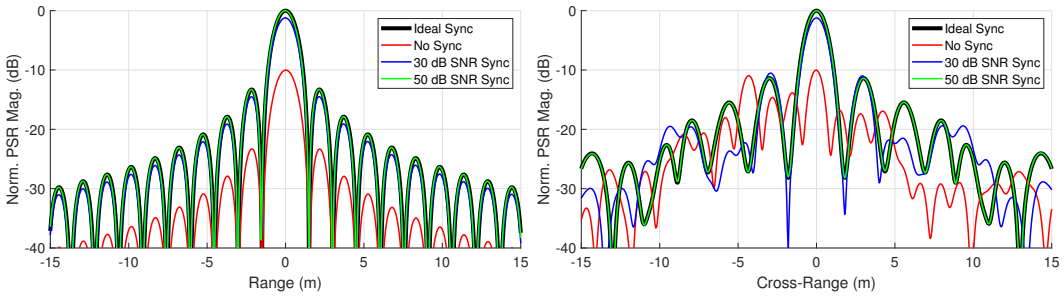


Fig. 4.30: Simulated PSR for different synchronization cases: perfect synchronization (a), no synchronization (b), synchronization performed with 30 dB of SNR (c), and synchronization performed with 50 dB of SNR (d). From [4] ©2024 IEEE.

synchronization cases are shown in Fig. 4.30. Note that although the practical synchronization cases are simulated with 30 dB and 50 dB of SNR for the purposes of synchronization, noise is not included in the radar simulation to enable clear observation of the PSR since simulated noise would only raise the noise floor of this image and provide no additional information on the performance of the radar systems due to synchronization accuracy. The PSR images are formed using the backprojection algorithm, and it is assumed that the locations of the radar systems on each pulse are known perfectly such that motion compensation contributes no error to the formation of the PSR. The ideal PSR shows a clear 2-D sinc function form centered at the origin, which is to be expected from a PSR. There are some distortions in the cross-range (or azimuth) dimension of the PSR due to the irregularities of the full MIMO aperture.

The case of no synchronization shows a severely smeared PSR with two peaks, neither centered at the origin as desired, indicating no useful summation of the radar returns from each different platform. An image formed with this PSR would be severely blurred and unrecognizable. The case of synchronization with 30 dB of SNR shows a PSR whose shape closely matches the ideal, though with more degradation in the sidelobe structure in azimuth due to imperfect coherent integration among platforms, indicating some minor residual phase errors. Finally, the case of synchronization with 50 dB of SNR shows a PSR that matches the ideal with no discernible error, indicating that with this relatively high level of SNR, a sufficiently high-quality synchronization can be achieved to effectively eliminate noticeable errors in the SAR image.

One-dimensional cuts of the cross-track and along-track dimensions of the PSRs are shown in Fig. 4.31. The magnitude of the PSRs shown in these plots are normalized to the peak of the ideal PSR and shifted such that the peak of each PSR is aligned with the others in the plot. As a result, these plots show not only the cross-track degradation of the PSR when the MIMO platforms are not properly synchronized but also the coherent gain lost when the radar returns cannot be coherently integrated. Table 4.1 lists three quality metrics for the different PSRs: the relative peak value (RPV), which describes the mainlobe peak value normalized to the ideal case; the peak sidelobe level ratio (PSLR), which describes the ratio of the peak sidelobe value of the PSR to the peak value in the mainlobe; and the integrated sidelobe level ratio (ISLR), which describes the ratio of the energy in the sidelobes to the energy in the mainlobe. For the RPV, a larger value is better (with the ideal value being 0 dB). Without synchronization, the RPV is less than -10 dB, indicating a 10 dB reduction in coherent gain. This dramatic loss in image power is a result not only of the out-of-phase summation of sub-images but also of the fact that the clock offsets between platforms



(a) (b)

Fig. 4.31: One-dimensional cuts of the simulated PSR for the cases of perfect synchronization, no synchronization, and synchronization performed with 30 and 50 dB of SNR. The cross-track cut is shown in (a) and the along-track cut is shown in (b). From [4] ©2024 IEEE.

Table 4.1: PSR Quality Metrics. From [4] ©2024 IEEE.

	RPV (dB)	**PSLR (dB)**	**ISLR (dB)**
Ideal Sync	0.000	-11.367	-4.457
No Sync	-10.022	-0.937	4.707
30 dB SNR Sync	-1.225	-9.295	-3.908
50 dB SNR Sync	-0.008	-11.343	-4.458

virtually guarantee that the MIMO returns between platforms will not appear at the correct range and will therefore not contribute significant energy to the PSR at all. As a result, the PSR is composed only of the summation of the returns from the three monostatic configurations. The case of synchronization with 30 dB SNR also shows a minor coherent gain reduction with an RPV of -1.2 dB, though this can be attributed primarily to phase errors rather than range misestimation. In the 50 dB SNR case, the PSR almost perfectly matches the ideal PSR and has an RPV of only -0.008 dB. For PSLR and ISLR, lower values that are as close as possible to the ideal synchronization case are better. For the unsynchronized case, the PSLR is greater than -1 dB, indicating a sidelobe level nearly matching the mainlobe (which is clear from Fig. 4.31b), while the ISLR is greater than 0 dB, indicating a larger amount of energy in the side-

lobes than the mainlobe. The synchronized cases improve these metrics substantially, reducing the PSLR to -9 and -11 dB and the ISLR to -4 dB. For the 50 dB SNR case, the PSLR and ISLR are very near to those of the ideal PSR.

The technique described in this chapter enables synchronization of distributed radar systems in time, phase, and frequency. However, as described here, it may only be performed once or in a sequence in which all prior synchronization information is discarded. To maximized synchronization quality over time, some form of Kalman filter should be implemented with the synchronization procedure to track the synchronization states over time. The process for integrating the synchronization technique with a Kalman filter is described in the next chapter.

Chapter 5

Filtered Synchronization

The synchronization technique, as described above in Chapter 4, is to be performed once and the different clock corrections are used perpetually. This solution is not practical in reality since imperfections in clock drift estimation will lead to an observed drift in clock bias and carrier phase over time. Furthermore, due to real clock considerations such as the Allan variance [112], the frequency offset of the oscillators manifesting in the clock drift will not remain constant over time, and such must be reevaluated regularly to maintain synchronization. The simplest approach to resolve this problem is to perform the synchronization process periodically. However, because the synchronization procedure relies on no previous information, the information on the previous synchronization iteration is discarded. By implementing a Kalman filter to track the clock state estimates, previous iterations of the clock parameter estimates may be leveraged. The main advantage of doing this is that the accuracy of each successive estimate of different clock parameters will generally increase over time. Furthermore, if the SNR of the signal exchanges fluctuates with time, the clock parameter estimates with high SNR can be prioritized, while those with low SNR can be ignored.

In this chapter, a framework for applying principles of the Kalman filter (with an

emphasis on the UKF for ease of integration later on with the cooperative navigation filter) is provided. First, the state model is described. This state model includes estimates of relative velocity and range, though it does not directly attempt to estimate platform position and velocity as this is addressed in Chapter 6. The initialization of the filter is discussed as well. The most difficult component of implementing the filter is properly obtaining meaningful measurements and deriving an accurate measurement model. Two methods for doing this are proposed in this section, which have different advantages and limitations. The first method uses time-delay, frequency, and phase measurements directly with a nonlinear relationship modeled between the measurements and the clock parameter state values. The second method simply reproduces the synchronization technique at each iteration and the clock parameter estimates are used directly as measurements. For each of these proposed measurement models, preliminary accuracy results are provided and discussed.

5.1 State Model

For the basic synchronization problem, it is sensible to track the clock drift, clock bias, and TX and RX carrier phases of each transceiver platform, as well as the range and relative radial velocities between each pair of platforms. For now, the tracking of the absolute positions and velocities of the platforms is omitted in favor of including only relative ranges and velocities between pairs of platforms. Define $\mathbf{x}_k^\alpha$ as

$$\mathbf{x}_\alpha^k = \begin{bmatrix} \alpha_1^k & \dots & \alpha_{N_p}^k \end{bmatrix}^T \tag{5.1}$$

where α_i^k is the drift value, α_i, for platform i on iteration k. For the estimate of the clock drift on iteration k, define $\hat{\alpha}_i^k$. Similarly, define $\mathbf{x}_k^\phi$ as

$$\mathbf{x}_\phi^k = \begin{bmatrix} \phi_1^k & \dots & \phi_{N_p}^k \end{bmatrix}^T, \tag{5.2}$$

where ϕ_i^k is the bias value, ϕ_i, for platform i on iteration k. For the estimate of the clock bias on iteration k, define $\hat{\phi}_i^k$. Finally, define $\mathbf{x}_\gamma^k$ as

$$\mathbf{x}_\gamma^k = \begin{bmatrix} \gamma_{1,k}^{\text{tx}} & \gamma_{1,k}^{\text{rx}} & \dots & \gamma_{N_p,k}^{\text{tx}} & \gamma_{N_p,k}^{\text{rx}} \end{bmatrix}^T, \tag{5.3}$$

where $\gamma_{i,k}^{\text{tx}}$ and $\gamma_{i,k}^{\text{rx}}$ are the TX and RX carrier phase values, γ_i^{tx} and γ_i^{rx}, respectively, for platform i on iteration k. For the estimates of the carrier on iteration k, define $\hat{\gamma}_{i,k}^{\text{tx}}$ and $\hat{\gamma}_{i,k}^{\text{rx}}$.

Further, define $\mathbf{x}_R^k$ as

$$\mathbf{x}_R^k = \begin{bmatrix} R_{1,2}^k & R_{1,3}^k & \dots & R_{1,N_p}^k & R_{2,3}^k & \dots & R_{N_p-1,N_p}^k \end{bmatrix}^T \tag{5.4}$$

and $\mathbf{x}_v^k$ as

$$\mathbf{x}_v^k = \begin{bmatrix} v_{1,2}^k & v_{1,3}^k & \dots & v_{1,N_p}^k & v_{2,3}^k & \dots & v_{N_p-1,N_p}^k \end{bmatrix}^T. \tag{5.5}$$

The full system state on iteration k ($\mathbf{x}_k^{\text{sync}}$) may then be described by

$$\mathbf{x}_k^{\text{sync}} = \begin{bmatrix} \mathbf{x}_\alpha^k \\ \mathbf{x}_v^k \\ \mathbf{x}_\phi^k \\ \mathbf{x}_R^k \\ \mathbf{x}_\gamma^k \end{bmatrix}. \tag{5.6}$$

Generally, the constituent variables of $\mathbf{x}_\alpha^k$, $\mathbf{x}_\phi^k$, and $\mathbf{x}_\gamma^k$ may evolve randomly with time but not in a way that may be modeled explicitly. Instead, these variables are modeled as constant with time and the random walk of the clock drift, bias, and carrier phases are described by the process covariance. Furthermore, the inter-platform range values may be described as evolving based on the relative radial velocities of the platforms, which in turn is assumed to be constant with some random change described in the process covariance. Thus, the full system update function, $f_k(\mathbf{x})$, may be described by

$$\begin{aligned} \mathbf{x}_\alpha^{k+1} &= \mathbf{x}_\alpha^k + \mathbf{w}_\alpha^k \\ \mathbf{x}_\phi^{k+1} &= \mathbf{x}_\phi^k + \mathbf{w}_\phi^k \\ \mathbf{x}_\gamma^{k+1} &= \mathbf{x}_\gamma^k + \mathbf{w}_\gamma^k \\ \mathbf{x}_R^{k+1} &= \mathbf{x}_R^k + \left(\mathbf{x}_v^k + \mathbf{w}_v^k\right)\Delta t \\ \mathbf{x}_v^{k+1} &= \mathbf{x}_v^k + \mathbf{w}_v^k \end{aligned} \tag{5.7}$$

where $\mathbf{w}_\alpha^k$ is the clock drift state process noise vector for iteration k with $\mathbf{w}_\alpha^k \sim \mathcal{N}\left(\mathbf{0},\, \mathbf{Q}_k^\alpha\right)$,

$\mathbf{w}_{\phi}^{k}$ is the clock drift state process noise vector for iteration k with $\mathbf{w}_{\phi}^{k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k}^{\phi}\right)$, $\mathbf{w}_{\gamma}^{k}$ is the clock drift state process noise vector for iteration k with $\mathbf{w}_{\gamma}^{k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k}^{\gamma}\right)$, and $\mathbf{w}_{v}^{k}$ is the velocity process noise vector with $\mathbf{w}_{v}^{k} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_{k}^{v}\right)$. The noise values are unknowable and are not included in the model during the state transition portion of the Kalman filter, and the noise is accounted for in the process covariance matrix.

Each of the covariance matrices described above will most likely be a diagonal matrix since it is assumed that each platform will have independent clock errors and generally independent motion characteristics. In a real system, $\mathbf{Q}_{k}^{\alpha}$ will have its entries defined by the Allan variance of each MO. Ideally, it would be expected that $\mathbf{Q}_{k}^{\phi}$ and $\mathbf{Q}_{k}^{\gamma}$ will be zero since the carrier phase and clock bias are modeled as constants. However, in practice, these values may actually drift somewhat as imperfections in the clock drift estimate will ultimately manifest practically as a slow drift of the time axis on each receiver related to the clock skew (drift in clock bias) and a slow drift of the carrier phase related to the carrier offset between platforms. The relative velocity covariance will be very difficult to define since it will generally depend on the expected motion (in particular, the expected accelerations) of each platform relative to one another. Modeling these matrices precisely is the subject of future work and will not be explored in detail in this dissertation. Assuming that some meaningful values for these covariance matrices are available, the process covariance for the synchronization

Kalman filter, $\mathbf{Q}_k^{\text{sync}}$, may be defined by

$$\mathbf{Q}_k^{\text{sync}} = \begin{bmatrix} \mathbf{Q}_k^{\alpha} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_k^{v} & \mathbf{0} & \mathbf{Q}_k^{v}\Delta t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{Q}_k^{\phi} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_k^{v}\Delta t & \mathbf{0} & \mathbf{Q}_k^{v}\Delta t^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Q}_k^{\gamma} \end{bmatrix}. \tag{5.8}$$

5.2 Initializing the Filter State and Covariance

Initializing the filter is straightforward. The initialization begins by running the synchronization algorithm once. This produces the estimates $\hat{\mathbf{x}}_f$ defined in (4.86) (which contains the clock drift estimates $\hat{\mathbf{x}}_\alpha$ and velocity estimates $\hat{\mathbf{x}}_v$), $\hat{\mathbf{x}}_\phi$ defined in (4.94), $\hat{\mathbf{x}}_\gamma$ defined in (4.96), as well as estimates of range in a vector $\hat{\mathbf{x}}_R$. The initial system state estimate, $\hat{\mathbf{x}}_0^{\text{sync}}$, may therefore be defined by

$$\hat{\mathbf{x}}_0^{\text{sync}} = \begin{bmatrix} \hat{\mathbf{x}}_\alpha \\ \hat{\mathbf{x}}_v \\ \hat{\mathbf{x}}_\phi \\ \hat{\mathbf{x}}_R \\ \hat{\mathbf{x}}_\gamma \end{bmatrix}. \tag{5.9}$$

The initial system state covariance, $\hat{\mathbf{C}}_0^{\text{sync}}$, is given by

$$\hat{\mathbf{C}}_0^{\text{sync}} = \begin{bmatrix} \Sigma_{f,v}^0 & \dots & \dots & \mathbf{0} \\ \vdots & \Sigma_\phi^0 & \ddots & \vdots \\ \vdots & \ddots & \Sigma_R^0 & \vdots \\ \mathbf{0} & \dots & \dots & \Sigma_\gamma^0 \end{bmatrix}, \tag{5.10}$$

where $\Sigma_{f,v}^k$ is the clock drift and velocity covariance computed from (4.93) using the SNR at iteration k; Σ_ϕ^k is the clock bias covariance computed from (4.95) using the SNR at iteration k; Σ_R^k is the range covariance computed from (4.117) using the SNR at iteration k; and Σ_γ^k is the carrier phase covariance computed from (4.97) using the SNR at iteration k.

5.3 Time, Phase, and Frequency Approach

After the initial synchronization process to initialize the filter, signals must continue to be periodically exchanged between platforms to continue tracking the synchronization quality of the network and correcting errors. After the initial synchronization, even with clock errors, the clock drift will be sufficiently corrected such that signals are virtually guaranteed to fall within the receive window of each receiver. As a result, there is no downside to transmitting a single-tone signal for frequency extraction along with a pulse-compression waveform for time and phase extraction, resulting in a 50% reduction in the amount of time required for each iteration of the synchronization. Define the signal broadcast by transmitter j as

$$s_j(\tau_j) = s_j^d(\tau_j) + s_j^t(\tau_j), \tag{5.11}$$

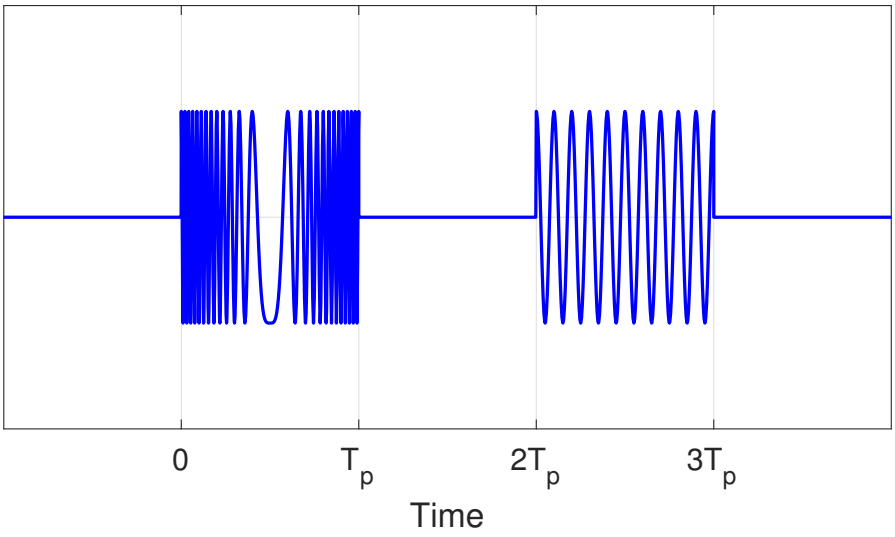


Fig. 5.1: An example of $s_j(\tau_j)$ using a LFM pulse as for the pulse-compression component.

where $s_j^d(\tau_j)$ is some pulse-compression signal with a pulse width T_p, bandwidth B, an auto-correlation response of $d_j(\tau_j)$, and an ambiguity function $A_j(t, f)$. Furthermore, $s_j^t(\tau_j)$ is a rectangularly windowed pulse of a single tone frequency f_j^α delayed from $s_j^d(\tau_j)$ by some amount T_2, also with a pulse length of T_p, expressed by

$$s_j^t(\tau_j) = \exp\left(\mathrm{j}\, 2\pi f_j^\alpha \tau_j\right) \mathrm{rect}\left(\frac{\tau_j - \frac{3}{2}T_p - T_2}{T_p}\right). \tag{5.12}$$

In a practical scenario, the pulse length of $s_j^d(\tau_j)$ and $s_j^t(\tau_j)$ may be set differently, but for simplicity here they are assumed to both be the same value T_p. Further, since the full signal on receive will be matched-filtered using $s_j^d(\tau_j)$ as the filter template, T_2 should be set such that the matched-filter output due to $s_j^t(\tau_j)$ will not interfere with the peak value of $d_j(\tau_j)$. The easiest way to ensure this is by setting $T_2 = T_p$, and this assumption will be made moving forward. An example of $s_j(\tau_j)$ using this model with a LFM pulse for $s_j^d(\tau_j)$ is shown in Fig. 5.1.

Because each platform continues applying the synchronization corrections to all

transmitted and received signals, each receiver i observes the signal broadcast by transmitter j as the signal $\tilde{r}^{f}_{i,j}(\tau'_i)$ as described in (4.74). Let $\tilde{r}^{k}_{i,j}(\tau'_i)$ be the received signal at platform i and transmitted by platform j on iteration k including all TX and RX corrections. In each of the Kalman filtering approaches to synchronization described here, the final corrected signal in (4.74) will be described in terms of the previous clock state estimates (which are known constants) and the current true clock state values (which are unknowns). Furthermore, to be given entirely in terms of the state values, the Doppler shift frequency $f^{d}_{i,j}$ is replaced by the Doppler shift expression in terms of the radial velocity state variable given by

$$f^{d}_{i,j} = -v^{k}_{i,j}\frac{f^{c}}{c}. \tag{5.13}$$

Additionally, let $T^{tx}_{j,k}$ be the transmit time for platform j at iteration k, which must be included since it will impact the time-delay estimate due to time-axis dilation and the phase estimate due to offsets in carrier frequency. In this way, (4.74) can be modified to give $\tilde{r}^{k}_{i,j}(\tau'_i)$ as

$$\begin{aligned}
\tilde{r}^{k}_{i,j}(\tau'_i) = s_j\Bigg(&\frac{\alpha^{k}_{j}\hat{\alpha}^{k-1}_{i}}{\hat{\alpha}^{k-1}_{j}\alpha^{k}_{i}}\tau'_i - \left(\frac{\alpha^{k}_{j}}{\hat{\alpha}^{k-1}_{j}\alpha^{k}_{i}}\left(\phi^{k}_{i}-\hat{\phi}^{k-1}_{i}\right) - \frac{1}{\hat{\alpha}_{j}}\left(\phi^{k}_{j}-\hat{\phi}^{k-1}_{j}\right) + \frac{\alpha^{k}_{j}}{\hat{\alpha}^{k-1}_{j}}\frac{R^{k}_{i,j}}{c}\right) - T^{\mathrm{tx}}_{j,k}\Bigg)\\
&\cdot \exp\left(-\mathrm{j}\,2\pi f^{c}\left(\frac{\alpha^{k}_{j}}{\hat{\alpha}^{k-1}_{j}\alpha^{k}_{i}}\left(\phi^{k}_{i}-\hat{\phi}^{k-1}_{i}\right) - \frac{1}{\hat{\alpha}^{k-1}_{j}}\left(\phi^{k}_{j}-\hat{\phi}^{k-1}_{j}\right) + \frac{\alpha^{k}_{j}}{\hat{\alpha}^{k-1}_{j}}\frac{R^{k}_{i,j}}{c}\right)\right)\\
&\cdot \exp\left(\mathrm{j}\,2\pi f^{c}\left(\frac{\alpha^{k}_{j}\hat{\alpha}^{k-1}_{i}}{\hat{\alpha}^{k-1}_{j}\alpha^{k}_{i}} - 1\right)\tau'_i\right)\exp\left(-\mathrm{j}\,2\pi\frac{\hat{\alpha}^{k-1}_{i}}{\alpha^{k}_{i}}v^{k}_{i,j}\frac{f^{c}}{c}\left(\tau'_i - T^{\mathrm{tx}}_{j,k}\right)\right)\\
&\cdot \exp\left(\mathrm{j}\,2\pi\frac{1}{\alpha^{k}_{i}}v^{k}_{i,j}\frac{f^{c}}{c}\left(\phi^{k}_{i}-\hat{\phi}^{k-1}_{i}\right)\right)\exp\left(\mathrm{j}\,\left(\left(\gamma^{\mathrm{tx}}_{j,k}-\hat{\gamma}^{\mathrm{tx}}_{j,k-1}\right)-\left(\gamma^{\mathrm{rx}}_{i,k}-\hat{\gamma}^{\mathrm{rx}}_{i,k-1}\right)\right)\right).
\end{aligned} \tag{5.14}$$

The above expression will be valid for this time, phase, and frequency approach, as well as the iterative synchronization approach described in Section 5.4.

The process of computing measurement inputs to the UKF will involve extracting the time-delay, phase, and frequency of the received signals in much the same way as the original synchronization procedure. Define $\hat{\tau}_{i,j}^k$, $\hat{\gamma}_{i,j}^k$, and $\hat{f}_{i,j}^k$ as the time-delay, phase, and frequency estimates, respectively, extracted from $\tilde{r}_{i,j}^k(\tau_i')$ on iteration k between transmitter j and receiver i. Let $d_{i,j}^k(\tau_i')$ be the cross-correlation of $s_j^d(\tau_i')$ with $\tilde{r}_{i,j}^k(\tau_i')$. Similarly to the original synchronization time-delay estimate, $\hat{\tau}_{i,j}^k$ is determined as the time value which maximizes $d_{i,j}^k(\tau_i')$, and the phase estimate, $\hat{\gamma}_{i,j}^k$, is given as the phase of the cross-correlation at this point, $\angle d_{i,j}^k(\hat{\tau}_{i,j}^k)$. Using the time-delay estimate, the received signal should be truncated to include only the single-tone pulse such that the frequency domain estimation will meet the CRLB (recall from Section 4.4.2 that the sinc NL-LS frequency domain estimation only meets the CRLB when the capture length is as long as the pulse length). Finally, after this truncation, the FFT of the truncated data should be computed and the frequency estimate, $\hat{f}_{i,j}^k$, can be computed as the solution to the sinc NL-LS frequency domain peak estimation. These estimates will be directly used to form the measurement vector for this approach to the synchronization UKF, and will also be used in forming the measurement vector in the iterative synchronization approach described in Section 5.4.

Define $\mathbf{z}_{\tau,\text{TPF}}^k$ as the measurement vector for iteration k containing time-delay measurements obtained from the signal exchanged between transmitter j and receiver i, given by

$$\mathbf{z}_{\tau,\text{TPF}}^k = \begin{bmatrix} \hat{\tau}_{1,2}^k & \hat{\tau}_{1,3}^k & \cdots & \hat{\tau}_{1,N_p}^k & \hat{\tau}_{2,1}^k & \hat{\tau}_{2,3}^k & \cdots & \hat{\tau}_{N_p-1,N_p}^k \end{bmatrix}^T. \tag{5.15}$$

Similarly, define a phase measurement vector, $\mathbf{z}^k_{\gamma,\text{TPF}}$, by

$$\mathbf{z}^k_{\gamma,\text{TPF}} = \begin{bmatrix} \hat{\gamma}^k_{1,2} & \hat{\gamma}^k_{1,3} & \cdots & \hat{\gamma}^k_{1,N_p} & \hat{\gamma}^k_{2,1} & \hat{\gamma}^k_{2,3} & \cdots & \hat{\gamma}^k_{N_p-1,N_p} \end{bmatrix}^T. \tag{5.16}$$

Finally, define a frequency measurement vector, $\mathbf{z}^k_{f,\text{TPF}}$, by

$$\mathbf{z}^k_{f,\text{TPF}} = \begin{bmatrix} \hat{f}^k_{1,2} & \hat{f}^k_{1,3} & \cdots & \hat{f}^k_{1,N_p} & \hat{f}^k_{2,1} & \hat{f}^k_{2,3} & \cdots & \hat{f}^k_{N_p-1,N_p} \end{bmatrix}^T. \tag{5.17}$$

The full measurement vector for the time, phase, and frequency approach, $\mathbf{z}^{\text{TPF}}_k$, may then be composed of these measurements by

$$\mathbf{z}^{\text{TPF}}_k = \begin{bmatrix} \mathbf{z}^k_{\tau,\text{TPF}} \\ \mathbf{z}^k_{\gamma,\text{TPF}} \\ \mathbf{z}^k_{f,\text{TPF}} \end{bmatrix}. \tag{5.18}$$

With the state model described here, and the measurement model derived below, the full UKF may be run exactly as described in Section 2.1.3.

5.3.1 Derivation of Measurement Model

This section derives the measurement function, $h_k(\mathbf{z})$, for the time, phase, and frequency approach. By observation of the fully corrected signal model in (4.74), a measurement model may be constructed for $\hat{\tau}^k_{i,j}$, $\hat{\gamma}^k_{i,j}$, and $\hat{f}^k_{i,j}$. Note that each signal exchange will leverage the clock error estimates from the previous iteration to compensate the signal on the current iteration, and thus the measurement model will depend on the previous (known) estimates and current (unknown) state parameters.

First, the time-delay estimate, $\hat{\tau}_{i,j}^{k}$, will be computed as the delay parameter of the baseband signal in (4.74). This is computed by determining the peak value of $d_{i,j}^{k}(\tau_i')$ through sinc NL-LS interpolation. Because it is not assumed that the pulse-compression signal is not Doppler-tolerant, it is possible that an additional frequency shift-induced time shift will be present. This additional shift can be computed from the peak of the ambiguity function $A_j(t, f)$ at the f value equal to the frequency shift in (4.74). Ignoring the RX signal contribution of $s_j^t(t)$ and focusing on the RX component corresponding to $s_j^d(t)$, the cross-correlation will be given by

$$\begin{aligned} d_{i,j}^{k}(\tau_i') = A_j\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k}\tau_i' - \tilde{\rho} - T_{j,k}^{\mathrm{tx}},\ f^c\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k} - 1\right) - \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}v_{i,j}^k\frac{f^c}{c}\right)\exp\left(-\mathrm{j}\,2\pi f^c\tilde{\rho}\right) \\ \exp\left(\mathrm{j}\,2\pi f^c\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k} - 1\right)T_{j,k}^{\mathrm{tx}}\right)\cdot\exp\left(\mathrm{j}\,2\pi\frac{1}{\alpha_i^k}v_{i,j}^k\frac{f^c}{c}\left(\phi_i^k - \hat{\phi}_i^{k-1}\right)\right) \\ \cdot\exp\left(\mathrm{j}\left(\left(\gamma_{j,k}^{\mathrm{tx}} - \hat{\gamma}_{j,k-1}^{\mathrm{tx}}\right) - \left(\gamma_{i,k}^{\mathrm{rx}} - \hat{\gamma}_{i,k-1}^{\mathrm{rx}}\right)\right)\right), \end{aligned} \tag{5.19}$$

where the auxiliary variable $\tilde{\rho}$ is defined by

$$\tilde{\rho} = \frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}\alpha_i^k}\left(\phi_i^k - \hat{\phi}_i^{k-1}\right) - \frac{1}{\hat{\alpha}_j}\left(\phi_j^k - \hat{\phi}_j^{k-1}\right) + \frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}}\frac{R_{i,j}^k}{c}. \tag{5.20}$$

The expression in (5.19) includes a phase term related to the carrier offset and the transmit time since the carrier offset phase will be set approximately based on the transmit time since in general the carrier offset term after correction will be very low-frequency. Note that the expression in (5.19) makes the simplifying assumption that the scaling ratio of clock drifts is close enough to one that there will not be an impact

to the pulse compression operation – that is, that

$$\frac{\alpha_j^k \hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1} \alpha_i^k} \approx 1. \tag{5.21}$$

Recall that the ambiguity function defines the cross-correlation response of a pulse compression signal with its frequency-shifted counterpart. Thus, the result in (5.19) can be interpreted as providing the cross-correlation function in the case of a specific frequency shift induced by the Doppler shift and the carrier offset between platforms, with the remaining phase modifications remaining the same as in (5.14). As dictated by the ambiguity function, the cross-correlation will also be shifted in time. Thus, the total measured time-delay will include the true time-delay given by $\tilde{\rho}$ as well as the time shift due to the frequency shift.

Define the time shift due to the frequency shift as

$$\tau_{i,j}^{A,k} = \arg\max_{\tau_i'} \left\{ A_j \left(\tau_i', \, f^c \left(\frac{\alpha_j^k \hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1} \alpha_i^k} - 1 \right) - \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k} v_{i,j}^k \frac{f^c}{c} \right) \right\}, \tag{5.22}$$

where the value in the frequency argument of $A_j(t, f)$ is computed based on the two sinusoidal terms in (5.14) multiplied with the original baseband signal. To determine the time-delay estimate, the time argument of the ambiguity function in (5.19) must be solved such that it equals zero, which requires scaling the delay parameters by the inverse of the clock drift ratio. The model for $\hat{\tau}_{i,j}^k$ may therefore be expressed by

$$\begin{aligned} \hat{\tau}_{i,j}^k &= \frac{\hat{\alpha}_j^{k-1} \alpha_i^k}{\alpha_j^k \hat{\alpha}_i^{k-1}} \left(\tilde{\rho} + \tau_{i,j}^{A,k} + T_{j,k}^{\text{tx}} \right) \\ &= \frac{1}{\hat{\alpha}_i^{k-1}} \left(\phi_i^k - \hat{\phi}_i^{k-1} \right) - \frac{\alpha_i^k}{\alpha_j^k \hat{\alpha}_i^{k-1}} \left(\phi_j^k - \hat{\phi}_j^{k-1} \right) + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k} \frac{R_{i,j}^k}{c} + \frac{\hat{\alpha}_j^{k-1} \alpha_i^k}{\alpha_j^k \hat{\alpha}_i^{k-1}} \left(\tau_{i,j}^{A,k} + T_{j,k}^{\text{tx}} \right). \end{aligned} \tag{5.23}$$

Next, the phase measurement, $\hat{\gamma}_{i,j}^{k}$, will be computed as the phase of the received baseband signal in (5.14). Similarly to the Doppler-intolerant case for time-delay measurement, the ambiguity function phase must be included assuming that the phase along the peak value is not constant as a function of f. The model for $\hat{\gamma}_{i,j}^{k}$ may therefore be expressed by

$$\begin{aligned}\hat{\gamma}_{i,j}^{k} = &-2\pi f^{c}\left(\frac{\alpha_{j}^{k}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}\left(\phi_{i}^{k}-\hat{\phi}_{i}^{k-1}\right)-\frac{1}{\hat{\alpha}_{j}}\left(\phi_{j}^{k}-\hat{\phi}_{j}^{k-1}\right)+\frac{\alpha_{j}^{k}}{\hat{\alpha}_{j}^{k-1}}\frac{R_{i,j}^{k}}{c}\right)\\ &+2\pi f^{c}\left(\frac{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}-1\right)T_{j,k}^{\mathrm{tx}}+2\pi f^{c}\frac{1}{\alpha_{i}^{k}}v_{i,j}^{k}\frac{f^{c}}{c}\left(\phi_{i}^{k}-\hat{\phi}_{i}^{k-1}\right)\\ &+\angle A_{j}\left(\tau_{i,j}^{A,k},\ f^{c}\left(\frac{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}-1\right)-\frac{\hat{\alpha}_{i}^{k-1}}{\alpha_{i}^{k}}v_{i,j}^{k}\frac{f^{c}}{c}\right)\\ &+\left(\left(\gamma_{j,k}^{\mathrm{tx}}-\hat{\gamma}_{j,k-1}^{\mathrm{tx}}\right)-\left(\gamma_{i,k}^{\mathrm{rx}}-\hat{\gamma}_{i,k-1}^{\mathrm{rx}}\right)\right).\end{aligned} \tag{5.24}$$

The frequency measurement is extracted from the peak value of the Fourier transform of the single-tone component of $\tilde{r}_{i,j}^{k}(\tau_{i}')$. Let $\tilde{r}_{i,j}^{t}(\tau_{i}')$ be the single-tone component of $\tilde{r}_{i,j}^{k}(\tau_{i}')$ after it is truncated. This is computed by inserting (5.12) into (5.14), yielding

$$\begin{aligned}\tilde{r}_{i,j}^{t}(\tau_{i}') = &\exp\left(\mathrm{j}\,2\pi f_{j}^{\alpha}\frac{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}\tau_{i}'\right)\exp\left(\mathrm{j}\,2\pi f^{c}\left(\frac{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}-1\right)\tau_{i}'\right)\\ &\cdot\exp\left(-\mathrm{j}\,2\pi\frac{\hat{\alpha}_{i}^{k-1}}{\alpha_{i}^{k}}v_{i,j}^{k}\frac{f^{c}}{c}\left(\tau_{i}'-T_{j,k}^{\mathrm{tx}}\right)\right)\exp\left(-\mathrm{j}\,2\pi f_{j}^{\alpha}\left(\tilde{\rho}+T_{j,k}^{\mathrm{tx}}\right)\right)\exp\left(-\mathrm{j}\,2\pi f^{c}\tilde{\rho}\right)\\ &\cdot\exp\left(\mathrm{j}\,2\pi\frac{1}{\alpha_{i}^{k}}v_{i,j}^{k}\frac{f^{c}}{c}\left(\phi_{i}^{k}-\hat{\phi}_{i}^{k-1}\right)\right)\exp\left(\mathrm{j}\,\left(\left(\gamma_{j,k}^{\mathrm{tx}}-\hat{\gamma}_{j,k-1}^{\mathrm{tx}}\right)-\left(\gamma_{i,k}^{\mathrm{rx}}-\hat{\gamma}_{i,k-1}^{\mathrm{rx}}\right)\right)\right)\\ &\cdot\mathrm{rect}\left(\frac{\tau_{i}'-\frac{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}\left(\tilde{\rho}+T_{i,j}^{k}\right)-\frac{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}\left(\frac{3}{2}T_{p}+T_{2}\right)}{\frac{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}T_{p}}\right).\end{aligned} \tag{5.25}$$

The frequency measurement, $\hat{f}_{i,j}^k$, will be computed as the frequency of the single-tone signal, using sinc NL-LS interpolation and thus the model for $\hat{f}_{i,j}^k$ may be expressed by

$$\hat{f}_{i,j}^k = f_j^\alpha \frac{\alpha_j^k \hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1} \alpha_i^k} + f^c \left(\frac{\alpha_j^k \hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1} \alpha_i^k} - 1 \right) - \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k} v_{i,j}^k \frac{f^c}{c}. \tag{5.26}$$

5.3.2 Covariance of Measurements

The observation covariance matrix for iteration k of the synchronization UKF, $\mathbf{R}_k^{\text{TPF}}$, is computed directly from the variance values σ_τ^2, σ_γ^2, and σ_f^2, which will in turn be computed from the associated CRLB expressions. It is assumed that all measurements are independent of one another, and thus $\mathbf{R}_k^{\text{TPF}}$ will be diagonal. Define $\mathbf{R}_k^\tau$ as the covariance of the time-delay measurements given by

$$\mathbf{R}_k^\tau = \text{diag}\left(\left[V_{1,2}^{\tau,k} \; V_{1,3}^{\tau,k} \; \dots \; V_{1,N}^{\tau,k} \; V_{2,1}^{\tau,k} \; V_{2,3}^{\tau,k} \; \dots \; V_{N-1,N}^{\tau,k} \right] \right) \tag{5.27}$$

where $V_{i,j}^{\tau,k}$ is defined as $\text{var}\left(\hat{\tau}_{i,j}^k\right)$ computed using (4.103). Similarly, define $\mathbf{R}_k^\gamma$ as the covariance of the phase measurements given by

$$\mathbf{R}_k^\gamma = \text{diag}\left(\left[V_{1,2}^{\gamma,k} \; V_{1,3}^{\gamma,k} \; \dots \; V_{1,N}^{\gamma,k} \; V_{2,1}^{\gamma,k} \; V_{2,3}^{\gamma,k} \; \dots \; V_{N-1,N}^{\gamma,k} \right] \right) \tag{5.28}$$

where $V_{i,j}^{\gamma,k}$ is defined as $\text{var}\left(\hat{\gamma}_{i,j}^k\right)$ computed using (4.114). Finally, define $\mathbf{R}_k^f$ as the covariance of the phase measurements given by

$$\mathbf{R}_k^f = \text{diag}\left(\left[V_{1,2}^{f,k} \; V_{1,3}^{f,k} \; \dots \; V_{1,N}^{f,k} \; V_{2,1}^{f,k} \; V_{2,3}^{f,k} \; \dots \; V_{N-1,N}^{f,k} \right] \right) \tag{5.29}$$

where $V_{i,j}^{f,k}$ is defined as $\text{var}\left(\hat{f}_{i,j}^{k}\right)$ computed using (4.102). The full observation covariance is then structured by

$$\mathbf{R}_k^{\text{TPF}} = \begin{bmatrix} \mathbf{R}_k^{\tau} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_k^{\gamma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_k^{f} \end{bmatrix}. \tag{5.30}$$

5.3.3 Preliminary Results

In this section, some preliminary results demonstrating the efficacy, and limitations, of this approach are provided. A set of four platforms with randomly generated clock error parameters and positions are simulated. For these simulations, the simplifying assumption is made that there is no drift of the clock parameters over time – that is, that the clock parameter covariance matrices in the process covariance in (5.8) (Σ_{α}^{k}, Σ_{ϕ}^{k}, and Σ_{γ}^{k}) are equal to zero. Future work will address the addition of more realistic clock components such as Allan variance. In practice, this approach has shown to be limited to relatively high values of SNR before diverging, so the simulations are generally performed at these high values of SNR. A discussion of specific advantages and limitations of this approach is provided in Section 5.3.4.

