3D INDOOR STATE ESTIMATION FOR RFID-BASED MOTION-CAPTURE SYSTEMS

A Dissertation
Presented to
The Academic Faculty

By

Qian Yang

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology

December 2021

3D INDOOR STATE ESTIMATION FOR RFID-BASED MOTION-CAPTURE SYSTEMS

Thesis committee:

Dr. Gregory D. Durgin
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Andrew F. Peterson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. David G. Taylor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Mary Ann Weitnauer
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Ying Zhang
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Billy Kihei
School of Electrical and Computer Engineering
Kennesaw State University

Date approved: August 10 2021

ACKNOWLEDGMENTS

I want to thank my advisor, Prof. Gregory Durgin, for his kind help, unreserved support, efforts and devotions throughout the journey. Without Prof. Gregory Durgin, it's impossible for me to reach the shore and make progresses at every single step to make this journey possible. I learned a great deal from him, not just about the research, but also enthusiasm and dedications to RF community, kindness to help others, and viewing every result as a big achievement in progress. He considers all of his students part of his family and does everything he can to help each student. He saves me from seeking study and changes my way of doing things. I am super grateful.

I would also like to my advisor, Prof. David Taylor for his kind guidance, patient advising, detailed and devoted instructions in research thinking, derivation, and explorations. I have leaned a lot from him in conducting research in a scientific way. His rigorousness and kindness totally changes my mind and way of thinking, impacting my life and being meaningful throughout the whole career and life. I'm also touched by his kindness behind his preciseness. It's amazing to have so many improvements in such limited time. I have been impressed by the gaining from continuous reading, working, and exploration following each rule. I am super grateful.

I am extremely grateful to my committee members, Prof. Ying Zhang, Prof. Andrew Peterson, Prof. Mary Ann Weitnauer and Prof. Billy Kihei for their help and kindness. Prof. Billy Kihei is super kind, nice, and have taught me the whole philosophy and the broad view. I like Prof. Mary Ann Weitnauer a lot and thanks her very much for her kindness and warm help, really appreciate time together and with her student Qiongjie Lin. Thanks her for providing the 3D positioner in the experiment. Prof. Andrew Peterson has nicely asking how I am doing throughout the year and his kindness lights up days. Prof. Zhang helps me a lot and her kindness and warm heart help me pull out from the difficulties. Time flies and so little time for being together.

I owe a great deal of thanks to my colleagues from the Propagation Group for their support and assistance throughout my study. Thanks very much for Mohamad Bashir Akbar's data and setup files, making this topic possible. Thanks a lot for Cheng Qi for his time, readers, and kind help in the measurements. Thanks Mohammad Alhassoun, Michael Varner, and Francesco Amato, three outstanding personalities for their kind support. I'm also thankful to my great roommate Chenyu Li, my friend Xia Ma for over six years company and getting along. Thanks my friends from GT Shenzhen and staff from ECE department. Thanks Yuting Hu, YiFei Fan, Yirui Li, Han Yu, Ming Liu for their valuable friendships. I'm also grateful for NCR for almost three years of support.

I would like to thank Polaris Wireless for providing the 3D localization research in the Covid-19 pandemic that matches the topic. Thanks Jet Zhu very much for his continuous help and valuable suggestions at every step in the project. It's a very enjoyable and successful collaboration, which I really appreciate the trust and working together. I hope to be helpful in future in this domain.

Finally, I am grateful to most valuable people, my family of my life. Thanks for giving me life and give unreserved love and freedom to pursue life I want with fully support both financially and emotionally. They're always there to listen and help. I'm very grateful for the freedom to choose whatever I want to study and whichever career I want to pursue, making my life of no regrets and everyday the best. Not only the support to pursue dream but also mind in education and attitudes toward life help me stand by myself in the world. Feeling very lucky to be born in the family surrounded by unconditional support and love at every stage. There is no way to acknowledge and thank them for their countless support.

It seems that throughout difficulties and is finally luckily caught by kindness, making myself the luckiest in twists and turns. I'd like to thank friends, teachers, and people unknown bringing lots of miracles into my life.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

In this research, 3D indoor state estimation for RFID-based motion-capture systems is built. The state estimation is based on sensor fusion by combining RF signal with IMU data. The estimation handles both LOS and multipath indoor environment. In this way, state-space model for sensor fusion and nonlinear state estimation methods are built.

As for the results, in 3D motion with indoor multipath, RMS error before estimation is 71.99 cm, in which 34.99 cm in xy- plane and 62.92 cm along z- axis. After NLE estimation using RF signal combined with IMU data, RMS error of 3D coordinates decreases to 31.90 cm, with 22.50 cm in xy- plane and 22.61 cm along z- axis, achieving a factor of 2 enhancement which is similar to the 2D estimation. In addition, using RF signal only obtains similar estimation results to using both RF and IMU, i.e., 3D RMS error of 31.90 cm, where 22.48 cm in xy- plane and 22.62 cm along z- axis. Hence, RF signal only is able to achieve fine-scale RFID-based motion capture in 3D motion, in consistency with the conclusion arrived at in 2D estimation. In this way, RFID-based motion capture systems can be simplified from embedding inertial sensors. EKF derives close results with 2 cm larger RMS error. In addition, ToF based position sensor in tracking achieves comparable and higher accuracy compared to RSS based position sensor based on the multipath simulation model, enabling ToF to be applied in fine-scale motion capture and tracking.

In summary, fine-scale 3D indoor localization for RFID-based motion capture systems is achieved. In addition, RF signal only is able to achieve fine-scale localization in both LOS and multipath indoor environment based on sensor fusion estimation, which simplifies RFID-based motion capture systems.

CHAPTER 1

INTRODUCTION AND BACKGROUND

RFID-Based motion capture has lots of applications, for instance, real-time tracking of NFL players, obtaining not only location, acceleration, distance, but also orientation in the motion [1] as shown in Figure 1.1. It can also be used in supply chain management [2], tracking and managing people and personnel safety and security [3], and gaming and motion capture for body movement [4].

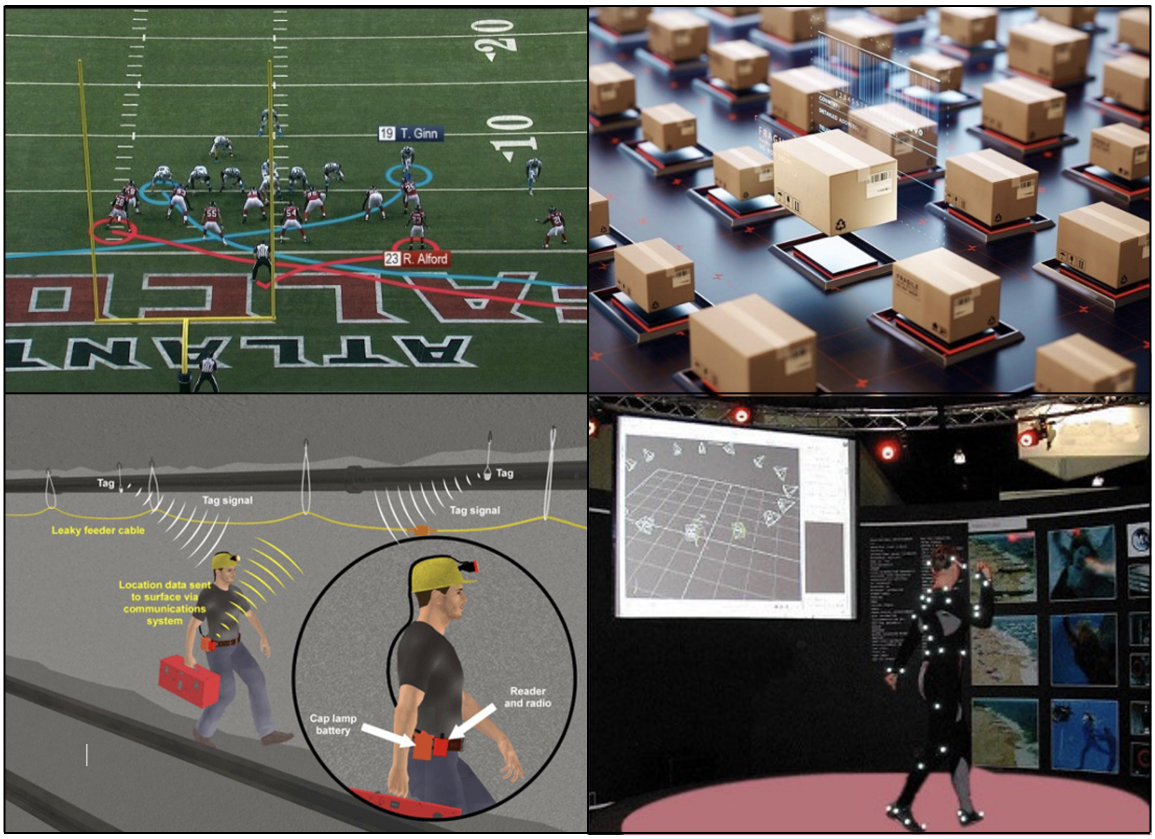


Figure 1.1: RFID-Based Motion Capture Applications. Real-time Tracking of NFL players [1], Supply Chain Management [2], People Safety and Security [3], Gaming and Motion Capture [4].

Motion capture systems need centimeter-level accuracy with good spatial and temporal resolution and hence state estimation is needed. RFID-based localization system may be

affected by small scale fading and multi-path effects in the indoor environment.

1.1 Summary of Localization and Motion Capture Systems

Localization and motion capture techniques mainly include four categories: electromagnetic measurement systems (EMS), optoelectronic measurement systems (OMS), ultrasonic localization systems (UMS) and inertial sensory systems (IMU) [5]. OMS includes Active Badge [6] and indoor GPS (iGPS) [7], which depend on line-of-sight (LOS) and will be interrupted when the markers are blocked or out of sight. UMS like Cricket Location Support System [8] and Active Bat location system [9] require a great deal of infrastructure including ultrasonic sensors and scalability. IMU [5] consisting of an accelerometer, gyroscope, and often a magnetometer [10] are most commonly used in short-range measurements. In addition, an IMU could be used to determine position by performing a double integration but will suffer from large integration drifts over even modest time intervals.

Table 1.1: Summary of Localization and Motion Capture Systems

Localization and Motion Capture Systems			
EMS	**OMS**	**UMS**	**IMU**
Large Volume	LOS	Lots of	Short Range
LOS		Infrastructure	Integration Drift
NLOS			

EMS solutions, especially RFID-based localization systems, have better detection range, blockage transmission, and flexible layout compared to OMS, UMS and IMU. RFID system provides large capture volume and no line-of-sight (NLOS) is necessary to find the positions of the transponders. Therefore, EMS is suitable for large measurement volumes and situations where occlusions are common. They can identify targets without visualization within clutter and through walls [11]. Out of these characteristics, RFID enabled robots not only to do human things, but you can also enable them to do superhuman things like identifying behind walls in a scalable way [11]. Also, RFID can be designed as non-

intrusive and does not have as many privacy concerns.

1.2 RFID-Based 3D Localization System

Various systems have been built for RFID-Based Indoor 3D Localization System. The first RFID-based 3D localization scheme SpotON uses most classic trilateration method with signal strength [12]. 3D-BATL utilizes four readers that are placed on the vertices of a tetrahedron [13]. A fingerprint approximate approach integrates hundreds of readers on the ceiling and floor as reference positions [14]. Holographic image is generated from phase values measured by one antenna at multiple locations [15]. The height is derived by spatial domain phase difference. The based hologram method of holographic image is originally proposed in [16].

1.2.1 3D Space Beams from Phase Array Antenna

In mWaveLoc system, 4 $\times$ 8 element phased-array antenna generates beams in 3D space for directional communication of millimeter-wave networks to achieve centimeter-level 3D localization of Access Point (AP)-client distance estimation in LOS settings [17]. The system is based on time-of-flight.

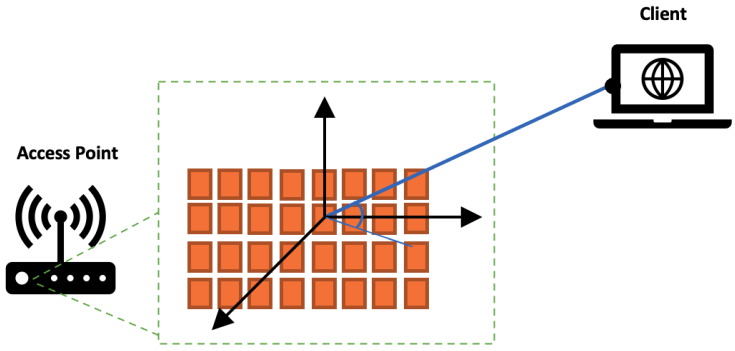


Figure 1.2: Scheme of mWaveLoc System. 4 $\times$ 8 Element Phased-Array Antenna Generates Beams in 3D Space Between Access Point and Client [17].

1.2.2 Triangulation Based on Multiple Base Stations

WiTrack tracks the 3D motion of user without carrying device using radio reflected from human body in occlusion scenarios based on time-of-flight [18]. WiTrack shows that 3D localization can translate into an intersection of three ellipsoids, which uses directional antennas: one for transmitting and three for receiving reflection off human body. A 3D real-time indoor localization system using broadband nonlinear backscatter in passive devices achieves centimeter-precision. The system utilizes one broadband transmitter antenna of fundamental frequency and four receiver antennas radiating the second harmonic signals [19].

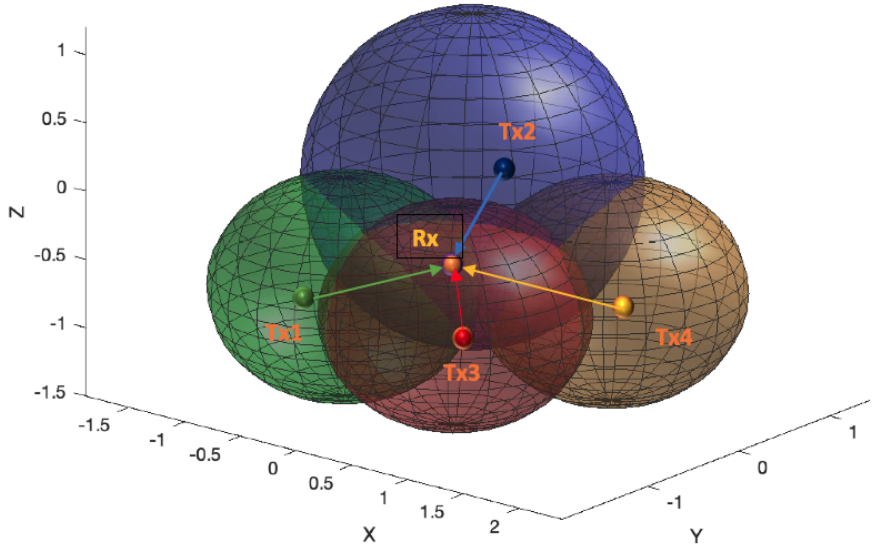


Figure 1.3: Scheme of 3D Localization Based on Four Antennas: Tx1, Tx2, Tx3, Tx4 as Base Stations for One Target Antenna Rx.

1.2.3 Anchor Sensor Nodes or Access Points

In umbrella, moving target nodes are attained using single reference node (access point) in an anisotropic network. The distance between the reference node and target node is calculated based on RSS. The umbrella projects virtual anchor nodes for better centroid

localization as shown in Figure 1.4 [20].

Given a set of RFID tags and/or readers deployed as reference points at known locations in a hexahedron, passive or active methods can be used for localization in as DeB [21]. In the active scheme of DeB, a set of reference tags with known coordinates are placed on two or more planes in any place of the space. The target reader activates passive tags with power at different levels without using RSS or phase information as shown in Figure 1.4.

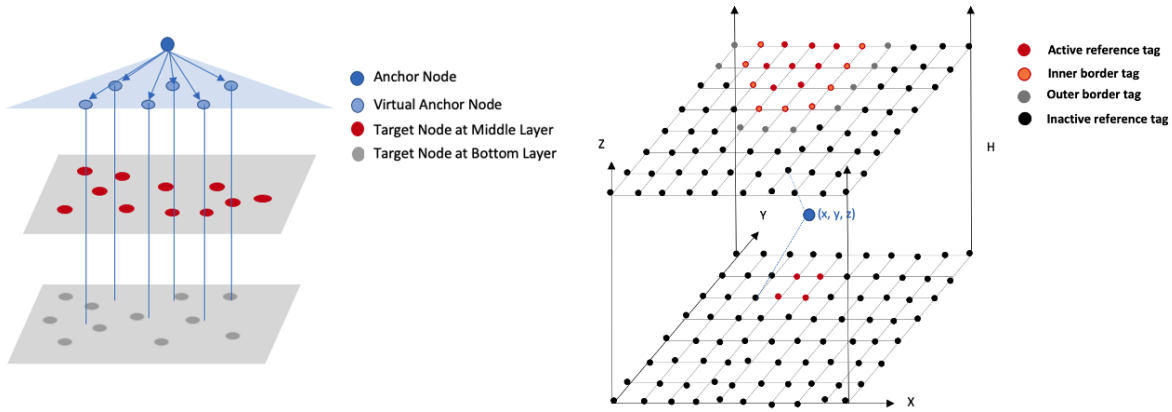


Figure 1.4: Left: Umbrella. Moving Target Nodes are Attained Using Single Reference Node as well as Projected Virtual Nodes [20]. Right: DeB System. Reader with Unknown Coordinates (x, y, z) Activate Reference Tags of Known Coordinates for Localization [21].

1.2.4 Indoor Localization Systems

Traditional solutions for location-based services use 802.11, visible light, or acoustics. In addition, RF-Compass navigates robot, recognizes, and grabs objects by dividing space into partitions based on RF signals [22]. Moreover, radio map for device free passive localization uses WLAN signal strength [23]. Prototypes use "fusion" algorithms to combine data from both radios and IMUs sometimes magnetometers, pressure sensors, and digital compasses together. Also, indoor localization using smartphone has been studied in [24]. 3D localization of unknown sensor using RSS from static and moving sensors with particle filters is built in [25]. Inertial sensor based localization has been investigated in [26, 27, 28].

1.3 Three Principal Techniques for Localization

Localization techniques may employ individually or in combination [29].

1.3.1 Triangulation, Scene Analysis, and Proximity

Triangulation, scene analysis, and proximity are the three principal techniques for automatic location-sensing [29]. The three principal techniques are summarized in Table 1.2.

Table 1.2: Three Principal Techniques for Localization

Principal Techniques	Division	Subdivision	Description	Example
Triangulation	Lateration	Direct	Direct distance	Tape measure
		Time-of -Flight	Travelling time	GPS, Active Bat [9] Cricket [8]
		Attenuation	Signal decreases with distance	Radio signal
	Angulation	-	Determine position using angle	Phased antenna arrays
Scene Analysis	Static Scene	Looked up features in predefined dataset to map		MSR RADAR [30] LANDMARC [31]
	Differential Scene	Tracks the difference between successive scenes		
Proximity	Detecting Physical Contact			Pressure sensors
	Monitoring Wireless Cellular Access Points			Active Badge [6]
	Observing Automatic ID Systems			Highway E-Toll systems

The triangulation location-sensing technique uses the geometric properties of triangles to compute object locations, which includes Lateration that computes the position of an object by measuring its distance from multiple reference positions and Angulation that uses primarily angle or bearing measurements like direction of arrival (DOA) [32]. Two challenges of time-of-flight in Lateration are: synchronization and indirect paths by reflections. Challenge for attenuation in Lateration is signal propagation issues such as reflection, refraction, and multipath in environments with many obstructions [29].

Scene analysis location-sensing technique uses features of a scene observed from a particular vantage point to draw conclusions about the location of the observer or the object in the scene [29]. Therefore, the location of objects can be inferred using passive observation and features that do not correspond to geometric angles or distances. At the same time, changes to the environment that alters the perceived features of the scenes may necessitate reconstruction of the predefined dataset or retrieval of an entirely new dataset [29].

A proximity location-sensing technique entails determining when an object is ”near” a known location. An example of monitoring wireless cellular access points is a mobile device in range of one or more access points in a wireless cellular network like Active Badge [6, 29].

1.3.2 Sensor Fusion and HIMR System

RFID systems can also be fused with other sensors to enhance accuracy. RFID information is combined with laser data in [33] to estimate the pose of the robot and localize tags. RFID signal is fused with location estimations from a wideband rebounding signal comprising multiple frequencies to track the distance of tag and reconstruct position of tag in the 3D space [11]. RFID RSS image is combined with camera and LIDAR to estimates the 3D location of the object [34]. Moreover, an RFID system is fused with an ultrasonic sensor to partially remove the uncertainties of RFID systems by using distance data obtained from ultrasonic sensors for mobile robot localization [35]. In addition, a wearable prototype that fuses the drift-sensitive IMU with a RFID tag reader in [36] corrects positional errors. Activity recognition system combining RFID and accelerometers shows that accuracy can be significantly improved [37]. Higher level of combining RF signal with images is employed in Camera-Radar RODNet [38].

HIMR combines RF signals with inertial sensor information in microwave range for motion capture and localization, which provides better and finer accuracy in the range of 1mm-20mm in 1D and 2D motion as compared to other schemes [39].

1.4 Reducing Multipath Effects in RFID-Based Indoor Localization

1.4.1 Decreasing Noise Level of Received Signal Strength (RSS)

Noise spectrum of stationary RSS is clustered mostly around DC level [40]. To decrease the noise level of RSS from scattering, multipath, and delay spread of indoor propagation, Hanning window with a sharp cutoff frequency response, finite impulse response low-pass filter, and Savizky-Golay smoothing filters known as polynomial smoothing have been applied to decrease noise level of RSS.

1.4.2 Using Received Signal Phase (RSP) Difference in Replace of Received Signal Strength

To achieve better accuracy, robustness, and sensitivity for localization, phase difference in replacement of RSS between two or more receiving antennas is used [41].

1.4.3 Frequency Modulated Continuous Wave (FMCW)

Frequency Modulated Continuous Wave (FMCW) transmits a narrowband signal, e.g., a few KHz whose carrier frequency changes linearly with time. To identify the distance from a reflector, FMWC compares the carrier frequency of the reflected signal to that of the transmitted signal [18].

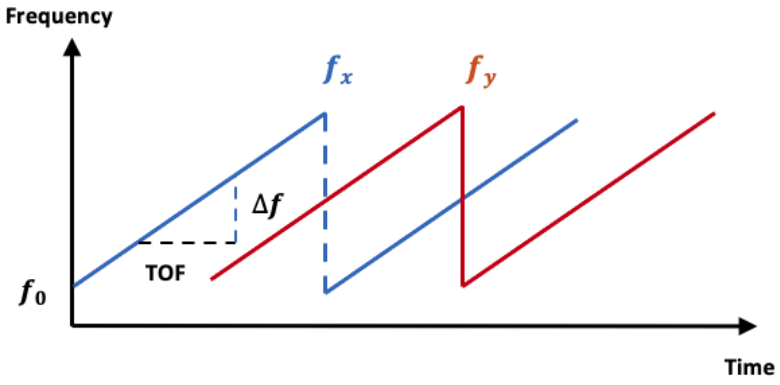


Figure 1.5: Scheme of Frequency Sweeping w.r.t. Time. The Frequency Difference at the Receiver Can Be Transferred to ToF. Sweeping Frequencies Along x and y Axes are f_x and f_y respectively [18].

1.4.4 Heuristic Multi Frequency Continuous Wave (HMFCW)

In order to achieve accurate ranging, broad bandwidth can help tolerate multi-path phase errors. Such trade-off has been exploited in FMCW radar and Ultra Wideband (UWB) systems [42]. A Heuristic Multi Frequency Continuous Wave (HMFCW) ranging was proposed in [43]. In order to tolerate large phase error while solving phase cycle integers correctly, optimal utilization of the broad bandwidth which is enabled by nonlinear backscatter tag and reader design is applied [19].