While the UKF update equations are sound from a mathematical standpoint, their implementation on a computer may run into numerical issues leading to errors and potential instability. This problem is exacerbated when the values of the variables in the state vector (and their associated variance values in the state covariance) are separated by many orders of magnitude, which is the case here. For instance, the estimated clock drift values will vary on the ppm scale while range values will vary on the hundreds or thousands of meters scale, leading to 9 or 10 orders of magnitude difference. Thus, to alleviate these problems, the values of the different state values (and some of the mea-

surements are normalized). The clock drift and clock bias values are normalized by a factor of 10, while the range and velocity values are normalized by a factor of $1/c$. The carrier phase state values are normalized by a factor of 10^{-3}. Furthermore, the phase measurements are normalized by a factor of $1/c$ and the frequency measurements are normalized by a factor of $1/f^c$.

For each of the simulation case studies below, the following simulation parameters are used. Unless otherwise stated, the SNR is set to 30 dB. The carrier frequency f^c is set to 3 GHz. The pulse length is set to 100 μs, while the sampling frequency is set to 200 MHz. The capture length in the initial synchronization is set to 400 μs, while it is set to 600 μs in the subsequent iterations of the UKF to account for the longer signal. The signal bandwidth in the LFM waveforms is set to 50 MHz. The TDMA scheme is used with the time slot set to 1 ms. The clock drift errors are drawn from a uniform distribution limited to within 25 ppm of 1. The clock bias errors are drawn from a Gaussian distribution with a standard deviation of 100 ns. The full simulations are conducted over 30 s with iterations taking place at 10 Hz, unless otherwise stated.

5.3.3.1 Stationary Systems, Assuming No Motion

In this first case study, a network of four stationary platforms with randomly distributed positions is used. In this scenario, the UKF is formatted to assume the platforms are stationary, such that the relative platform velocities are omitted from the state vector, and the processing is slightly modified to assume that there will be no Doppler shifts.

Fig. 5.2 shows the error results for a single platform in the network of four platforms, while Fig. 5.3 shows these same results for the mean error for the full network of four platforms. There are several noteworthy aspects of these results. First, it can be

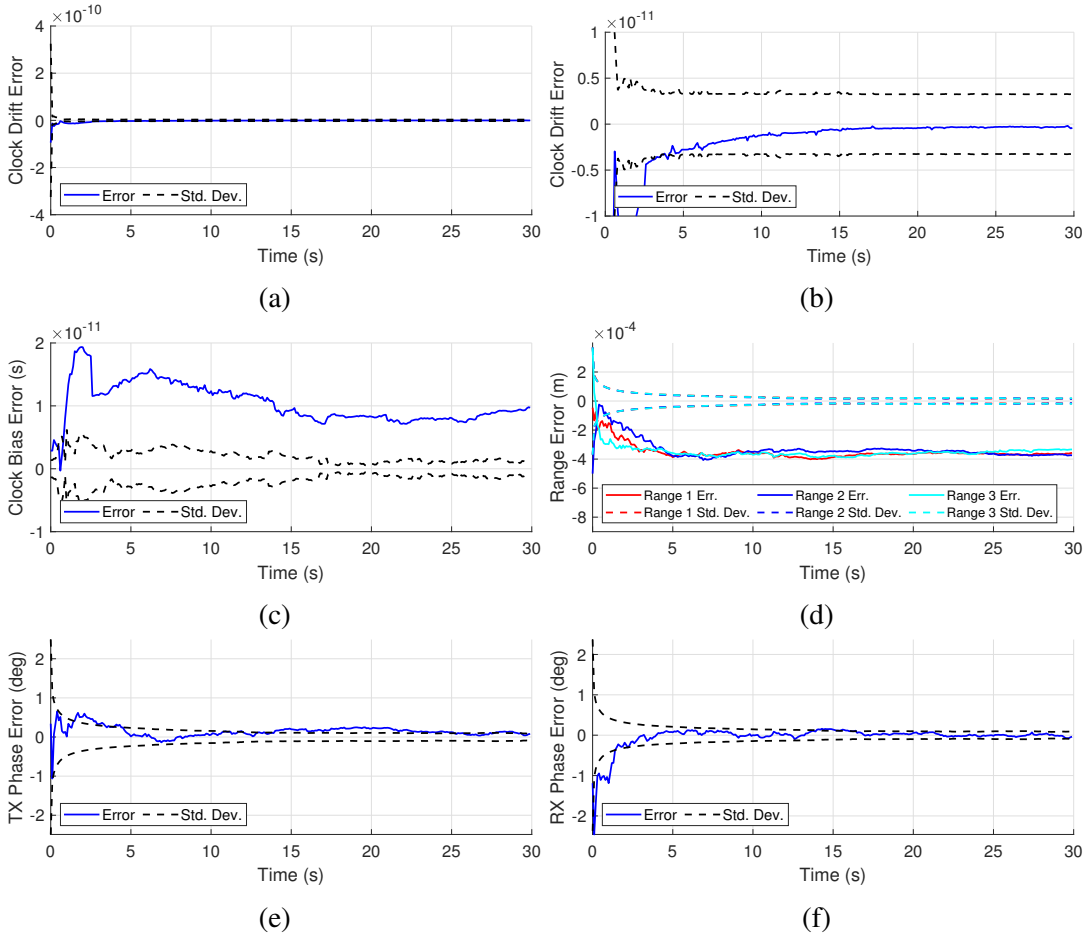


Fig. 5.2: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed for a single platform in a stationary network of four platforms with relative velocities assumed in the UKF to be zero. The above plots show the error for a single platform of clock drift (a), clock drift (zoomed) (b), clock bias (c), range with the other three platforms (d), TX carrier phase (e), and RX carrier phase (f). The solid lines show the error, while the dashed lines indicate the expected standard deviation of the error bound predicted by the UKF.

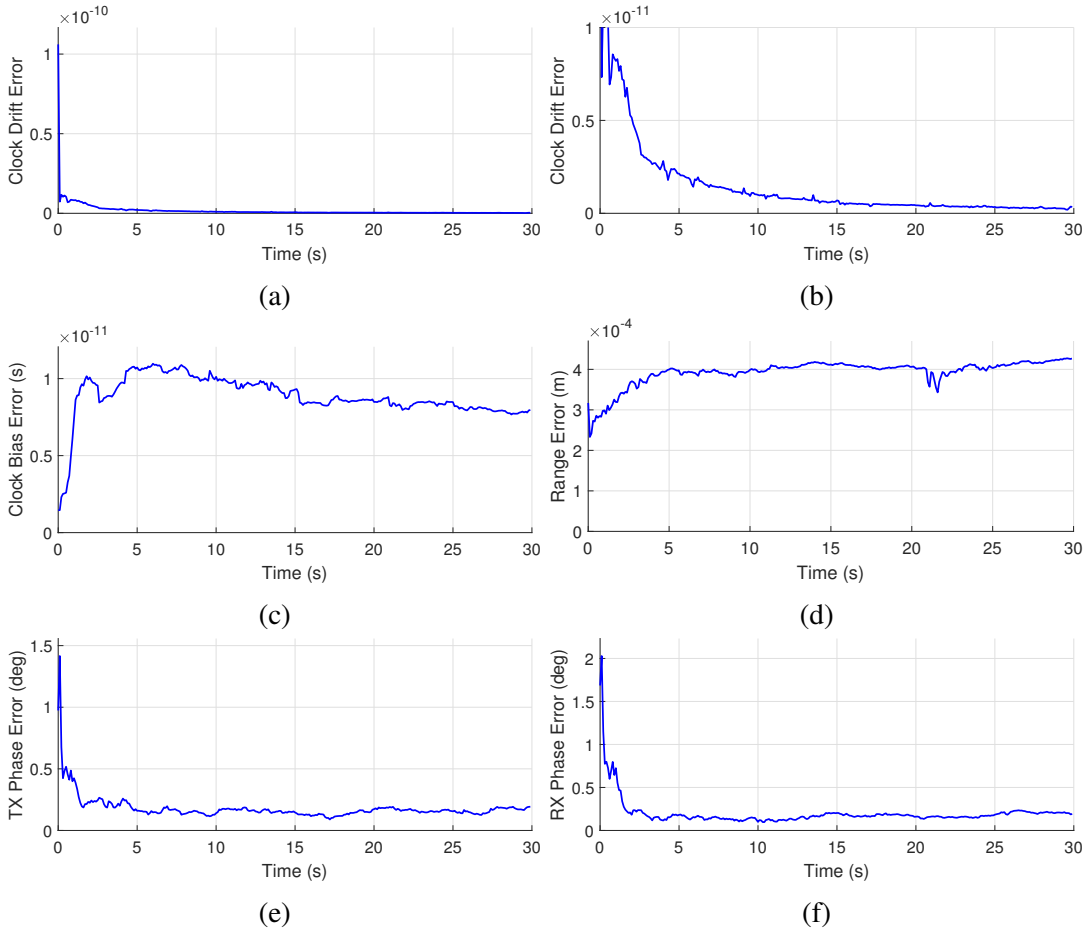


Fig. 5.3: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed as the mean error for a stationary network of four platforms with relative velocities assumed in the UKF to be zero. The above plots show the mean error for clock drift (a), clock drift (zoomed) (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

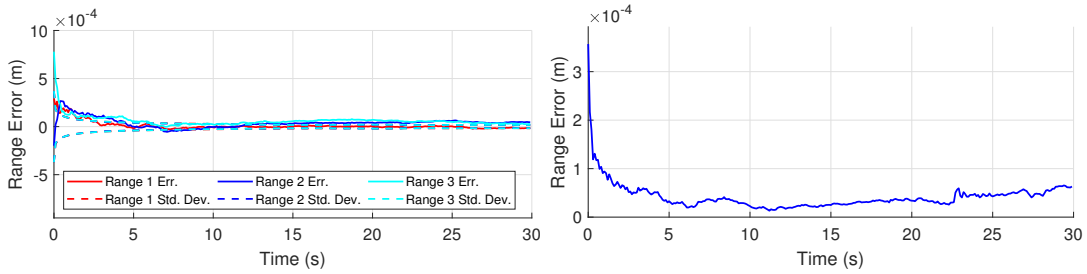


Fig. 5.4: The range error when the true value of α_1 = such that there is no range scaling – the single platform (a) and the mean over the full network of four platforms (b).

seen that the clock drift and carrier phase error generally converge to zero over time, indicating that the reduction of error expected from the Kalman filtering is occurring. In particular, the clock drift estimate error rapidly converges, falling to less than 1 ppt over the 30 second simulation time.

However, the clock bias and range values do not converge to zero. The clock bias can be explained as the clock drift error not exactly being zero – as a result, the signals will still have a small time offset which grows over time with $T^{\text{tx}}_{j,k}$. The range error values do seem to converge, but not to zero. This is because it is assumed that $\alpha_1 = 1$, which is not true, and as a result the estimate of range is scaled by the ratio of $\alpha_j/\hat{\alpha}_j$. This small error (0.4 mm) will grow linearly with range for networks with large distances between platforms. Fig. 5.4 shows the range error for a simulation case in which the true value of α_1 is actually 1 – in this case, it can be seen that the bias in the range error is removed, and the range error between the platform and all the other platforms converges toward zero.

Fig. 5.5 shows the innovation values in $\tilde{\mathbf{z}}_k$ for time, phase, and frequency. It can be seen that the time and phase innovations quickly converge to zero, implying that the system state value is predicted well enough at the beginning of each iteration such that

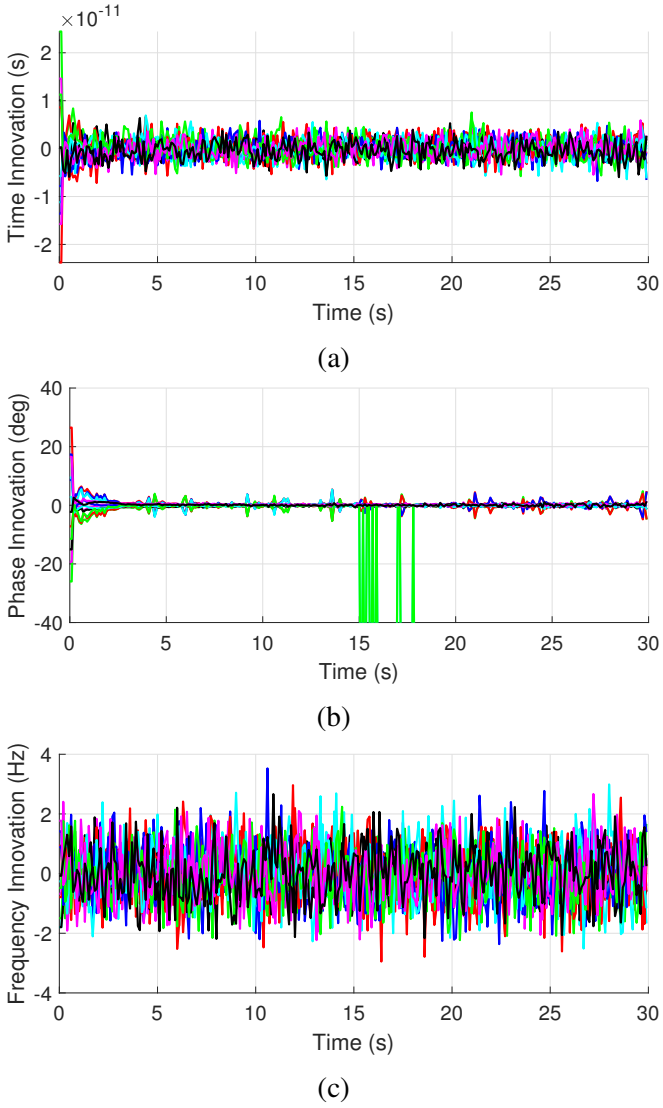


Fig. 5.5: The innovation values in $\tilde{\mathbf{z}}_k$ – time (a), phase (b), and frequency (c).

the measurement estimates closely match the true measurement, minus the noise on the measurements. There are some isolated instances in which the phase innovations jump to higher values, which is likely a result of a single wrapped phase value in the sigma points such that the weighted average in computing that particular entry of $\hat{\mathbf{z}}_k$ does not lead to zero. However, these isolated errors do not seem to impact the UKF operation significantly. The frequency innovations do not get better after the initial

iterations and are instead dominated by the noise on the measurements throughout the UKF operation. This implies that the tighter synchronization as the UKF operation continues does not help the frequency estimation, unlike time and phase, where the estimate of time and phase improves as the clock drift is better known. This facet of the frequency estimation is why it is such an effective way of producing clock drift estimates before coarse time synchronization can be achieved.

Of interest is the fact that the clock drift estimate converges so rapidly. In this example, the initial error of close to 100 ppt is reduced after only one iteration to close to 10 ppt – a reduction in error of a whole order of magnitude. By the end of the 30 s simulation (300 iterations), the error has reduced to 0.4 ppt – a reduction of nearly three orders of magnitude from the original estimate. This is likely because of the introduction of time-delay information. In the original synchronization algorithm using time-delay information is not possible due to the lack of a guarantee that the transmit and receive windows will align well, and therefore using frequency estimation to infer clock drift from the carrier skew is used. Once this initial synchronization is performed, the RX signal is essentially guaranteed to fall within the receive window. However, due to the non-zero clock drift error at each iteration, the signal will still drift by small amounts. This small amount of drift can be more easily detected by time-delay measurements than frequency measurements because the time-delay measurements obtain high-resolution information due to the ease with which larger bandwidths can be obtained when compared to longer signal times required to obtain high-resolution information in the frequency domain.

5.3.3.2 Stationary Systems, Assuming Motion

In the second case study, the same network of four stationary platforms is used. In this scenario, the UKF is formatted to assume the platforms are not stationary as described in the problem formulation previously in this chapter. Thus, the relative platform velocities are estimated by the UKF. Although most of $\mathbf{Q}_k^{\text{IS}}$ is assumed to be zero since the clock parameters are simulated as constant, the radial velocity covariance Σ_v^k in general will not be zero since the changing geometry of the network over time as platforms move implies that the relative velocities will not be constant. Because no assumptions are made on the initial positioning of the platforms and there is no direct estimation of the platform positions over time, there is not a good way to estimate the correct value for Σ_v^k. For this simulation, Σ_v^k is assumed to be the scalar matrix $\Sigma_v^k = 10^{-3}\mathbf{I}$.

Fig. 5.6 shows the error results for a single platform in the network of four platforms, while Fig. 5.7 shows these same results for the mean error for the full network of four platforms. These simulation results align in terms of general performance with the results where the stationary platforms were assumed – the clock drift error rapidly decreases to zero, the range and clock bias errors converge but not to zero (for identical reasons to the case where stationary platforms were assumed), and the carrier phase errors both diminish to zero. The radial velocity estimation error also generally approaches zero, with some spikes in error which are eliminated after one or two iterations.

Interestingly, the clock drift error reduces more quickly in this case than in the case where stationary platforms were assumed – this is perhaps contrary to intuition since there is more information to estimate in this case where error could be introduced. Nevertheless, the clock drift error for the single platform in this case is reduced to

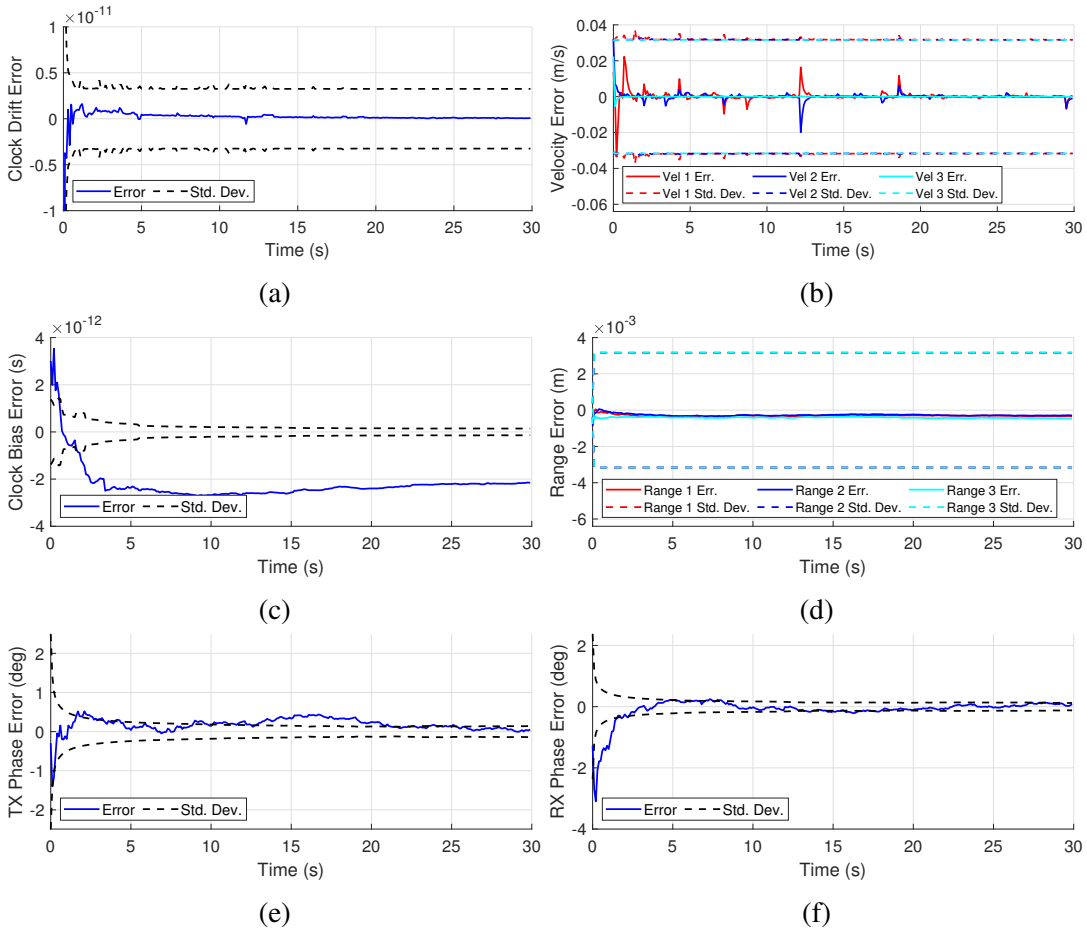


Fig. 5.6: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed for a single platform in a stationary network of four platforms with relative velocities being estimated by the UKF. The above plots show the error for a single platform of clock drift (a), relative velocity with the other three platforms (b), clock bias (c), range with the other three platforms (d), TX carrier phase (e), and RX carrier phase (f). The solid lines show the error, while the dashed lines indicate the expected standard deviation of the error bound predicted by the UKF.

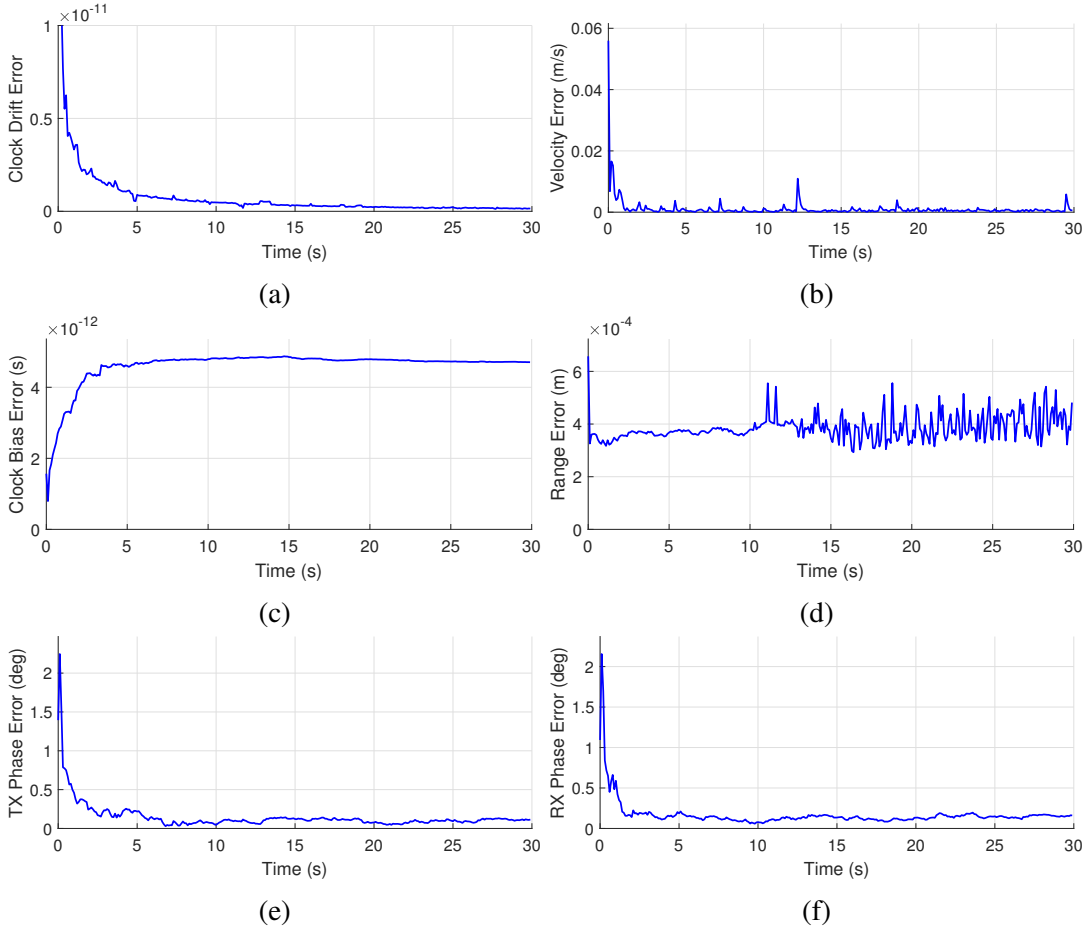


Fig. 5.7: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed as the mean error for a stationary network of four platforms with relative velocities being estimated by the UKF. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

0.06 ppt by the end of the 30 s simulation in contrast to the 0.4 ppt for the case in which the platforms were assumed stationary, giving nearly an order of magnitude in improvement.

5.3.3.3 Impact of Moderate to Low SNR

Next, the same network of four stationary platforms is simulated to perform the synchronization UKF procedure but with a reduced SNR value of 20 dB. Fig. 5.8 shows the mean error for the different estimation quantities for this scenario. It is quite clear from this simulation that the estimation falls apart rapidly – in fact, outside of the initial synchronization, the synchronization is completely erroneous, leading to purely random phase errors, large velocity errors and range errors, and non-convergence of the clock drift and clock bias values. More investigation is necessary to determine the point of failure, as it is unlikely that this algorithm can be deployed reliably on any network if it cannot tolerate even moderately low SNR. This conclusion is the primary impetus for developing the stabler iterative synchronization UKF technique described in Section 5.4.

5.3.3.4 Impact of Iteration Frequency

Another drawback to this approach is that it requires relatively quick updates to the UKF, which may be a problem in applications with low downtime requirements. The previous successful demonstrations of the UKF synchronization using this approach use an update rate of 10 Hz. Fig. 5.9 shows the mean error results when the update rate is reduced to 1 Hz. As with the low SNR case, the reduction in sampling frequency leads to a radical increase in mean error for all the variables. It may therefore be concluded that even with moderately high SNR, this technique does require a relatively

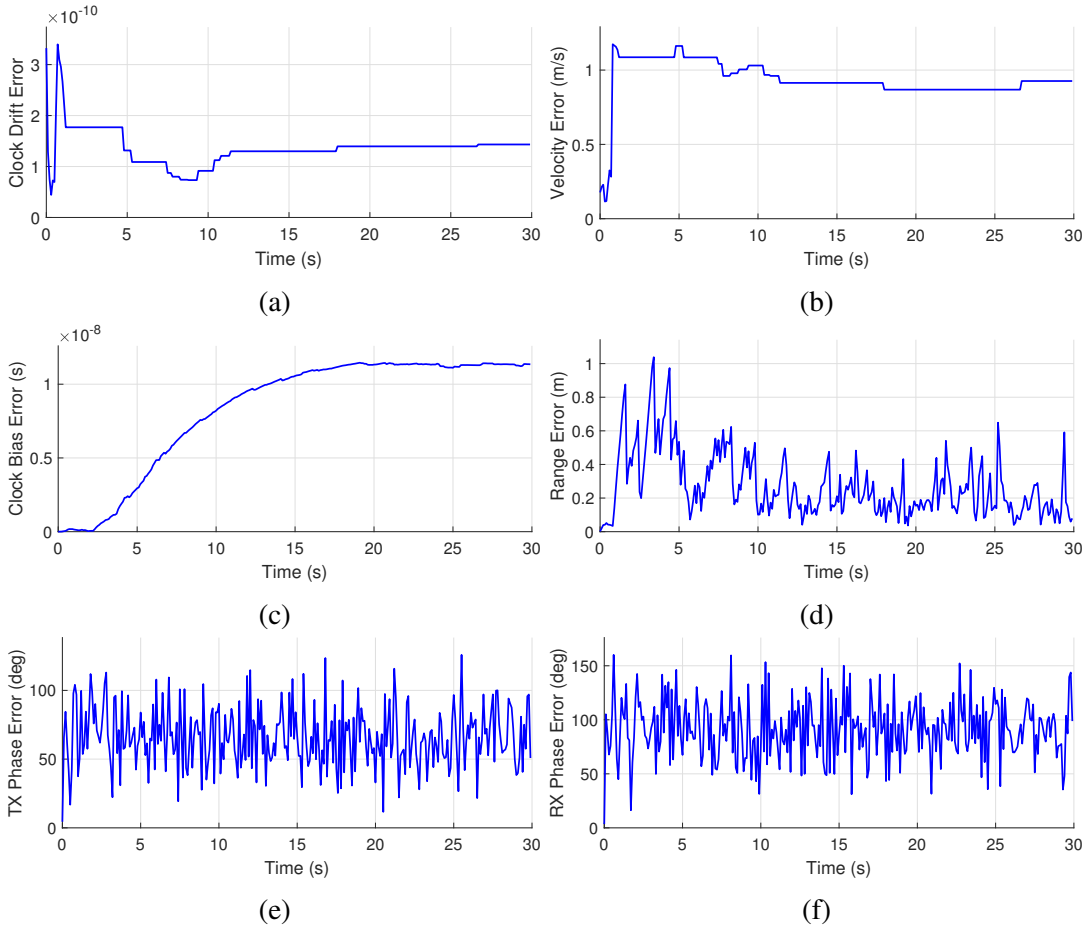


Fig. 5.8: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed using an SNR of 20 dB, leading to significant error and a general collapse of the estimation process. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

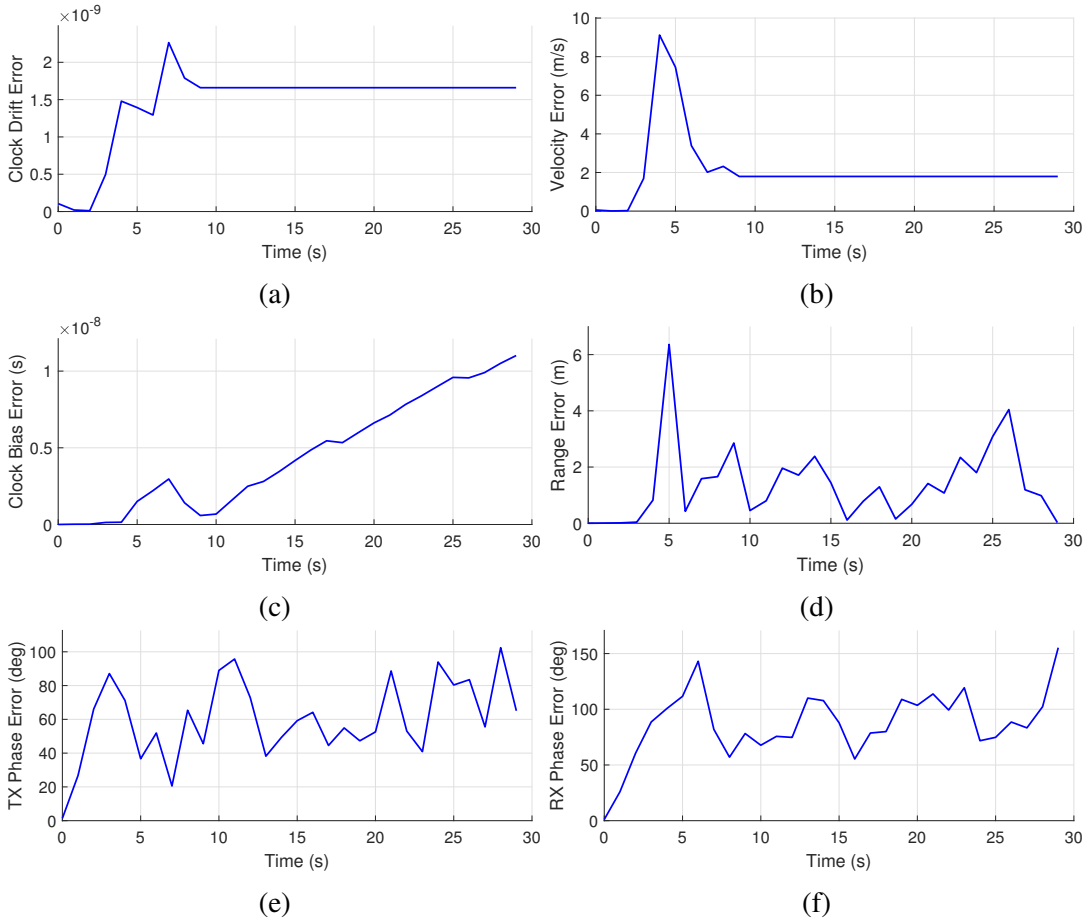


Fig. 5.9: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed with a UKF update rate of 1 Hz, leading to significant error and a general collapse of the estimation process. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

high update rate. This is likely due to the measured phase value changing due to the clock drift error in the carrier correction. Even a good estimate of clock drift will leave residual error in the carrier frequency correction, which manifests as a low-frequency skew. If the low-frequency skew is not sampled fast enough, the phase will alias, and the estimation cannot be performed correctly.

5.3.3.5 Impact of Platform Motion

Finally, Fig. 5.10 shows the simulated mean error in a scenario of four platforms where there are relative velocities between platforms. As with the other disruptions, this causes the error to grow unacceptably. This is likely due to an error with how the Doppler shift frequency is compensated to accurately compute phase on each iteration as well as how the relative motion is computed alongside range, ultimately leading to divergence. Ultimately, this limitation is possibly the most limiting as the eventual goal is to add the synchronization together with the navigation, which by nature requires motion to be included.

5.3.4 Discussion of Advantages and Limitations

This approach has several implementation issues which ultimately make it unsuitable for use in a practical scenario. The results when using this approach diverge significantly even in fairly straightforward scenarios, such that the error increases significantly for even moderately low SNR values. Furthermore, the algorithm requires a relatively quick update rate of the UKF, with performance degrading significantly when the update rate is less than a few Hz. This may be a significant issue in networks where downtime is a concern since the exchange of signals can take several milliseconds. This will be particularly pronounced if TDMA is required to multiplex the

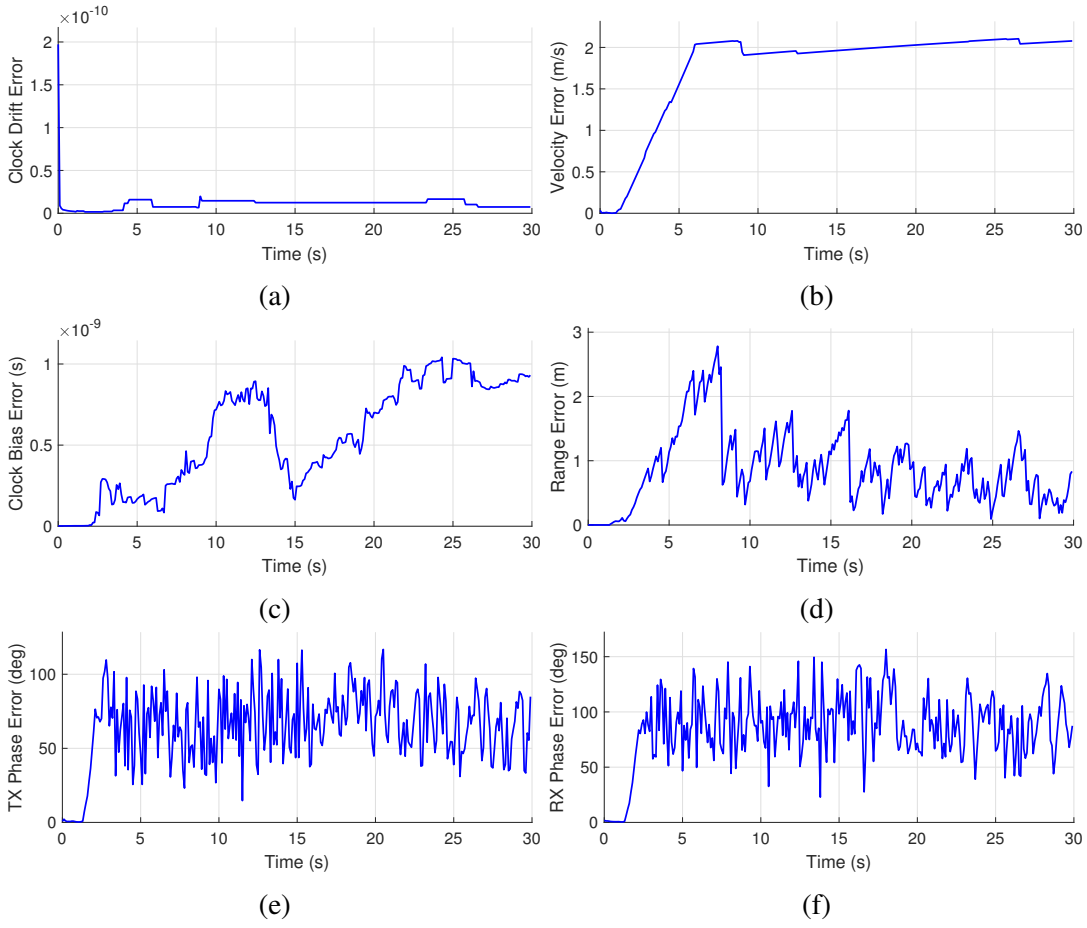


Fig. 5.10: Simulated UKF synchronization error with the time, phase, and frequency approach. These results are computed with small but non-zero motion between platforms, leading to significant error and a general collapse of the estimation process. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

signals, which highlights the desirability of other forms of orthogonality [98]. Most crucially, this approach stops working well when actual platform motion is included – it can correctly estimate the synchronization states and relative velocities when those velocities are zero, but ultimately cannot work well when there are accelerations between platforms.

The primary advantage of this approach is that the clock drift estimation begins to use time and phase information in addition to frequency information. This is hugely helpful due to the increased information resolution provided by pulse compression waveforms, which can have high bandwidths rather than requiring long pulse lengths – this is in contrast to frequency estimation, which can only have increased information resolution by increasing the pulse length, which, per the discussion above, is not desirable due to downtime considerations. However, this advantage is considerable since the clock drift error in this approach (when the proper conditions are met) approaches sub-ppt level, which would lead to extraordinarily well-synchronized systems.

5.4 Iterative Synchronization Approach

Due to the limitations of the time, phase, and frequency approach, another approach to the synchronization filtering is proposed here. In this approach, the full synchronization algorithm described at the beginning of this chapter is performed at regular intervals, producing direction one-to-one measurements of the state variables in the clock parameter estimates as well as the range and relative velocity estimates. By doing so with properly computed covariance, the UKF (or theoretically a linear Kalman filter) can be used to fuse the measurements with the state estimates to produce a more accurate estimate over time.

This approach uses the same transmit signal form of $s_j(t)$ described in (5.11) since

this signal contains all the information required to perform the synchronization algorithm. Furthermore, as with the time, phase, and frequency approach, because clock drift is aligned well enough from the initial synchronization, all the information from both components of the signal will be virtually guaranteed to fall within the properly timed receive window.

The primary change for this synchronization filtering approach when compared to the time, phase, and frequency approach is in the measurement model. Rather than using the direct time, phase, and frequency estimates as measurements, the UKF measurements are taken from the output of the synchronization procedure. Define $\mathbf{z}^k_{\alpha,\mathrm{IS}}$ as the measurement vector for iteration k containing the clock drift measurements obtained from the k^{th} iteration synchronization, given by

$$\mathbf{z}^k_{\alpha,\mathrm{IS}} = \begin{bmatrix} \hat{\alpha}^{z,k}_1 & \dots & \hat{\alpha}^{z,k}_{N_p} \end{bmatrix}^T, \tag{5.31}$$

where $\hat{\alpha}^{z,k}_i$ is the clock drift estimate for platform i produced as a measurement for the update step of the synchronization UKF on iteration k. In a similar manner, define $\mathbf{z}^k_{\phi,\mathrm{IS}}$ as

$$\mathbf{z}^k_{\phi,\mathrm{IS}} = \begin{bmatrix} \hat{\phi}^{z,k}_1 & \dots & \hat{\phi}^{z,k}_{N_p} \end{bmatrix}^T, \tag{5.32}$$

define $\mathbf{z}^k_{\gamma,\mathrm{IS}}$ as

$$\mathbf{z}^k_{\gamma,\mathrm{IS}} = \begin{bmatrix} \hat{\gamma}^{\mathrm{tx},z}_{1,k} & \hat{\gamma}^{\mathrm{rx},z}_{1,k} & \dots & \hat{\gamma}^{\mathrm{tx},z}_{N_p,k} & \hat{\gamma}^{\mathrm{rx},z}_{N_p,k} \end{bmatrix}^T, \tag{5.33}$$

define $\mathbf{z}^k_{R,\mathrm{IS}}$ as

$$\mathbf{z}^k_{R,\mathrm{IS}} = \begin{bmatrix} \hat{R}^{z,k}_{1,2} & \hat{R}^{z,k}_{1,3} & \dots & \hat{R}^{z,k}_{1,N_p} & \hat{R}^{z,k}_{2,3} & \dots & \hat{R}^{z,k}_{N_p-1,N_p} \end{bmatrix}^T, \tag{5.34}$$

and define $\mathbf{z}^k_{v,\mathrm{IS}}$ as

$$\mathbf{z}^k_{v,\mathrm{IS}} = \begin{bmatrix} \hat{v}^{z,k}_{1,2} & \hat{v}^{z,k}_{1,3} & \dots & \hat{v}^{z,k}_{1,N_p} & \hat{v}^{z,k}_{2,3} & \dots & \hat{v}^{z,k}_{N_p-1,N_p} \end{bmatrix}^T. \tag{5.35}$$

In (5.32) and (5.33), $\hat{\phi}^{z,k}_i$, $\hat{\gamma}^{\mathrm{tx},z}_{i,k}$, and $\hat{\gamma}^{\mathrm{rx},z}_{i,k}$ are the clock bias, TX carrier phase, and RX carrier phase, respectively, of platform i produced as measurements for the update step of the synchronization UKF on iteration k, and in (5.34) and (5.35), $\hat{R}^{z,k}_{i,j}$ and $\hat{v}^{z,k}_{i,j}$ are the range and velocity, respectively, between platforms i and j produced as measurements for the update step of the synchronization UKF on iteration k. Finally, these vectors may be constructed to form the full measurement vector $\mathbf{z}^{\mathrm{IS}}_k$ given by

$$\mathbf{z}^{\mathrm{IS}}_k = \begin{bmatrix} \mathbf{z}^k_{\alpha,\mathrm{IS}} \\ \mathbf{z}^k_{v,\mathrm{IS}} \\ \mathbf{z}^k_{\phi,\mathrm{IS}} \\ \mathbf{z}^k_{R,\mathrm{IS}} \\ \mathbf{z}^k_{\gamma,\mathrm{IS}} \end{bmatrix}. \tag{5.36}$$

Because the measurement vector above will be formed using the WLS process in which clock drifts and velocities are solved for together, it will be convenient to define a measurement vector $\mathbf{z}^k_{f,\mathrm{IS}}$ where

$$\mathbf{z}^k_{f,\mathrm{IS}} = \begin{bmatrix} \mathbf{z}^k_{\alpha,\mathrm{IS}} \\ \mathbf{z}^k_{v,\mathrm{IS}} \end{bmatrix}. \tag{5.37}$$

Because the measurement vector aligns exactly with the values expected in the state vector, the measurement function h_k will be the identity function.

5.4.1 Modifications to Synchronization Procedure

To use the iterative synchronization approach, the original synchronization procedure must be slightly modified to account for the fact that the transmit and receive corrections for the different clock parameters are being applied, and that to accurately track the clock parameters, these corrections must be "undone" during each subsequent iteration's estimation. This is accomplished simply by acknowledging that the synchronization procedure, when performed on the compensated signals, produces estimates of the corrected clock parameters rather than the actual clock parameters, and by backing the previous estimate of the clock parameters out, updated estimates of the true clock parameters can be produced.

From the original synchronization procedure, the clock parameter, range, and velocity estimates are all produced from direct estimates of time, phase, and frequency. Therefore, the estimates of these quantities computed by (5.23), (5.24), and (5.26) may be used for this purpose.

To begin with the frequency estimation, (5.26) may be rearranged to give

$$\frac{\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}}\left(f_j^{\alpha}+f^c\right)\alpha_j^k-\left(\hat{f}_{i,j}^k+f^c\right)\alpha_i^k-\hat{\alpha}_i^{k-1}v_{i,j}^k\frac{f^c}{c}=0. \tag{5.38}$$

This equation may be solved as a system of equations as in (4.92) to produce the measurement vector $\mathbf{z}_{f,\mathrm{IS}}^k$.

Next, consider the time-delay estimate $\hat{\tau}_{i,j}^k$ given in (5.23). Define the bias error for platform i on iteration k as $\phi_i^{e,k}$ given by

$$\phi_i^{e,k}=\phi_i^k-\hat{\phi}_i^{k-1}. \tag{5.39}$$

The time-delay estimate in (5.23) may therefore be expressed by

$$\hat{\tau}_{i,j}^{k} = \frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}}\phi_j^{e,k} + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}\frac{R_{i,j}^k}{c} + \frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}}\left(\tau_{i,j}^{A,k} + T_{j,k}^{\mathrm{tx}}\right). \tag{5.40}$$

Under the assumption made in (4.43), this can be simplified to

$$\hat{\tau}_{i,j}^{k} = \frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{1}{\hat{\alpha}_j^{k-1}}\phi_j^{e,k} + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}\frac{R_{i,j}^k}{c} + \tau_{i,j}^{A,k} + T_{j,k}^{\mathrm{tx}}. \tag{5.41}$$

Finally, assuming that $\tau_{i,j}^{A,k}$ can be estimated from the frequency estimate and that $T_{j,k}^{\mathrm{tx}}$ is known, these can be subtracted off to produce an augmented time-delay estimate $\tilde{\tau}_{i,j}^{k}$

$$\tilde{\tau}_{i,j}^{k} = \frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{1}{\hat{\alpha}_j^{k-1}}\phi_j^{e,k} + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}\frac{R_{i,j}^k}{c}. \tag{5.42}$$

This simplifying assumption removes the erroneous clock drift scaling of $\tau_{i,j}^{A,k}$, and more importantly, $T_{j,k}^{\mathrm{tx}}$. Because $T_{j,k}^{\mathrm{tx}}$ will quickly grow large relative to the clock drift error ratio, the error resulting from this simplification will become significant – however, this significant error can be absorbed into the clock bias estimate and will not impact the range estimate, as will be shown in Section 5.4.1.1. Noting that the new expression in (5.42) has an identical form to the original expression in (4.49), the range estimates $\hat{R}_{i,j}^{z,k}$ may be formed by computing

$$\hat{R}_{i,j}^{z,k} = \frac{\tilde{\tau}_{i,j}^{k} + \tilde{\tau}_{j,i}^{k}}{2}. \tag{5.43}$$

Similarly define $\phi_{i,j}^{e,k}$ as the bias error difference between platforms i and j on iteration k given by

$$\phi_{i,j}^{e,k} = \frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{1}{\hat{\alpha}_j^{k-1}}\phi_j^{e,k}. \tag{5.44}$$

An estimate of the bias error difference $\hat{\phi}_{i,j}^{e,k}$ may be computed by

$$\hat{\phi}_{i,j}^{e,k} = \frac{\tilde{\tau}_{i,j}^{k} - \tilde{\tau}_{j,i}^{k}}{2}. \tag{5.45}$$

Estimates of the clock bias errors, $\hat{\phi}_{i}^{e,k}$, may be formed by solving the system of equations in (5.44) in the same manner as in (4.94). Finally, the true clock bias measurements, $\hat{\phi}_{i}^{z,k}$, may be formed for all platforms by computing

$$\hat{\phi}_{i}^{z,k} = \hat{\phi}_{i}^{e,k} + \hat{\phi}_{i}^{k-1}. \tag{5.46}$$

Finally, consider the phase estimate $\hat{\gamma}_{i,j}^{k}$ given in (5.24). Define the transmit carrier phase error for platform i on iteration k as $\gamma_{i,k}^{\mathrm{tx},e}$ given by

$$\gamma_{i,k}^{\mathrm{tx},e} = \gamma_{i,k}^{\mathrm{tx}} - \hat{\gamma}_{i,k-1}^{\mathrm{tx}} \tag{5.47}$$

and the receive carrier phase error for platform i on iteration k as $\gamma_{i,k}^{\mathrm{rx},e}$ given by

$$\gamma_{i,k}^{\mathrm{rx},e} = \gamma_{i,k}^{\mathrm{rx}} - \hat{\gamma}_{i,k-1}^{\mathrm{rx}}. \tag{5.48}$$

Using these phase error values, the phase estimate in (5.24) may be re-written as

$$\begin{aligned}\hat{\gamma}_{i,j}^{k} = & -2\pi f^{c}\left(\frac{\alpha_{j}^{k}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}\phi_{i}^{e,k} - \frac{1}{\hat{\alpha}_{j}}\phi_{j}^{e,k} + \frac{\alpha_{j}^{k}}{\hat{\alpha}_{j}^{k-1}}\frac{R_{i,j}^{k}}{c}\right) + 2\pi f^{c}\frac{1}{\alpha_{i}^{k}}v_{i,j}^{k}\frac{f^{c}}{c}\phi_{i}^{e,k} + \left(\gamma_{j,k}^{\mathrm{tx},e} - \gamma_{i,k}^{\mathrm{rx},e}\right) \\ & + \angle A_{j}\left(\tau_{i,j}^{A,k},\ f^{c}\left(\frac{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}} - 1\right) - \frac{\hat{\alpha}_{i}^{k-1}}{\alpha_{i}^{k}}v_{i,j}^{k}\frac{f^{c}}{c}\right) + 2\pi f^{c}\left(\frac{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}} - 1\right)T_{j,k}^{\mathrm{tx}}.\end{aligned} \tag{5.49}$$

As with the time-delay estimate, the phase due to the ambiguity function phase can be

compensated for directly. It may also be assumed that if (5.21) holds, the phase term due to the carrier offset and the transmit time will be zero and can be ignored. Thus, a new augmented phase estimate, $\tilde{\gamma}^k_{i,j}$, may be produced, which is given by

$$\tilde{\gamma}^k_{i,j} = -2\pi f^c \left(\frac{\alpha^k_j}{\hat{\alpha}^{k-1}_j \alpha^k_i} \phi^{e,k}_i - \frac{1}{\hat{\alpha}_j} \phi^{e,k}_j + \frac{\alpha^k_j}{\hat{\alpha}^{k-1}_j} \frac{R^k_{i,j}}{c} \right) + 2\pi f^c \frac{1}{\alpha^k_i} v^k_{i,j} \frac{f^c}{c} \phi^{e,k}_i + \left(\gamma^{\mathrm{tx},e}_{j,k} - \gamma^{\mathrm{rx},e}_{i,k} \right). \tag{5.50}$$

As with the time-delay estimate, the simplifying assumption which removes $T^{\mathrm{tx}}_{j,k}$ will not actually hold in practice and will introduce phase errors since $T^{\mathrm{tx}}_{j,k}$ grows large relative to the clock drift error ratio. However, the errors will not impede the proper estimation of the carrier phase values since they will ultimately be canceled out. This is discussed further in Section 5.4.1.1. Define the carrier phase error difference term between platforms i and j on iteration k, $\gamma^{\mathrm{err},e}_{i,j,k}$, as

$$\gamma^{\mathrm{err},e}_{i,j,k} = \gamma^{\mathrm{tx},e}_{j,k} - \gamma^{\mathrm{rx},e}_{i,k}. \tag{5.51}$$

This error term describes the phase difference between the compensated TX carrier phase of platform j and the compensated RX carrier phase of platform i in the same manner as $\gamma^{\mathrm{err}}_{i,j}$ for the uncompensated case. In the same manner as (4.68), the expression in (5.50) can be rearranged to produce

$$\gamma^{\mathrm{err},e}_{i,j,k} = \tilde{\gamma}^k_{i,j} + 2\pi f^c \left(\frac{\alpha^k_j}{\hat{\alpha}^{k-1}_j \alpha^k_i} \phi^{e,k}_i + \frac{1}{\hat{\alpha}_j} \phi^{e,k}_j + \frac{\alpha^k_j}{\hat{\alpha}^{k-1}_j} \frac{R^k_{i,j}}{c} \right) - 2\pi f^c \frac{1}{\alpha^k_i} v^k_{i,j} \frac{f^c}{c} \phi^{e,k}_i. \tag{5.52}$$

Noting that the delay phase term in the expression above is approximately equivalent to the carrier frequency multiplied by the delay estimate $\tilde{\tau}^k_{i,j}$ in (5.42), an estimate $\hat{\gamma}^{\mathrm{err},e}_{i,jk}$

may be computed by

$$\hat{\gamma}_{i,j,k}^{\text{err},e} = \tilde{\gamma}_{i,j}^{k} + 2\pi f^{c}\left(\tilde{\tau}_{i,j}^{k} - \hat{v}_{i,j}^{z,k}\frac{f^{c}}{c}\hat{\phi}_{i}^{e,k}\right), \tag{5.53}$$

which has the same form as (4.68) producing the original phase difference estimate, with the minor change that the Doppler shift frequency is replaced with the expression in terms of the relative platform velocity. From these values of $\hat{\gamma}_{i,j,k}^{\text{err},e}$, the system of equations in (5.51) may be solved in the same manner as in (4.96) to produce estimates of the carrier phase errors, $\hat{\gamma}_{i,k}^{\text{tx},e}$ and $\hat{\gamma}_{i,k}^{\text{rx},e}$. Note that as with the original solution for carrier phase, this must be solved to account for phase wrapping as described in 4.4.1.4. Finally, the direct measurements of the carrier phases may be formed by

$$\hat{\gamma}_{i,k}^{\text{tx},z} = \hat{\gamma}_{i,k}^{\text{tx},e} + \hat{\gamma}_{i,k-1}^{\text{tx}}, \tag{5.54}$$

and

$$\hat{\gamma}_{i,k}^{\text{rx},z} = \hat{\gamma}_{i,k}^{\text{rx},e} + \hat{\gamma}_{i,k-1}^{\text{rx}}. \tag{5.55}$$

5.4.1.1 TX Time Errors in Clock Bias and Carrier Phase

As discussed in the previous subsection, the transmit time term $T_{j,k}^{\text{tx}}$ is removed from the clock bias and carrier phase computations. However, because it is scaled by the clock drift ratio term, there will be a small but ultimately non-negligible error in the time-delay and phase terms. However, the structure of these errors is such that they will be canceled out. The un-simplified time-delay measurement in (5.40) may be differently expressed as

$$\hat{\tau}_{i,j}^{k} = \frac{1}{\hat{\alpha}_{i}^{k-1}}\phi_{i}^{e,k} - \frac{\alpha_{i}^{k}}{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}}\phi_{j}^{e,k} + \frac{\hat{\alpha}_{i}^{k-1}}{\alpha_{i}^{k}}\frac{R_{i,j}^{k}}{c} + \tau_{i,j}^{A,k} + T_{j,k}^{\text{tx}} + \left(\frac{\hat{\alpha}_{j}^{k-1}\alpha_{i}^{k}}{\alpha_{j}^{k}\hat{\alpha}_{i}^{k-1}} - 1\right)\left(\tau_{i,j}^{A,k} + T_{j,k}^{\text{tx}}\right). \tag{5.56}$$

When the ambiguity function time shift and transmit time are removed, the true value of this augmented time-delay will be given by

$$\tilde{\tau}_{i,j}^{k} = \frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}}\phi_j^{e,k} + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}\frac{R_{i,j}^k}{c} + \left(\frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}} - 1\right)\left(\tau_{i,j}^{A,k} + T_{j,k}^{\text{tx}}\right). \tag{5.57}$$

Crucially, it can be shown that

$$\frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}} - 1 \approx 1 - \frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k}. \tag{5.58}$$

To show this, let z_r be a temporary variable defined by

$$z_r = \frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}} \approx 1. \tag{5.59}$$

The degree to which $z_r \approx 1$ will generally follow the accuracy of the estimates of the clock drift – for instance, if the standard deviation of the clock drift error is 1 ppb, z_r has a high likelihood to fall within 1 ppb of 1. To show that (5.58) is true, it must be shown that when $z_r \approx 1$,

$$z_r - 1 \approx 1 - \frac{1}{z_r}. \tag{5.60}$$

First, it is trivial to show that

$$(z_r - 1)\Big|_{z_r=1} = \left(1 - \frac{1}{z_r}\right)\Big|_{z_r=1} = 0. \tag{5.61}$$

Furthermore, differentiating with respect to z_r shows that

$$\frac{d}{dz_r}(z_r - 1)\Big|_{z_r=1} = 1 \tag{5.62}$$

$$\frac{d}{dz_r}\left(1-\frac{1}{z_r}\right)\Big|_{z_r=1} = \left(\frac{1}{z_r^2}\right)\Big|_{z_r=1} = 1. \tag{5.63}$$

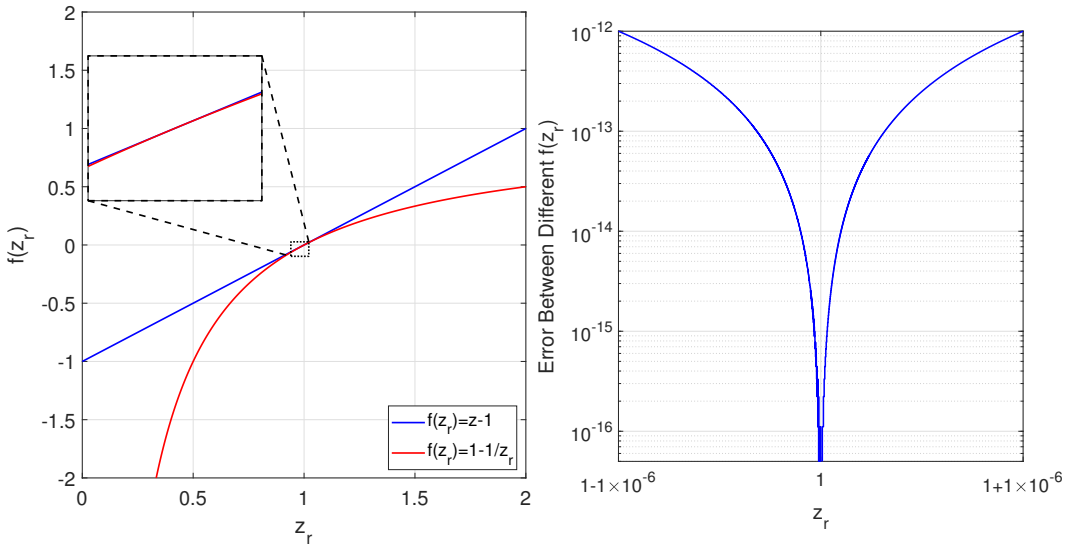
Thus, the function $z_r - 1$ and $1 - 1/z_r$ are tangent to each other at $z_r = 1$, and as such (5.60) is true so long as z_r is close to 1. Therefore, it follows that (5.58) is true. Figure. 5.11 shows plots of these two functions and how they are tangent to each other at $z_r = 1$. It also shows that for small deviations of z_r around 1 (within 1×10^{-6} of 1), the error between the functions does not exceed 1×10^{-12}, or one ppt.

Therefore, there is negative symmetry in this error term across $\hat{\tau}_{i,j}^k$ and $\hat{\tau}_{j,i}^k$ in a similar manner to the clock biases. As a result, in computing the range in (5.43), the clock drift offset induced error will cancel out, while it will appear in the clock bias difference estimate – that is, the true clock bias difference estimate may be more accurately expressed as

$$\phi_{i,j}^{e,k} = \frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{1}{\hat{\alpha}_j^{k-1}}\phi_j^{e,k} + \left(\frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}} - 1\right)\left(\tau_{i,j}^{A,k} + T_{j,k}^{\text{tx}}\right). \tag{5.64}$$

As a result, the additional time-delay error due to the clock drift offset and will not appear in the range estimates, ensuring that the range measurements can still be accurately used in relative navigation. However, the clock bias estimates will absorb this error. This is an intuitive result, since clock drift errors over time will appear as a "drift" in the clock bias.

Next, the carrier phase estimates will also have error as a result of the residual uncompensated clock drifts. This manifests as very low-frequency carrier offsets which cause the apparent carrier phases between platforms to drift between synchronization iterations on the longer time scales seen in $T_{j,k}^{\text{tx}}$. Generally, the phase due to the ambiguity function will be small enough that ignoring it will not cause noticeable error. However, the carrier phase drift will be significant. To account for this, define $\check{\gamma}_{i,j}^k$ as



(a) (b)

Fig. 5.11: Plots of the functions $z_r - 1$ and $1 - 1/z_r$ (a) and the error between them locally around $z_r = 1$ (b).

the true value of $\tilde{\gamma}_{i,j}^k$, given by

$$\begin{aligned}
\check{\gamma}_{i,j}^k &= \tilde{\gamma}_{i,j}^k + 2\pi f^c \left(\frac{\alpha_j^k \hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1} \alpha_i^k} - 1 \right) T_{j,k}^{\text{tx}} \\
&= -2\pi f^c \left(\frac{\alpha_j^k}{\hat{\alpha}_j^{k-1} \alpha_i^k} \phi_i^{e,k} - \frac{1}{\hat{\alpha}_j} \phi_j^{e,k} + \frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}} \frac{R_{i,j}^k}{c} \right) + 2\pi f^c \frac{1}{\alpha_i^k} v_{i,j}^k \frac{f^c}{c} \phi_i^{e,k} + \left(\gamma_{j,k}^{\text{tx},e} - \gamma_{i,k}^{\text{rx},e} \right) \\
&\quad + 2\pi f^c \left(\frac{\alpha_j^k \hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1} \alpha_i^k} - 1 \right) T_{j,k}^{\text{tx}},
\end{aligned} \tag{5.65}$$

and therefore the phase error difference is more accurately expressed by

$$\begin{aligned}\gamma_{i,j,k}^{\mathrm{err},e} &= \breve{\gamma}_{i,j}^{k} + 2\pi f^{c}\left(\frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}\alpha_i^k}\phi_i^{e,k} + \frac{1}{\hat{\alpha}_j}\phi_j^{e,k} + \frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}}\frac{R_{i,j}^k}{c}\right) - 2\pi f^{c}\frac{1}{\alpha_i^k}v_{i,j}^k\frac{f^c}{c}\phi_i^{e,k} \\ &= \tilde{\gamma}_{i,j}^{k} + 2\pi f^{c}\left(\frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}\alpha_i^k}\phi_i^{e,k} + \frac{1}{\hat{\alpha}_j}\phi_j^{e,k} + \frac{\alpha_j^k}{\hat{\alpha}_j^{k-1}}\frac{R_{i,j}^k}{c}\right) - 2\pi f^{c}\frac{1}{\alpha_i^k}v_{i,j}^k\frac{f^c}{c}\phi_i^{e,k} \\ &\quad + 2\pi f^{c}\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k} - 1\right)T_{j,k}^{\mathrm{tx}}.\end{aligned} \tag{5.66}$$

The actual estimate of this phase error difference then is expressed by

$$\hat{\gamma}_{i,j,k}^{\mathrm{err},e} = \tilde{\gamma}_{i,j}^{k} + 2\pi f^{c}\left(\tilde{\tau}_{i,j}^{k} - \hat{v}_{i,j}^{z,k}\frac{f^c}{c}\hat{\phi}_i^{e,k}\right) + 2\pi f^{c}\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k} - 1\right)T_{j,k}^{\mathrm{tx}}. \tag{5.67}$$

However, from (5.57), the time-delay will also contain a similar error. When the full expression for $\tilde{\tau}_{i,j}^{k}$ is included, (5.67) will be given by

$$\begin{aligned}\hat{\gamma}_{i,j,k}^{\mathrm{err},e} &= \tilde{\gamma}_{i,j}^{k} + 2\pi f^{c}\left(\frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}}\phi_j^{e,k} + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}\frac{R_{i,j}^k}{c} + \left(\frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}} - 1\right)\left(\tau_{i,j}^{A,k} + T_{j,k}^{\mathrm{tx}}\right)\right) \\ &\quad - 2\pi f^{c}\left(\hat{v}_{i,j}^{z,k}\frac{f^c}{c}\hat{\phi}_i^{e,k}\right) + 2\pi f^{c}\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k} - 1\right)T_{j,k}^{\mathrm{tx}} \\ &= \tilde{\gamma}_{i,j}^{k} + 2\pi f^{c}\left(\frac{1}{\hat{\alpha}_i^{k-1}}\phi_i^{e,k} - \frac{\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}}\phi_j^{e,k} + \frac{\hat{\alpha}_i^{k-1}}{\alpha_i^k}\frac{R_{i,j}^k}{c}\right) - 2\pi f^{c}\left(\hat{v}_{i,j}^{z,k}\frac{f^c}{c}\hat{\phi}_i^{e,k}\right) \\ &\quad + 2\pi f^{c}\left(\left(\frac{\alpha_j^k\hat{\alpha}_i^{k-1}}{\hat{\alpha}_j^{k-1}\alpha_i^k} - 1\right)T_{j,k}^{\mathrm{tx}} + \left(\frac{\hat{\alpha}_j^{k-1}\alpha_i^k}{\alpha_j^k\hat{\alpha}_i^{k-1}} - 1\right)\left(\tau_{i,j}^{A,k} + T_{j,k}^{\mathrm{tx}}\right)\right).\end{aligned} \tag{5.68}$$

Assuming that $\tau_{i,j}^{A,k}$ is small compared to $T_{j,k}^{\mathrm{tx}}$ (which is true unless the Doppler velocity is unrealistically large), it can be ignored. As a result, based on the equivalence in (5.58), the error terms at the end of the expression in (5.68) cancel each other out,

and the phase error difference estimate will contain only the desired phase information.

5.4.2 Covariance of Measurements

The measurement covariance, $\mathbf{R}_k^{\text{IS}}$, for the iterative synchronization approach will be defined in an identical manner to the initial system state covariance, $\hat{\mathbf{C}}_0^{\text{sync}}$, in (5.10) because the measurements are obtained from the same operations as the initial system state. Therefore,

$$\mathbf{R}_k^{\text{IS}} = \begin{bmatrix} \Sigma_{f,v}^k & \dots & \dots & \mathbf{0} \\ \vdots & \Sigma_{\phi}^k & \ddots & \vdots \\ \vdots & \ddots & \Sigma_R^k & \vdots \\ \mathbf{0} & \dots & \dots & \Sigma_{\gamma}^k \end{bmatrix}, \tag{5.69}$$

where the constituent covariance matrices must be recomputed based on the measured SNR at iteration k.

5.4.3 Preliminary Results

In this section, some preliminary results demonstrating the efficacy of the iterative synchronization approach are provided. A set of four platforms with randomly generated clock error parameters and positions are simulated. As with the results for the time, phase, and frequency approach in Section 5.3.3, the clock parameters are assumed to remain constant throughout the simulation time.

The following normalizations of the state variables and measurements are used to ensure numerical stability of the UKF. The clock drift and clock bias state values and measurements are normalized by a factor of 10, while the range and velocity state estimates and measurements are normalized by a factor of $1/c$. The carrier phase state

values and measurements are normalized by a factor of 10^{-3}.

The following parameters are used for the simulations, which are identical to the parameters used in the time, phase, and frequency approach. Unless otherwise stated, the SNR is set to 30 dB. The carrier frequency f^c is set to 3 GHz. The pulse length is set to 100 μs, while the sampling frequency is set to 200 MHz. The capture length in the initial synchronization is set to 400 μs, while it is set to 600 μs in the subsequent iterations of the UKF to account for the longer signal. The signal bandwidth in the LFM waveforms is set to 50 MHz. The TDMA scheme is used with the time slot set to 1 ms. The clock drift errors are drawn from a uniform distribution limited to within 25 ppm of 1. The clock bias errors are drawn from a Gaussian distribution with a standard deviation of 100 ns. The full simulations are conducted over 30 s with iterations taking place at 10 Hz, unless otherwise stated.

As will be seen, the clock drift estimation using this technique is not quite as accurate as with the time, phase, and frequency approach. As a result, as described above, the clock bias absorbs a large amount of error due to the compounding error of the clock drift error multiplied into $T^{\text{tx}}_{j,k}$. However, the measurements of clock bias at each iteration will still be highly accurate and will generally capture this error such that synchronization can still be achieved. Therefore, a process covariance term for clock bias is added here so that the previous state estimate for clock bias is ignored in favor of the most recent measurements. This is not strictly necessary since the measurement itself may be used as the true clock bias value and the tracked state may generally be ignored or omitted. The downside of the apparent shift in clock bias at each iteration is that the changing values cannot be tracked effectively and thus the accuracy will not build over time. Fortunately, because the original time-delay estimation is so accurate, this will not typically be necessary and will only pose problems in scenarios with

extremely low SNR.

5.4.3.1 Stationary Systems, Assuming No Motion

In this first case study, a network of four stationary platforms with randomly distributed positions is used. The platform positions are identical to the stationary platform positions used in the time, phase, and frequency results in Section 5.3.3. In this scenario, the UKF is formatted to assume the platforms are stationary, such that the relative platform velocities are omitted from the state vector and the processing is slightly modified to assume that there will be no Doppler shifts.

Fig. 5.12 shows the error results for a single platform in the network of four platforms, while Fig. 5.13 shows these same results for the mean error for the full network of four platforms. There are several noteworthy aspects of these results. As expected, the clock drift and carrier phase errors tend to converge toward zero. Interestingly, the range errors don't change significantly over the course of the simulation, indicating that the initial covariance is heavily weighted enough toward confidence that there is no need to overwrite the results. This phenomenon does not carry over when motion is introduced. The clock bias error in Figs. 5.12c and 5.13c is plotted in two ways. The first, in blue, shows the absolute clock bias error. The second, in cyan, shows the clock bias error normalized to the additional time shift error due to the clock drift estimation error. These plots show that although the absolute error will randomly walk, the normalized error (which accounts for the actual necessary clock bias compensation due to clock drift error) will remain very low, falling down to the picosecond level.