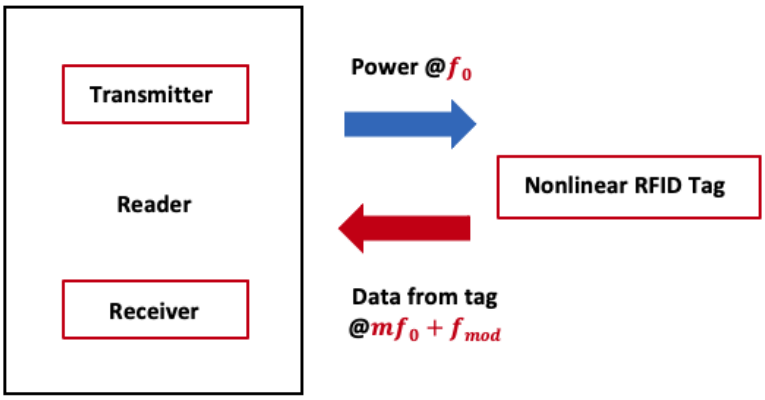


Figure 1.6: Scheme of HMFCW. The Optimal Frequency Sequence is Sent to the Nonlinear RFID Tag at Fundamental Frequency f_0 in the Downlink. The Second Harmonic Signal $2f_0$ was Sent to Receiver in the Uplink [19].

1.5 State Estimation for Nonlinear Systems

Motion-capture installations need centimeter-level spatial positioning and precise temporal resolution [29]. However, when the distance between the base station and the transponder increases, noise increases and the quality of the signal decreases [5, 29, 44].

When the state of a system is estimated from noisy sensor information, state estimator is employed to fuse together data from different sensors to produce an accurate estimate of the true system state. When the system dynamics and observation models are linear, the minimum mean squared error (MMSE) estimate may be computed using Kalman filter.

Table 1.3: Summary of State Estimation Methods for Nonlinear Systems

1. Nonlinear Kalman Filter	
Linearized Kalman Filter:	Expand around nominal on priori guesses of system trajectory.
Extended Kalman Filter:	First-order Taylor approximations of state and observation equations.
Higher-Order Approaches:	More refined linearization for highly nonlinear systems.
2. Unscented Kalman Filter	
Nonlinear Transformation:	Mean and Variance: Unscented transformation approximation.
3. Nonlinear Least-Squares Estimation	
Gauss-Newton:	Q-quadratically convergent to the root of nonlinear equation with good start.
Levenberg-Marquardt:	Interpolate Gauss-Newton and Gradient Descent to explore start value.
4. Partical Filter:	
Bayesian state estimation based on conditional relative likelihood of each particle (state vector) with preknown pdf of measurement noise.	

However, in most applications of interest, the system dynamics and observation equations are nonlinear [45]. State estimators for nonlinear systems are derived based on polynomial approximations with multi-dimensional interpolation formula [46].

The optimal solution to the nonlinear filtering problem requires maintaining a complete description of the conditional probability density. Unfortunately this exact description requires a potentially unbounded number of parameters (such as moments) and a number of suboptimal approximations have been proposed [47] as summarized in Table 1.3.

1.5.1 Linearized Kalman Filter

More specifically, linearized Kalman filter uses Taylor series to expand state and measurement equations around a nominal control, state, nominal output, and nominal noise values. These nominal values are based on a priori guesses of the system trajectory.

1.5.2 Extended Kalman Filter

Since nominal state trajectory is not straightforward, the most widely used filter, the Extended Kalman Filter (EKF) uses sort of bootstrap method: linearize the nonlinear system around Kalman filter estimate and Kalman filter estimate is based on the linearized system [48]. Each linear step is replaced by its linearized equivalent [45]. The EKF is also based on first-order Taylor approximations of state transition and observation equations [46].

1.5.3 Higher-Order Approaches

More refined linearization techniques including iterated and second-order EKF are used to reduce the linearization error in EKF for highly nonlinear systems.

1.5.4 Unscented Kalman Filter

Unscented Kalman filter (UKF) is an extension of Kalman Filter that reduces the linearization errors of the EKF, using unscented transformation that approximates mean and variance change in a nonlinear transformation [48].

1.5.5 Nonlinear Least-Squares Estimation

Least square methods including weighted least square, recursive least square estimations, and Wiener filtering have been applied to state estimation [48]. Nonlinear least-squares estimation (NLE) proposed in [49] is applied to nonlinear state estimation in a 2D motion. NLE is developed from the perspective of nonlinear least-squares optimization and the map-inversion philosophy based on discrete-time dynamics of the system [49].

Nonlinear equations can be solved using Newton's method based algorithms called Newton-like or quasi-Newton [50]. Newton's method is Q-quadratically convergent for most problems, although it will not necessrily achieve global convergence [50]. The Levenberg-Marquardt method interpolates between the Gauss–Newton algorithm and the gradient descent method, shares with the gradient methods their ability to converge from an initial

guess which may be outside the region of convergence and also shared with the Taylor series methods the ability to converge rapidly in the vicinity of the converged value [51].

1.5.6 Particle Filter

The Particle filter is a probability-based estimator derived based on Bayesian approach. State vectors called particles are generated randomly. At each time stamp, particles are propagated to the next time step. Based on measurement, conditional relative likelihood of each particle is evaluated based on pdf of measurement noise, which increases the computational effort of Particle filtering [48].

CHAPTER 2

RFID LOCALIZATION SYSTEM MODEL

In this chapter, we built a model for the received backscatter signals for semi-passive switched modulator and semi-active tunnel diode modulator to be used in simulation. The model is built based on the link budgets for backscatter-radio and RFID systems [52].

2.1 Localization System Model Based on Semi-Passive Switched Modulator Tag

According to the Friis transmission equation, power received by a reader from another tag in free space is

$$P_R(r) = \frac{P_T G_T G_R G_t^2 \lambda^4 X^2 M}{(4\pi r)^4} \tag{2.1}$$

where P_T is reader's transmitted power, G_T is reader's transmitter antenna gain, G_R is reader's receiver antenna gain, G_t is tag's antenna gain, λ is carrier frequency wavelength, X is the polarization mismatch factor, M is tag's modulation factor, and r is the reader-tag separation in radial distance. Additional parameters such as on-object antenna and path blockages have been omitted, but can be re-introduced for more complicated studies [52].

In addition to the signal model, the round-trip phase of the channel must also be calculated. The reader receives a complex baseband signal after demodulation that can be characterized as [39]

$$S(r) = H(r) \exp\left(-j\left[\frac{4\pi r}{\lambda} + \psi_0\right]\right) = I + jQ \tag{2.2}$$

where ψ_0 is the accumulated phase offset, which is assumed to be a constant in this simulation. I is the in-phase channel and Q is the quadrature-phase channel of the back-scattered

signal. Therefore, the magnitude and phase of the measured signal can be calculated as

$$H(r) = \sqrt{I^2 + Q^2} \tag{2.3}$$

$$\psi = \frac{4\pi r}{\lambda} = \begin{cases} \tan^{-1}\left(\frac{Q}{I}\right) & , \ I \geq 0 \\ \pi + \tan^{-1}\left(\frac{Q}{I}\right) & , \ I < 0 \end{cases} \tag{2.4}$$

Furthermore, the signal amplitude $H(r)$ and received power $P_R(r)$ are related by

$$P_R(r) = \frac{H^2(r)}{2R} \tag{2.5}$$

where R is the RF impedance of the receiver hardware. Hence, the transformation from signal amplitude $H(r)$ to reader-tag separation r is

$$r = \sqrt[4]{\frac{2P_T G_T G_R G_t^2 \lambda^4 X^2 M R}{(4\pi)^4 H(r)^2}} \tag{2.6}$$

Equation 2.6 allows an estimate of tag position based on signal strength alone. Additionally, phase difference provides an estimate of tag velocity, where $\Delta\psi$ is phase difference over time interval Δt

$$\dot{r} = \frac{\lambda}{4\pi}\frac{\Delta\psi}{\Delta t} \tag{2.7}$$

2.2 Localization System Model Based on Semi-Active Tunnel Diode Modulator Tag

For tunnel diode modulator tag, the magnitude and phase of received signal relates to normal tag as

$$H_t(r) = \frac{H(r)}{|H(r)|} H(r) \tag{2.8}$$

$$\psi_t = \psi = \frac{4\pi r}{\lambda} \tag{2.9}$$

Power received by reader from the tunnelling tag backward in free space is

$$P_t(r) = \frac{P_T G'_T G'_t \lambda^2 X'}{(4\pi r)^2} \tag{2.10}$$

where P_T is reader's transmitted power, G'_T is reader's transmitter antenna gain, G'_t is tunnelling tag's antenna gain, λ is carrier frequency wavelength, X' is the polarization mismatch factor, and r is the reader-tag separation.

Out of saturation at the diode, power received by tunnelling tag from a reader in the downlink in free space is

$$P_R(r) = \frac{P_T G'_T G'_t \lambda^2 X' M}{(4\pi r)^2} \tag{2.11}$$

Regardless of incident power, reflected power starts at P_{tl}. Hence, reader received power is inversely proportional to second order of distance r. The modulation factor $M = \frac{1}{4}|\Gamma_1 - \Gamma_2|^2$ can be greater than 1 [53]. A quantum tunneling-based RFID tag achieves free-space long-range backscattering communication links as wide as 1.2 km.

The reader receives a complex baseband signal after demodulation that can be charac-

terized as [39]

$$S(r) = H_t(r) \exp\left(-j\left[\frac{4\pi r}{\lambda} + \psi_0\right]\right) = I + jQ \tag{2.12}$$

where ψ_0 is the accumulated phase offset, which is assumed to be a constant in this simulation. I is the in-phase channel and Q is the quadrature-phase channel of the back-scattered signal. Hence, the transformation from signal amplitude $H_t(r)$ to reader-tag separation r is

$$r = \frac{\sqrt{2P_T G'_T G'_t \lambda^2 X' R M}}{4\pi H_t(r)} \tag{2.13}$$

Equation 2.13 allows an estimate of tag position based on signal strength alone.

In the experiment of 3D motion in Figure 6.1 and Figure 6.2, to avoid interference among readers, four readers operate independently on four different frequencies 5.80 GHz, 5.82 GHz, 5.83 GHz, and 5.85 GHz in 3D motion. Then received signal amplitudes and phases from four readers are identified based on Equation 2.13 and Equation 2.7 to derive radial distances and velocities. Afterwards, the derived radial distances and velocities are input into estimation algorithm to get position and velocity of the tag as described in the following chapters.

2.3 Noise Distributions for Magnitude and Phase of Received Baseband Signal

In this section, we model the noise distribution for the received backscatter signals to be used in simulation. Assume that the thermal noise in channels I and Q is independent, identically-distributed (i.i.d.) white Gaussian noise with variance σ^2 [39]. Then the pdf of the measured signal magnitude follows Rician distribution [54]

$$f_H(x) = \frac{x}{\sigma^2} \exp\left(-\frac{(x^2 + H^2)}{2\sigma^2}\right) I_0\left(\frac{xH}{\sigma^2}\right) \tag{2.14}$$

Rician factor K proposed in [54] indicates the ratio of power in dominant multi-path to the power in remaining nonspecular multi-path

$$K = \frac{H^2}{2\sigma^2} \tag{2.15}$$

In simulation, $K-$factor can be calculated from signal amplitude $H(r)$ based on the tag-reader distance r and σ. At low SNR and small K value, $f_H(x)$ follows Rayleigh distribution. When $K \gg 1$ ($H \gg \sigma$) with high SNR, I_0 can be approximated as

$$I_0\left(\frac{xH}{\sigma^2}\right) \approx \sqrt{\frac{\sigma^2}{2\pi xH}} \exp\left(\frac{xH}{\sigma^2}\right) \tag{2.16}$$

Therefore, f_H is approximated as Gaussian distribution [54]

$$f_H(x) \approx \sqrt{\frac{x}{2\pi H\sigma^2}} \exp\left(-\frac{(x-\sqrt{\sigma^2+H^2})^2}{2\sigma^2}\right) \tag{2.17}$$

$$\approx \frac{1}{\sigma\sqrt{(2\pi)}} \exp\left(-\frac{(x-H)^2}{2\sigma^2}\right) \tag{2.18}$$

Different from previous derivation [39], a transformation of lower complexity to the log domain [55] is proposed

$$f_{\frac{Q}{I}}(I,Q) \approx \frac{\mu_Q + N(0,\sigma_Q^2)}{\mu_I + N(0,\sigma_I^2)} = \frac{\mu_Q}{\mu_I} \frac{1+\frac{N(0,\sigma_Q^2)}{\mu_Q}}{1+\frac{N(0,\sigma_I^2)}{\mu_I}} \tag{2.19}$$

Since $\log_e(1+\delta) = \delta - \frac{\delta^2}{2} + \frac{\delta^3}{3} + \cdots$ and $\arctan(x) \approx \frac{\pi}{4}x$, pdf of phase distribution can be characterized as

$$f_\psi(I,Q) = \frac{\pi}{4} e^{\log_e\left(f_{\frac{Q}{I}}(I,Q)\right)} \tag{2.20}$$

$$\approx \frac{\pi}{4} e^{\log_e\left(\frac{\mu_Q}{\mu_I}\right) + \frac{N(0,\sigma_Q^2)}{\mu_Q} - \frac{N(0,\sigma_I^2)}{\mu_I}} \tag{2.21}$$

$$= \frac{\pi}{4} e^{\log_e\left(\frac{\mu_Q}{\mu_I}\right) + N\left(0, \frac{\sigma_Q^2}{\mu_Q^2} + \frac{\sigma_I^2}{\mu_I^2}\right)} \tag{2.22}$$

$$= \frac{\pi}{4}\frac{\mu_Q}{\mu_I} e^{N\left(0, \frac{\sigma_Q^2}{\mu_Q^2} + \frac{\sigma_I^2}{\mu_I^2}\right)} \approx \psi e^{N\left(0, \frac{\sigma_Q^2}{\mu_Q^2} + \frac{\sigma_I^2}{\mu_I^2}\right)} \tag{2.23}$$

Therefore, f_ψ is approximated as phase multiplied by exponential order of Gaussian distribution.