The primary disadvantage here is that the clock drift error does not reach the same low level as it does in the time, phase, and frequency approach, achieving a mean error of 5 ppt over the 30 s simulation time compared to the $<$ 1 ppt in the time, phase, and

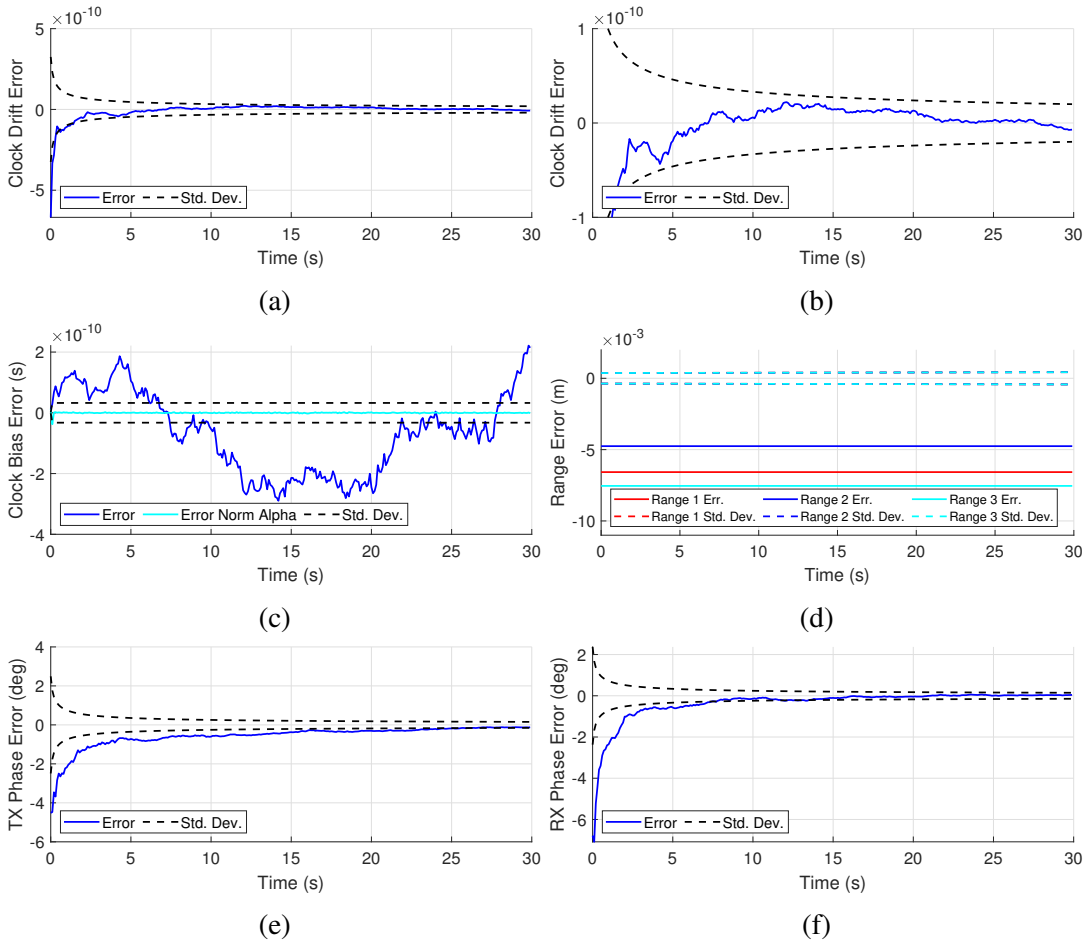


Fig. 5.12: Simulated UKF synchronization error with the iterative synchronization approach. These results are computed for a single platform in a stationary network of four platforms with relative velocities assumed in the UKF to be zero. The above plots show the error for a single platform of clock drift (a), clock drift (zoomed) (b), clock bias (c), range with the other three platforms (d), TX carrier phase (e), and RX carrier phase (f). The solid lines show the error, while the dashed lines indicate the expected standard deviation of the error bound predicted by the UKF.

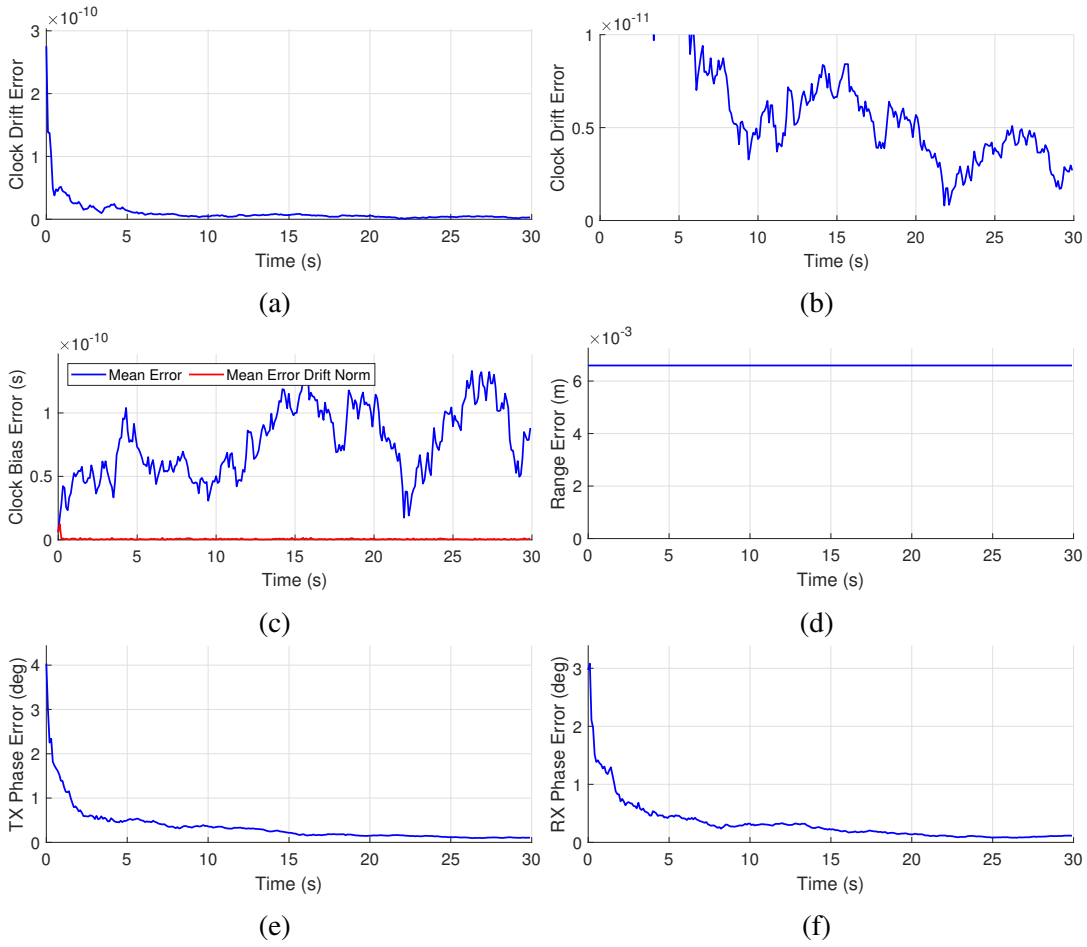


Fig. 5.13: Simulated UKF synchronization error with the iterative synchronization approach. These results are computed as the mean error for a stationary network of four platforms with relative velocities assumed in the UKF to be zero. The above plots show the mean error for clock drift (a), clock drift (zoomed) (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

frequency approach. As discussed above, this is because the iterative synchronization approach only leverages the frequency domain information for clock drift estimation inherent to the original synchronization procedure. The lack of high-resolution time-domain information causes the error to reduce at a slower rate and to converge toward a generally higher error value. While this is a major limitation when compared to the time, phase, and frequency approach, the remaining test cases that would not work properly with that approach will still enable correct tracking of variables with the iterative synchronization approach.

5.4.3.2 Stationary Systems, Assuming Motion

In this example, the same network of four stationary platforms is used. In this scenario, the UKF is formatted to assume the platforms are not stationary as is described in the problem formulation previously in this chapter. As with this similar case for the time, phase, and frequency approach in Section 5.3.3.2, Σ_v^k is assumed to be the scalar matrix $\Sigma_v^k = 10^{-3}\mathbf{I}$ to account for the unknown relative accelerations between platforms.

Fig. 5.14 shows the error results for a single platform in the network of four platforms, while Fig. 5.15 shows these same results for the mean error for the full network of four platforms. These results align with the performance when stationary platforms were assumed, with the clock drift and carrier phase estimate errors approaching zero. The clock bias results continue to show low but not convergent error when accounting for the impact of non-zero clock drift error. The relative velocity errors drop very near to zero after only a few iterations, showing that the UKF quickly converges on these values. The range errors tend to stay relatively constant, but with some noise added due to the small errors in the relative velocity estimation. As with the time, phase, and

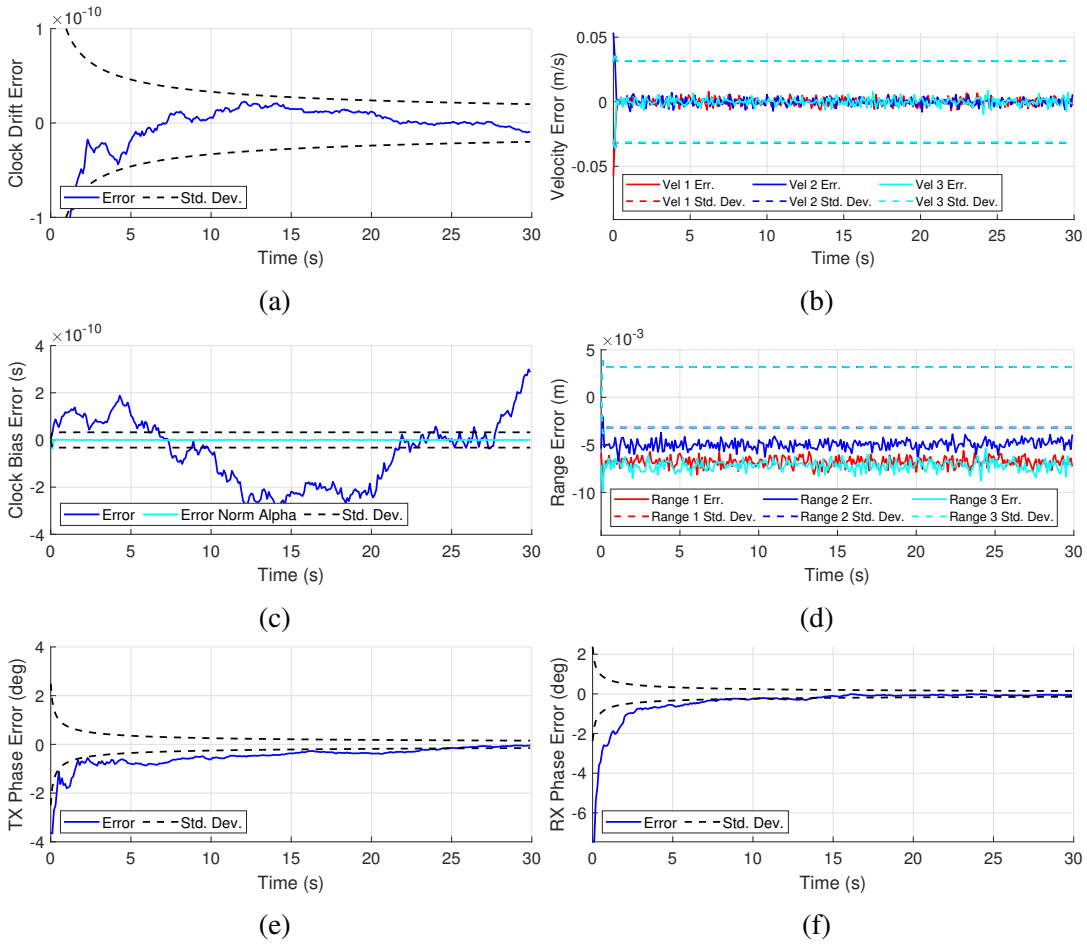


Fig. 5.14: Simulated UKF synchronization error with the iterative synchronization approach. These results are computed for a single platform in a stationary network of four platforms with relative velocities being estimated by the UKF. The above plots show the error for a single platform of clock drift (a), relative velocity with the other three platforms (b), clock bias (c), range with the other three platforms (d), TX carrier phase (e), and RX carrier phase (f). The solid lines show the error, while the dashed lines indicate the expected standard deviation of the error bound predicted by the UKF.

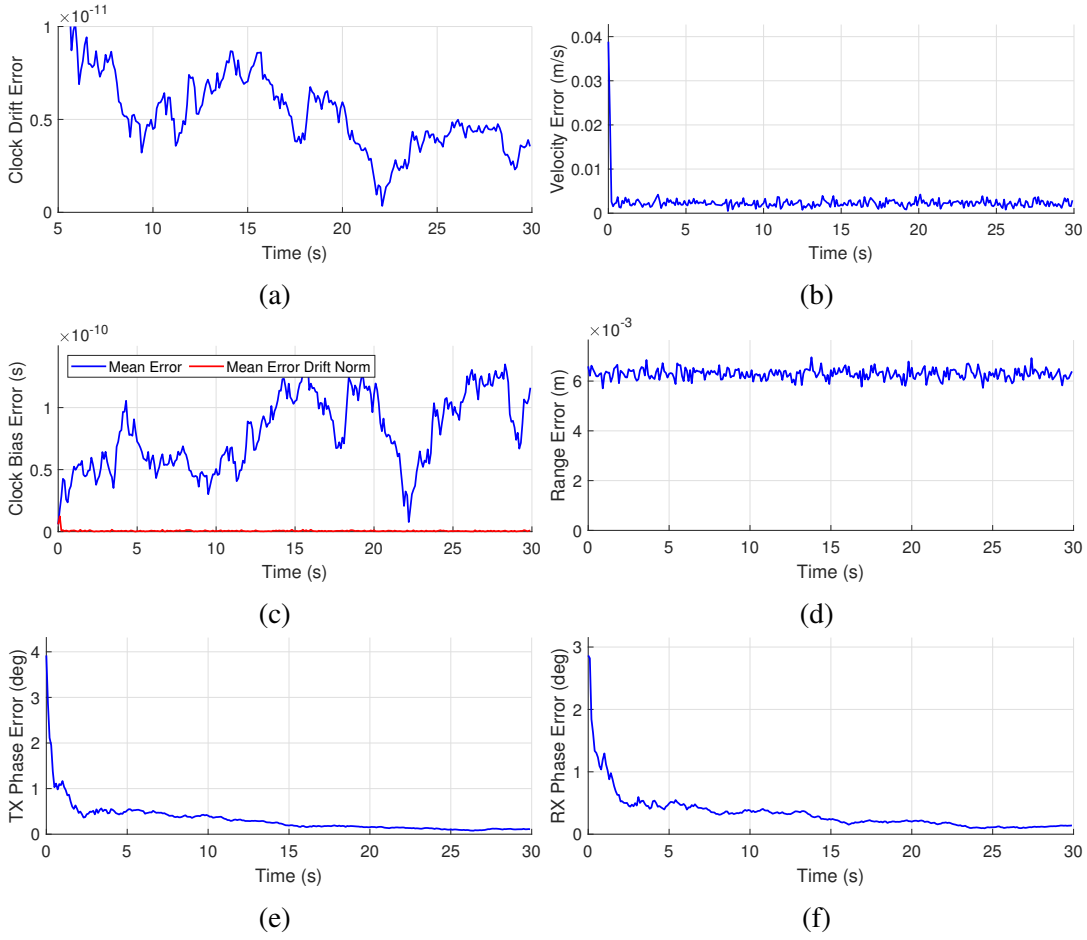


Fig. 5.15: Simulated UKF synchronization error with the iterative synchronization approach. These results are computed as the mean error for a stationary network of four platforms with relative velocities being estimated by the UKF. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

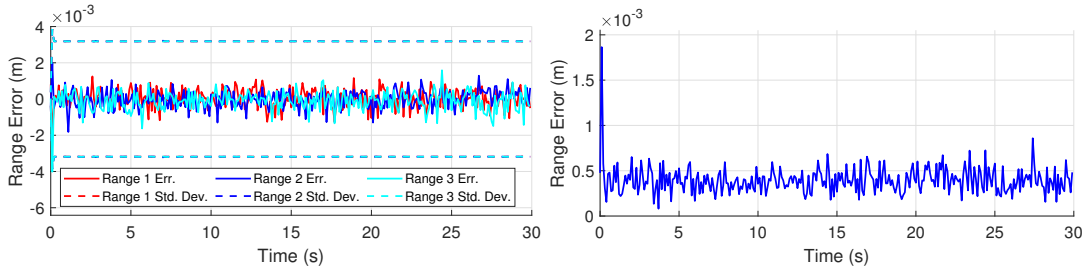
frequency results, the range estimates are biased due to the false assumption of $\alpha_1 = 1$. Fig. 5.16 shows the range error results when α_1 is intentionally simulated to equal 1 such that the range bias disappears. In this case, it can be seen that the range error approaches zero after only a few iterations but retains some noise due to imperfect relative velocity information.

Fig. 5.17 shows the error of the simulated state estimates along with the associated measurements of the corresponding variables for a single platform. Due to the lack of filtering capability for clock bias, the clock bias plot is omitted. Furthermore, for readability, only a single range and relative velocity measurement set is used rather than all three. Because the SNR does not change over time, the noise on the error function for each variable maintains more or less the same magnitude. However, due to the UKF filtering the noise over time, the estimate errors converge to zero (with the exception of range due to the range bias) despite the continued presence of noise on the measurements.

5.4.3.3 Impact of Low SNR

In this example, the SNR is lowered from 30 dB to 10 dB to demonstrate the resilience of the iterative synchronization procedure to low SNR. In contrast to the time, phase, and frequency approach, the iterative synchronization approach continues to produce error that converges to zero over time even with lower values of SNR.

Consider Fig. 5.18, which shows the mean error results over the simulation time for the network with 10 dB of SNR, and Fig. 5.19, which shows the state estimates and measurement values of these variables for a single platform. Although the error admittedly converges much slower in this case when compared to the 30 dB SNR case above, the general trend of the error to reduce significantly over time is still



(a) (b)

Fig. 5.16: The range error when the true value of α_1 = such that there is no range scaling – the single platform (three range values and standard deviations) (a) and the mean over the full network of four platforms (b).

present. It should be pointed out as well that based on the high error after only a single iteration as seen at $t = 0$ of the error plots, the initial synchronization would not produce estimates with high enough quality to perform distributed radar tasks, while the filtered synchronization does over time. Furthermore, as seen throughout the measurements in Fig. 5.19, the measurements produced by single iterations of the synchronization procedure do not improve over time since the SNR does not go up, while the filtered output of the UKF continues to converge to zero while using those same measurements.

5.4.3.4 Impact of Iteration Frequency

In this example, the iteration frequency is lowered from 10 Hz to 0.2 Hz to demonstrate the resilience of the iterative synchronization procedure to lower UKF update rates. In contrast to the time, phase, and frequency approach, the iterative synchronization approach continues to produce error that converges to zero over time even with a much lower iteration frequency. This ability owes to the canceling of the time and phase errors due to the transmit time $T^{\text{tx}}_{j,k}$. To compensate for the much lower update frequency, the simulation time is increased from 30 s to 300 s.

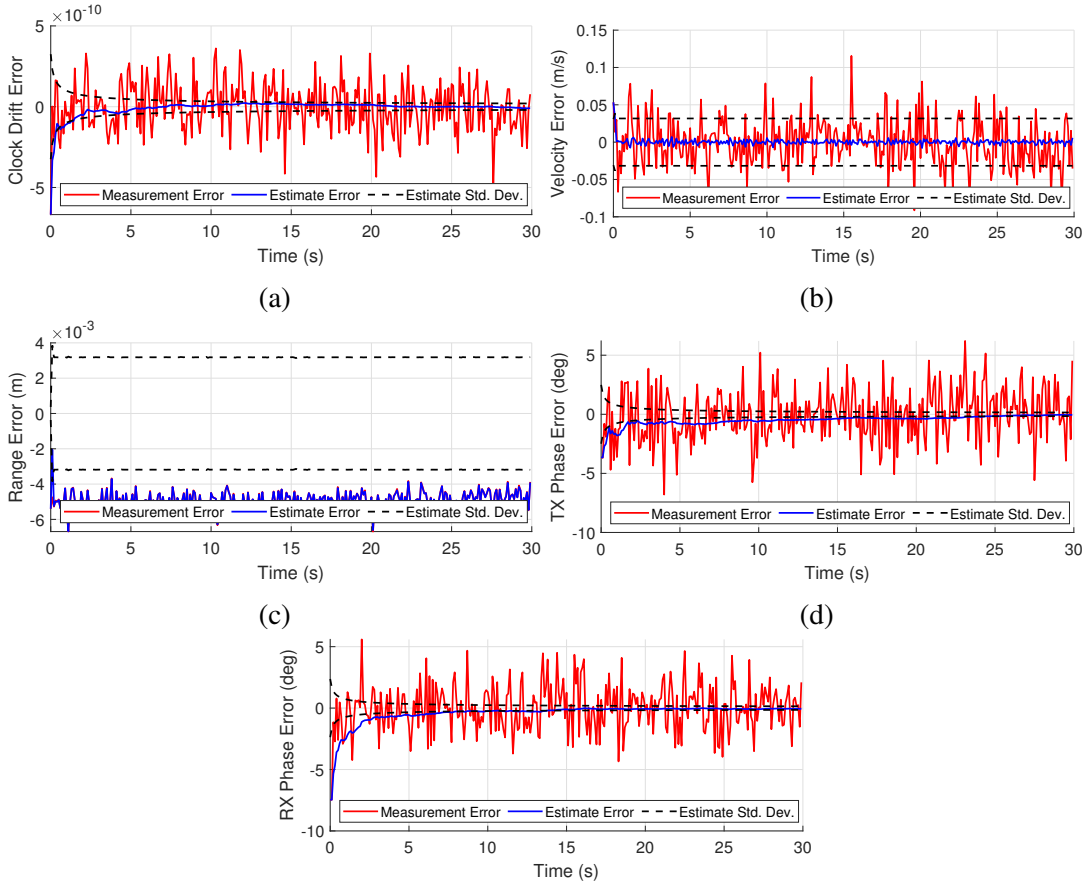


Fig. 5.17: Simulated UKF synchronization error of the system state and measurements with the iterative synchronization approach. These results are computed for a single platform in a stationary network of four platforms with relative velocities being estimated by the UKF. The above plots show the error for a single platform of clock drift (a), relative velocity with the other three platforms (b), range with the other three platforms (c), TX carrier phase (d), and RX carrier phase (e). The solid lines show the error, while the dashed lines indicate the expected standard deviation of the error bound predicted by the UKF.

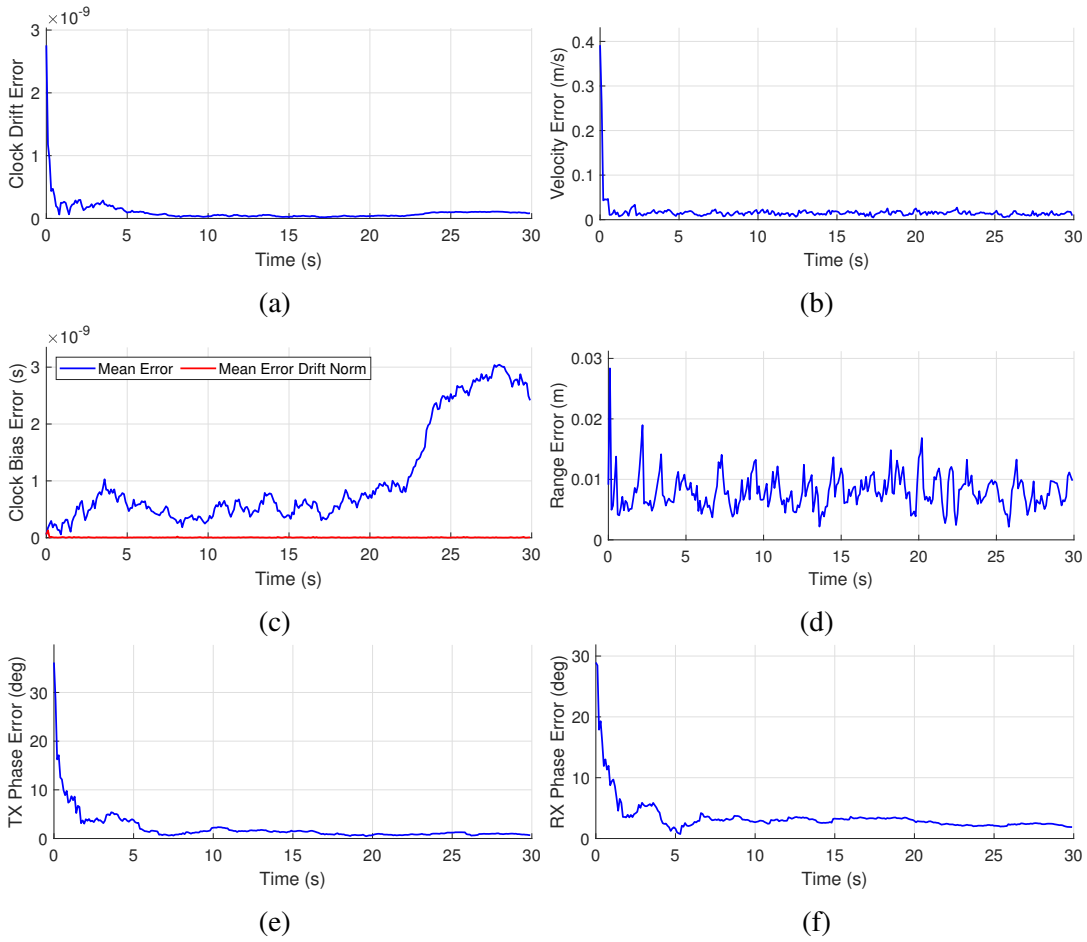


Fig. 5.18: Simulated UKF synchronization error with iterative synchronization approach. These results are computed using an SNR of 10 dB. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

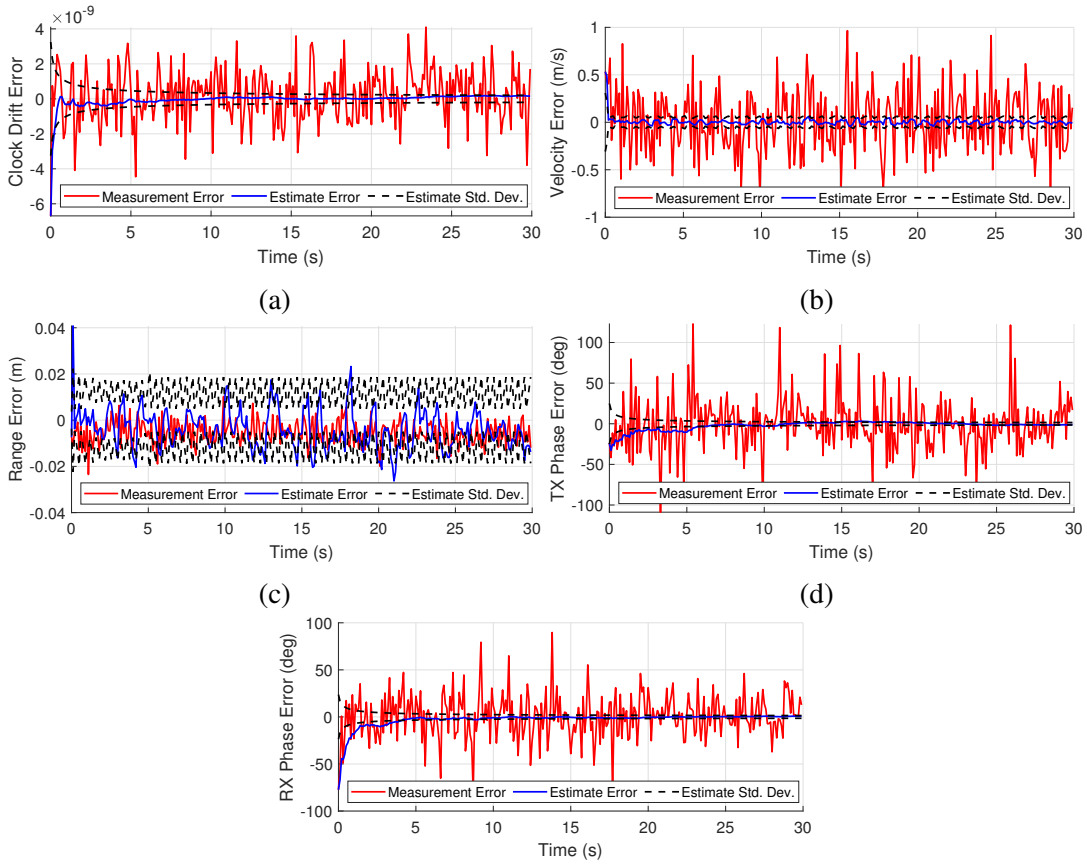


Fig. 5.19: Simulated UKF synchronization error of the system state and measurements with the iterative synchronization approach. These results are computed using an SNR of 10 dB. The above plots show the error for a single platform of clock drift (a), relative velocity with the other three platforms (b), range with the other three platforms (c), TX carrier phase (d), and RX carrier phase (e). The solid lines show the error, while the dashed lines indicate the expected standard deviation of the error bound predicted by the UKF.

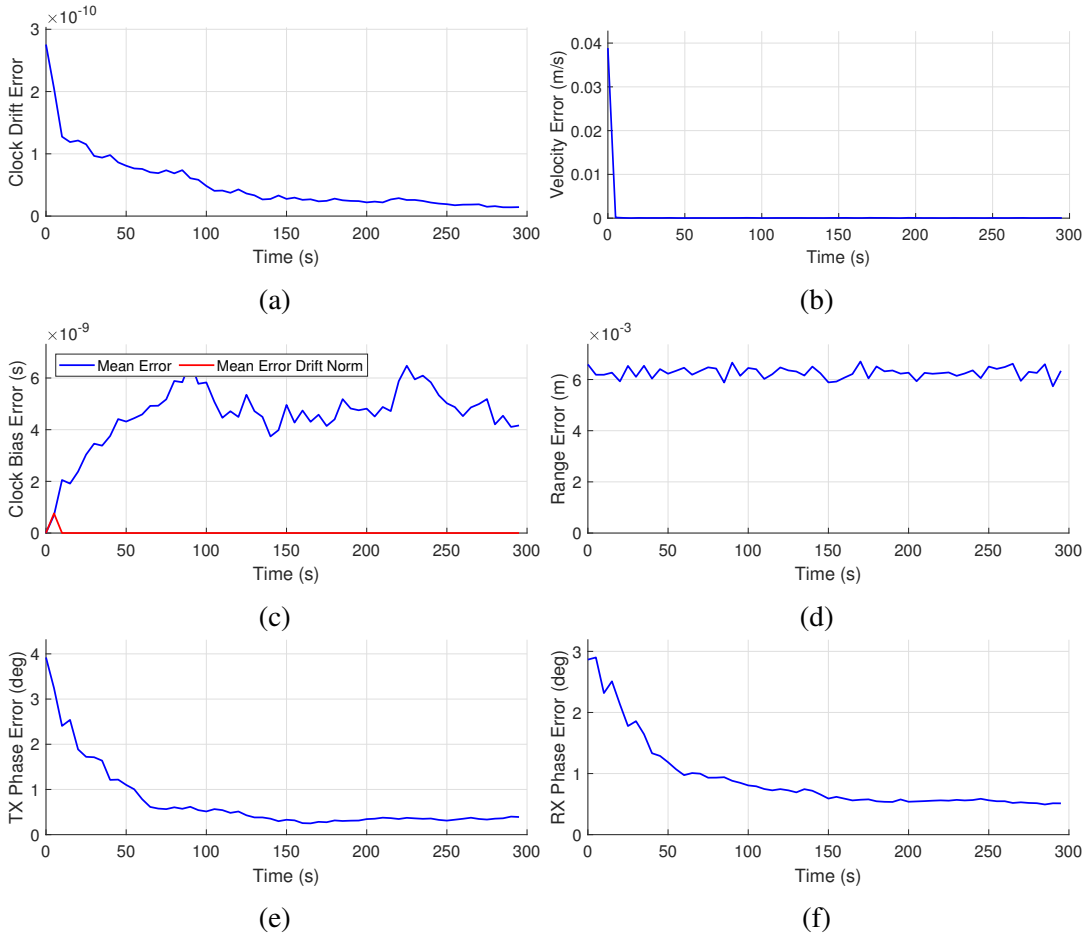


Fig. 5.20: Simulated UKF synchronization error with iterative synchronization approach. These results are computed using an SNR of 10 dB. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

Fig. 5.20 shows the mean error for the different state variables for the full network with the lowered UKF update frequency. Although the general performance is lower than the 10 Hz update rate in the other simulation scenarios, the trend of the error reduction for all the variables remains the same, with the clock drift error ultimately reducing close to 20 ppt. In contrast to the time, phase, and frequency approach, these results indicate that this approach will still be applicable in situations where low downtime is a system requirement leading to lower update rates.

5.4.3.5 Impact of Platform Motion

In this final study, relative platform motion, including accelerations, is added to the simulation. The results demonstrate that the iterative synchronization procedure will lead to a reduction in error over time even when non-constant relative velocities are included.

Fig. 5.21 shows the mean error for the system state of the full four-platform network with motion included between platforms. The clock drift and carrier phase errors follow the same trend as before, generally converging to zero relatively quickly. There are some velocity estimation errors related to the unanticipated accelerations since the velocity estimation is not a high enough quality at each iteration while the UKF state prediction step is not capable of predicting the change in velocity. These velocity estimation errors lead to small errors in clock bias (even the clock drift normalized clock bias) and range but do not have a noticeable negative impact on the carrier phase and clock drift.

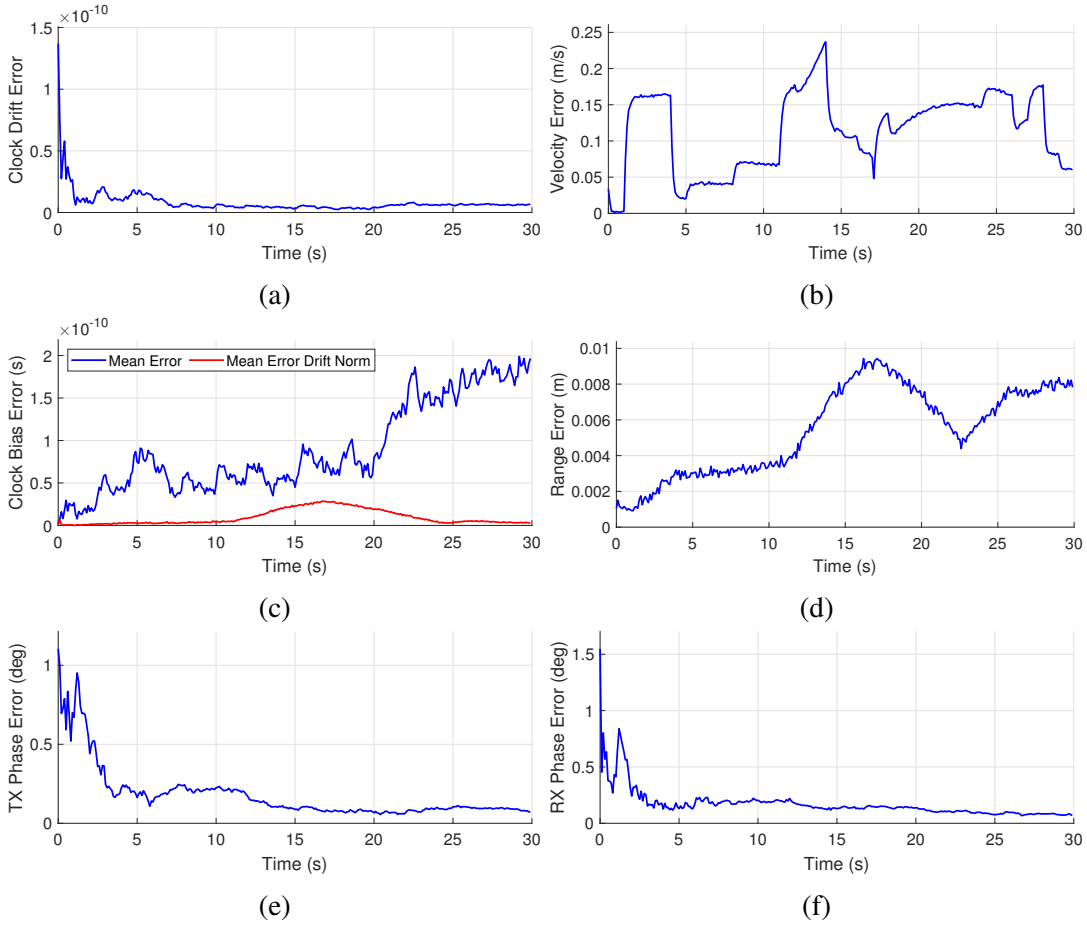


Fig. 5.21: Simulated UKF synchronization error with iterative synchronization approach. These results are computed with relative motion and accelerations between the platforms. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

5.4.4 Discussion of Advantages and Limitations

Contrary to the time, phase, and frequency approach, the iterative synchronization approach is a robust and simple implementation of the UKF for tracking clock parameters over time, and it would prove highly useful in a wide variety of use cases. The approach operates with high-quality results even for low values of SNR, making it applicable for situations with longer-range links between platforms when high receive power cannot be guaranteed. The approach also maintains relatively high performance when the UKF update rate is very low. This would be a major advantage in scenarios where the radar systems in the network cannot tolerate downtime in the radar operation, such as in electronic warfare (EW) contexts. Another important advantage is that the iterative synchronization UKF approach can tolerate relative motion between platforms without a significant loss of performance. This key advantage will ultimately make it the choice for simultaneous navigation and synchronization described in Chapter 6.

The only real disadvantage of this approach when compared to the time, phase, and frequency approach is that the clock drift estimation continues to rely exclusively on frequency-domain estimation, which lacks the information resolution of the time-domain resolution. This leads to higher error and slower convergence, although the error achievable by this approach is still serviceable for a practical network of radar systems. Future work will benefit from finding a way to incorporate time-domain estimation for clock drift estimation once coarse time synchronization is achieved in the initial synchronization iteration.

The theory and simulations provided in this chapter provide a framework for adapting the proposed synchronization techniques into a Kalman filter for iterative tracking

of the synchronization states over time. In the next chapter, this adaptation will be extended to enable simultaneous navigation and synchronization using this technique integrated with the work done in Chapter 3.

Chapter 6

Simultaneous Navigation and Synchronization

The ultimate goal of developing the cooperative navigation and synchronization approaches described in the rest of this dissertation is to enable simultaneous navigation and synchronization for mobile distributed systems. This chapter presents preliminary results in the combining of cooperative navigation described in Chapter 3 with the UKF synchronization described in Chapter 5 into a single cohesive state estimator to this end. Because of the current limitations of the time, phase, and frequency approach to filtered synchronization, the iterative synchronization approach described in Section 5.4 is chosen to be integrated into this full framework. In the following sections, the full modifications to the state models of the individual UKF components are described as well as the changes made to the measurement model to accommodate the new state model. Results are then given to demonstrate the overall accuracy of the simultaneous navigation and synchronization approach for several different cases.

6.1 Full State Model

As with previous descriptions of a network, it is assumed that there is a distributed radar network comprising N_p radar nodes. The total system state vector on iteration k,

$\mathbf{x}_k^{\text{ns}}$, is constructed by

$$\mathbf{x}_k^{\text{ns}} = \begin{bmatrix} \mathbf{x}_{1,k}^{\text{ns}} \\ \vdots \\ \mathbf{x}_{N_p,k}^{\text{ns}} \end{bmatrix}, \tag{6.1}$$

where $\mathbf{x}_{i,k}^{\text{ns}}$ is the single-platform state-vector for platform i on iteration k constructed by

$$\mathbf{x}_{i,k}^{\text{ns}} = \begin{bmatrix} \bar{p}_{i,k} \\ \bar{v}_{i,k} \\ \bar{\theta}_{i,k} \\ \alpha_i^k \\ \phi_i^k \\ \gamma_{i,k}^{\text{tx}} \\ \gamma_{i,k}^{\text{rx}} \end{bmatrix}. \tag{6.2}$$

It is assumed that there is no coupling between the state transition functions of the navigation variables and synchronization variables. Therefore, the state transition function f_k for this application may be described by the navigation state update in (2.48) and the synchronization state update described in (5.7). Similarly, the process covariance $\mathbf{Q}_k^{\text{ns}}$ can be constructed from the associated navigation and synchronization process covariance matrices, structured as

$$\mathbf{Q}_k^{\text{ns}} = \begin{bmatrix} \mathbf{Q}_{1,k}^{\text{ns}} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \cdots & \mathbf{Q}_{N_p,k}^{\text{ns}} \end{bmatrix}. \tag{6.3}$$

The individual constituent matrices of $\mathbf{Q}_k^{\text{ns}}$ will be structured by

$$\mathbf{Q}_{i,k}^{\text{ns}} = \begin{bmatrix} \mathbf{Q}_{i,k}^{\text{nav}} & \cdots & \cdots & \cdots & \mathbf{0} \\ \vdots & \left[\mathbf{Q}_k^{\alpha}\right]_{i,i} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \left[\mathbf{Q}_k^{\phi}\right]_{i,i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \left[\mathbf{Q}_k^{\gamma}\right]_{2i-1,2i-1} & \vdots \\ \mathbf{0} & \cdots & \cdots & \cdots & \left[\mathbf{Q}_k^{\gamma}\right]_{2i,2i} \end{bmatrix}, \tag{6.4}$$

where the notation $[\mathbf{Q}]_{i,j}$ indicates the entry in the i^{th} row and j^{th} column of an arbitrary matrix $\mathbf{Q}$.

6.2 Full Measurement Model

The measurement vector for the simultaneous navigation and synchronization will include GPS measurements (if available) and will include the measurement vector for the chosen synchronization approach since the synchronization procedure generates the range and relative velocity values to be ultimately used in the cooperative navigation. As discussed in Chapter 5, the time, phase, and frequency approach experienced unacceptably high levels of error for any reasonable and practical scenario. As a result, the iterative synchronization approach is the best choice. Therefore, the measurement vector on iteration k, $\mathbf{z}_k^{\text{ns}}$, will be given by

$$\mathbf{z}_k^{\text{ns}} = \begin{bmatrix} \mathbf{z}_k^{\text{GPS}} \\ \\ \mathbf{z}_k^{\text{IS}} \end{bmatrix}. \tag{6.5}$$

As with the cooperative navigation, the values of $\mathbf{z}_k^{\text{GPS}}$ may be assumed to align one-to-one with the corresponding position variables for each platform, and as such this component of the measurement function, h_k, may be assumed to be the identity.

Furthermore, as with the synchronization UKF, the measurements corresponding to the clock parameters (α_i, ϕ_i, γ_i^{tx}, and γ_i^{rx}) will line up exactly with the state variables, and as such the portion of h_k concerning these variables will also be the identity. The range measurements, $\hat{R}_{i,j}^{z,k}$, and the relative velocity measurements, $\hat{v}_{i,j}^{z,k}$, will no longer directly align with state variables and must instead be related to the cartesian position and velocity variables. These may be computed in an identical manner to (3.5) and (3.19) for range and relative velocity, respectively.

6.3 Results

In this section, some preliminary results demonstrating the efficacy of the simultaneous navigation and synchronization approach are provided. A set of four platforms with randomly generated clock error parameters and positions are simulated. As with the synchronization filtering results in Chapter 5, the clock parameters are assumed to remain constant throughout the simulation time.

As with the filtered synchronization, the state variables, covariance values, and measurements must be normalized to limit the impact of numerical errors in the computation of the unscented transform due to the large order-of-magnitude difference between several different variables. The following normalizations of the state variables and measurements are used to ensure numerical stability of the UKF. The clock drift state values and measurements are normalized by a factor of 10, while the clock bias state values and measurements are normalized by a factor of 1000. The range and velocity measurements are normalized by a factor of $1/c$, as are the position and ve-

locity state variables. When applicable, the GPS measurements are scaled by a factor of $1/c$. The Euler angle state variables are normalized by a factor of 10^{-3}. The carrier phase state values and measurements are normalized by a factor of 10^{-3}.

The following parameters are used for the simulations, which are identical to the parameters used in the UKF synchronization approach shown in Chapter 5. Unless otherwise stated, the SNR is set to 30 dB. The carrier frequency f^c is set to 3 GHz. The pulse length is set to 100 μs, while the sampling frequency is set to 200 MHz. The capture length in the initial synchronization is set to 400 μs, while it is set to 600 μs in the subsequent iterations of the UKF to account for the longer signal. The signal bandwidth in the LFM waveforms is set to 50 MHz. The TDMA scheme is used with the time slot set to 1 ms. The clock drift errors are drawn from a uniform distribution limited to within 25 ppm of 1. The clock bias errors are drawn from a Gaussian distribution with a standard deviation of 100 ns. To produce results comparable to the cooperative navigation results in Chapter 3, the full simulations are conducted over 300 s. To shorten the simulation times, the navigation simulation is run at 10 Hz rather than 100 Hz, and the synchronization is produced every 10 iterations (1 Hz) unless otherwise stated. The motion paths of the four platforms are shown in Fig. 6.1. Unless otherwise stated, the IMU accuracy on each platform is the low-accuracy IMU described in Chapter 3.

6.3.1 Initial Study

For the default parameters outlined above, the simulation results are provided below. Fig. 6.2 shows the synchronization variable errors for a single platform in the network as an example, along with standard deviation bounds on the state estimates for the variables for which direct state estimates are produced (recall that range and

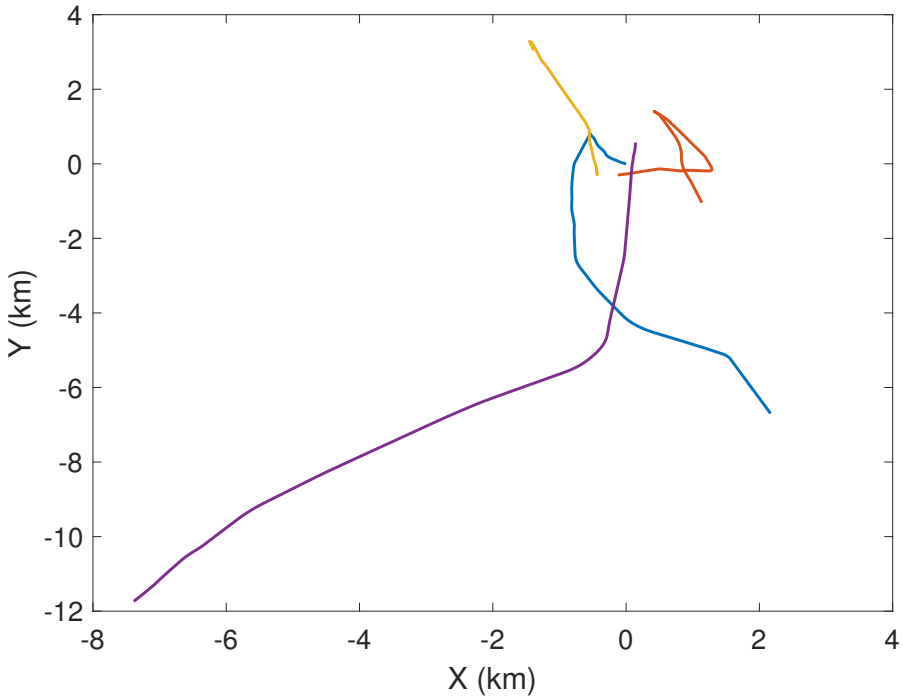


Fig. 6.1: The motion paths for the four platforms used to evaluate the simultaneous navigation and synchronization.

relative velocity are not included in the state vector directly in this approach). For the case of range and relative velocities, the estimates are computed from the actual estimates of position and velocity in the ECEF Cartesian coordinate system. Fig. 6.3 shows these same variables but plots the average error over the entire network.

The results shown in Fig. 6.4 and Fig. 6.1 are highly similar in quality to those of the iterative synchronization demonstrated for a similar case in Fig. 5.21. Importantly, the estimates of the clock drift and carrier phases tend to converge such that the error approaches zero. There are some notable distinctions, some of which are due to differences in the techniques and some of which are artifacts of the different simulation parameters. First, note how the clock bias estimate with drift compensation has higher error than in the iterative synchronization procedure. This results from the modification to the process covariance not being sufficient to allow the clock bias estimates to

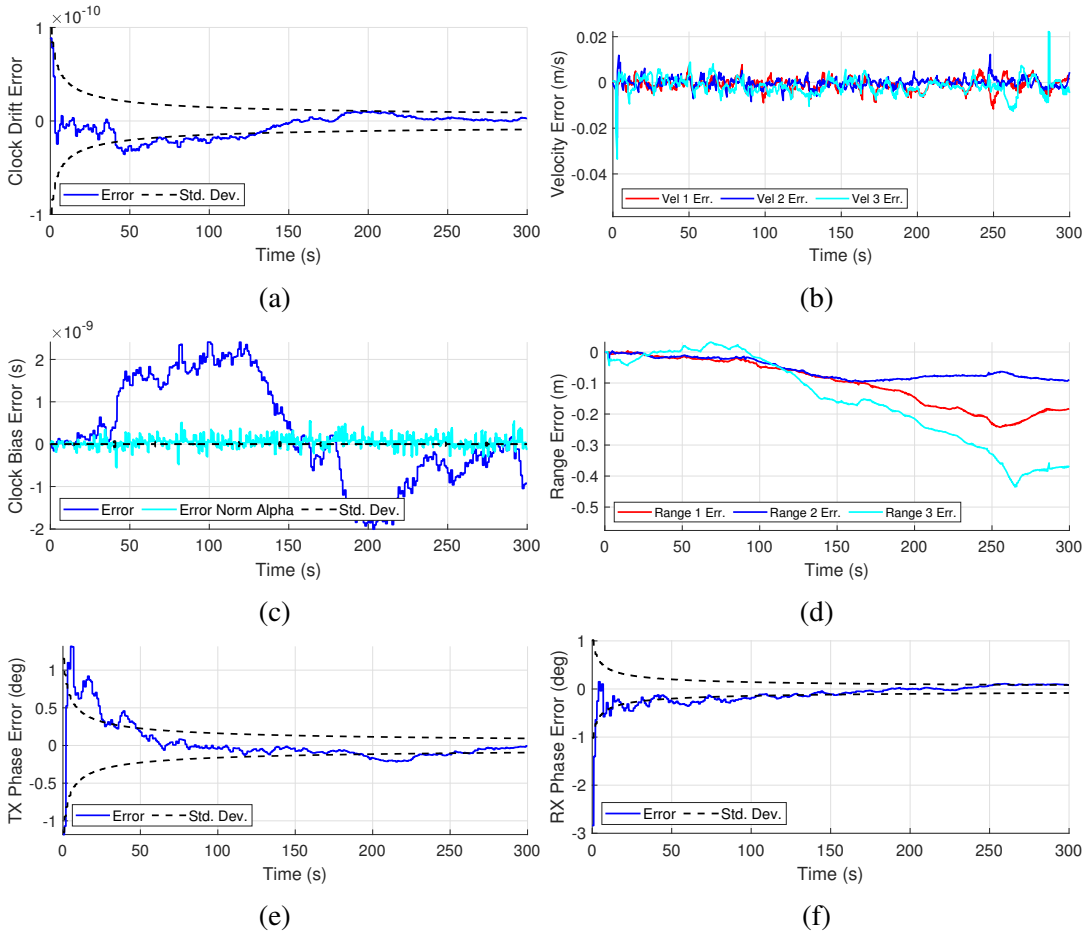


Fig. 6.2: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed for a single platform in a network of four platforms. The above plots show the error for a single platform of clock drift (a), relative velocity with the other three platforms (b), clock bias (c), range with the other three platforms (d), TX carrier phase (e), and RX carrier phase (f). The solid lines show the error, while the dashed lines indicate the expected standard deviation of error bound predicted by the UKF.

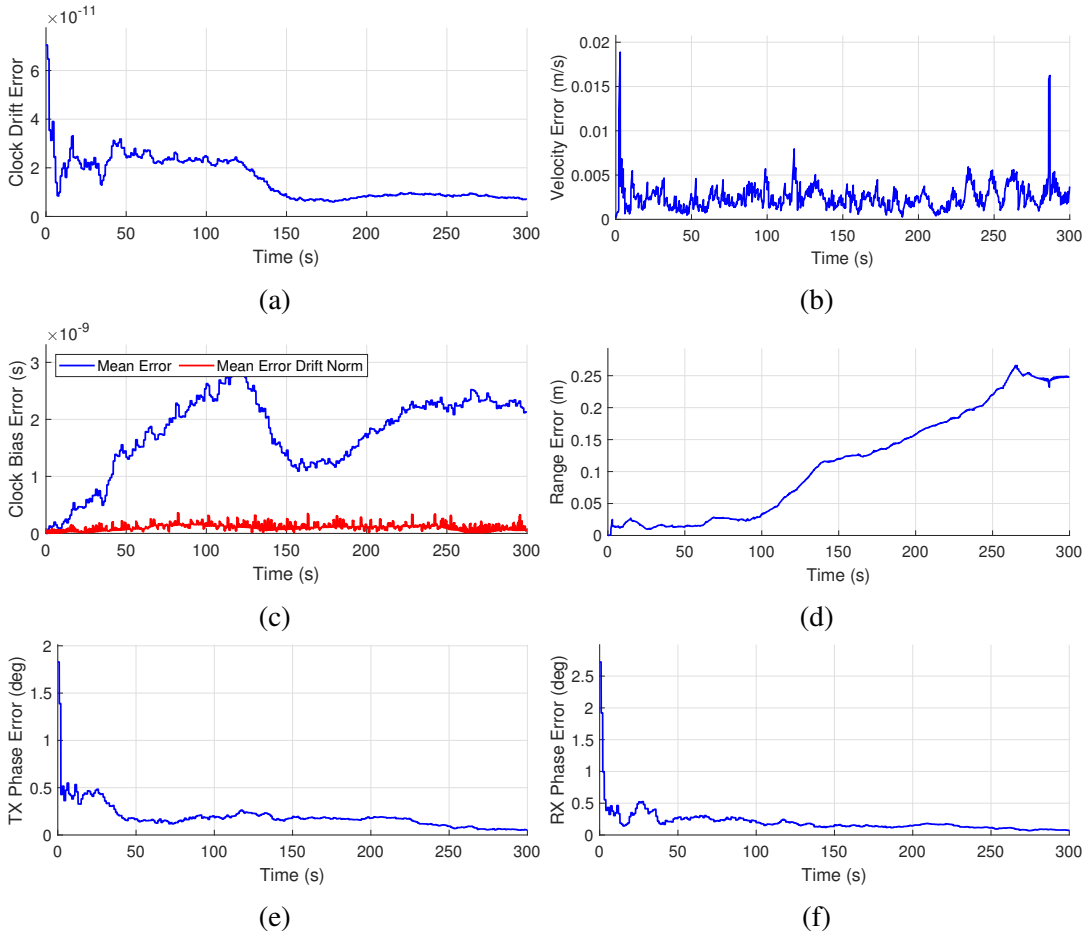


Fig. 6.3: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed as the mean error for a network of four platforms. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

match the measurements properly, which is likely caused by the underlying update rate of the UKF being higher than the measurement availability rate, which is not the case in the iterative synchronization UKF. To fix this, the measurements of clock bias are directly substituted for the state variables, leading to some additional error. However, because the additional compensated error is still within a fraction of a nanosecond, this is considered sufficient.

Another notable distinction is the increased range error. This is an artifact of the simulation, since this simulation is run over 300 s rather than 30 s, and the platforms move much further away from one another as the simulation carries on. The resulting increased distances lead to much higher biasing of the true range measurements due to the clock drift error arising from the incorrect assumption that $\alpha_1 = 1$.

In addition to the synchronization variables, the network mean error for navigation is shown in Fig. 6.4, with comparisons between the dead-reckoning navigation and cooperative navigation. The reduction in error is similar in magnitude to the four-platform cooperative navigation case in Fig. 3.2, though the overall error for both cases is increased due to the reduced simulation update rate. Generally, it would be expected that the cooperative navigation quality would be increased rather than decreased due to the expected range variance at 30 dB being much lower than the 0.1 m standard deviation assumed in the simulations in Chapter 3. However, as demonstrated in Fig. 4.24, the true range measurement accuracy tapers off after a fairly low SNR due to error introduced by the ambiguity function and bias introduced by the clock drift error. As a result, the higher accuracy in theory does not translate to practice. Future work may need to consider practical ways to estimate the true range variance given these additional practical effects on top of SNR.

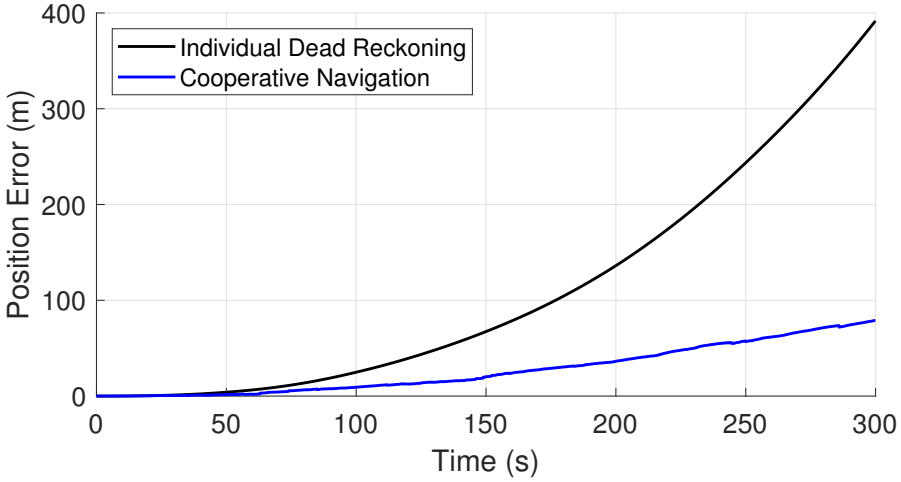


Fig. 6.4: Average platform navigation error for 4 platforms using simultaneous navigation and synchronization, with individual dead-reckoning (black) and using cooperative navigation (blue).

6.3.2 Results with Lower SNR

In this example, the SNR is lowered from 30 dB to 10 dB to demonstrate the synchronization and cooperative navigation performance in lower SNR scenarios. Consider Fig. 6.5, which shows the mean error for the synchronization variables in this case, and Fig. 6.6, which shows the cooperative navigation and dead-reckoning navigation errors. The synchronization variable errors are increased in a similar manner to those in Fig. 5.18, though there are some distinctions due to the simulation differences described above. The range error is very similar to the 30 dB SNR case, since the error is dominated by the increased range between platforms and those errors rather than the SNR, are described in Fig. 4.24.

Interestingly, the cooperative navigation error in this case is reduced from the higher SNR case, as seen in Fig. 6.6. This is likely because in the 10 dB SNR case, the estimated range variance described by the (4.116) will more closely match reality, as opposed to the 30 dB case in which the estimated variance will be much lower than

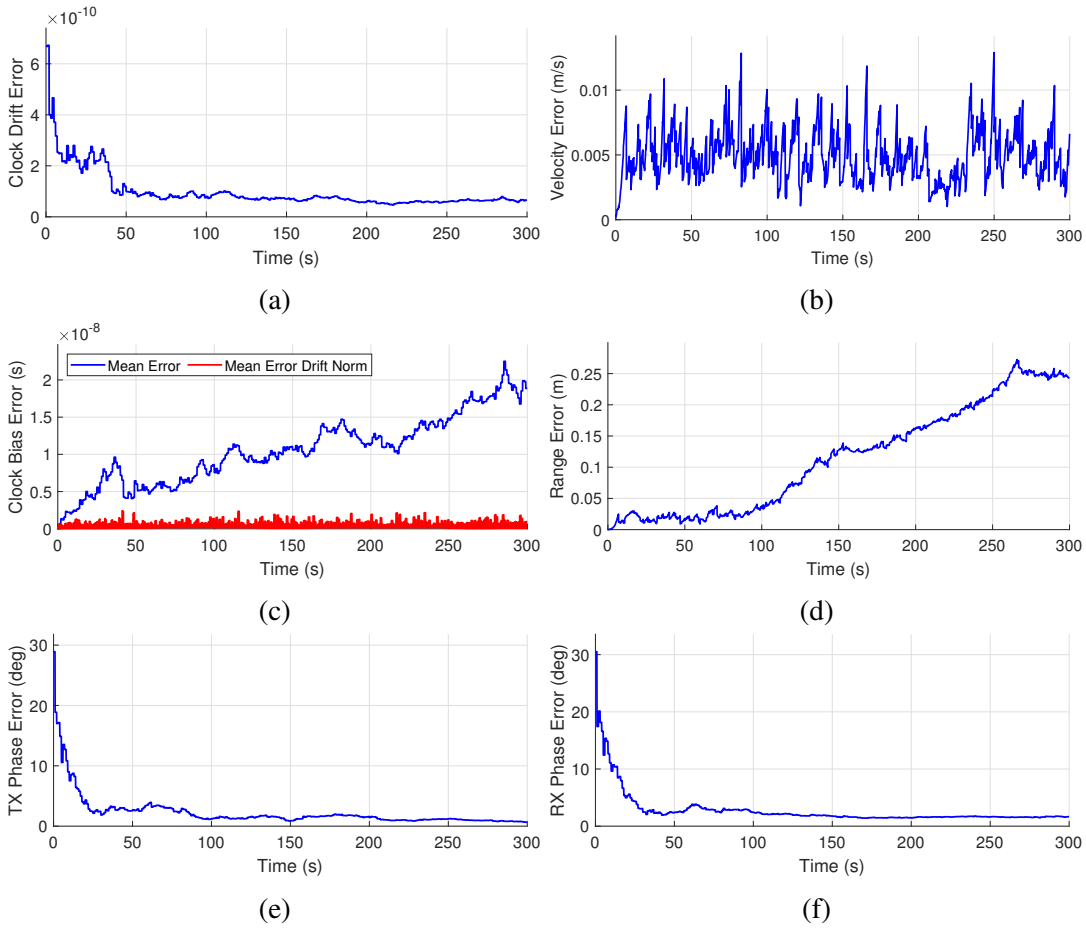


Fig. 6.5: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed as the mean error for a network of four platforms with 10 dB of SNR. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

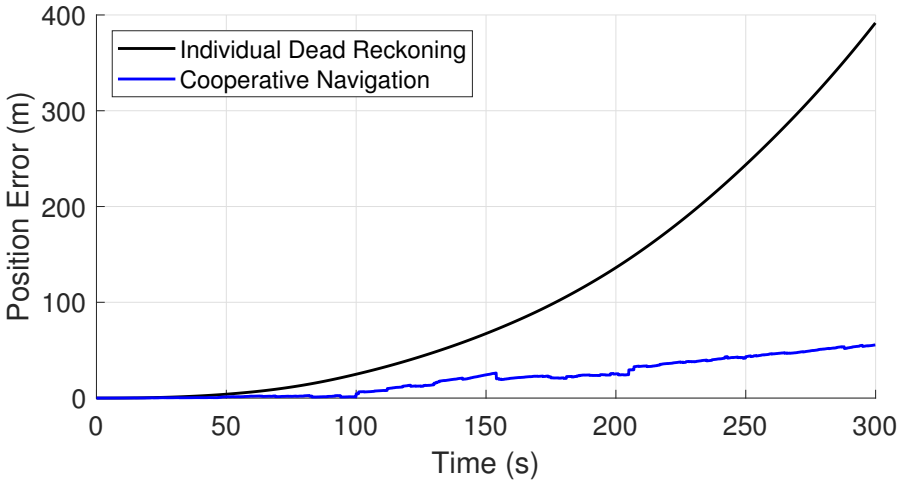


Fig. 6.6: Average platform navigation error for 4 platforms using simultaneous navigation and synchronization with 10 dB of SNR, with individual dead-reckoning (black) and using cooperative navigation (blue).

the error variance in reality. Thus, the measurements are more appropriately weighted in the update step in comparison to the IMU updates, leading to overall increased accuracy. This scenario highlights the importance of properly weighting the covariance of measurements to avoid over-trusting inaccurate measurements.

6.3.3 Results with Realistic Variable SNR

In this next scenario, the SNR values are adjusted based on the expected SNR based on the Friis formula and SNR computation described in Section 2.2.4. For this simulation, it is assumed that each platform transmits with a power of $P_t = 10$ W. Isotropic antennas are assumed such that $G_t = G_r = 0$ dB. For simplicity, the RF front-end gain and noise figure are both assumed to be 0 dB such that the SNR at the antenna is the observed SNR at the ADC. The SNR values are estimated using the technique referenced in Section 4.4.4.

Error results in the synchronization states for a single platform, along with stan-

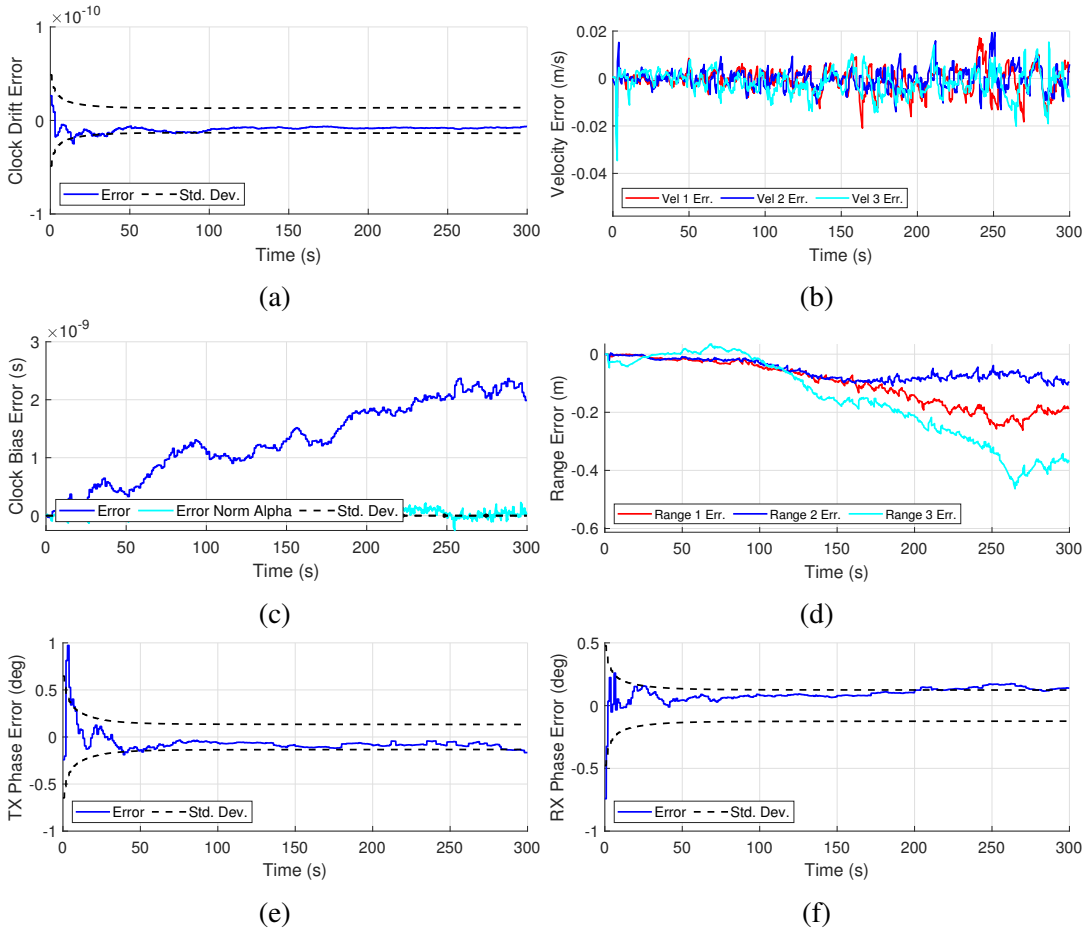


Fig. 6.7: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed for a single platform in a network of four platforms with variable SNR based on the Friis formula. The above plots show the error for a single platform of clock drift (a), relative velocity with the other three platforms (b), clock bias (c), range with the other three platforms (d), TX carrier phase (e), and RX carrier phase (f). The solid lines show the error, while the dashed lines indicate the expected standard deviation of error bound predicted by the UKF.

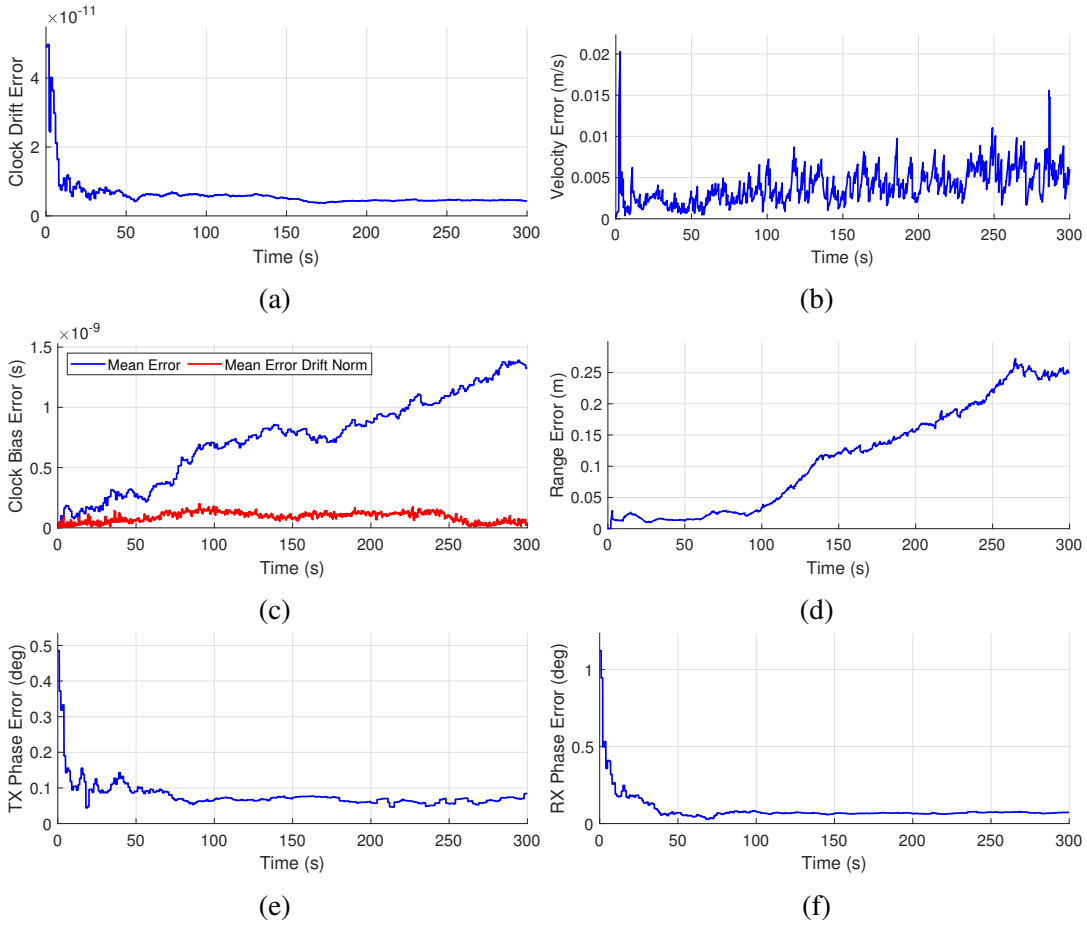


Fig. 6.8: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed as the mean error for a network of four platforms with variable SNR based on the Friis formula. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

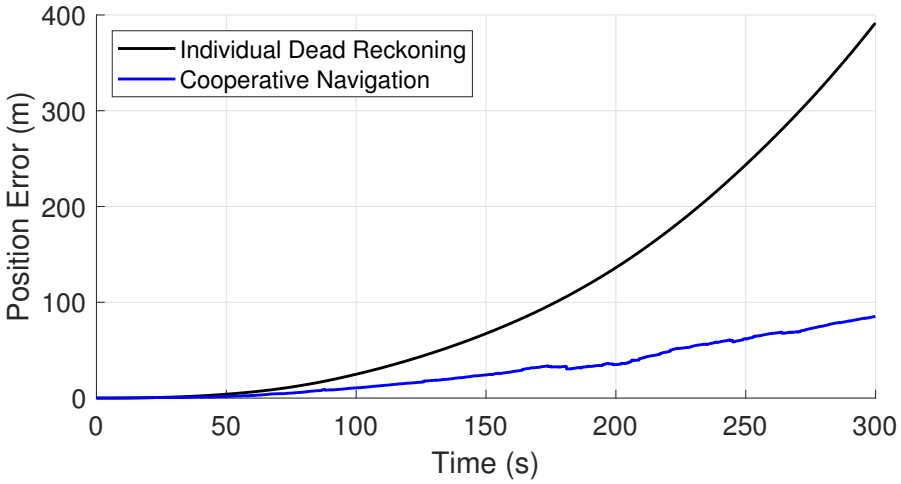


Fig. 6.9: Average platform navigation error for 4 platforms using simultaneous navigation and synchronization with variable SNR based on the Friis formula, with individual dead-reckoning (black) and using cooperative navigation (blue).

dard deviation lines, are shown in Fig. 6.7, while the mean errors for the network in estimating the synchronization variables in this case are shown in Fig. 6.8. Particularly in Fig. 6.7 where the standard deviation bounds are shown, it can be seen that the errors estimates of clock drift and carrier phase generally do not change much after the 75 s point, indicating that the estimates are not updated strongly. This is a logical result since as the simulation continues, the platforms spread out and the range between them increases, leading to a large reduction in SNR. Thus, the measurement covariance is large and the result is that the measurements will be weighted much lower than the original state estimates during the update phase of the UKF.