In simulation, received signal amplitude $H(r)$ and phase ψ are generated following above distributions. The noise distributions for magnitude and phase are on the basis of the thermal channel noise assumption. If the dominant channel noise of RFID system is not thermal, simulation results may deviate from measurements.

2.4 Noise Distributions for Baseband Signal on Tunnel Diode Modulator Tag in 3D Motion

Out of the magnitude change, the reader receives a complex baseband signal after demodulation that can be characterized as [39]

$$S(r) = H_t(r)\exp\left(-j\left[\frac{4\pi r}{\lambda} + \psi_0\right]\right) = I + jQ \tag{2.24}$$

where ψ_0 is the accumulated phase offset, which is assumed to be a constant in this simulation. I is the in-phase channel and Q is the quadrature-phase channel of the back-scattered signal.

After sending to and reflecting back from the tunnel diode modulator tag in the 3D localization, two transmissions are applied in power as

$$S(r)^2 = \left[H_t(r)\exp\left(-j\left[\frac{4\pi r}{\lambda}+\psi_0\right]\right)\right]^2 = H_t^2(r)\exp\left(-2j\left[\frac{4\pi r}{\lambda}+\psi_0\right]\right) \quad (2.25)$$

Then the pdf of the quared measured signal magnitude follows square of Rician distribution [54]

$$f_H^2(x) = \left[\frac{x}{\sigma^2}\exp\left(-\frac{(x^2+H^2)}{2\sigma^2}\right)I_0\left(\frac{xH}{\sigma^2}\right)\right]^2 \quad (2.26)$$

Hence the square root of measured magnitude follows Rician distribution. The K factor of Rician distribution is derived in subsection 2.4.1.

2.4.1 K Factor Calculation Based on Moments Method

The moments of Rician distribution, expressed in terms of σ^2 and K are given by [56]

$$\mu_n = E[R^n(t)] = (\sigma^2)^{n/2}\Gamma(n/2+1)\exp(-K){}_1F_1(n/2+1;1;K) \quad (2.27)$$

where ${}_1F_1(;;;)$ is the confluent hypergeometric function, and $\Gamma()$ is the gamma function [57]. Then

$$f_{2,4}(K) = \frac{\mu_n^m}{\mu_m^n} = \left[\frac{(K+1)^2}{K^2+4K+2}\right]^2 \quad (2.28)$$

One of the root is always negative. Since $K \geq 0$, yielding a unique nonnegative solution for $\hat{K}_{2,4}$ [58, 59, 60] as

$$\hat{K}_{2,4} = \frac{-2\hat{\mu}_2^2+\hat{\mu}_4-\hat{\mu}_2\sqrt{2\hat{\mu}_2^2-\hat{\mu}_4}}{\hat{\mu}_2^2-\hat{\mu}_4} \quad (2.29)$$

Calculated K values from square root of magnitude of the four readers in the motion is shown in Figure A.1. As shown in the figure, K factors from different readers follow the same trend in the motion. In motion, K value is around 10 - 40. In the static states, K values is larger than 80.

2.5 Multipath Effects on Magnitude and Phase of Received Baseband Signal

Moreover, power received by a reader from another tag in multipath propagation is

$$P_R(r) = \frac{P_T G_T G_R G_t^2 \lambda^4 X^2 M}{(4\pi)^4} \left| \frac{1}{r_0} \exp(-j\frac{2\pi r_0}{\lambda}) + \sum_{i=1}^{N} \Gamma_i \frac{1}{r_i} \exp(-j\frac{2\pi r_i}{\lambda}) \right|^4 \tag{2.30}$$

where r_0 is the length of the direct path, r_i, $i = 1, ..., N$ is the length of ith reflection path, N is the total number of reflection paths, and Γ_i is the complex reflection coefficient for the ith reflection path [61]. For one reflection with N = 1, Equation 7.1 gives

$$\begin{aligned} P_R(r) &= \frac{P_T G_T G_R G_t^2 \lambda^4 X^2 M}{(4\pi)^4} \left| \frac{1}{r_0} \exp(-j\frac{2\pi r_0}{\lambda}) + \Gamma_1 \frac{1}{r_1} \exp(-j\frac{2\pi r_1}{\lambda}) \right|^4 \\ &= \frac{P_T G_T G_R G_t^2 \lambda^4 X^2 M (A^2 + B^2)}{(4\pi)^4} \exp(\mp j(\frac{8\pi r_0}{\lambda} + 4 \arctan \phi)) \\ &= \frac{P_T G_T G_R G_t^2 \lambda^4 X^2 M (A^2 + B^2)}{(4\pi)^4} \exp(\mp j(\frac{8\pi r_0}{\lambda} + 4 \arctan(\frac{\sin \frac{2\pi \Delta r}{\lambda}}{\frac{r_1}{\Gamma_1 r_0} + \cos \frac{2\pi \Delta r}{\lambda}}))) \end{aligned} \tag{2.31}$$

where $A = \frac{\Gamma_1}{r_1} \sin \frac{2\pi \Delta r}{\lambda}$, $B = \frac{1}{r_0} + \frac{\Gamma_1}{r_1} \cos \frac{2\pi \Delta r}{\lambda}$, $\phi = \arctan(\frac{A}{B}) = \arctan(\frac{\sin \frac{2\pi \Delta r}{\lambda}}{\frac{r_1}{\Gamma_1 r_0} + \cos \frac{2\pi \Delta r}{\lambda}})$. The detailed derivations are in Appendix B.

2.5.1 Power Received by a Reader from Tunnel Diode Modulator Tag in Multipath Propagation

Power received by a reader from tunnel diode modulator tag in multipath propagation is

$$P_R(r) = \frac{P_T G'_T G'_t \lambda^2 X' M}{(4\pi)^2} \left| \frac{1}{r_0} \exp(-j\frac{2\pi r_0}{\lambda}) + \sum_{i=1}^{N} \Gamma_i \frac{1}{r_i} \exp(-j\frac{2\pi r_i}{\lambda}) \right|^2 \tag{2.32}$$

Similarly, r_0 is the length of the direct path, r_i, $i = 1, ..., N$ is the length of ith reflection path, N is the total number of reflection paths, and Γ_i is the complex reflection coefficient for the ith reflection path [61]. In case of one reflection, the phase can also be derived as in Equation 2.31.

2.5.2 Multipath Effects on Magnitude of Received Baseband Signal

As shown in Equation 2.31, since $A^2 + B^2 = (\frac{\Gamma_1}{r_1})^2 + (\frac{1}{r_0} + \frac{\Gamma_1}{r_1}\cos\frac{2\pi\Delta r}{\lambda})^2$ is multiplied to the magnitude. Then the effects of multipath on magnitude decreases with the increase of length of multipath. In addition, it is of sinusoidal waves in continuous motions.

2.5.3 Multipath Effects on Phase of Received Baseband Signal

In Equation 2.31, if $\frac{r_1}{r_0}\Gamma_1 \ll \cos\frac{2\pi\Delta r}{\lambda}$, then $\frac{2\pi r_0}{\lambda} + 2\arctan(\frac{\sin\frac{2\pi\Delta r}{\lambda}}{\frac{r_1}{r_0}\Gamma_1 + \cos\frac{2\pi\Delta r}{\lambda}}) = \frac{2\pi(r_0+\Delta r)}{\lambda} = \frac{2\pi r_1}{\lambda}$. Otherwise, the phase angle is smaller than $\frac{2\pi r_1}{\lambda}$. If Δr changes continuously, phase also changes sinusoidal accordingly.

CHAPTER 3

SIMULATION VERIFICATION ON 1D AND 2D LINE-OF-SIGHT (LOS) DATA

In this chapter, sample parameters referring to system in [39] are chosen in the simulation for semi-passive switched modulator tag for 1D and 2D motions. Based on model and noise distribution in chapter 2 and section 2.3, Received Signal Strength (RSS) and Received Signal Phase (RSP) of tag from given reader are simulated. The parameters of the simulated reader are shown in Table 3.1: [1]

Table 3.1: Simulation Template with Sample Parameters in 2D Motion

P_T	Reader's Transmitted Power	1W
G_T	Reader's Transmitter Antenna Gain	9.5 dBi
G_R	Reader's Receiver Antenna Gain	9.5 dBi
G_t	Tag's Antenna Gain	9.5 dBi
X	Polarization Mismatch Between Reader's and Tag's Antenna	0.5
M	Load Modulation Factor of the Tag	0.25
R	Input Impedance of the System	15 Ω
f	Carrier Frequency	5.8 GHz
σ	Standard Deviation of Added Noise	2×10^{-4}

3.1 1D Simulation Verification

To verify the value and noise distribution of simulated states, states of 1D motion based on the experiment in [62] are simulated. More specifically, the tag initially located at a distance of 0.99 m from the reader. Starting from rest, the tag accelerated at 7.35 m/s^2 to attain a velocity of 1.2 m/s and keeps the velocity for 355 ms before decelerating at a constant rate of -7.35 m/s^2 to static, travelling a total distance of 0.64 m away from the

[1] Simulation Model is in: `https://github.com/qq3575022/RFID_SimulationModel`

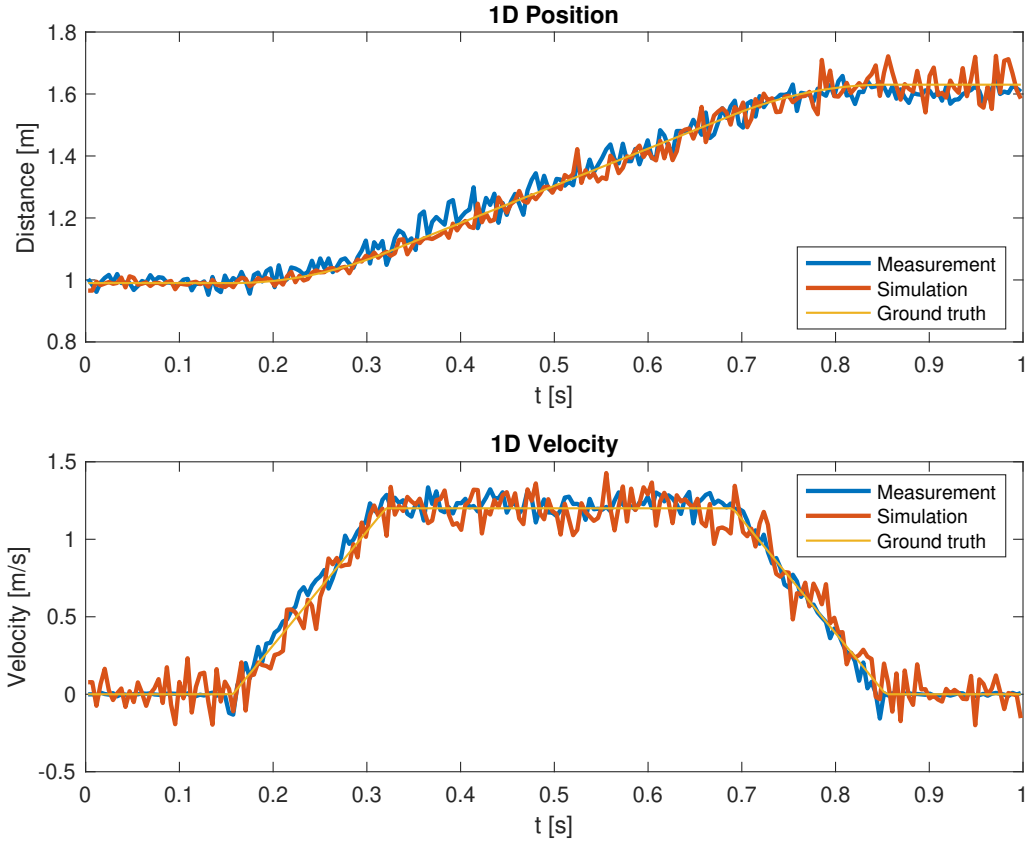


Figure 3.1: Simulation Verification of Position and Velocity in 1D Motion. Upper: Position from Measurement, Simulated Position based on Signal Magnitude, and Ground Truth States. Lower: Velocity from Measurement, Simulated Velocity based on Phase, and Ground Truth States.

reader. The input of the simulation model are parameters of the reader in the Table 3.1 and commanded 1D motion described above. The output of the model are RSS and RSP with channel noise received by the reader from tag in the 1D motion based on models in chapter 2 and section 2.3.