The navigation error results are shown in Fig. 6.9. Despite the much lower SNR toward the end of the simulation, the cooperative navigation error still is close to the error in the 30 dB case and significantly outperforms the dead-reckoning error.

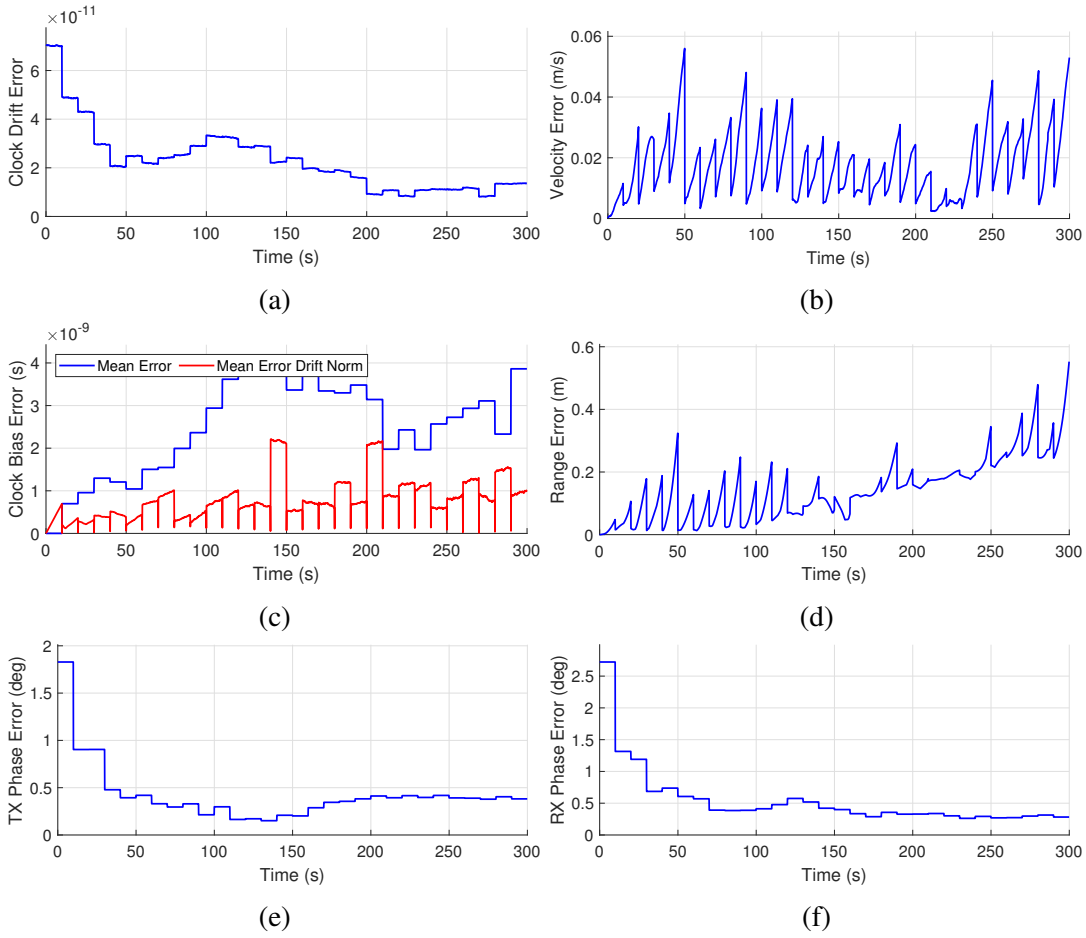


Fig. 6.10: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach with a synchronization update rate of 0.1 Hz. These results are computed as the mean error for a network of four platforms. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

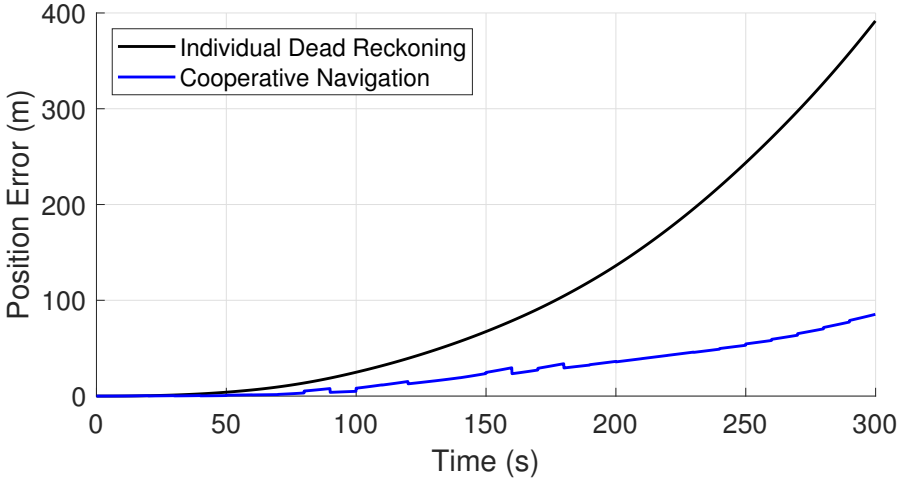


Fig. 6.11: Average platform navigation error for 4 platforms using simultaneous navigation and synchronization with a synchronization update rate of 0.1 Hz, with individual dead-reckoning (black) and using cooperative navigation (blue).

6.3.4 Results with Lower Update Rate

This scenario shows the performance of the synchronization and navigation when the rate at which the synchronization procedure is performed is lowered to 0.1 Hz (once every 10 s). The network mean errors for the synchronization states are shown in Fig. 6.10. The errors in general have the same quality as the 1 Hz update rate in Fig. 6.3, which is unsurprising since it was confirmed in Chapter 5 that the update rate would not significantly impact the synchronization quality (though this may prove untrue if the clock states were not simulated to remain constant). The only visible distinction is that the estimate error displays a "stair step" form with jumps at the update times. The main issue is that the clock bias errors make a significant jump, even going above a single ns – this clock bias error may be significant enough to introduce phase errors in signals. This will be the primary limitation of decreasing the update rate unless a solution to this issue is found in future work.

The navigation results with the lower update rate are shown in Fig. 6.11. Interest-

ingly, the update rate also does not seem to impact the cooperative navigation accuracy severely, indicating that the cooperative navigation quality will remain high even with a decrease in the availability of ranging signals.

6.3.5 Results with GPS

This scenario shows the impact of including GPS measurements in the simultaneous navigation and synchronization procedure. The GPS measurements are included at the same update rate as the synchronization measurements on the same iterations, and the variance of the GPS measurements is identical to the variance of the GPS measurements described in Chapter 3. Fig. 6.12 shows the mean errors for the synchronization states with the GPS enabled. In general, these results show very similar quality synchronization to the results without GPS in Fig. 6.3, with the only noticeable improvement being the clock drift error being reduced. Fig. 6.13 shows the navigation error for this case; unsurprisingly, the cooperative navigation with GPS far outperforms dead-reckoning. Because the point of this simulation is to show the simultaneous navigation and synchronization, which by nature includes the ranging signals, it is not evident how to draw a comparison between GPS with and without the cooperative navigation as in Fig. 3.6.

6.3.6 Results with a High-Accuracy IMU

In this final scenario, the results of the simultaneous navigation and synchronization approach are produced using an IMU with high accuracy. As with the lower-accuracy IMU used in each other scenario, the IMU properties are described in Chapter 3. Fig. 6.14 shows the mean errors for the synchronization states using the higher-accuracy IMU. Similar to the GPS case, there is not a significant increase in syn-

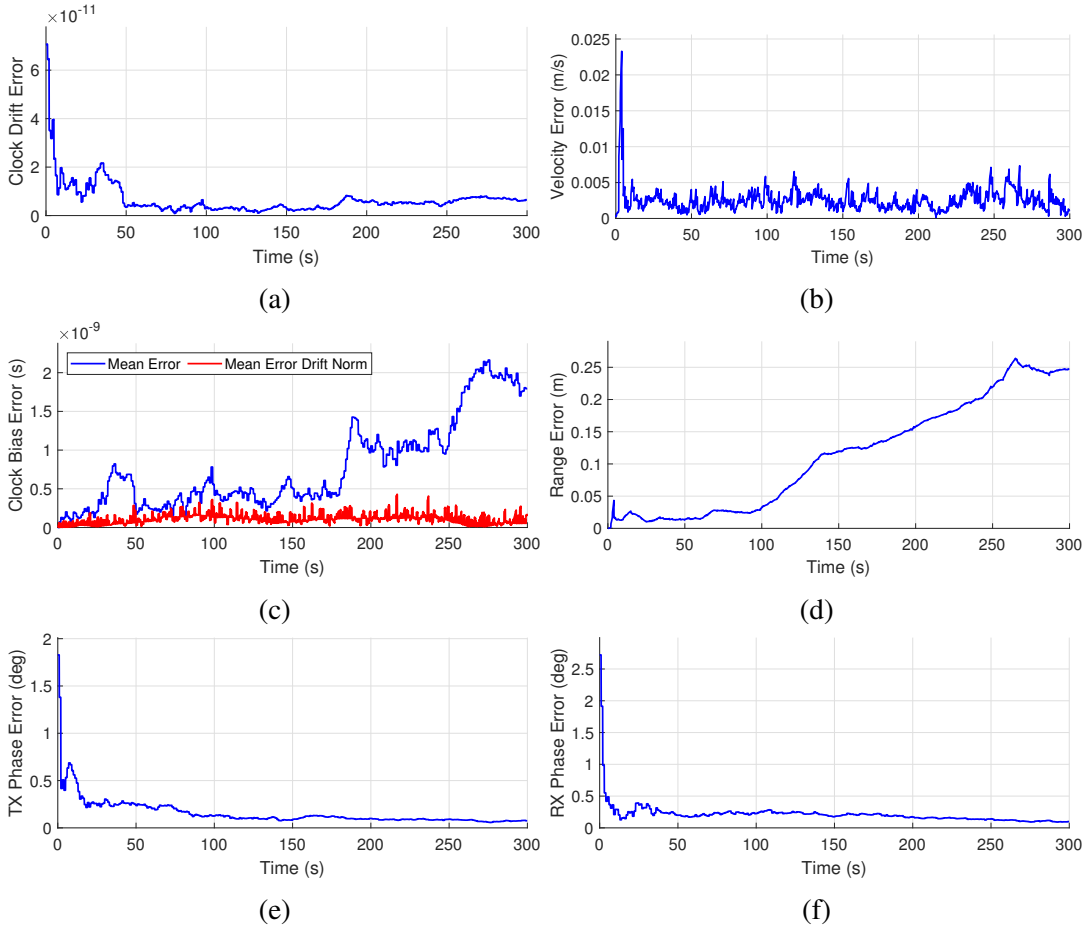


Fig. 6.12: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed as the mean error for a network of four platforms with access to GPS. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

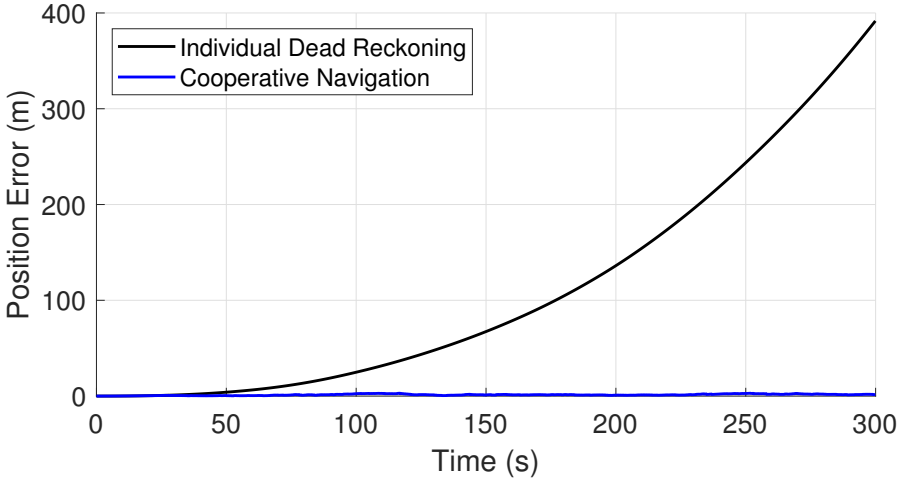


Fig. 6.13: Average platform navigation error for 4 platforms using simultaneous navigation and synchronization with access to GPS, with individual dead-reckoning (black) and using cooperative navigation (blue).

chronization accuracy, although the clock drift estimate error is somewhat reduced compared to Fig. 6.3. One interesting anomaly is the jumps in phase error around 50 and 200 s. Although these jumps are relatively small (only slightly larger than a degree), they do represent a departure from the pattern of general reduction in error over time. More analysis of the phase error will be necessary to determine the source of these errors.

Fig. 6.15 show the navigation error for the dead-reckoning and cooperative case with the high-accuracy IMU. In contrast to the cooperative navigation scenario in Fig. 3.4, the dead-reckoning closely matches the cooperative navigation case to some extent. This is likely due to the range measurement bias due to relative velocity and clock drift error present in the realistic ranging signals, in which case the ranging signals must be ignored to some extent in favor of the highly accurate IMU navigation updates.

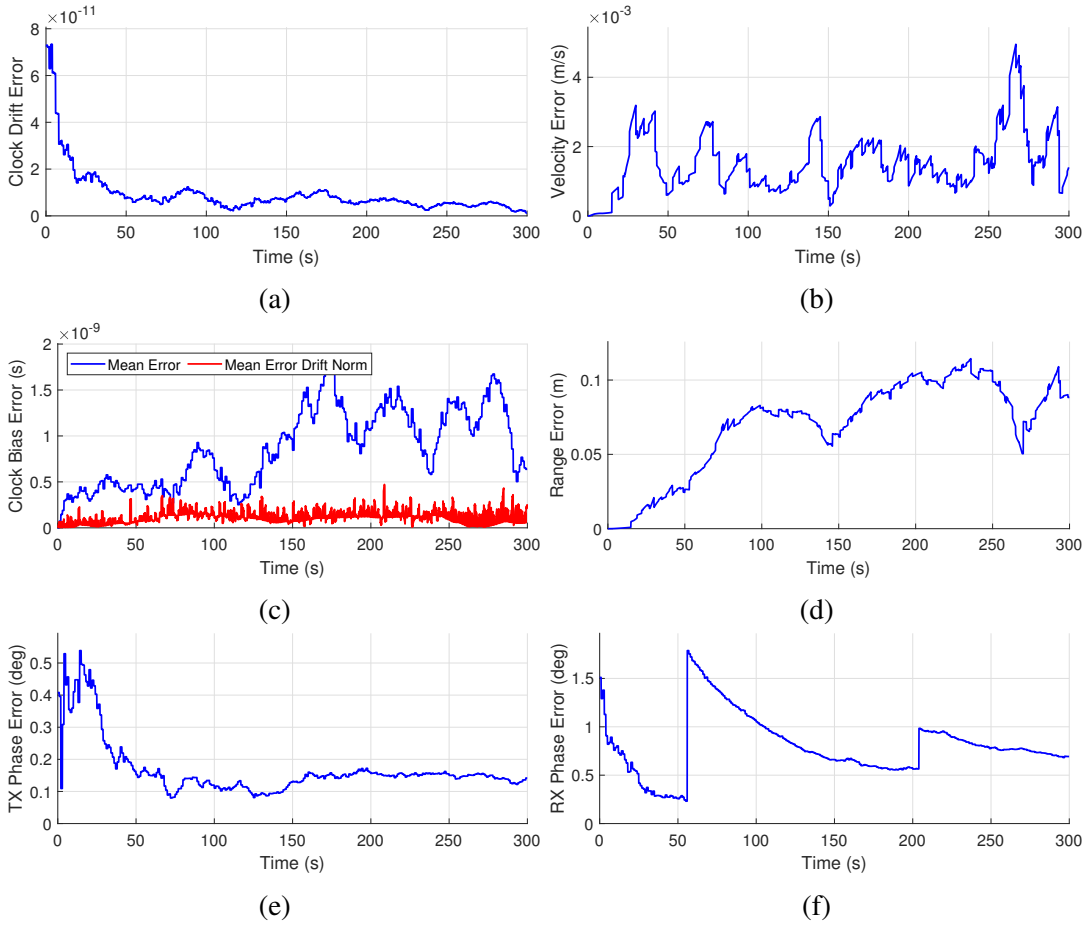


Fig. 6.14: Simulated UKF synchronization error with the simultaneous navigation and synchronization approach. These results are computed as the mean error for a network of four platforms with high-accuracy IMUs. The above plots show the mean error for clock drift (a), relative velocities between platforms (b), clock bias (c), range between platforms (d), TX carrier phase (e), and RX carrier phase (f).

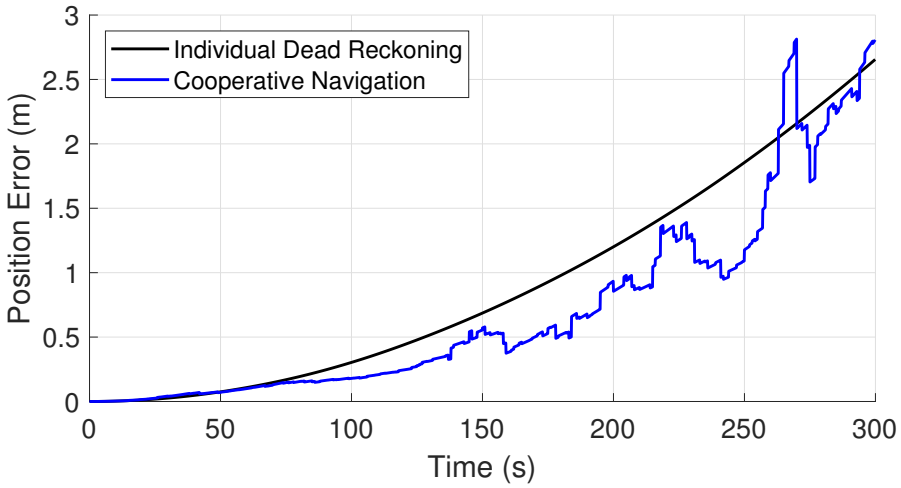


Fig. 6.15: Average platform navigation error for 4 platforms using simultaneous navigation and synchronization with high-accuracy IMUs, with individual dead-reckoning (black) and using cooperative navigation (blue).

6.4 Discussion

This chapter does demonstrate that the simultaneous navigation and synchronization approach works while using the iterative synchronization approach to achieve filtered synchronization and ranging and relative velocity measurements. In terms of cooperative navigation accuracy, similar results to those in Chapter 3 can be achieved using this approach, while similar results in synchronization to those in Chapter 5 can also be achieved using this approach.

There are some important sources of error that reduce the quality of the results presented in this chapter. First, the proposed digital synchronization technique has the drawback of introducing clock drift bias by making the assumption that $\alpha_1 = 1$, which introduces ranging errors. The ranging errors are compounded by minor time-delay shifts due to errors in Doppler shift compensation, although these errors will be present in any technique relying on the exchange of such signals in the presence of unknown Doppler shifts. These ranging errors ultimately degrade the performance of

the navigation and make it difficult to perfectly estimate the ranging variance required to correctly construct the measurement covariance matrix.

The second problem is that the iterative synchronization approach, although stable and functional, discards a significant amount of information. Theoretically, the time, phase, and frequency approach does not abstract away any of the information that can be extracted from signals, such that the impact on the ranging and single-tone frequency signals due to the contribution of every state variable, including navigation states and clock parameters, can be modeled and extracted through the measurement function. As a result, in theory, the dependencies of all system state variables on one another can be quantified correctly through the measurement function, which would ultimately lead to higher accuracy overall, which can be seen in some way in the time, phase, and frequency results showing how the time-delay measurement allows for very accurate estimates of clock drift. When using highly-accurate navigation equipment, such as GPS (or RTK GPS, for instance) or high-accuracy IMUs, these could even be refined in such a way as to compute and compensate for the clock drift biases; that is, the assumption that $\alpha_1 = 1$ could be discarded in favor of pulling out the ranging biases over time and compensating for them.

In contrast to these possibilities, the iterative synchronization approach which is used here discards time-delay information when computing clock drift and discards additional frequency information when computing clock biases, ranges, and phases. Because there is no additional information, the proposed simultaneous navigation approach is not much different from running the iterative synchronization UKF separately and simply pulling the range and relative velocity measurements over to a concurrent cooperative navigation UKF. Therefore, future efforts will focus on correcting the errors in the time, phase, and frequency approach such that the benefits it endows

to synchronization accuracy, and ideally, navigation accuracy, can be fully realized in future iterations of this work.

Chapter 7

Preliminary Hardware Demonstration

This chapter provides some preliminary results demonstrating the hardware implementation of the synchronization algorithms described in this work. To date, the only component of the synchronization that has been fully and reliably implemented on the hardware setup is the carrier frequency synchronization accomplished through the estimation of the clock drifts as described in Chapter 4. Implementation of other parts of the algorithm has already been accomplished in [92] and as such this is not deemed a crucial part of this demonstration. Future work on the implementation will involve implementing all components of the synchronization, including reliable time and phase synchronization, as well as implementation of the time-series UKF estimation described in Chapter 5.

7.1 Hardware Setup

To validate the proposed frequency synchronization technique, two custom-built software-defined radios (SDRs) were employed. The core of each SDR unit is an Analog Devices AD9361 dual-channel transceiver, which comprises all components of the transceiver chain including digital-to-analog converters (DACs), analog-to-digital converters (ADCs), and the RF up-conversion and down-conversion. The transceiver

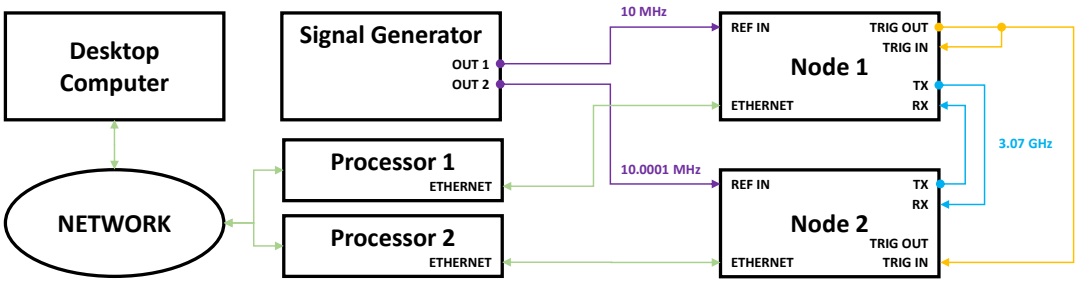


Fig. 7.1: A block diagram of the hardware used in this experiment. From [5] ©2024 IEEE.

is governed by a Zynq 7020 integrated field-programmable gate array (FPGA) and dual-core ARM processor, which configures the transceiver chip, loads the programmable transmit waveform to the transceiver, collects the received digital signal, and interfaces with the external central processing computer. Each SDR communicates through an Ethernet interface with a dedicated processing computer, which collects the received data, loads waveforms and configurations defined by the user, and communicates through a network to the user's computer where all pre- and post-processing of the waveforms and received data is accomplished.

Each SDR is capable of running on an internal 10 MHz oscillator or an external clock source. Each node may also be programmed to output a trigger signal after the system is reset to begin a new session. This trigger signal can be input to all other SDRs in the network to reset the FPGA clock counter. This coarse synchronization step is important for properly implementing the frequency synchronization procedure since the received signals will quickly drift outside the receive window before the FPGA clock counter can be scaled to compensate for clock drift. This coarse synchronization requirement is a limitation of the hardware implementation rather than the proposed synchronization procedure since the SDR data capture length is currently limited to 100 μs due to restrictions on the data communication rate.

The two SDRs have their transmitters and receivers connected via coaxial cables

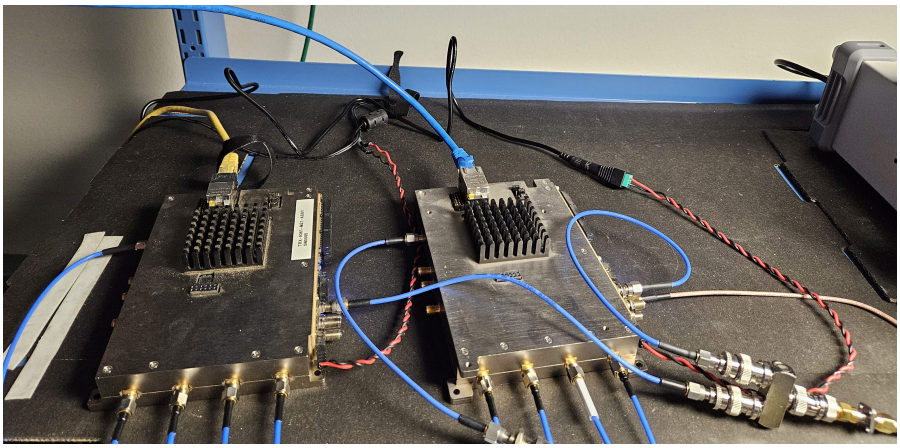


Fig. 7.2: An image of the connected SDR units. From [5] ©2024 IEEE.

in a loopback mode. At the beginning of the experiment, a trigger from SDR 1 is input back into SDR 1 and SDR 2 to reset the FPGA clock counters and ensure that the clocks start aligned. A Siglent SDG 1032X two-channel arbitrary waveform generator is used to generate reference clock inputs to both SDRs. SDR 1 receives a 10 MHz clock input, giving a drift value of $\alpha_1 = 1$, while SDR 2 receives a 10.0001 MHz clock input, giving a drift value of $\alpha_2 = 1.00001$. This emulates a clock error of 10 ppm, which is a reasonable performance to expect from a typical crystal oscillator. The SDRs transmit with a nominal carrier frequency of $f^c = 3.07$ GHz. A block diagram of the hardware configuration is shown in Fig. 7.1, and an image of the connected pair of SDRs is shown in Fig. 7.2.

After the initial coarse synchronization, each node transmits the rectangularly-windowed synchronization pulse, with SDR 1 transmitting at a baseband frequency $f_1^\alpha = 1$ MHz and SDR 2 transmitting at a baseband frequency $f_2^\alpha = -1$ MHz. Each transmitted pulse has a pulse length of 60 μs. Each SDR receives in a 100 μs window sampled at 30 MHz, with the receive window aligned such that the received pulse will fit entirely in the window with at least 15 μs cushion on either side without accounting for any propagation delay. With a clock drift of 1.00001, it is guaranteed that the full

pulse will appear in the receive window so long as the pulse is transmitted within 1.5 s of the initial coarse synchronization. After this, the clock drift value is estimated and used to compensate the carrier frequency and the FPGA clock, which ensures that future transmitted signals will be aligned in the receive window of the other node. All clock drift estimations and waveform compensations are performed on a desktop computer. For simplicity, the communication of the frequency estimates is achieved through the computer link rather than through a modulated communications waveform over the coaxial link between the SDRs.

7.2 Experimental Results

To demonstrate the statistical accuracy of the proposed technique, a sample set of 100 single-tone pulse exchanges is produced to generate 100 independent clock drift estimates. The FPGA clock is aligned using the first clock drift estimate to ensure the subsequent pulses will fall within the receive window, though no carrier corrections are applied such that the quality of each drift estimate will still be identical to the first estimate. Because there are only 2 nodes in the network and it is assumed $\alpha_1 = 1$, only an estimate of α_2 is produced. The 100 clock drift estimates have a mean of $\mu_\alpha = 1.000009999818253$, giving a mean error of $\mu_\epsilon = -1.81746824257 \times 10^{-10}$. The standard deviation of the clock drift estimate errors is $\sigma_\epsilon = 9.75395662 \times 10^{-9}$, which translates to <10 ppb error. At the carrier frequency of this experiment, the result is an average corrected carrier frequency misalignment of 29.94 Hz.

Next, the system was reset and another clock drift estimate was produced, this time with a value of $\hat{\alpha}_2 = 1.000010000252$. To demonstrate the carrier frequency corrections in (4.36) and (4.47), the same single-tone waveforms with frequencies of $f_1^\alpha = 1$ MHz and $f_2^\alpha = -1$ MHz were transmitted with the transmit and receive

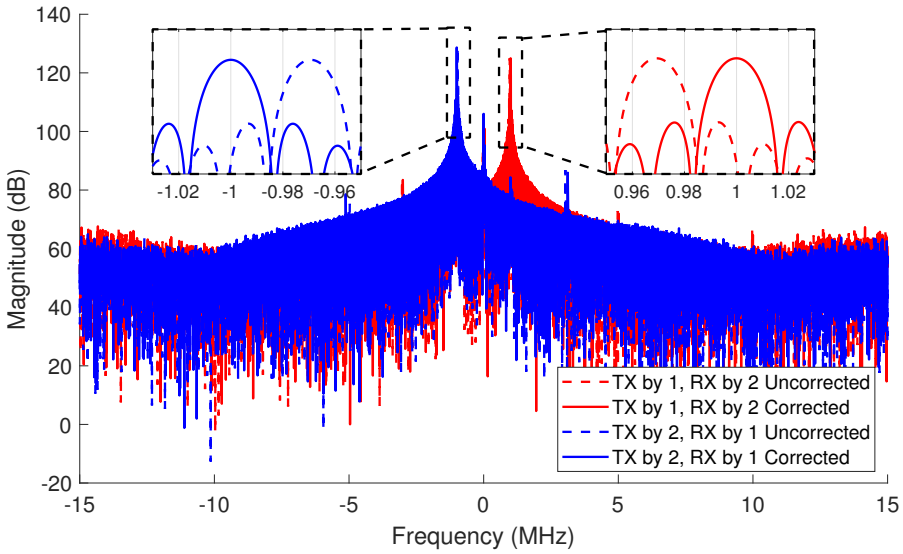
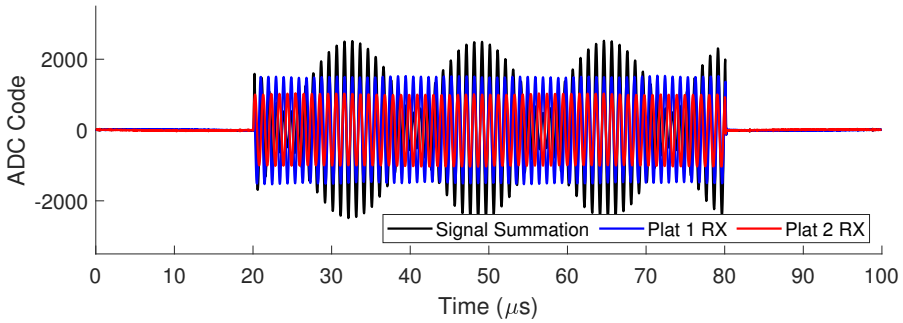


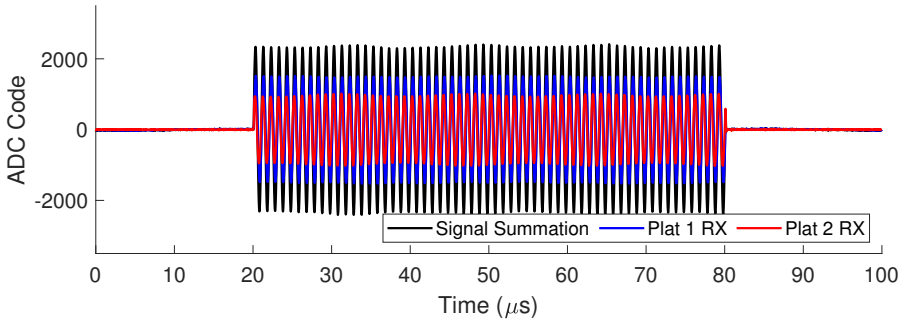
Fig. 7.3: The FFT of the single-tone signals transmitted and received by each platform, before and after the carrier frequency corrections were applied. The FFT is computed with an up-sampling factor of 200. From [5] ©2024 IEEE.

corrections applied by both platforms. Fig. 7.3 shows the FFT of the signals received by both nodes from one another, before and after the carrier frequency corrections. The FFT is taken after zero-padding to increase the frequency resolution. Because of the precisely known clock drift, the expected frequency error before corrections may be computed to be 30.7 kHz, which matches closely to the frequency offsets observed in Fig. 7.3. After correction, the frequency error of the tone received by node 1 is 22 Hz and the error of the tone received by node 2 is 31 Hz, which are larger errors than the expected 0.773 Hz computed from the clock drift estimate. These errors are attributable to the small but non-negligible noise power on the signals, which may slightly shift the expected peak positions during the up-sampling process.

Finally, in a distributed radar application, it is desirable to coherently sum signals received by disparate platforms. For this to happen accurately, frequency errors between signals must be eliminated. Fig. 7.4 shows the summation of two signals transmitted and received by the SDRs from one another, both with and without frequency



(a)



(b)

Fig. 7.4: The signals received by nodes 1 and 2 summed without frequency corrections (a) and with frequency corrections (b). For clarity, only the real part of the signals are shown. From [5] ©2024 IEEE.

corrections. The baseband signals are single-tone pulses with nominal frequencies of 500 kHz. When the frequency corrections are not applied, the summation of the offset sinusoids produces a modulating effect, while the summation of the corrected sinusoids is clearly at the correct frequency with no modulation over the width of the pulse.

Chapter 8

Conclusion

The goal of the research in this dissertation is to develop practical techniques for cooperative navigation and synchronization of wirelessly distributed radar platforms in a cooperative radar sensor network. Advances in radar remote sensing will need to consider distributed and MIMO capabilities to maintain the modern trend of increased performance. As radar engineers continue to probe the upper limits on performance in a monostatic system, it is anticipated that next-generation systems will see the proliferation of distributed architectures since not only do they provide increased performance in a traditional sense at a potentially lower per-unit cost, but they will also provide capabilities that are not possible from a monostatic point of view due to their many geometric degrees of freedom. To enable the true capabilities of such systems, integrated localization and synchronization routines such as those described in this dissertation are of paramount importance.

First, a cooperative navigation routine is presented in Chapter 3. This procedure builds very simply on the existing principles of the UKF, using inter-platform ranging measurements in conjunction with traditional navigation measurements such as acceleration and angular velocity measurement inputs from an IMU, as well as GPS positioning signals when available, to track platforms' positions over time in an absolute

manner. It has been shown that even when using a low-quality IMU, the cooperative positioning can improve navigation error by upwards of 90% in a moderately sized network of navigating platforms in a GPS-denied scenario. Even in a scenario where GPS is available, the cooperative navigation framework enables a halving of the overall error. In both cases, the cooperative navigation case will generally outperform the non-cooperative case in terms of SAR performance when a system is included on one or all of the platforms.