Simulation outputs including RSS and RSP with added noise are applied to derive position and velocity based on equations Equation 2.6 and Equation 2.7. Simulated position and velocity as well as ground truth states are shown and compared in Figure 3.1.

As shown in Figure 3.1, simulated position and velocity states are of the same shape and similar noise level as measurements [39]. In addition, both of simulation and measurement fit ground truth states. Thus, the physical-based simulator appears to faithfully reproduce

Table 3.2: Statistics of Error for Simulation and Measurement Position States in 1D Motion

Position Error	Mean [m]	Variance [m^2]	Standard Deviation [m]
Measurement	-0.0024	0.0009	0.0295
Simulation	0.0035	0.0009	0.0306

the behaviors of the measurements. In some instances, the physical simulator could be used to test algorithms faster and in a wider variety of scenarios compared to measurements-only tests. To make a more detailed comparison, noises of position states in simulation and measurement deviated from ground truth are shown in Table 3.2.

Statistics of e, i.e. Mean$= E(e) = \bar{e}$, Variance$= \frac{\sum_{t=1}^{N}(e_t - E(e))^2}{N}$, and standard deviation, the root square of variance, are shown in Table 3.2. It can be inferred Table 3.2 that simulated error is of similar distribution and variance to that of measurement error on position states in 1D motion [39].

3.2 2D Simulation Verification

To verify the simulation for nonlinear states, 2D motion based on the experiment in [39] is simulated. The motion of the apparatus is commanded by a reference trajectory for angular position in [39]. Rotation of the beam will occur in the horizontal plane. Physical parameters of the apparatus are characterized by α, β [39]. More specifically, starting at time $t_{start} = 0.12$ s, a positive acceleration lasting $t_{acc} = 0.58$ s is applied. After a cruise time of $t_{crz} = 0.28$ s, a negative acceleration is applied. Both acceleration occur on a time interval of length t_{acc}, with constant accelerations equal to $\ddot{\psi} = \pm 12.6$ rad/s^2. The distance traveled, in radians, is 2π. From measurement, the radius of the beam is 0.31 m and the rotation is around center at $(1.95$ m, 1.15 m). Please refer to [39] for more parameters.

The setup of the two-dimensional motion is in Figure 3.2. Radial distances r_1, r_2, and r_3 are distances from the tag to three reader pylons: Reader #1 (Rx #1), Reader #2 (Rx #2), and Reader #3 (Rx #3), whose locations are in Table 3.3. The origin is O and the tag is rotating counter-clockwise in the plane. The initial position of the tag has body axis

x^B pointing along y axis. The offset angle for motion in [39] is $\pi/2$ from x axis for both x and y coordinates as shown in Figure 3.2. Similar to 1D simulation, parameters of the

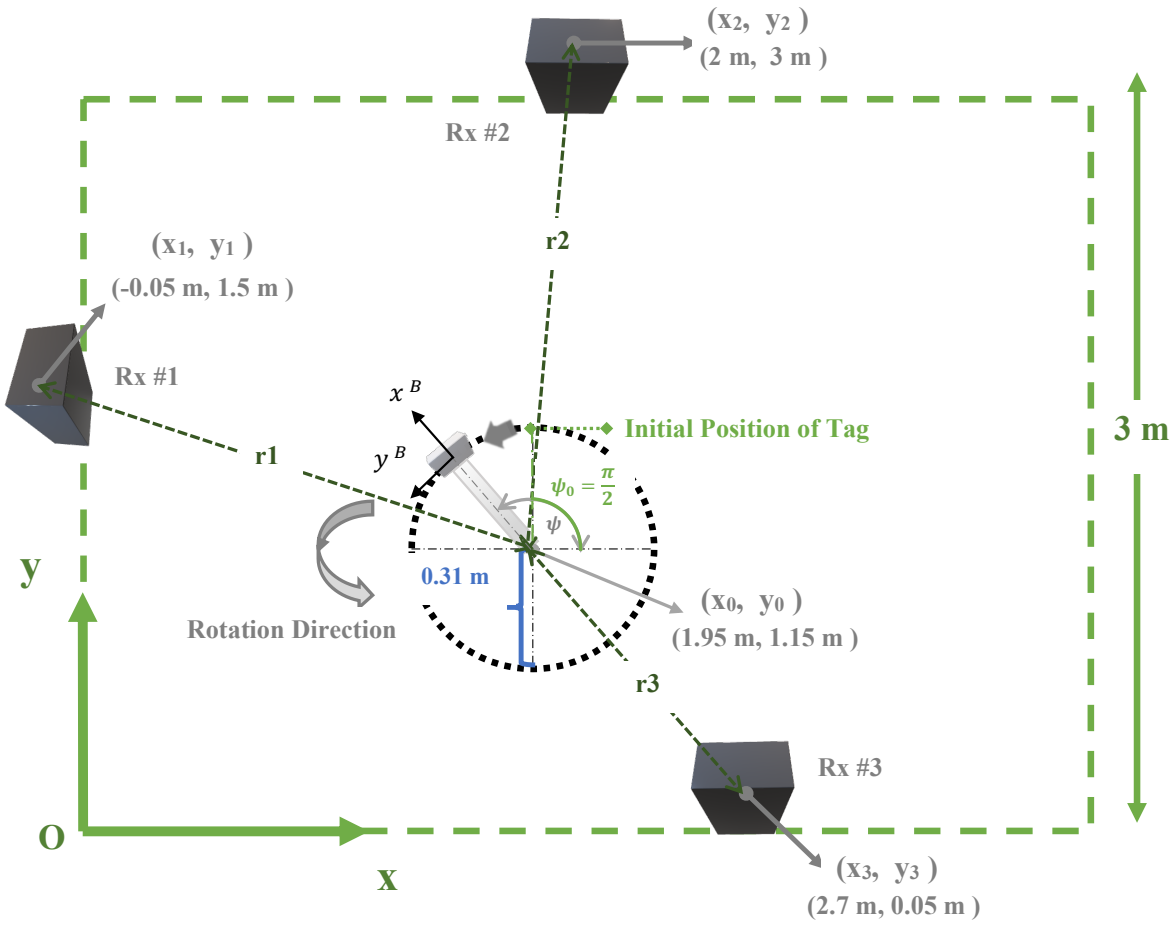


Figure 3.2: Two-dimensional Motion Setup. The Three 5.8 GHz RFID Reader Pylons are Installed at Known Locations with Fixed Coordinates as Shown in Table 3.3 Whose Origin is Defined by O. An RF Tag Considered as a Rigid Body with Its Own Body Axes x^B and y^B Moves Around $(1.95$ m, 1.15 m) Counter-Clockwise in the Plane. The Radial Distance to the Tag from Each Pylon is Defined by r_1, r_2 and r_3 Respectively.

reader in **??** and commanded 2D motion coordinates (x, y) described above are input into the simulation model, and the output of the model is RSS and RSP from three readers at different locations. The locations of three readers are listed in Table 3.3.

Based on simulated RSS and RSP from three readers, radial distances r_1 r_2 r_3 and radial velocities $\dot{r}_1$ $\dot{r}_2$ $\dot{r}_3$ from the moving tag to three readers can be derived based on Equation 2.6 and Equation 2.7, which are shown in Figure 3.3 and Figure 3.4. In this way, simulated

Table 3.3: Locations of Three Readers in 2D Motion

Reader Index	Coordinates	Location [m]
Reader #1	(x_1, y_1)	(-0.05, 1.5)
Reader #2	(x_2, y_2)	(2, 3)
Reader #3	(x_3, y_3)	(2.7, 0.05)

radial distances r_1 r_2 r_3 and velocities $\dot{r}_1$ $\dot{r}_2$ $\dot{r}_3$ can be compared with measurement states in [39] and ground truth states for verification. Simulation and comparison results of distances and velocities are shown in Figure 3.3 and Figure 3.4. It can be inferred from Figure 3.3

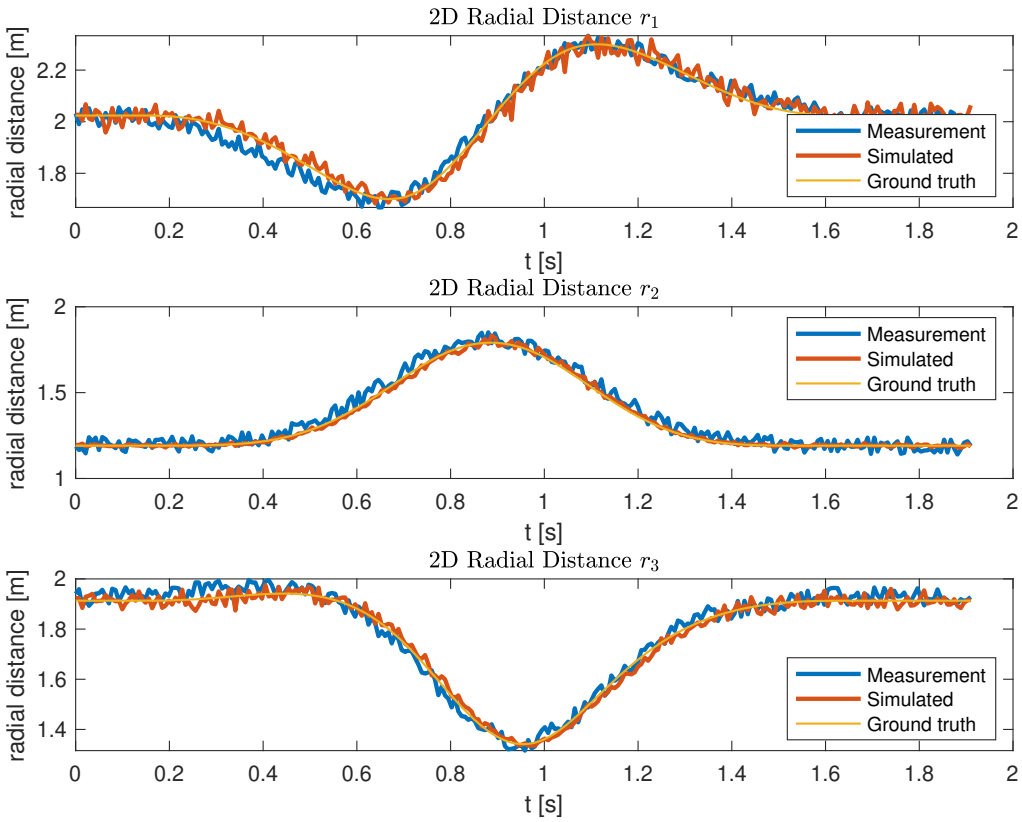


Figure 3.3: Simulation Verification of Radial Distances in 2D Motion: Comparison of Radial Distances r_1 r_2 r_3 Among Measurement, Simulation, and Ground Truth States.

and Figure 3.4 that simulated radial distances and radial velocities are of the same shape as measurement states. In addition, both of simulation and measurement states fit ground truth states generated from description of 2D motion [39].