Chapter 4 presents a newly developed algorithm for digitally synchronizing distributed networks of RF systems – although the emphasis in this dissertation is on radar, it must be noted that this technique is applicable to the synchronization of any network of RF systems requiring synchronization on transmit and receive, such as distributed communication arrays. This technique is decentralized, making it robust to any single point of failure. The technique can also be run very quickly, requiring relatively few transmissions from each system and only requiring significant amounts of time to complete in large networks when TDMA is required. In future iterations, other methods of signal orthogonality will be implemented, such as MIMO ranging waveforms and full exploitation of the frequency multiplexing of the single-tone waveforms, allowing the systems to receive the transmitted waveforms from all other platforms simultaneously and dramatically reducing the required runtime. The technique enables highly accurate estimation of clock drift, clock bias, and carrier phase, which may be compensated for to achieve frequency, time, and phase synchronization. Notably, this technique is entirely digital, which is a significant improvement to the state-of-the-art in this area which either cannot be accomplished without additional hardware, requires an external frequency reference, or can only be accomplished on receive but not on transmit, obviating use in systems requiring concentrated energy

on target such as directed communications or transmit beamforming for radar. It has been shown that this technique performs well enough with moderate SNR to maintain synchronization over a second or more without requiring updates, enabling such techniques as distributed transmit beamforming and high-quality MIMO SAR from platforms with relative motion since Doppler shifts do not adversely impact the frequency transfer performance. In addition to deriving and testing the algorithm extensively, a thorough theoretical analysis is provided through comparison to the CRLB, showing that the estimation technique closely approaches or meets the CRLB for the estimation of all clock parameters. Finally, the procedure produces estimates of range and relative velocity which may be used in cooperative navigation.

This synchronization algorithm is only a "one-shot" solution, meaning that the clock parameter estimates are only valid for one point in time and will become unusable quickly, either due to drift of the clock parameters or errors in the original estimates. To override this issue, tracking approaches using the UKF are discussed in Chapter 5. First, the state model is derived for tracking the clock parameters and other relevant state variables over time. Next, two measurement models are proposed: the time, phase, and frequency approach, and the iterative synchronization approach. In the time, phase, and frequency approach, general signals are transmitted and the time, phase, and frequency of the received signals at each platform are measured. With the nonlinear measurement model relating these measurements back to the original clock parameters, the UKF is able to track the parameters over time. However, this approach, while extremely accurate with respect to clock drift in some scenarios, falls apart quickly in the presence of relative motion and low SNR, making it unsuitable for use in its present form. The iterative synchronization approach, on the other hand, essentially runs the original synchronization algorithm at a set interval, and subsequent

measurements of clock parameters are integrated into the current estimates through the UKF. Results are provided which show that while the clock drift estimates are not as accurate as the time, phase, and frequency approach due to the lack of time-delay information in forming the estimates, all clock parameters may be reliably tracked in every circumstance, including low SNR, relative motion, and low update rates.

As a result of the iterative synchronization approach's success, this technique is selected for the integration of synchronization and navigation, which is described in Chapter 6. In this chapter, the iterative synchronization measurement model is integrated into the cooperative navigation simulation developed in Chapter 3. The only minor change required is to modify the state model to include cartesian position, velocity, and orientation rather than ranges and relative velocities. Instead, these range and relative velocities are used directly as measurements such that navigation may still be performed exactly as before. It is shown that by integrating these two procedures together, the performance of each procedure individually is maintained in all scenarios, including low SNR, variable SNR, and various motion paths. Future work will involve correcting the errors in the time, phase, and frequency approach since the additional information available in this approach is expected to increase the accuracy of both synchronization and navigation.

Finally, a first-time demonstration of the synchronization technique described in Chapter 4 is provided in Chapter 7. The results are demonstrated for a two-node network of SDRs with a known offset in clock drift. The clock drift is compensated for with ppb level accuracy, leading to a measured frequency correction of close to 30 Hz. Future iterations of this will focus on improving this accuracy since this will likely not be sufficient for obtaining high-accuracy Doppler phase measurements in a real radar system.

8.1 Contributions of this Work

The following significant contributions are made in this work:

- Development of UKF-based cooperative navigation
- Demonstration of benefits of navigation in cooperative networks with heterogeneous GPS access
- Utilization of cooperative navigation in single-aperture and distributed SAR
- First-time demonstration of a fully-digital synchronization algorithm for time, phase, and frequency
- Development of multiple frequency synchronization approaches tailored to different system requirements and tolerances
- Derivation of anticipated synchronization performance based on the CRLB
- Derivation of technique for solving for wrapped phase values using OLS and WLS
- Demonstration of synchronization for use in high-quality distributed transmit beamforming and MIMO SAR
- Development of two UKF-based approaches for tracking synchronization states over time
- Demonstration of UKF-based tracking of synchronization states, showing reduction in error over time

- First-time demonstration of simultaneous tracking of navigation and synchronization clock states
- First-time demonstration of the proposed all-digital synchronization approach in real hardware

8.2 Future Work

This dissertation provides a foundation and framework for navigation and synchronization in mobile distributed radar networks. However, there are still many tasks to be completed related to this research and a large body of future research to be performed. This section lays out some of these future efforts.

8.2.1 Kalman Filtering of Wrapped Phase

One key problem with the implementation of the iterative synchronization UKF approach is that if the true wrapped carrier phase is close enough to π or $-\pi$, the noisy estimates will occasionally wrap around, leading to a large apparent jump in phase which biases the mean value. The result is that the UKF will converge to a value between the true value and the noisy wrapped estimates. This problem is illustrated in Fig. 8.1. To make this technique universally applicable, a solution will need to be derived such that noisy estimates can be unwrapped correctly to ensure the true mean phase will be converged upon. This may be as simple as un-dynamically sequential phase estimates but may also require a more complex approach since there is a non-zero chance in low-SNR situations that an apparent wrapping of phase will be a legitimate jump in phase due to noise. An analysis of this problem and, ideally, a functional solution will be the subject of a future study.

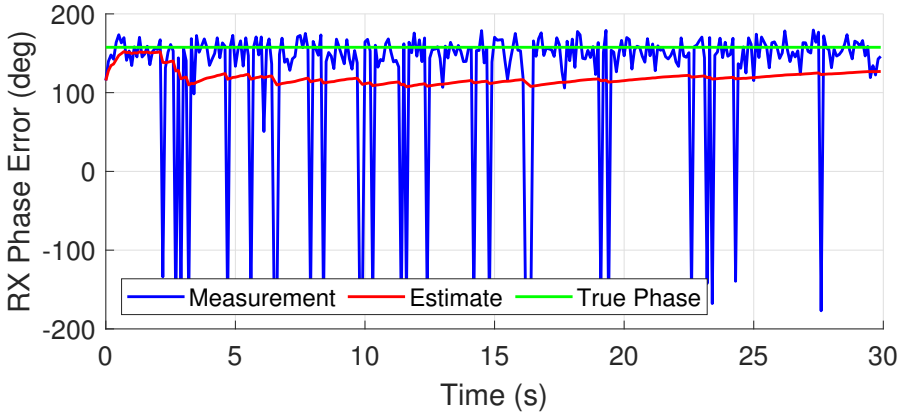


Fig. 8.1: A plot illustrating the problem of wrapped noisy phase estimates on UKF phase estimation when the carrier phase value is near the $\pm\pi$ value.

8.2.2 Proper Implementation of Time, Phase, and Frequency

As alluded to in Chapters 5 and 6, the iterative synchronization approach is serviceable for reducing synchronization error over time and providing measurements for cooperative navigation. However, in directly producing the clock state, range, and relative velocity estimates, it discards a significant amount of information which can increase overall estimation accuracy. This is most notable in the clock drift estimation accuracy, which is significantly lower in the time, phase, and frequency estimation procedure, which leverages time, phase, and frequency information rather than just frequency alone to produce high-quality estimates. Furthermore, the accuracy of other clock parameter estimates, along with range and relative velocity estimates, is dependent on clock drift synchronization being performed with high accuracy. Therefore, future work must focus on properly characterizing the sources of error in the time, phase, and frequency approach in non-ideal scenarios such as lower SNR and the presence of relative motion. One possible approach to this is to dynamically alter the update rate of the UKF based on the estimated SNR and estimated clock drift variance to ensure that there will not be aliasing of the phase estimates from itera-

tion to iteration since it is possible that the estimation error is due to aliasing phase information when there are Doppler shifts and carrier frequency skew.

Included in this future work will be the exploration of techniques for extracting range bias due to incorrect estimation of clock drift based on the assumption of $\alpha_1 = 1$. If the navigation system is accurate enough, and particularly if high-quality GPS measurements are available, it may be possible to extract the ranging bias over time and from there estimate the true values of α_i rather than just relative ones.

8.2.3 Demonstrate Technique with Practical Clock Simulations

As discussed in Chapter 5, the Kalman filtering of the synchronization state estimates serves not only to converge on clock parameters estimates in the presence of noise but also to track non-constant clock states as they slowly evolve with time, for instance, as a result of the frequency drift of an oscillator described by its Allan variance. Therefore, future work will involve simulations including additional realistic clock errors to demonstrate the UKF's ability to track changing errors over time while simultaneously reducing error. This will include both building a simulation that considers these practical clock errors as well as developing methods for characterizing the clock errors in the design of the process covariance matrix.

8.2.4 Full Hardware Demonstration of Synchronization

The hardware demonstration of the synchronization algorithm in Chapter 7 provides a fairly limited demonstration of the synchronization capabilities described in previous chapters. In general, this demonstration only demonstrates the ability of the single-tone with exchange technique to synchronize in frequency, but due to some practical effects in the SDR units, the phase synchronization is not consistent. This

inconsistency is primarily due to the FPGA sampling time, which restricts TX and RX frames to discrete multiples of 8 ns, leading to large unanticipated changes in delay phase, and the amount of time required to buffer waveforms such that the carrier phase will change due to carrier skew before the carrier phase can be estimated and compensated correctly. Future work in this area will correct these issues by addressing the underlying FPGA limitations. First, work must be done on the TX buffering of the waveform to compensate for the time and phase change required due to the potential offset between the desired TX time and the actual TX time aligning with the FPGA clock. The slow buffering time of the waveforms must be addressed by moving the components of the waveform generation from MATLAB directly to the FPGA itself to eliminate the slow transfer of data required for each waveform correction. The resulting corrections will enable a more consistent and convincing demonstration of the proposed synchronization technique which will correct for all of time, phase, and frequency. This completed system will also include a Kalman filtering component in which the UKF techniques discussed in Chapter 5 will be demonstrated in real hardware. Finally, a third SDR unit will be procured such that the synchronization techniques described in this dissertation can be tested in a network of more than two nodes.

8.2.5 Navigation and Synchronization Demonstration

While a basic hardware demonstrator has been constructed, all navigation work to date has been done using only simulation. Implementing a distributed navigation demonstration will be significantly more challenging than the synchronization alone since it will still require the implementation of all the synchronization algorithms to obtain the ranging measurements. However, this must also be done on top of imple-

menting some way of moving the radar platforms with a ground truth available. This will possibly be done using some form of track-mounted system as this will enable precise control and ground truth position measurements with which to compare. Another option is to simply place platforms on different vehicles and use high-precision RTK GPS units to produce the ground truth position measurements. This will need to be implemented with four units at a minimum to match the simulated performance shown in Chapter 3.

8.2.6 Distributed Radar Demonstration

In addition to the full navigation and synchronization implementation, a full distributed radar demonstration using the proposed synchronization algorithm is proposed. This proposed demonstration may use the same SDR implementation in Chapter 7 since these systems are fully functional radar units. As with the desired full synchronization demonstration, it is anticipated that this demonstration be done with at least three units since most of the targeted radar applications (distributed beamforming, MIMO SAR, etc.) will benefit from more than two nodes. A good starter demonstration would be a stationary network demonstrating coherent TX beamforming similar in nature to the simulation in Chapter 4, while long-term demonstrations would possibly include some form of MIMO SAR to include all of synchronization and motion compensation provided by the proposed simultaneous navigation and synchronization approach.

8.2.7 Decentralized Implementation of Filtering

One major challenge in developing the proposed distributed estimation approach toward implementation in practical systems is enabling a decentralized approach. While

the synchronization approach described in Chapter 4 is decentralized in that no centralized computations need to take place, the method does not consider synchronization with any directional communications links as may appear in some networks [113]. Furthermore, the UKF approach to navigation, synchronization, and their combination as proposed in this dissertation is exclusively centralized such that all computations will be performed on one (or all) nodes, and all state vectors and covariance matrices comprise the states and correlations among all platform states. This will ultimately lead to infeasible implementation for even moderately large networks.

To overcome this, decentralized approaches will be explored in future research. The distributed and decentralized Kalman filters have been previously explored in the literature [114, 115] to handle the networks of sensors attempting to apply Kalman filtering to reach consensus on a system state estimate. However, these particular implementations make some assumptions that would make them infeasible for the problem at hand. Efforts in potentially adapting the framework provided by these Kalman filter implementations toward the synchronization and cooperative navigation problems will be explored.

References

[1] R. H. Kenney and J. W. McDaniel, "Cooperative navigation of mobile radar sensors using time-of-arrival measurements and the unscented Kalman filter," *IEEE Trans. Radar Syst.*, vol. 1, pp. 435–447, 2023.

[2] R. H. Kenney, R. E. Jarvis, H. H. Sigmarsson, and J. W. McDaniel, "Efficient time-domain tunin of microwave filters using concepts from the unscented Kalman filter," *IEEE Trans. Microw. Theory Tech.*, vol. 72, no. 4, pp. 2391–2413, Apr. 2024.

[3] R. H. Kenney, J. G. Metcalf, and J. W. McDaniel, "Decentralized digital clock drift compensation in distributed radar sensor networks through single-tone frequency broadcasts," in *2023 IEEE Int. Radar Conf. RADAR*, Sydney, Australia, Nov. 2023, pp. 1–6.

[4] ——, "Wireless distributed frequency and phase synchronization for mobile platforms in cooperative digital radar networks," *IEEE Trans. Radar Syst.*, vol. 2, pp. 268–287, 2024.

[5] R. H. Kenney and J. W. McDaniel, "All-digital carrier frequency synchronization for distributed radar sensor networks," in *2024 IEEEMTT- Int. Microw. Symp. IMS*, Washington, D.C., USA, 2024, pp. 1–4.

[6] C. Baker and A. Hume, "Netted radar sensing," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 18, no. 2, pp. 3–6, Feb. 2003.

[7] T. M. Comberiate, K. S. Zilevu, J. E. Hodkin, and J. A. Nanzer, "Distributed transmit beamforming on mobile platforms using high-accuracy microwave wireless positioning," in *Radar Sens. Technol. XX*, vol. 9829. SPIE, May 2016, pp. 549–556.

[8] E. Fishler, A. Haimovich, R. Blum, L. Cimini, D. Chizhik, and R. Valenzuela, "Spatial diversity in radars—models and detection performance," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 823–838, Mar. 2006.

[9] H. Godrich, A. M. Haimovich, and R. S. Blum, "Target localization accuracy gain in MIMO radar-based systems," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2783–2803, Jun. 2010.

[10] S. Yang, W. Yi, and A. Jakobsson, "Multitarget detection strategy for distributed MIMO radar with widely separated antennas," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.

[11] K. Xiong, G. Cui, W. Yi, S. Wang, and L. Kong, "Distributed localization of target for MIMO radar with widely separated directional transmitters and omnidirectional receivers," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 3, pp. 3171–3187, Jun. 2023.

[12] J. Liang, Q. Liang, and Z. Zhou, "Radar sensor network design and optimization for blind speed alleviation," in *2007 IEEE Wirel. Commun. Netw. Conf.*, Mar. 2007, pp. 2643–2647.

[13] A. D. Byrd, R. D. Palmer, and C. J. Fulton, "Development of a low-cost multistatic passive weather radar network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2796–2808, Apr. 2020.

[14] H. Ochiai, P. Mitran, H. Poor, and V. Tarokh, "Collaborative beamforming for distributed wireless ad hoc sensor networks," *IEEE Trans. Signal Process.*, vol. 53, no. 11, pp. 4110–4124, Nov. 2005.

[15] R. Mudumbai, D. R. Brown Iii, U. Madhow, and H. V. Poor, "Distributed transmit beamforming: challenges and recent progress," *IEEE Commun. Mag.*, vol. 47, no. 2, pp. 102–110, Feb. 2009.

[16] S. Hanna and D. Cabric, "Distributed transmit beamforming: Design and demonstration from the lab to UAVs," *IEEE Trans. Wirel. Commun.*, vol. 22, no. 2, pp. 778–792, Feb. 2023.

[17] A. M. Haimovich, R. S. Blum, and L. J. Cimini, "MIMO radar with widely separated antennas," *IEEE Signal Process. Mag.*, vol. 25, no. 1, pp. 116–129, 2008.

[18] Q. He, N. H. Lehmann, R. S. Blum, and A. M. Haimovich, "MIMO radar moving target detection in homogeneous clutter," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 3, pp. 1290–1301, Jul. 2010.

[19] P. Wang, H. Li, and B. Himed, "Moving target detection using distributed MIMO radar in clutter with nonhomogeneous power," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4809–4820, Oct. 2011.

[20] N. Goodman, S. C. Lin, D. Rajakrishna, and J. Stiles, "Processing of multiple-receiver spaceborne arrays for wide-area SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 4, pp. 841–852, Apr. 2002.

[21] G. Krieger and A. Moreira, "Spaceborne bi- and multistatic SAR: potential and challenges," *IEE Proc. - Radar Sonar Navig.*, vol. 153, no. 3, pp. 184–198, Jun. 2006.

[22] G. Krieger, I. Hajnsek, K. P. Papathanassiou, M. Younis, and A. Moreira, "Interferometric synthetic aperture radar (SAR) missions employing formation flying," *Proc. IEEE*, vol. 98, no. 5, pp. 816–843, May 2010.

[23] Y. Wang, Z. Ding, L. Li, M. Liu, X. Ma, Y. Sun, T. Zeng, and T. Long, "First demonstration of single-pass distributed SAR tomographic imaging with a P-band UAV SAR prototype," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–18, 2022.

[24] R. H. Kenney, K. Konyalioglu, M. Yeary, H. H. Sigmarsson, and J. W. McDaniel, "An all-COTS high sampling frequency pulse-doppler imaging radar," in *2020 IEEE Radar Conf. RadarConf20*, Sep. 2020, pp. 1–6.

[25] D. Moller, S. Hensley, G. A. Sadowy, C. D. Fisher, T. Michel, M. Zawadzki, and E. Rignot, "The glacier and land ice surface topography interferometer: An airborne proof-of-concept demonstration of high-precision ka-band single-pass elevation mapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 2, pp. 827–842, Feb. 2011.

[26] R. Feger, A. Haderer, and A. Stelzer, "Experimental verification of a 77-GHz synthetic aperture radar system for automotive applications," in *2017 IEEE MTT- Int. Conf. Microw. Intell. Mobil. ICMIM*, Mar. 2017, pp. 111–114.

[27] M. E. Yanik, D. Wang, and M. Torlak, "Development and demonstration of MIMO-SAR mmWave imaging testbeds," *IEEE Access*, vol. 8, pp. 126 019–126 038, 2020.

[28] C. Waldschmidt, J. Hasch, and W. Menzel, "Automotive radar — from first efforts to future systems," *IEEE J. Microw.*, vol. 1, no. 1, pp. 135–148, Jan. 2021.

[29] W. Q. Wang, C. B. Ding, and X. D. Liang, "Time and phase synchronisation via direct-path signal for bistatic synthetic aperture radar systems," *IET Radar Sonar Navig.*, vol. 2, no. 1, pp. 1–11, Feb. 2008.

[30] W.-Q. Wang, "GPS-based time & phase synchronization processing for distributed SAR," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 45, no. 3, pp. 1040–1051, Jul. 2009.

[31] J. Liang and Q. Liang, "Design and analysis of distributed radar sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 11, pp. 1926–1933, Nov. 2011.

[32] J. A. Nanzer, S. R. Mghabghab, S. M. Ellison, and A. Schlegel, "Distributed phased arrays: Challenges and recent advances," *IEEE Trans. Microw. Theory Tech.*, vol. 69, no. 11, pp. 4893–4907, Nov. 2021.

[33] P. Chatterjee and J. A. Nanzer, "Effects of time alignment errors in coherent distributed radar," in *2018 IEEE Radar Conf. RadarConf18*, Apr. 2018, pp. 0727–0731.

[34] R. Kenney, "Design and implementation of an all-COTS digital back-end for a pulse-doppler synthetic aperture radar," Master's thesis, University of Oklahoma, May 2020.

[35] A. Naouri, L. Ortega, J. Vilà-Valls, and E. Chaumette, "A multidimensional scaling approach for cooperative GNSS navigation," in *2021 Int. Conf. Localization GNSS ICL-GNSS*, Jun. 2021, pp. 1–6.

[36] S. Sand, S. Zhang, M. Mühlegg, G. Falconi, C. Zhu, T. Krüger, and S. Nowak, "Swarm exploration and navigation on Mars," in *2013 Int. Conf. Localization GNSS ICL-GNSS*, Jun. 2013, pp. 1–6.

[37] M. G. Amin, P. Closas, A. Broumandan, and J. L. Volakis, "Vulnerabilities, threats, and authentication in satellite-based navigation systems [scanning the issue]," *Proc. IEEE*, vol. 104, no. 6, pp. 1169–1173, Jun. 2016.

[38] N. Zhu, J. Marais, D. Bétaille, and M. Berbineau, "GNSS position integrity in urban environments: A Review of literature," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 9, pp. 2762–2778, Sep. 2018.

[39] I. Skog and P. Handel, "In-car positioning and navigation technologies—a survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 4–21, Mar. 2009.

[40] J. Wen, L. He, and F. Zhu, "Swarm robotics control and communications: Imminent challenges for next generation smart logistics," *IEEE Commun. Mag.*, vol. 56, no. 7, pp. 102–107, Jul. 2018.

[41] Y. Gu, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks," *IEEE Commun. Surv. Tutor.*, vol. 11, no. 1, pp. 13–32, 2009.

[42] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, pp. 131–149, Jan. 2014.

[43] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4409–4420, Nov. 2012.

[44] I. Ohya, A. Kosaka, and A. Kak, "Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing," *IEEE Trans. Robot. Autom.*, vol. 14, no. 6, pp. 969–978, Dec. 1998.

[45] C. Toth, D. A. Grejner-Brzezinska, and Y.-J. Lee, "Terrain-based navigation: Trajectory recovery from LiDAR data," in *2008 IEEEION Position Locat. Navig. Symp.*, May 2008, pp. 760–765.

[46] A. Wahdan, J. Georgy, W. F. Abdelfatah, and A. Noureldin, "Magnetometer calibration for portable navigation devices in vehicles using a Fast and autonomous technique," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 5, pp. 2347–2352, Oct. 2014.

[47] P. Groves, *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd ed. Artech House, Mar. 2013.

[48] L. Chang, K. Li, and B. Hu, "Huber's M-estimation-based process uncertainty robust filter for integrated INS/GPS," *IEEE Sens. J.*, vol. 15, no. 6, pp. 3367–3374, Jun. 2015.

[49] G. Guo and J. Liu, "A stochastic model-based fusion algorithm for enhanced localization of land vehicles," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–10, 2022.

[50] A. Noureldin, T. B. Karamat, M. D. Eberts, and A. El-Shafie, "Performance enhancement of MEMS-based INS/GPS integration for low-cost navigation applications," *IEEE Trans. Veh. Technol.*, vol. 58, no. 3, pp. 1077–1096, Mar. 2009.

[51] "A double-stage Kalman filter for orientation tracking with an integrated processor."

[52] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, Mar. 1960.

[53] M. Athans, "The importance of Kalman filtering methods for economic systems," in *Annals of Economic and Social Measurement, Volume 3, number 1*. NBER, 1974, pp. 49–64.

[54] L. P. Perera, P. Oliveira, and C. Guedes Soares, "Maritime traffic monitoring based on vessel detection, tracking, state estimation, and trajectory prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1188–1200, Sep. 2012.

[55] S. Blackman, "Multiple hypothesis tracking for multiple target tracking," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 19, no. 1, pp. 5–18, Jan. 2004.

[56] A. Farina, B. Ristic, and D. Benvenuti, "Tracking a ballistic target: comparison of several nonlinear filters," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 3, pp. 854–867, Jul. 2002.

[57] S. Y. Chen, "Kalman filter for robot vision: A survey," *IEEE Trans. Ind. Electron.*, vol. 59, no. 11, pp. 4409–4420, Nov. 2012.

[58] D.-J. Jwo and S.-H. Wang, "Adaptive fuzzy strong tracking extended Kalman filtering for GPS navigation," *IEEE Sens. J.*, vol. 7, no. 5, pp. 778–789, May 2007.

[59] M. B. Alatise and G. P. Hancke, "Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter," *Sensors*, vol. 17, no. 10, p. 2164, Oct. 2017.

[60] J. Crassidis, "Sigma-point Kalman filtering for integrated GPS and inertial navigation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 42, no. 2, pp. 750–756, Apr. 2006.

[61] Z. Bodó and B. Lantos, "State estimation for UAVs using sensor fusion," in *2017 IEEE 15th Int. Symp. Intell. Syst. Inform. SISY*, Sep. 2017, pp. 000 111–000 116.

[62] J. Fang and X. Gong, "Predictive iterated Kalman filter for INS/GPS integration and its application to SAR motion compensation," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 4, pp. 909–915, Apr. 2010.

[63] A. C. B. Chiella, B. O. S. Teixeira, and G. A. S. Pereira, "Quaternion-based robust attitude estimation using an adaptive unscented Kalman filter," *Sensors*, vol. 19, no. 10, p. 2372, Jan. 2019.

[64] F. Daum, "Nonlinear filters: beyond the Kalman filter," *IEEE Aerosp. Electron. Syst. Mag.*, vol. 20, no. 8, pp. 57–69, Aug. 2005.

[65] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Trans. Autom. Control*, vol. 45, no. 3, pp. 477–482, Mar. 2000.

[66] E. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. IEEE 2000 Adapt. Syst. Signal Process. Commun. Control Symp. Cat No00EX373*, Oct. 2000, pp. 153–158.

[67] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[68] P. Djuric, J. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo, and J. Miguez, "Particle filtering," *IEEE Signal Process. Mag.*, vol. 20, no. 5, pp. 19–38, Sep. 2003.

[69] H. Khazraj, F. Faria da Silva, and C. L. Bak, "A performance comparison between extended Kalman Filter and unscented Kalman Filter in power system dynamic state estimation," in *2016 51st Int. Univ. Power Eng. Conf. UPEC*, Sep. 2016, pp. 1–6.

[70] M. Partovibakhsh and G. Liu, "An adaptive unscented Kalman filtering approach for online estimation of model parameters and state-of-charge of lithium-ion batteries for autonomous mobile robots," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 1, pp. 357–363, Jan. 2015.

[71] J. Zhao and L. Mili, "Robust unscented Kalman filter for power system dynamic state estimation with unknown noise statistics," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1215–1224, Mar. 2019.

[72] M. St-Pierre and D. Gingras, "Comparison between the unscented Kalman filter and the extended Kalman filter for the position estimation module of an integrated navigation information system," in *IEEE Intell. Veh. Symp. 2004*, Jun. 2004, pp. 831–835.

[73] P. Zhang, J. Gu, E. Milios, and P. Huynh, "Navigation with IMU/GPS/digital compass with unscented Kalman filter," in *IEEE Int. Conf. Mechatron. Autom. 2005*, vol. 3, Jul. 2005, pp. 1497–1502 Vol. 3.

[74] W. You, F. Li, L. Liao, and M. Huang, "Data fusion of UWB and IMU based on unscented Kalman filter for indoor localization of quadrotor UAV," *IEEE Access*, vol. 8, pp. 64 971–64 981, 2020.

[75] W. Carrara, R. Goodman, and R. Majewski, *Spotlight synthetic aperture radar: signal processing algorithms*. Artech House, 1995.

[76] P. H. Eichel, D. C. Ghiglia, and C. V. Jakowatz, "Speckle processing method for synthetic-aperture-radar phase correction," *Opt. Lett., OL*, vol. 14, no. 1, pp. 1–3, Jan. 1989.

[77] B. Sun, M. Yeary, H. H. Sigmarsson, and J. W. McDaniel, "Fine resolution position estimation using Kalman filtering," in *2019 IEEE Int. Instrum. Meas. Technol. Conf. I2MTC*, May 2019, pp. 1–5.

[78] B. M. Sun, R. H. Kenney, M. B. Yeary, H. H. Sigmarsson, and J. W. McDaniel, "An Up-Sampled Particle Filter Fusion Technique and Its Application in Synthetic Aperture Radar Imaging," *IEEE J. Microw.*, vol. 2, no. 1, pp. 108–122, Jan. 2022.

[79] B. Sun, "Fusion of multiple inertial measurements units and its application in reduced cost, size, weight, and power synthetic aperture radars," Ph.D. dissertation, University of Oklahoma, May 2022.

[80] M. Xing, X. Jiang, R. Wu, F. Zhou, and Z. Bao, "Motion compensation for UAV SAR based on raw radar data," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 8, pp. 2870–2883, Aug. 2009.

[81] S. Werness, W. Carrara, L. Joyce, and D. Franczak, "Moving target imaging algorithm for SAR data," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 26, no. 1, pp. 57–67, Jan. 1990.

[82] S. Werness, M. Stuff, and J. Fienup, "Two-dimensional imaging of moving targets in SAR data," in *1990 Conf. Rec. Twenty-Fourth Asilomar Conf. Signals Syst. Comput. 1990*, vol. 1, Oct. 1990, pp. 16–.

[83] H. Mu, T. Bailey, P. Thompson, and H. Durrant-Whyte, "Decentralised solutions to the cooperative multi-platform navigation problem," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 2, pp. 1433–1449, Apr. 2011.

[84] T. Qin, M. Macdonald, and D. Qiao, "Fully decentralized cooperative navigation for spacecraft constellations," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 4, pp. 2383–2394, Aug. 2021.

[85] H. Mokhtarzadeh and D. Gebre-Egziabher, "Performance of networked dead reckoning navigation system," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 5, pp. 2539–2553, Oct. 2016.

[86] K. Yu, Y. J. Guo, and M. Hedley, "TOA-based distributed localisation with unknown internal delays and clock frequency offsets in wireless sensor networks," *IET Signal Process.*, vol. 3, no. 2, pp. 106–118, Mar. 2009.

[87] N. Patwari, J. Ash, S. Kyperountas, A. Hero, R. Moses, and N. Correal, "Locating the nodes: cooperative localization in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 54–69, Jul. 2005.

[88] H. Wymeersch, J. Lien, and M. Z. Win, "Cooperative localization in wireless networks," *Proc. IEEE*, vol. 97, no. 2, pp. 427–450, Feb. 2009.

[89] M. Z. Win, A. Conti, S. Mazuelas, Y. Shen, W. M. Gifford, D. Dardari, and M. Chiani, "Network localization and navigation via cooperation," *IEEE Commun. Mag.*, vol. 49, no. 5, pp. 56–62, May 2011.

[90] M. Z. Win, Y. Shen, and W. Dai, "A theoretical foundation of network localization and navigation," *Proc. IEEE*, vol. 106, no. 7, pp. 1136–1165, Jul. 2018.

[91] M. Ilyas, J. Lim, J. G. Lee, and C. G. Park, "Federated unscented Kalman filter design for multiple satellites formation flying in LEO," in *2008 Int. Conf. Control Autom. Syst.*, Oct. 2008, pp. 453–458.

[92] S. Prager, M. S. Haynes, and M. Moghaddam, "Wireless subnanosecond RF synchronization for distributed ultrawideband software-defined radar networks," *IEEE Trans. Microw. Theory Tech.*, vol. 68, no. 11, pp. 4787–4804, Nov. 2020.

[93] S. Mghabghab and J. A. Nanzer, "A self-mixing receiver for wireless frequency synchronization in coherent distributed arrays," in *2020 IEEEMTT- Int. Microw. Symp. IMS*, Aug. 2020, pp. 1137–1140.

[94] S. R. Mghabghab and J. A. Nanzer, "Open-loop distributed beamforming using wireless frequency synchronization," *IEEE Trans. Microw. Theory Tech.*, vol. 69, no. 1, pp. 896–905, Jan. 2021.

[95] S. R. Mghabghab, S. M. Ellison, and J. A. Nanzer, "Open-loop distributed beamforming using wireless phase and frequency synchronization," *IEEE Microw. Wirel. Compon. Lett.*, vol. 32, no. 3, pp. 234–237, Mar. 2022.

[96] J. M. Merlo, A. Schlegel, and J. A. Nanzer, "High accuracy wireless time-frequency transfer for distributed phased array beamforming," in *2023 IEEEMTT- Int. Microw. Symp. - IMS 2023*, Jun. 2023, pp. 109–112.

[97] S. M. Ellison, S. Mghabghab, J. J. Doroshewitz, and J. A. Nanzer, "Combined wireless ranging and frequency transfer for internode coordination in open-loop coherent distributed antenna arrays," *IEEE Trans. Microw. Theory Tech.*, vol. 68, no. 1, pp. 277–287, Jan. 2020.

[98] B. D. Carlton, J. W. McDaniel, and J. G. Metcalf, "Optimizing the tradeoff between radar waveform resolution and sidelobe level using a Dolph-Chebyshev approach," in *2023 IEEE Radar Conf. RadarConf23*, May 2023, pp. 1–6.

[99] W.-Q. Wang, "Carrier frequency synchronization in distributed wireless sensor networks," *IEEE Syst. J.*, vol. 9, no. 3, pp. 703–713, Sep. 2015.

[100] M. Younis, R. Metzig, and G. Krieger, "Performance prediction of a phase synchronization link for bistatic SAR," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 3, pp. 429–433, Jul. 2006.

[101] Y. Yang and R. S. Blum, "Phase synchronization for coherent MIMO radar: Algorithms and their analysis," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5538–5557, Nov. 2011.

[102] S. Julier, "The scaled unscented transformation," in *Proc. 2002 Am. Control Conf. IEEE Cat NoCH37301*, vol. 6, May 2002, pp. 4555–4559 vol.6.

[103] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Trans. Autom. Control*, vol. 54, no. 6, pp. 1254–1269, Jun. 2009.

[104] Benoit, "Note sur une méthode de résolution des équations normales provenant de l'application de la méthode des moindres carrés a un système d'équations linéaires en nombre inférieur a celui des inconnues (procédé du commandant cholesky)," *Bull. Géodésique*, vol. 2, no. 1, pp. 67–77, Apr. 1924.

[105] Å. Björck and S. Hammarling, "A Schur method for the square root of a matrix," *Linear Algebra and its Applications*, vol. 52–53, pp. 127–140, Jul. 1983.

[106] E. Deadman, N. J. Higham, and R. Ralha, "Blocked Schur algorithms for computing the matrix square root," in *Appl. Parallel Sci. Comput.*, ser. Lecture Notes in Computer Science, P. Manninen and P. Öster, Eds. Berlin, Heidelberg: Springer, 2013, pp. 171–182.

[107] D. M. Pozar, *Microwave Engineering*, 4th ed. John Wiley & Sons Inc., Nov. 2011.

[108] S. Coutts, K. Cuomo, J. McHarg, F. Robey, and D. Weikle, "Distributed coherent aperture measurements for next generation BMD radar," in *Fourth IEEE Workshop Sens. Array Multichannel Process. 2006*, Jul. 2006, pp. 390–393.

[109] S. M. Kay, *Fundamentals of Statistical Processing, Volume I: Estimation Theory*, 1st ed. Prentice-Hall, 1993, vol. 1.

[110] D. Pauluzzi and N. Beaulieu, "A comparison of SNR estimation techniques for the AWGN channel," *IEEE Trans. Commun.*, vol. 48, no. 10, pp. 1681–1691, Oct. 2000.

[111] S. R. Mghabghab and J. A. Nanzer, "High accuracy adaptive microwave ranging using SNR-based perception for coherent distributed antenna arrays," *IEEE Trans. Circuits Syst. Regul. Pap.*, vol. 67, no. 12, pp. 5540–5549, Dec. 2020.

[112] D. Allan, "Statistics of atomic frequency standards," *Proc. IEEE*, vol. 54, no. 2, pp. 221–230, Feb. 1966.

[113] M. Rashid and J. A. Nanzer, "Frequency and Phase Synchronization in Distributed Antenna Arrays Based on Consensus Averaging and Kalman Filtering," *IEEE Trans. Wirel. Commun.*, vol. 22, no. 4, pp. 2789–2803, Apr. 2023.