To make a more detailed comparison, noises of radial position r_1, r_2, and r_3 in simula-

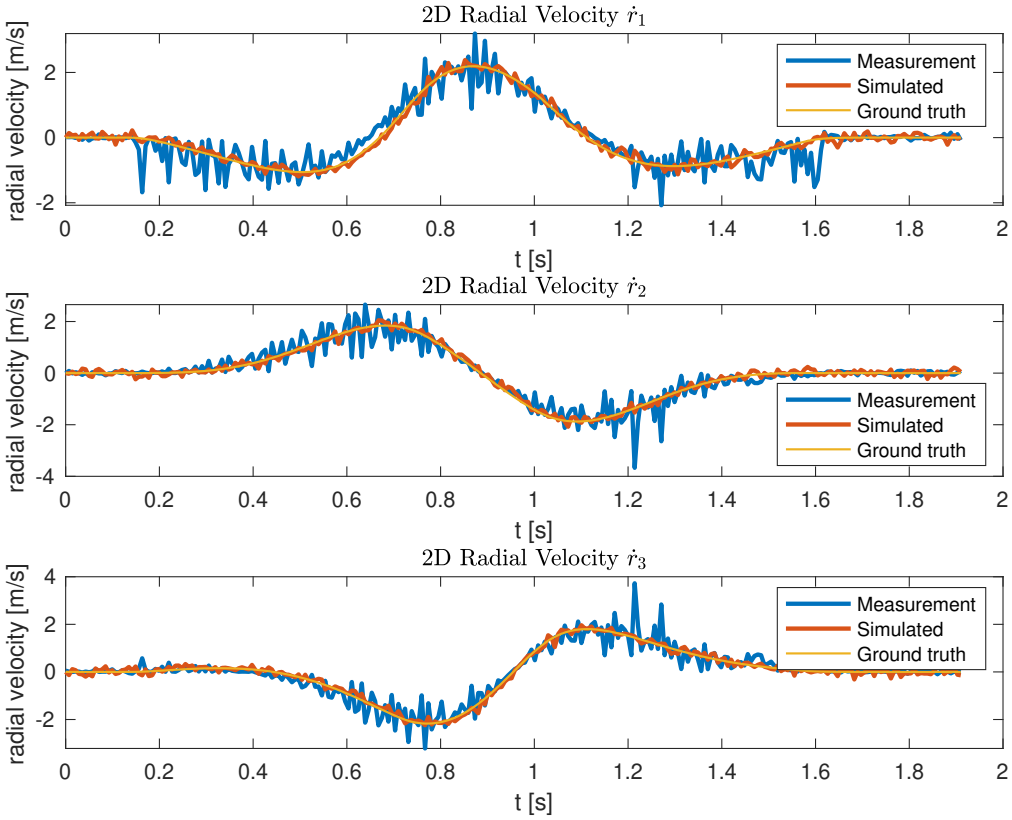


Figure 3.4: Simulation Verification of Radial Velocity in 2D Motion: Comparison of Radial Velocities $\dot{r}_1$ $\dot{r}_2$ $\dot{r}_3$ Among Measurement, Simulation, and Ground Truth States.

tion and measurement from ground truth states are shown in Table 3.4, from which it can be inferred that simulated errors are of similar variance to that of measurement errors on radial position in 2D motion.

Table 3.4: Statistics of Error for Simulation and Measurement Radial Distances in 2D Motion

Radial Distance	Error	Mean [m]	Variance [m^2]	Standard Deviation [m]
r_1	Measurement	-0.0062	0.0009	0.0303
	Simulation	0.0016	0.0006	0.0253
r_2	Measurement	0.0195	0.0010	0.0321
	Simulation	-0.0001	0.0001	0.0102
r_3	Measurement	0.0097	0.0007	0.0262
	Simulation	0.0023	0.0003	0.0185

CHAPTER 4

DESIGN OF 2D NONLINEAR LEAST-SQUARES ESTIMATION

In this chapter, moving forward from linear state estimation in 1D motion, nonlinear state-space model and nonlinear state estimation for real-time RFID-based 2D localization are built. Nonlinear state-space model is built with different state vectors and sensor outputs. Nonlinear Least-Squares Estimation with stacked measurement vector is applied to nonlinear state estimation. Gauss-Newton and Levenberg-Marquardt methods are designed for estimation implementation. Further, hypothesis of the positive correlation between the stack length and sensor noise level in Nonlinear Least-Squares Estimation is proposed. [1]

4.1 2D State-Space Model

In this section, nonlinear state-space model of the 2D motion using different state vectors and sensor outputs is built. The tagged object is modeled as a rigid body experiencing 2D motion whose motion in the plane is governed by Newton's Law [39]. The motion equations of the tag are in the form

$$m\ddot{x} = F_x \tag{4.1}$$

$$m\ddot{y} = F_y \tag{4.2}$$

$$I_z\ddot{\psi} = T_z \tag{4.3}$$

where m is the mass of the rigid body, $\ddot{x}$ and $\ddot{y}$ are the accelerations along x and y axes. $\ddot{\psi}$ is the angular acceleration. F_x and F_y are the resultant forces applied to the mass along x and y axes. I_z and T_z are the inertial moment and resultant torque around z axis. The force/mass and torque/moment ratios determine the 2D motion of the tag. The explained

[1] NLE is in: `https://github.com/qq3575022/Nonlinear_Least_Square_Estimation`

motion profiles are unknowns in the experiment. In addition, unlike estimators in [39], initial conditions are also unknown in the estimation.

The nonlinear state-space model describing the 2D motion with different state vectors and sensor outputs is shown in Equation 4.4 - Equation 4.5

$$\dot{X} = AX + Bw \tag{4.4}$$

$$Y = h(X) + v \tag{4.5}$$

where X is the fixed-length state vector composed of position, velocity, and acceleration states along x, y axes without and with orientation and angular velocity along z axis, which is shown in Equation 4.6 and Equation 4.9 respectively. Y denotes the vector of noisy sensor outputs to estimate the state vector. Six sets of sensor outputs Y: $\{r\}$, $\{r, \dot{r}\}$ for outputs without acceleration, orientation, or angular velocity are shown in Equation 4.7; $\{r, \theta_z, a_x, a_y\}$, $\{r, \theta_z, \omega_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, \omega_z, a_x, a_y\}$ for outputs with acceleration, orientation, and angular velocity are shown in Equation 4.10. Vector $h(X)$ describes the nonlinear transformation between state vector and the noise-free sensor outputs vector. w is the disturbance vector which accounts for the forces $(\frac{\dot{F}_x}{m}, \frac{\dot{F}_y}{m}, \frac{\dot{T}_z}{I_z})$ and v is the vector of sensor noises.

For state vector without orientation or angular velocity, the state vector X and output vector Y are composed of

$$X = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} \end{bmatrix}' \tag{4.6}$$

$$Y = \begin{bmatrix} r_1 & \dot{r}_1 & r_2 & \dot{r}_2 & r_3 & \dot{r}_3 \end{bmatrix}' \tag{4.7}$$

Output vector Y in Equation 4.7 corresponds to sensor outputs of $\{r\}$, $\{r, \dot{r}\}$. In Equation 4.7, r_i, $\dot{r}_i$ are radial distances and radial velocities measured from reader i. If only radial distances are used, $\dot{r}_i$, $\forall i = 1, 2, 3$ are removed from Equation 4.7. Corresponding

coefficient matrices A, B in the state-space model are

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.8}$$

For state vector with orientation and angular velocity, the state and output vectors X, Y are composed of

$$X = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} & \psi & \dot{\psi} \end{bmatrix}' \tag{4.9}$$

$$Y = \begin{bmatrix} r_1 & \dot{r}_1 & r_2 & \dot{r}_2 & r_3 & \dot{r}_3 & \theta_z & \omega_z & a_x & a_y \end{bmatrix}' \tag{4.10}$$

Output vector Y in Equation 4.10 corresponds to sensor outputs of $\{r, \theta_z, a_x, a_y\}$, $\{r, \theta_z, \omega_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, \omega_z, a_x, a_y\}$. Similarly, r_i, $\dot{r}_i$ are radial distances and radial velocities measured from reader i. θ_z is the orientation angle. ω_z is the angular velocity. a_x and a_y are accelerations along x^B and y^B axes in Equation 4.26 and Equation 4.27. More specifically, detailed descriptions can be inferred in Equation 9.10 and Equation 9.11. If angular velocity is not used, then ω_z is removed from Equation 4.10. If radial velocities are not used, $\dot{r}_i$ where $i = 1, 2, 3$ in the four sets of sensor outputs are

removed from Equation 4.10. Coefficient matrices A, B in the state-space model become

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{4.11}$$

Suppose that values of the continuous signal are sampled at the sampling instants k with sampling period T. Corresponding discrete-time state-space model of the tag in the 2D motion is shown in Equation 4.12 - Equation 4.13.

$$X_k = FX_{k-1} + Gw_k \tag{4.12}$$

$$Y_k = h(X_k) + v_k \tag{4.13}$$

where X_k is discrete-time state vector of X composed of position, velocity, and acceleration states along x and y axes at time-stamp k. Y_k denotes the vector of noisy sensor outputs at time-stamp k. w_k and v_k are the discrete-time disturbance vector and the discrete-time sensor noises vector. Similarly, for state vector without orientation or angular velocity, discrete-time state vector and output vector corresponding to vectors in continuous time Equation 4.6 - Equation 4.8 are

$$X_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k & y_k & \dot{y}_k & \ddot{y}_k \end{bmatrix}' \tag{4.14}$$

$$Y_k = \begin{bmatrix} r_1 & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} \end{bmatrix}' \tag{4.15}$$

If only radial distances are used, $\dot{r}_{ik}$ where $i = 1, 2, 3$ are removed from Equation 4.15.

Coefficient matrices F and G are shown in Equation 4.16, where T is the sampling period.

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & T \end{bmatrix} \tag{4.16}$$

For state vector with orientation and angular velocity, discrete-time state and output vectors corresponding to vectors in Equation 4.9 - Equation 4.11 become

$$X_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k & y_k & \dot{y}_k & \ddot{y}_k & \psi_k & \dot{\psi}_k \end{bmatrix}' \tag{4.17}$$

$$Y_k = \begin{bmatrix} r_{1k} & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} & \theta_z & \omega_z & a_{xk} & a_{yk} \end{bmatrix}' \tag{4.18}$$

If angular velocity is not used, then ω_z is removed from Equation 4.18. If radial velocities are not used, $\dot{r}_{ik}$ where $i = 1, 2, 3$ in the four sets of sensor outputs are removed from Equation 4.18. Corresponding coefficient matrices F and G are

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, G = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ T & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & T \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{4.19}$$

Nonlinear transformation between state vector and noise-free sensor outputs vector are

presented as following. For state vector without orientation or angular velocity,

$$h(X_k) = \begin{bmatrix} h_1 & \dot{h}_1 & h_2 & \dot{h}_2 & h_3 & \dot{h}_3 \end{bmatrix}' \tag{4.20}$$

$\dot{h}_i$ where $i = 1, 2, 3$ are removed from Equation 4.20 if only radial distances are used. For state vector with orientation and angular velocity,

$$h(X_k) = \begin{bmatrix} h_1 & \dot{h}_1 & h_2 & \dot{h}_2 & h_3 & \dot{h}_3 & h_4 & h_5 & h_6 & h_7 \end{bmatrix}' \tag{4.21}$$

Likewise, if angular velocity is not used, then h_5 is removed from Equation 4.21. If radial velocities are not used, $\dot{h}_i$ where $i = 1, 2, 3$ in the four sets of sensor outputs are removed from Equation 4.21. In both situations,

$$h_u = \sqrt{(x_k - x_u)^2 + (y_k - y_u)^2}, \quad u = 1, 2, 3 \tag{4.22}$$

$$\dot{h}_u = \frac{(x_k - x_u)\dot{x}_k + (y_k - y_u)\dot{y}_k}{\sqrt{(x_k - x_u)^2 + (y_k - y_u)^2}}, \quad u = 1, 2, 3 \tag{4.23}$$

$$h_4 = \psi_k \tag{4.24}$$

$$h_5 = \dot{\psi}_k \tag{4.25}$$

$$h_6 = \ddot{x}_k \cos(\psi_k) + \ddot{y}_k \sin(\psi_k) \tag{4.26}$$

$$h_7 = -\ddot{x}_k \sin(\psi_k) + \ddot{y}_k \cos(\psi_k) \tag{4.27}$$

in which x_u y_u, $u = 1, 2, 3$ are the locations of each reader inferred from Table 3.3.