[114] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proc. 44th IEEE Conf. Decis. Control*, Dec. 2005, pp. 8179–8184.

[115] ——, "Distributed Kalman filtering for sensor networks," in *2007 46th IEEE Conf. Decis. Control*, Dec. 2007, pp. 5492–5498.

Appendix A

Notation

The following notation and symbols are used:

$\bar{a}_{i,k}$	–	A 3×1 vector denoting the body-frame acceleration, as measured by an accelerometer, of platform i at time-series point k
$\bar{a}^{r}_{i,k}$	–	A 3×1 vector denoting the acceleration of platform i at time-series point k in the ECEF frame
$A_j(t, f)$	–	The ambiguity function of $s^d_j(t)$
$A_x(\tau)$	–	The amplitude of a received signal as a function of slow-time
$\mathbf{A}_f$	–	The matrix describing the coefficients of the system of linear equations solving for clock drift in the stationary case
$\mathbf{A}_{f,d}$	–	The matrix describing the coefficients of the system of linear equations solving for clock drift in the Doppler shift case
$\mathbf{A}_{f,v}$	–	The matrix describing the coefficients of the system of linear equations solving for clock drift in the Doppler shift case with coefficients solving for relative radial velocity directly rather than Doppler shift
$\mathbf{A}_\gamma$	–	The matrix describing the coefficients of the system of linear equations solving for carrier phase in the stationary case
$\mathbf{A}_\phi$	–	The matrix describing the coefficients of the system of linear equations solving for clock bias
$b_i(t)$	–	The carrier frequency compensation term on receive for platform i
B	–	Bandwidth
$\mathbf{b}_f$	–	A vector containing the frequency estimates $\hat{f}^{\alpha}_{i,j}$
$\mathbf{b}_{f,d}$	–	A vector containing the frequency estimates $\hat{f}^{\alpha,d}_{i,j}$
$\mathbf{b}_\gamma$	–	A vector containing the clock bias difference estimates $\hat{\gamma}^{\text{err}}_{i,j}$
$\mathbf{b}_\phi$	–	A vector containing the clock bias difference estimates $\hat{\phi}_{i,j}$
c	–	The speed of light c = 299792458 m/s (approximately 3×10^8 m/s)

$\mathbf{C}_f$ – The covariance matrix of the frequency estimates vector, $\mathbf{b}_f$, in the stationary case

$\mathbf{C}_{f,d}$ – The covariance matrix of the frequency estimates vector, $\mathbf{b}_{f,d}$, in the Doppler shift case

$\mathbf{C}_k^-$ – The *a priori* covariance matrix (the covariance of $\mathbf{x}_k^-$) in a Kalman filter at iteration k

$\hat{\mathbf{C}}_k$ – The covariance matrix of the state estimate (the covariance of $\hat{\mathbf{x}}_k$) of a system in a Kalman filter at iteration k

$\hat{\mathbf{C}}_k^{\text{sync}}$ – The covariance matrix of the synchronization state estimate at iteration k

$\mathbf{C}_{k,xz}$ – The cross-covariance matrix between the state estimate and measurement estimate (the cross-covariance of $\hat{\mathbf{x}}_k$ and $\hat{\mathbf{z}}_k$) in a Kalman filter at iteration k

$\mathbf{C}_{pp,i}$ – The position constituent covariance matrix of the cooperative navigation process covariance for platform i

$\mathbf{C}_{vp,i}$, $\mathbf{C}_{pv,i}$ – The position and velocity constituent cross-covariance matrix of the cooperative navigation process covariance

$\mathbf{C}_{vv,i}$ – The velocity constituent covariance matrix of the cooperative navigation process covariance for platform i

$\mathbf{C}_\gamma$ – The covariance matrix of the clock bias difference estimates vector $\mathbf{b}_\gamma$

$\mathbf{C}_\phi$ – The covariance matrix of the clock bias difference estimates vector $\mathbf{b}_\phi$

$\mathbf{C}_{\omega\omega,i}$ – The orientation constituent covariance matrix of the cooperative navigation process covariance for platform i

$d_{i,j}(t)$ – The cross-correlation signal between the signal received by platform i and the signal transmitted by platform j

$d_{i,j}^k(t)$ – The cross-correlation signal between the signal received by platform i ($\tilde{r}_{i,j}^k(\tau_i')$) and the pulse-compression signal (($s_j^d(\tau_i')$) transmitted by platform j on iteration k of the synchronization UKF

$d_j(t)$ – The auto-correlation response of the pulse compression component $s_j^d(t)$ transmitted by platform j during the iterative synchronization process

f – The continuous frequency variable in Hz

f^c – Nominal carrier frequency

f_i^c – True carrier frequency of platform i

f^d – Doppler shift frequency

$f_{i,j}^{d}$ – Doppler shift frequency between platforms i and j due to relative velocity $v_{i,j}$ – the convention is defined in this dissertation such that platforms moving away from one another have a negative Doppler shift frequency

$\hat{f}_{i,j}^{d}$ – An estimate of the Doppler shift frequency between platforms i and j due to relative velocity $v_{i,j}$

$\hat{f}_{i,j}^{k}$ – The frequency estimate of the single-tone component of $\tilde{r}_{i,j}^{k}(\tau_i')$

f^{MO} – The nominal main oscillator frequency of a platform

f_i^{MO} – The true main oscillator frequency of platform i

f^{s} – Nominal sampling frequency

f_i^{s} – True sampling frequency of platform i

f_j^{α}, f_j^{β} – A single-tone frequency transmitted by platform j

$f_{i,j}^{\alpha}$, $f_{i,j}^{\beta}$ – The true frequency of the observed single tone observed by platform i and transmitted by platform j

$f_{i,j}^{\alpha}$, $f_{i,j}^{\beta}$ – The estimated frequency of the observed single tone observed by platform i and transmitted by platform j

$f_{i,j}^{\alpha,d}$, $f_{i,j}^{\beta,d}$ – The true frequency of the observed single tone observed by platform i and transmitted by platform j including the Doppler shift frequency

$\hat{f}_{i,j}^{\alpha,d}$, $\hat{f}_{i,j}^{\beta,d}$ – The estimated frequency of the observed single tone observed by platform i and transmitted by platform j including the Doppler shift frequency

$f_k(\mathbf{x})$ – The state transition function in nonlinear Kalman filtering at iteration k

$\mathbf{F}_k$ – The state transition matrix in a linear Kalman filter at iteration k

F_N – The noise figure of an RF front-end

g_0 – The scalar acceleration due to gravity

$g(t, \tau)$ – A range-compressed and motion compensated baseband receive signal for a radar system as a function of fast-time t and slow-time τ

$\tilde{g}(t, \tau)$ – The ideal range-compressed and motion compensated baseband receive signal for a radar system as a function of fast-time t and slow-time τ assuming perfect motion compensation

$\mathbf{g}$ – The acceleration due to gravity in the ECEF frame

$\mathbf{g}_{\mathrm{ned}}$ – The acceleration due to gravity in the NED frame

G_{r} – Receiver antenna gain

G_{RF} – RF front-end gain

G_{t} – Transmitter antenna gain

$h_k(\mathbf{x})$	–	The measurement function in nonlinear Kalman filtering at iteration k
$\mathbf{H}_k$	–	The observation matrix in a linear Kalman filter at iteration k
i	–	An index unit
$\mathbf{I}$	–	The identity matrix
j	–	An index unit
j	–	The imaginary unit $\mathrm{j} = \sqrt{-1}$
J	–	A Jacobian matrix
k	–	The iteration index in a Kalman filter
k	–	The frequency index of the DFT
k_b	–	Boltzmann's constant $k_b \approx 1.38 \times 10^{-23}$ J/K
$\mathrm{k_{pk}}$	–	The index of the peak magnitude value in a DFT
$\mathbf{K}_k$	–	The Kalman gain in a Kalman filter at iteration k
l	–	An index unit
L_b	–	Latitude
L	–	The number of links between platforms in the network
$m_{i,j}$	–	The peak value of the cross-correlation during the TWTT process
$m'_{i,j}$	–	The peak value of the cross-correlation during the TWTT process adjusted to account for asymmetric range between moving platforms in a TDMA ranging scheme
M_K	–	The length of the measurement vector in a Kalman filter
n	–	An index unit, and the index for a discrete time-domain signal
N_K	–	The length of the state vector in a Kalman filter
N_p	–	The number of radar platforms in a network
p	–	An index unit
$\bar{p}_{i,k}$ or $\bar{p}_i(\tau)$	–	A 3×1 vector denoting the position of platform i at time-series point k (discrete) or slow-time τ (continuous)
P_D	–	Power at the system ADC
P_N	–	Noise power
P_r	–	Received signal power
P_t	–	Transmitted signal power
$\mathbf{p}_0$	–	The scene center position for motion compensation
$\mathbf{Q}_k$	–	The process covariance in a Kalman filter at iteration k
$\mathbf{Q}_k^{\mathrm{nav}}$	–	The process covariance in the navigation UKF at iteration k

$\mathbf{Q}_{i,k}^{\text{nav}}$ – The process covariance associated with platform i in the navigation UKF at iteration k

$\mathbf{Q}_{k}^{\text{ns}}$ – The process covariance in the navigation and synchronization UKF at iteration k

$\mathbf{Q}_{i,k}^{\text{ns}}$ – The process covariance associated with platform i in the navigation and synchronization UKF at iteration k

$\mathbf{Q}_{k}^{\text{sync}}$ – The process covariance in the synchronization UKF at iteration k

$\mathbf{Q}_{k}^{v}$ – The radial velocity process covariance matrix for the synchronization UKF at iteration k

$\mathbf{Q}_{k}^{\alpha}$ – The clock drift process covariance matrix for the synchronization UKF at iteration k

$\mathbf{Q}_{k}^{\gamma}$ – The carrier phase process covariance matrix for the synchronization UKF at iteration k

$\mathbf{Q}_{k}^{\phi}$ – The clock bias process covariance matrix for the synchronization UKF at iteration k

$r_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i

$r'_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i with transmit drift compensation

$\tilde{r}_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i with transmit drift, bias, and phase compensation

$r^{b}_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i after carrier frequency compensation

$\tilde{r}^{b}_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i after carrier frequency compensation with transmit drift, bias, and phase compensation

$\tilde{r}^{c}_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i after carrier frequency compensation with transmit drift, bias, and phase compensation and receive bias correction

R, R_1, R_2, $R(t)$ – Generic range variables

$r^{d}_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i with the Doppler shift included

$\tilde{r}^{f}_{i,j}(t)$ – The baseband received signal transmitted by platform j and received by platform i after all transmit and receive clock corrections are applied

$\tilde{r}_{i,j}^{k}(t)$ – The baseband received signal transmitted by platform j and received by platform i after all transmit and receive clock corrections are applied on iteration k of the synchronization UKF

$\tilde{r}_{i,j}^{t}(t)$ – The truncated single-tone component of the baseband received signal transmitted by platform j and received by platform i after all transmit and receive clock corrections are applied on iteration k of the synchronization UKF

$r_{i,j}^{\text{tt}}(t)$ – The two-tone baseband received signal transmitted by platform j and received by platform i including the Doppler shift

$r_{i,j}^{\alpha}(t)$, $r_{i,j}^{\beta}(t)$ – The single-tone baseband received signal transmitted by platform j and received by platform i

$r_{i,j}^{\alpha,d}(t)$, $r_{i,j}^{\beta,d}(t)$ – The single-tone baseband received signal transmitted by platform j and received by platform i including the Doppler shift

$R_{i,j}$ – The range between platforms i and j

$\hat{R}_{i,j}$ – An estimate of the range between platforms i and j

$R_{i,j}^{k}$ – The range between platforms i and j on iteration k

$\hat{R}_{i,j}^{z,k}$ – The measurement of the range between platforms i and j at iteration k of the UKF for the iterative synchronization approach

$R(\tau)$ – The range to a target from a radar system as a function of slow-time

$R_c(\tau)$ – The true motion compensation range to a target from a radar system as a function of slow-time

$\hat{R}_c(\tau)$ – The estimated motion compensation range to a target from a radar system as a function of slow-time

$R_e(\tau)$ – The error in the estimated motion compensation range to a target from a radar system as a function of slow-time

$\mathbf{R}_k$ – The measurement covariance in a Kalman filter at iteration k

$\mathbf{R}_k^f$ – The covariance of the frequency measurements at iteration k of the synchronization UKF

$\mathbf{R}_k^{\text{GPS}}$ – The covariance of the GPS measurements ($\mathbf{z}_k^{\text{GPS}}$) at iteration k

$\mathbf{R}_k^{\text{IS}}$ – The covariance of the synchronization measurements ($\mathbf{z}_k^{\text{sync}}$) at iteration k for the iterative synchronization approach

$\mathbf{R}_k^{\text{nav}}$ – The covariance of the navigation measurements ($\mathbf{z}_k^{\text{nav}}$) at iteration k

$\mathbf{R}_k^R$ – The covariance of the range measurements ($\mathbf{z}_k^R$) at iteration k

$\mathbf{R}_k^{\text{TPF}}$ – The covariance of the synchronization measurements ($\mathbf{z}_k^{\text{sync}}$) at iteration k for the time, phase, and frequency approach

$\mathbf{R}_k^{\gamma}$ – The covariance of the phase measurements at iteration k of the synchronization UKF

$\mathbf{R}_k^{\tau}$	–	The covariance of the time-delay measurements at iteration k of the synchronization UKF
$\mathcal{R}_{i,j}^{\mathrm{tt}}(f)$, $\mathcal{R}_{i,j}^{\mathrm{tt}}[k]$	–	The Fourier transform (or DFT) of the received two-tone signal transmitted by platform j and received by platform i ($r_{i,j}^{\mathrm{tt}}(t)$)
$\mathcal{R}_{i,j}^{\alpha}(f)$, $\mathcal{R}_{i,j}^{\beta}(f)$, $\mathcal{R}_{i,j}^{\alpha}$[k], $\mathcal{R}_{i,j}^{\beta}$[k]	–	The Fourier transform (or DFT) of the received single-tone signal transmitted by platform j and received by platform i ($r_{i,j}^{\alpha}(t)$)
$\mathcal{R}_{i,j}^{\alpha,d}(f)$, $\mathcal{R}_{i,j}^{\beta,d}(f)$, $\mathcal{R}_{i,j}^{\alpha,d}$[k], $\mathcal{R}_{i,j}^{\beta,d}$[k]	–	The Fourier transform (or DFT) of the received single-tone signal transmitted by platform j and received by platform i including the Doppler shift ($r_{i,j}^{\alpha,d}(t)$)
$s_j(t)$	–	An arbitrary complex baseband waveform transmitted by platform j
$s'_j(t)$	–	An arbitrary complex baseband waveform transmitted by platform j with drift compensation
$s_j^d(t)$	–	The pulse compression component of the iterative synchronization signal transmitted by platform j
$s_j^t(t)$	–	The single-tone component of the iterative synchronization signal transmitted by platform j
$s_j^{\mathrm{tt}}(t)$	–	The two-tone baseband transmit signal transmitted by platform j
$s_j^{\alpha}(t)$, $s_j^{\beta}(t)$	–	The single-tone baseband transmit signal transmitted by platform j
S	–	A cost function metric
$\mathbf{S}_k$	–	The innovation covariance matrix (the covariance of $\tilde{\mathbf{z}}_k$) in a Kalman filter at iteration k
$\mathrm{SNR}_{\mathrm{ADC}}$	–	SNR at the input to a receiver's ADC
$\mathrm{SNR}_{\mathrm{IN}}$	–	SNR at the input to a receiver's antenna
t	–	The time variable used to describe fast-time and global true time for signals
t_j^{tx}	–	The transmit time of platform j according to global time
$t_{i,j}^{\mathrm{rx}}$	–	The receive time at platform i of the signal transmitted by platform j according to global time
T_c	–	Capture length
$T_{j,k}^{\mathrm{tx}}$	–	The transmit time of platform j on iteration k of the synchronization UKF
T_p	–	Pulse length
T_s	–	Sampling time

T_2 – The gap between $s_j^d(t)$ and $s_j^t(t)$

$\text{TOF}_{i,j}$ – The time-of-flight between platforms i and j

v – Generic velocity variable

$\bar{v}_{i,k}$ – A 3×1 vector denoting the velocity of platform i at time-series point k

$v_{i,j}$ – The relative velocity between platforms and i and j – the convention assumed in this dissertation is that positive velocity implies the platforms are moving apart from one another

$v_{i,j}^k$ – The relative velocity between platforms and i and j on iteration k of the UKF

$\bar{v}_{i,j}^k$ – The relative velocity vector between platforms and i and j on iteration k of the UKF

$\hat{v}_{i,j}$ – An estimate of the relative velocity between platforms and i and j

$\hat{v}_{i,j}^{z,k}$ – The measurement of the relative velocity between platforms i and j at iteration k of the UKF for the iterative synchronization approach

$V_{i,j}^{\text{err}}$ – The variance of the carrier phase difference estimates $\hat{\gamma}_{i,j}^{\text{err}}$

$V_{i,j}^{f}$ – The variance of the frequency estimates $\hat{f}_{i,j}^{\alpha}$ or $\hat{f}_{i,j}^{\alpha,d}$

$V_{i,j}^{f,k}$ – The variance of the frequency measurement $\hat{f}_{i,j}^{k}$

$V_{i,j}^{R}$ – The variance of the range estimates $\hat{R}_{i,j}$

$V_{i,j}^{\gamma}$ – The variance of the direct phase estimates $\angle d_{i,j}(m_{i,j})$

$V_{i,j}^{\gamma,k}$ – The variance of the phase measurement $\hat{\gamma}_{i,j}^{k}$

$V_{i,j}^{\tau}$ – The variance of the time-delay estimates $m_{i,j}$

$V_{i,j}^{\tau,k}$ – The variance of the time-delay measurement $\hat{\tau}_{i,j}^{k}$

$V_{i,j}^{\phi}$ – The variance of the clock bias difference estimates $\hat{\phi}_{i,j}$

$\mathbf{v}_k$ – Measurement noise in a Kalman filter at iteration k

$w_{i,c}$ – The sigma point weight for sigma point i used in computing the covariance of transformed sigma points in the unscented transform

$w_{i,m}$ – The sigma point weight for sigma point i used in computing the mean value of transformed sigma points in the unscented transform

$w_j(t)$ – An arbitrary complex RF up-converted signal transmitted by platform j

$w_j'(t)$ – An arbitrary complex RF up-converted signal transmitted by platform j with drift compensation

$\mathbf{w}_k$ – Process noise in a Kalman filter at iteration k

$\mathbf{w}_v^k$ – Radial velocity process noise in the synchronization UKF at iteration k

$\mathbf{w}_{\alpha}^{k}$ – Clock drift process noise in the synchronization UKF at iteration k

$\mathbf{w}_{\gamma}^{k}$ – Carrier phase process noise in the synchronization UKF at iteration k

$\mathbf{w}_{\phi}^{k}$ – Clock bias process noise in the synchronization UKF at iteration k

x – One dimension in Cartesian space

$x(t, \tau)$ – A range-compressed baseband receive signal for a radar system as a function of fast-time t and slow-time τ

$\tilde{x}(t, \tau)$ – The ideal range-compressed baseband receive signal for a radar with a single point scatterer at zero meters (i.e., the ideal matched filter response) as a function of fast-time t and slow-time τ

$\tilde{x}_R(t, \tau)$ – The ideal range-compressed baseband receive signal for radar with a single point scatterer at R meters as a function of fast-time t and slow-time τ

$\mathbf{x}_k$ – The true state vector of a system in a Kalman filter at iteration k

$\mathbf{x}_k^{\text{nav}}$ – The true state vector of the navigation UKF at iteration k

$\mathbf{x}_k^{\text{ns}}$ – The true state vector of the simultaneous navigation and synchronization UKF at iteration k

$\mathbf{x}_k^{\text{sync}}$ – The true state vector of the synchronization UKF at iteration k

$\mathbf{x}_k^{-}$ – The *a priori* state vector of a system in a Kalman filter at iteration k

$\mathbf{x}_{i,k}^{\text{nav}}$ – The navigational system state vector for the i^{th} platform at time series point k

$\mathbf{x}_{i,k}^{\text{ns}}$ – The simultaneous navigation and synchronization system state vector for the i^{th} platform at time series point k

$\mathbf{x}_R^k$ – The range state vector of the synchronization UKF at iteration k

$\mathbf{x}_v^k$ – The relative velocity state vector of the synchronization UKF at iteration k

$\mathbf{x}_{\alpha}^{k}$ – The clock bias state vector of the synchronization UKF at iteration k

$\mathbf{x}_{\gamma}^{k}$ – The carrier phase state vector of of the synchronization UKF at iteration k

$\mathbf{x}_{\phi}^{k}$ – The clock drift state vector of the synchronization UKF at iteration k

$\hat{\mathbf{x}}_d$ – The estimates of Doppler shift frequency produced by least-squares estimation

$\hat{\mathbf{x}}_f$ – The estimates of clock drift and Doppler shift frequency produced by least-squares estimation

$\hat{\mathbf{x}}_{f,v}$ – The estimates of clock drift and relative radial velocity produced by least-squares estimation

$\hat{\mathbf{x}}_k$ – The estimated state vector of a system in a Kalman filter at iteration k

$\hat{\mathbf{x}}_k^{\text{sync}}$ – The estimated synchronization state vector for the synchronization UKF at iteration k

$\hat{\mathbf{x}}_v$ – The estimates of relative radial velocity produced by least-squares estimation

$\hat{\mathbf{x}}_\alpha$ – The estimates of clock drift produced by least-squares estimation

$\hat{\mathbf{x}}_\gamma$ – The estimates of carrier phase produced by least-squares estimation

$\hat{\mathbf{x}}_\phi$ – The estimates of clock bias produced by least-squares estimation

y – One dimension in Cartesian space

z – One dimension in Cartesian space

$\mathbf{z}_{f,\text{IS}}^k$ – The synchronization UKF clock drift and velocity measurement vector at iteration k for the iterative synchronization approach

$\mathbf{z}_{f,\text{TPF}}^k$ – The synchronization UKF frequency measurement vector at iteration k for the time, phase, and frequency approach

$\mathbf{z}_k$ – The measurement vector in a Kalman filter at iteration k

$\mathbf{z}_k^{\text{GPS}}$ – The GPS measurement vector at iteration k

$\mathbf{z}_{p,k}^{\text{GPS}}$ – The GPS measurement vector for platform p at iteration k

$\mathbf{z}_k^{\text{IS}}$ – The synchronization measurement vector at iteration k for the iterative synchronization approach

$\mathbf{z}_k^{\text{nav}}$ – The navigation measurement vector at iteration k

$\mathbf{z}_k^{\text{ns}}$ – The simultaneous navigation and synchronization measurement vector at iteration k

$\mathbf{z}_k^{\text{TPF}}$ – The synchronization measurement vector at iteration k for the time, phase, and frequency approach

$\mathbf{z}_k^R$ – The range measurement vector at iteration k in the navigation UKF

$\mathbf{z}_k^v$ – The relative velocity measurement vector at iteration k in the navigation UKF

$\hat{\mathbf{z}}_k$ – The *a priori* estimate of the measurement vector in a Kalman filter at iteration k

$\tilde{\mathbf{z}}_k$ – The measurement innovation in a Kalman filter at iteration k

$\mathbf{z}_{R,\text{IS}}^k$ – The synchronization UKF range measurement vector at iteration k for the iterative synchronization approach

$\mathbf{z}_{v,\text{IS}}^k$ – The synchronization UKF and velocity measurement vector at iteration k for the iterative synchronization approach

$\mathbf{z}_{\alpha,\text{IS}}^k$ – The synchronization UKF clock drift measurement vector at iteration k for the iterative synchronization approach

$\mathbf{z}^k_{\gamma,\text{IS}}$ – The synchronization UKF carrier phase measurement vector at iteration k for the iterative synchronization approach

$\mathbf{z}^k_{\gamma,\text{TPF}}$ – The synchronization UKF phase measurement vector at iteration k for the time, phase, and frequency approach

$\mathbf{z}^k_{\tau,\text{TPF}}$ – The synchronization UKF time-delay measurement vector at iteration k for the time, phase, and frequency approach

$\mathbf{z}^k_{\phi,\text{IS}}$ – The synchronization UKF clock bias measurement vector at iteration k for the iterative synchronization approach

α_u – A tuning parameter of the UKF

$\bar{\alpha}_{i,k}$ – A 3×1 vector denoting the the Euler-angle rotation increment of platform i at time step k

α_i – The drift of the clock on platform i relative to global time t equal to $\alpha_i = f_i^{\text{MO}}/f^{\text{MO}}$

α_i^k – The clock drift of platform i at iteration k of the UKF

$\hat{\alpha}_i$ – The estimate of the clock drift of platform i

$\hat{\alpha}_i^k$ – The estimate of the clock drift of platform i at iteration k of the UKF

$\hat{\alpha}_i^{z,k}$ – The measurement of the clock drift of platform i at iteration k of the UKF for the iterative synchronization approach

β_u – A distribution-based tuning parameter of the UKF – for Gaussian distributions, $\beta_u = 2$

γ_i^{rx} – The receiver carrier phase for platform i

$\hat{\gamma}_i^{\text{rx}}$ – An estimate of the receiver carrier phase for platform i

$\gamma_{i,k}^{\text{rx}}$ – The receiver carrier phase of platform i at iteration k of the UKF

$\hat{\gamma}_{i,k}^{\text{rx}}$ – The estimate of the receiver carrier phase of platform i at iteration k of the UKF

$\gamma_{i,k}^{\text{rx},e}$ – The receiver carrier phase estimate error of platform i at iteration k of the UKF

$\hat{\gamma}_{i,k}^{\text{rx},e}$ – An estimate of the receiver carrier phase estimate error of platform i at iteration k of the UKF

$\hat{\gamma}_{i,k}^{\text{rx},z}$ – The measurement of the receiver carrier phase of platform i at iteration k of the UKF for the iterative synchronization approach

γ_i^{tx} – The transmitter carrier phase for platform i

$\hat{\gamma}_i^{\text{tx}}$ – An estimate of the transmitter carrier phase for platform i

$\gamma_{i,k}^{\text{tx}}$ – The transmit carrier phase of platform i at iteration k of the UKF

$\hat{\gamma}_{i,k}^{\text{tx}}$ – The estimate of the transmit carrier phase of platform i at iteration k of the UKF

$\gamma_{i,k}^{\mathrm{rx},e}$ – The transmit carrier phase estimate error of platform i at iteration k of the UKF

$\hat{\gamma}_{i,k}^{\mathrm{rx},e}$ – An estimate of the transmit carrier phase estimate error of platform i at iteration k of the UKF

$\hat{\gamma}_{i,k}^{\mathrm{tx},z}$ – The measurement of the transmit carrier phase of platform i at iteration k of the UKF for the iterative synchronization approach

$\gamma_{i,j}^{\mathrm{err}}$ – The difference between the transmitter carrier phase of platform j and the receiver carrier phase of platform i

$\hat{\gamma}_{i,j}^{\mathrm{err}}$ – An estimate of the difference between the transmitter carrier phase of platform j and the receiver carrier phase of platform i

$\gamma_{i,j,k}^{\mathrm{err},e}$ – The difference between the transmitter carrier phase error of platform j and the receiver carrier phase error of platform i

$\hat{\gamma}_{i,j,k}^{\mathrm{err},e}$ – An estimate of the difference between the transmitter carrier phase error of platform j and the receiver carrier phase error of platform i

$\hat{\gamma}_{i,j}^{k}$ – The phase estimate of the pulse-compression component of $\tilde{r}_{i,j}^{k}(\tau_i')$

$\tilde{\gamma}_{i,j}^{k}$ – The phase estimate of the pulse-compression component of $\tilde{r}_{i,j}^{k}(\tau_i')$ ignoring the ambiguity function phase and phase due to the carrier offset and transmit time

$\check{\gamma}_{i,j}^{k}$ – The phase estimate of the pulse-compression component of $\tilde{r}_{i,j}^{k}(\tau_i')$ accounting for the phase due to the carrier offset and transmit time

Δt – The amount of time in seconds between time steps $k-1$ and k

$\Delta t_{i,j}$ – The difference between the $t_{i,j}^{\mathrm{rx}}$ and $t_{j,i}^{\mathrm{rx}}$

Δ_{TDMA} – TDMA time-slot size

$\Delta\gamma$ – Carrier phase shift of a received radar signal

ϵ_f – An arbitrarily small variance value to approximate zero variance in clock drift least-squares estimation

ϵ_γ – An arbitrarily small variance value to approximate zero variance in carrier phase least-squares estimation

ϵ_ϕ – An arbitrarily small variance value to approximate zero variance in clock bias least-squares estimation

$\zeta_{k,i}$ – The i^{th} sigma point of the UKF at iteration k transformed through the measurement model h_k

$\eta_{i,j}^{\alpha}$ – The additive noise component of $\hat{f}_{i,j}^{\alpha}$

$\eta_{i,j}^{\alpha,d}$ – The additive noise component of $\hat{f}_{i,j}^{\alpha,d}$

θ – The pitch component of the three-angle Euler angle set

$\bar{\theta}_{i,k}$ – A 3×1 vector denoting the Euler angle attitude of platform i at time-series point k

κ_u – A tuning parameter of the UKF

λ_b – Longitude

λ_c – Carrier frequency wavelength

λ_u – A tuning parameter of the UKF

ρ – An auxiliary delay variable

$\tilde{\rho}$ – An auxiliary delay variable

$\sigma_{a,i}$ – The noise standard deviation for the accelerometer on platform i

σ_f – Frequency estimation standard deviation

$\sigma_{h,i}$ – The height standard deviation of the GPS measurement for the i^{th} platform in a network

$\sigma_{L,i}$ – The latitude standard deviation of the GPS measurement for the i^{th} platform in a network

$\sigma_{R,l}$ – The range standard deviation of the l^{th} link between platforms in a network

σ_v – Velocity standard deviation

σ_α – Clock drift standard deviation

σ_γ – Phase estimation standard deviation

$\sigma_{\lambda,i}$ – The longitude variance of the GPS measurement for the i^{th} platform in a network

σ_τ – Time-delay estimation standard deviation

σ_ϕ – Clock bias standard deviation

$\sigma_{\omega,i}$ – The noise standard deviation for the gyroscope on platform p

Σ_f – The covariance matrix of the clock drift and Doppler shift frequency estimates after least-squares estimation

$\Sigma_{f,v}$ – The covariance matrix of the clock drift and relative radial velocity estimates after least-squares estimation

$\Sigma_{f,v}^k$ – The covariance matrix of the clock drift and relative radial velocity estimates after least-squares estimation using the SNR at iteration k

Σ_i^{GPS} – The covariance of the 3-dimensional GPS measurements of platform i in the ECEF frame

Σ_i^{LLH} – The covariance of the 3-dimensional GPS measurements of platform i in terms of latitude, longitude, and height

Σ_R – The covariance matrix of the range estimates

Σ_R^k – The covariance matrix of the range estimates using the SNR at iteration k

Σ_α – The covariance matrix of the clock drift estimates after least-squares estimation in the stationary case

Σ_γ – The covariance matrix of the carrier phase estimates after least-squares estimation

Σ_γ^k – The covariance matrix of the carrier phase estimates after least-squares estimation using the SNR at iteration k

Σ_ϕ – The covariance matrix of the clock bias estimates after least-squares estimation

Σ_ϕ^k – The covariance matrix of the clock bias estimates after least-squares estimation using the SNR at iteration k

τ – The time variable generally used to refer to slow-time in a radar signal

τ_i – The local time variable for platform i taking into account the clock drift and bias

τ_i' – The local time variable for platform i taking into account the clock drift and bias after axis adjustment to compensate for clock drift

$\tau_{i,j}^{A,k}$ – The time shift of the pulse-compressed component of $\tilde{r}_{i,j}^k(\tau_i')$ due to the frequency shift as dictated by the ambiguity function $A_j(t, f)$

$\hat{\tau}_{i,j}^k$ – The time-delay estimate of the pulse-compression component of $\tilde{r}_{i,j}^k(\tau_i')$

$\tilde{\tau}_{i,j}^k$ – The time-delay estimate of the pulse-compression component of $\tilde{r}_{i,j}^k(\tau_i')$ compensated to remove the transmit time and frequency-shift delay

$\tau_{i,j}^{\mathrm{rx}}$ – The receive time at platform i of the signal transmitted by platform j according to the local time of receiving platform i

τ_j^{tx} – The transmit time of platform j according to the local time of transmitting platform j

ϕ – The roll component of the three-angle Euler angle set

ϕ_i – The clock bias of platform i

ϕ_i^k – The clock bias of platform i at iteration k of the UKF

$\hat{\phi}_i$ – An estimate of the clock bias of platform i

$\hat{\phi}_i^k$ – The estimate of the clock bias of platform i at iteration k of the UKF

$\phi_i^{e,k}$ – The clock bias error for platform i on iteration k

$\hat{\phi}_i^{e,k}$ – The estimated clock bias error for platform i on iteration k

$\phi_{i,j}^{e,k}$ – The clock bias error difference between platforms i and j on iteration k

$\hat{\phi}_{i,j}^{e,k}$ – An estimate of the clock bias error difference between platforms i and j on iteration k

$\hat{\phi}_{i}^{z,k}$ – The measurement of the clock bias of platform i at iteration k of the UKF for the iterative synchronization approach

$\tilde{\phi}_i$ – The network mean-adjusted clock drift for platform i

$\phi_{i,j}$ – The drift estimate scaled clock bias difference between platforms i and j

$\hat{\phi}_{i,j}$ – An estimate of the drift estimate scaled clock bias difference between platforms i and j

$\chi_{k,i}$ – The i^{th} sigma point of the UKF at iteration k

$\chi_{k,i}^{f}$ – The i^{th} sigma point of the UKF at iteration k transformed through the state model f_k

ψ – The yaw component of the three-angle Euler angle set

$\bar{\omega}_{i,k}$ – A 3×1 vector denoting the body-frame angular velocity, as measured by a gyroscope, of platform i at time-series point k

$\mathbf{\Omega}_e$ – The angular-rate rotation matrix for the Earth-rotation vector

$\mathbf{\Omega}_{i,k}$ – The rotation matrix corresponding to $\bar{\theta}_{i,k}$

$\mathbf{\Omega}_{i,k}^{a}$ – The rotation matrix corresponding to the transformation of the specific force measured in the body frame to the inertial frame for platform i at time step k

$\mathbf{\Omega}_{i,k}^{\delta}$ – The rotation matrix corresponding to the transformation of the body frame at the end of the navigation update step to the beginning for platform i at time step k

$\mathbf{\Omega}_{\text{ned}}$ – The rotation matrix to convert from the NED frame to the ECEF frame

Appendix B

Abbreviations

The following abbreviations are used:

ADC	Analog-to-digital converter
AWGN	Additive white Gaussian noise
CKF	Cubature Kalman filter
CPI	Coherent processing interval
CRLB	Cramér-Rao Lower Bound
DAC	Digital-to-analog converter
DFT	Discrete Fourier transform
DGPS	Differential GPS
ECEF	Earth-centered Earth-fixed
ECI	Earth-centered inertial
EKF	Extended Kalman filter
EW	Electronic warfare
FFT	Fast Fourier transform
FPGA	Field-programmable gate array
GPS	Global positioning system
IMU	Inertial measurement unit
INS	Inertial navigation system
ISLR	Integrated sidelobe level ratio
LFM	Linear frequency modulated/modulation
LiDAR	Light detection and ranging
LO	Local oscillator
LOS	Line-of-sight
LPF	Low-pass filter
MIMO	Multiple-input multiple-output

MMSE	Minimum mean squared error
NED	North-east-down
NL-LS	Non-linear least-squares
OLS	Ordinary least-squares
PDF	Probability density function
PF	Particle filter
ppb	Parts per billion
ppm	Parts per million
PPS	Pulse-per-second
ppt	Parts per trillion
PRF	Pulse repetition frequency
PRT	Pulse repetition time
PSD	Power spectral density
PSLR	Peak sidelobe level ratio
PSR	Point spread response
RF	Radio frequency
RMSE	Root-mean-squared error
RPV	Relative peak value
RTK	Real-time kinematic
RX	Receiver
SAR	Synthetic aperture radar
SDR	Software-defined radio
SNR	Signal-to-noise ratio
TDMA	Time-division multiple-access
TOA	Time-of-arrival
TOF	Time-of-flight
TWTT	Two-way time transfer
TX	Transmitter
UAV	Unmanned aerial vehicle
UKF	Unscented Kalman filter
WLS	Weighted least-squares