4.2 2D Nonlinear State Estimation Based on NLE

In this section, the Nonlinear Least-Squares Estimation (NLE) with stacked measurement vector, originally proposed in [49], is applied to the nonlinear state estimation of real-time RFID localization in 2D motion. NLE is developed from the perspective of nonlinear least-squares optimization and the map-inversion philosophy based on discrete-time dy-

namics of the system. More specifically, NLE determines an estimate $\hat{x}_k$ of x_k, using the l-length measurement set of inputs $\{y_k, y_{k-1}, ..., y_{k-l+1}\}$ to reduce the impact of sensor noise. Therefore, NLE includes a two-step procedure, wherein first $\hat{x}_{k-l+1}$ is determined from the measurement sets and second $\hat{x}_k$ is determined through forward propagation [49]. The stacked measurement vector of the tag in the 2D motion based on the discrete-time state-space model in section 4.1 can be represented as

$$Z_k = \begin{bmatrix} Y_{k-l+1}, & Y_{k-l+2}, & \dots & , Y_k \end{bmatrix}' \tag{4.28}$$

where Y_k is the noisy measurement outputs vector in Equation 4.15 or Equation 4.18, which is composed of the whole or part of

$$Y_k = \begin{bmatrix} r_{1k}, & \dot{r}_{1k}, & r_{2k}, & \dot{r}_{2k}, & r_{3k}, & \dot{r}_{3k}, & \theta_z & \omega_z & a_x, & a_y \end{bmatrix}' \tag{4.29}$$

in which r_{1k}, r_{2k}, r_{3k}, $\dot{r}_{1k}$, $\dot{r}_{2k}$, $\dot{r}_{3k}$ are simulated/measurement radial distances and radial velocities at time-stamp k with added noise, which are shown in Figure 3.3 and Figure 3.4. A mapping from state and input values to output values is denoted as H

$$H_k(X_{k-l+1}) = \begin{bmatrix} h(X_{k-l+1}) \\ h(FX_{k-l+1}) \\ .. \\ h(F^{l-1}X_{k-l+1}) \end{bmatrix} \tag{4.30}$$

where F is the same as in Equation 4.16 and Equation 4.19 for state vector without and with orientation and angular velocity. In the absence of modeling error and sensor noise [49],

$$Z_k = H_k(X_{k-l+1}) \tag{4.31}$$

In case of noise, NLE minimizes the sum of the squares for the residual errors

$$\hat{X}_{k-l+1} = \text{argmin}||E_k(X_{k-l+1})||^2 = \text{argmin}||Z_k - H_k(X_{k-l+1})||^2 \tag{4.32}$$

This least-squares state estimate is l samples delayed in time, so it is propagated forward in time using the nominal system dynamics and gets [49]

$$\hat{X}_k = F^{l-1}X_{k-l+1} \tag{4.33}$$

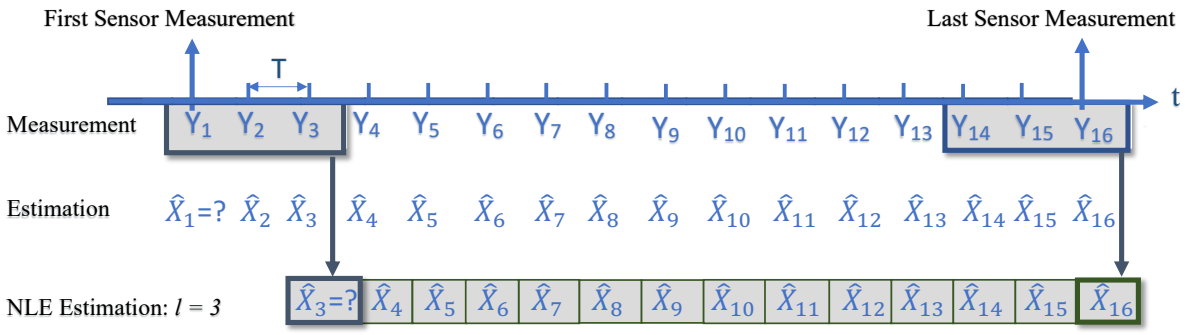
Suppose that the current iterate in the search for the minimizer of $||E_k(X_{k-l+1})||^2$ is denoted by $X^{(j)}_{k-l+1}$. Using Taylor series approximation, a corresponding local model for the iteration process would be [49]

$$\hat{E}_k(X_{k-l+1}) = E_k(X^{(j)}_{k-l+1}) + J_k(X^{(j)}_{k-l+1})(X_{k-l+1} - X^{(j)}_{k-l+1}) \tag{4.34}$$

where $J_k(X^{(j)}_{k-l+1})$ denotes the Jacobian matrix of $E_k(X^{(j)}_{k-l+1})$ with respect to $X^{(j)}_{k-l+1}$.

NLE-based estimation can be implemented in both causal and non-causal ways. For state estimation of RFID-based motion-capture and localization systems, we implement NLE causally using states before and at the time-stamp. The scheme of nonlinear state estimation based on NLE is shown in Figure 4.1. In Figure 4.1, $Y_1, Y_2, ..., Y_{16}$ are measurement states and $X_1, X_2, ..., X_{16}$ are states to be estimated. Estimated states are related measurement states by $Y_k = h(X_k) + v_k$. Furthermore, NLE estimates state X_k from a stack of measurement Y, i.e. $Z_k = [Y_{k-l+1}, Y_{k-l+2}, ..., Y_k]'$. The estimation is solved by minimizing nonlinear least square error.

In order to minimize nonlinear least square error $||E_k(X_{k-l+1})||^2 = ||Z_k - H_k(X_{k-l+1})||^2$, an iterative search of damped Gauss-Newton method is conducted, which is shown in subsection 4.2.1 [49]. Levenberg-Marquardt method interpolates between Gauss–Newton algorithm and the method of gradient descent. Hence, Levenberg-Marquardt method shares



Non-NLE: Given $Y_k = h(X_k) + v_k$, Estimate $\hat{X}_k$

NLE: Given $Z_k = \begin{bmatrix} Y_{k-l+1} \\ Y_{k-l+2} \\ \vdots \\ Y_k \end{bmatrix}_{N\times 1}$ where $Y_k = h(X_k) + v_k$, Estimate $\hat{X}_k$

Figure 4.1: Scheme of State Estimation Based on NLE.

with the gradient methods their ability to converge from an initial guess which may be outside the region of convergence and also shares with the Taylor series methods the ability to converge rapidly in the vicinity of the converged value. The Levenberg-Marquardt is more robust than the Gauss–Newton algorithm, which means that in many cases it finds a solution even if it starts very far off the final minimum [51]. Levenberg-Marquardt method is also applied to minimize the error of nonlinear least-squares in subsection 4.2.2.

4.2.1 Implementation of NLE Based on Gauss-Newton Method

In this section, an iterative search of damped Gauss-Newton method is conducted to minimize the nonlinear least-squares error in $||E_k(X_{k-l+1})||^2 = ||Z_k - H_k(X_{k-l+1})||^2$. The solution of Equation 4.34 may be computed using Gauss-Newton method as [49]

$$X_{k-l+1}^{(j+1)} = X_{k-l+1}^{(j)} - J_k(X_{k-l+1}^{(j)})^{-1} E_k(X_{k-l+1}^{(j)}) \tag{4.35}$$

To make sure that the inversion of Jacobian matrix can be calculated even for ill-conditioned matrix, the inversion is calculated based on pseudoinverse using QR decomposition as in [50]. To guarantee a descent direction and reduce the residual error, the step increment is defined as [49]

$$X_{k-l+1}^{(j+1)} = X_{k-l+1}^{(j)} - \gamma^{(j)} J_k(X_{k-l+1}^{(j)})^{-1} E_k(X_{k-l+1}^{(j)}) \tag{4.36}$$

where $0 < \gamma^{(j)} < 1$ is a scalar parameter that may be used to limit the length of each step.

Initial Guess: Initialization of state vectors affects the estimation accuracy. In the estimation, X_k, $k = 1, 2, ..., \textit{len}$ where *len* is the maximum time-stamp in the sequential state vector in Equation 4.14 and Equation 4.17, are initialized as $[1.5, -2, 0, 1, -2, 0]'$ and $[1.5, -2, 0, 1, -2, 0, 3, 0]'$.

Parameter Selection: From Dennis and Schnabel, if initialization is close enough to a solution X_* and Jocobian matrix with respect to $J(X_*)$ is nonsingular, Gauss-Newton method converges Q-quadratically to solution X_* [50]. In order to take step that does not exceed the Newton step, step size γ is chosen as 0.1 in the estimation. Maximum iteration number is set as 200, i.e., $\gamma = 0.1, \textit{limit} = 200$.

Convergence Criterion: Since fixed step size is applied to update states, if $E(X_k)$ does not decrease any more or the iteration exceeds the maximal limit, iteration of updating $\hat{X}_k$ breaks and proceeds to estimate the state at next time-stamp, i.e., $\hat{X}_{k+1}$, $k = l, l+1, ...,$ *len* - l where l is the stack length.

Estimation Pseudo Code: Pseudo code of implementing NLE based on Gauss-Newton method with Z_k and $H_k(X_{k-l+1})$ in Equation 4.28 and Equation 4.30 is shown below.

In the pseudo code, *len* is the maximum time-stamp in the sequential state vector. State vector X_k, $\forall k$ is initialized with initial guess. *limit* is the maximum number of iterations and γ is the step size. When error $E_k(X_{k-l+1})$ stops decreasing or exceeds the maximum

procedure NLE ESTIMATION BASED ON GAUSS-NEWTON
  for $k = l$, $k{+}{+}$, `while` $k < len$ **do**
    $X_{k-l+1} \leftarrow$ initial value
    for iter $<$ *limit* **do**
      $E_k(X_{k-l+1}) = Z_k - H_k(X_{k-l+1})$
      $X'_{k-l+1} = X_{k-l+1} - \gamma J_k(X_{k-l+1})^{-1} E_k(X_{k-l+1})$
      $E_k(X'_{k-l+1}) = Z_k - H_k(X'_{k-l+1})$
      if $E_k(X'_{k-l+1}) < E_k(X_{k-l+1})$ **then**
        $X_{k-l+1} \leftarrow X'_{k-l+1}$
      else break
    $X_k = F^{l-1} X_{k-l+1}$
  close

number of iterations, the loop of updating estimated $\hat{X}_{k-l+1}$ breaks. Inversion of Jacobian matrix $J_k(X_{k-l+1})^{-1}$ is computed using the pseudo inverse $\backslash$ based on QR decomposition in Matlab. Finite-difference approximation is applied to compute Jocobian matrix referring [49]. Finite-difference approximation also works for cases where derivatives are not analytically available [50]. When iteration finishes or breaks, estimated state $\hat{X}_k$ is calculated based on forward propagation as: $\hat{X}_k = F^{l-1}\hat{X}_{k-l+1}$.

4.2.2 Implementation of NLE Based on Levenberg-Marquardt Method

Globally convergent modifications of Newton's method have been elaborated in [50]. In this estimation, Levenberg-Marquardt method is applied to minimize the nonlinear least-squares error. Levenberg replaces the inversion from Equation 4.35

$$[J_k(X^{(j)}_{k-l+1})^T J_k(X^{(j)}_{k-l+1})](X^{(j)}_{k-l+1} - X^{(j+1)}_{k-l+1}) = J_k(X^{(j)}_{k-l+1})^T E_k(X^{(j)}_{k-l+1}) \tag{4.37}$$

by a "damped version" as

$$[J_k(X^{(j)}_{k-l+1})^T J_k(X^{(j)}_{k-l+1}) + \lambda I](X^{(j)}_{k-l+1} - X^{(j+1)}_{k-l+1}) = J_k(X^{(j)}_{k-l+1})^T E_k(X^{(j)}_{k-l+1}) \tag{4.38}$$

The non-negative damping factor λ is adjusted at each iteration. Mores specifically, if reduction of E_k is rapid, a smaller value can be used, making algorithm closer to the

Gauss–Newton algorithm. If the reduction is insufficient, λ can be increased, giving a step closer to the gradient-descent [63]. Marquardt recommended starting with a value λ_0 and a factor v ($v > 1$) and computing square error based on damping factors λ_0 and λ_0/v. If both of these are worse than the initial point, then the damping factor is increased by successive multiplication by v until a better point is found with a new damping factor of $\lambda_0 v^k$ for some k. If use of the damping factor λ_0/v results in a reduction in squared residual, then this is taken as the new value of λ_0 and the process continues; if using λ_0/v resulted in a worse residual but using λ_0 resulted in a better residual, then λ_0 is unchanged and the new optimum is taken as the value obtained with λ_0 as damping factor [51]. Since only if square error is less than the initial point, the new optimum will be updated, every iteration of Levenberg-Marquardt method is in a descent direction. Therefore, Equation 4.39 is positive definite [50], retaining fast local convergence.

$$J_k(X_{k-l+1}^{(j)})^T J_k(X_{k-l+1}^{(j)}) + \lambda I \tag{4.39}$$

Comparing square error using damping factor and a fraction of damping factor gives a more proper step size at each iteration.

Implementation: Both step-by-step design of Levenberg-Marquardt method and built-in function `lsqnonlin` in Matlab have been applied to the estimation. The two implementations give close enough results. Built-in function in Matlab $\texttt{lsqnonlin}(@(X)E_k(Y, X, l), X_k)$ is implemented in the estimation. Input $E_k(X)$ is the residual error in Equation 4.34, state vector X_k, $k = 1, 2, ...,$ *len* is initialized the same as in subsection 4.2.1.

4.3 Stack Length of NLE at Different Sensor Noise Levels

If sensor noise v is large, longer stack length of NLE gets higher estimation accuracy; If sensor noise v is small, shorter stack length gets higher estimation accuracy.

CHAPTER 5

RESULTS OF 2D NONLINEAR LEAST-SQUARES ESTIMATION

In this chapter, estimation based on NLE is implemented on both measurement data from Bashir's work of HIMR system in 2D motion [39] and simulated states from the model of RFID system in Chapter 2. Estimation results with Gauss-Newton and Levenberg-Marquardt methods using different sensor outputs are compared and evaluated. From estimation results, RF-signal achieves similar estimation results to RF-signal with added inertial information. In addition, stacked states in NLE maintain the motion-capture grade, i.e., 10 mm estimation accuracy for states at various noise levels. Estimation results also verify the positive correlation hypothesis between noise level and the stack length in NLE.

NLE using different sensor information including instantaneous trilateration based on measured radial distances from three readers, radial velocities based on phase difference from three readers, orientation from magnetometer, angular velocity from gyroscope, and acceleration from accelerometer: $\{r\}$, $\{r, \dot{r}\}$, $\{r, \theta_z, a_x, a_y\}$, $\{r, \theta_z, \omega_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, \omega_z, a_x, a_y\}$ corresponding to output vectors in Equation 4.15 or Equation 4.18 are implemented following estimation steps in subsection 4.2.2. Gauss-Newton and Levenberg-Marquardt methods obtain very close estimation results. Numerically, mean error and RMS error before and after estimation are compared in Table 5.1.

5.1 2D State Estimation Results Using Measurement States with Gauss-Newton and Levenberg-Marquardt Methods

Instantaneous trilateration in Table 5.1 is tag position (x, y) of minimum rms error to the three intersections of circles whose centers are the reader locations in Table 3.3 and radii are radial distances calculated based on RSS from three readers. State vectors in Equation 4.14 and Equation 10.21 are estimated from different measurement sensor outputs in the table.

Table 5.1: Mean Error and RMS Error of Position States from Instantaneous Trilateration and Estimation Results Using Different Sensor Outputs in Measurement Based on Nonlinear Least-Squares Estimation with Gauss-Newton and Levenberg-Marquardt Methods

Sensor Outputs	Mean Error [mm]	RMS Error [mm]	Stack Length	Enhancement Factor for Mean	Enhancement Factor for RMS
Instantaneous Trilateration	**15.9**	**38.8**	-	-	-
Estimation with Stack Length of 10					
r	16.0	33.2	10	1	1
$r\ \dot{r}$	**4.7**	**27.1**	10	**3**	**1**
$r\ \theta_z\ a_x\ a_y$	13.3	29.3	10	1	1
$r\ \theta_z\ \omega_z\ a_x\ a_y$	13.3	29.3	10	1	1
$r\ \dot{r}\ \theta_z\ a_x\ a_y$	**4.49**	**27.3**	10	**4**	**1**
$r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y$	**4.50**	**27.3**	10	**4**	**1**
Estimation with Optimal Stack Length for Mean Error					
r	12.5	35.1	4	1	1
$r\ \dot{r}$	**4.66**	**26.5**	14	**3**	**2**
$r\ \theta_z\ a_x\ a_y$	11.0	19.0	29	1	2
$r\ \theta_z\ \omega_z\ a_x\ a_y$	11.0	19.1	29	1	2
$r\ \dot{r}\ \theta_z\ a_x\ a_y$	**4.49**	**27.1**	11	**4**	**1**
$r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y$	**4.50**	**27.3**	10	**4**	**1**
Estimation with Optimal Stack Length for RMS Error					
r	14.0	28.7	34	1	1
$r\ \dot{r}$	**4.68**	**26.3**	17	**3**	**2**
$r\ \theta_z\ a_x\ a_y$	11.0	**18.9**	31	1	**2**
$r\ \theta_z\ \omega_z\ a_x\ a_y$	11.0	**19.0**	30	1	**2**
$r\ \dot{r}\ \theta_z\ a_x\ a_y$	**4.51**	**26.7**	13	**4**	**2**
$r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y$	**4.58**	**26.6**	15	**4**	**2**

Estimated mean error and RMS error of position states are calculated based on ground truth in Chapter 2 by combining errors along x and y axes. From Table 5.1, we can get that:

1. NLE achieves 4.49 mm mean error and 18.9 mm RMS error estimation, reducing mean error and RMS error to orders up to 4 and 2 respectively.

2. Using RF signals only (radial distance r and radial velocity $\dot{r}$, i.e., $\{r, \dot{r}\}$) achieves similar estimation results to using both RF and inertial signals $\{r\ \dot{r}\ \theta_z\ a_x\ a_y\}$, $\{r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y\}$ of 4.6 mm mean error with optimal stack length, which holds true for RMS error as well.

3. From estimation results with stack length 10, it can be inferred that the factor of enhancement for RMS error from different sensor outputs follows: $\{r\} < \{r, \theta_z, a_x, a_y\}$, $\{r, \theta_z, \omega_z, a_x, a_y\} < \{r, \dot{r}, \theta_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, \omega_z, a_x, a_y\} < \{r, \dot{r}\}$.

4. In both Gauss-Newton and Levenberg-Marquardt methods, NLE reduces RMS errors, i.e., enhancement factor for RMS error > 1 holds true for various sensor outputs.

5. Stack length does not affect too much on estimation accuracy, e.g., 27.1 mm RMS estimation error with stack length 10 compared to 26.3 mm RMS estimation error with optimal stack length 17 using sensor outputs $\{r, \dot{r}\}$.

5.2 2D State Estimation Results with Different Stack Lengths in Measurement

To investigate the impact of stack length in NLE, stack lengths of $1-40$ are iterated in NLE based estimation on measurement states. Mean Error and RMS error of estimation from different sensor outputs with different stack lengths are shown in Figure 5.1. In Figure 5.1, dotted lines are the measurement instantaneous trilaterations along x and y axes, i.e., 15.9 mm and 38.8 mm as shown in Table 5.1. Lines under the dotted line mean the stack length range in which estimation errors are smaller than measurement errors. Vice versa, lines above the dotted line mean that estimation errors are larger than measurement errors. It can be inferred from Figure 5.1 that with optimal stack length, estimation based on $\{r, \dot{r}\}$ is close to estimation based on $\{r, \dot{r}, \theta_z, a_x, a_y\}$, $\{r, \dot{r}, \theta_z, \omega_z, a_x, a_y\}$.

5.3 2D State Estimation Results with Different Sensor Outputs

It can be inferred from Figure 5.2 that 2D estimation results with RF signals only $\{r\ \dot{r}\}$ gain similar estimation results to signals with acceleration, orientation, and angular velocity $\{r\ \dot{r}\ \theta_z\ a_x\ a_y\}$, $\{r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y\}$. RF signals can achieve localization that is comparable to using both RF and inertial signals, similar to 1D results in [62].

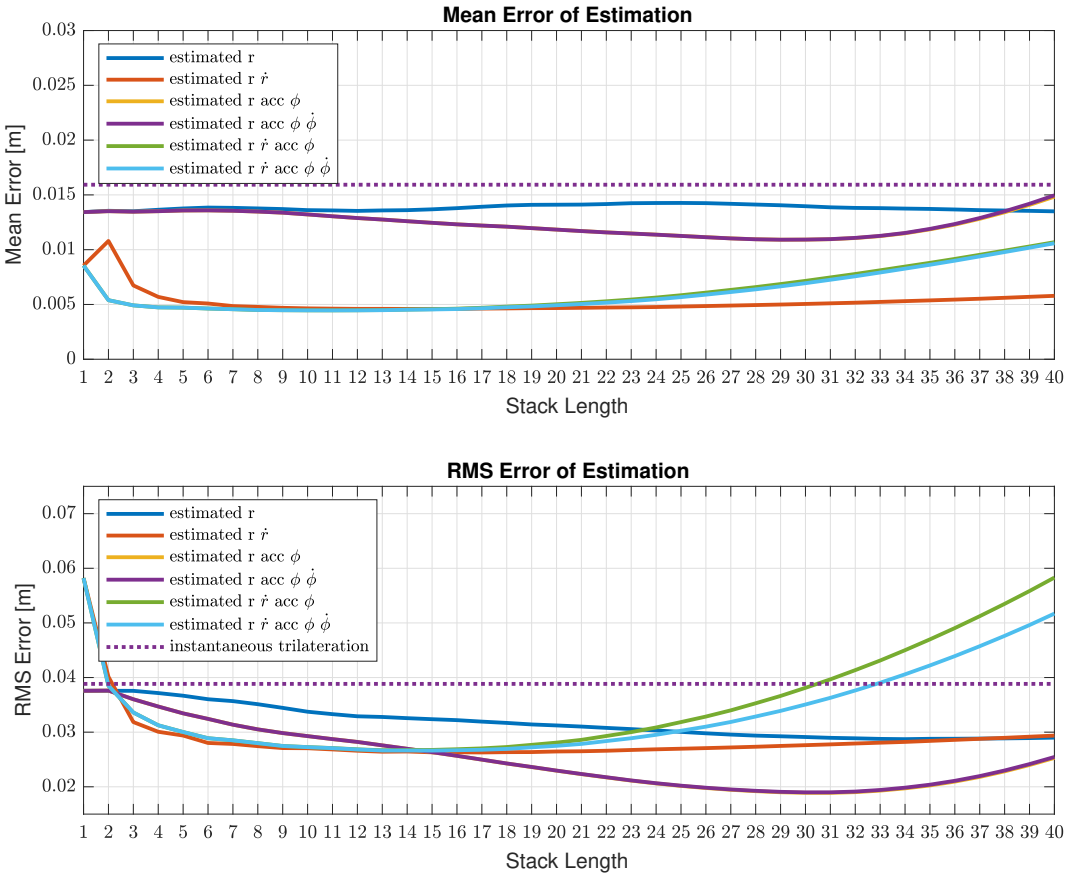
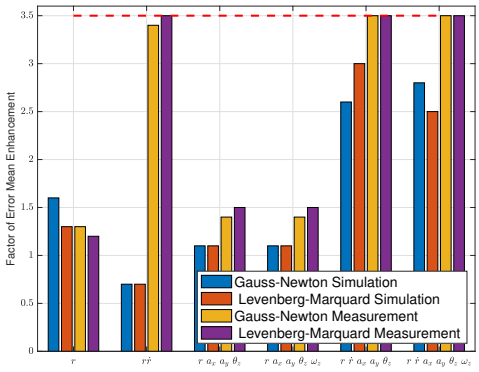


Figure 5.1: Mean Error and RMS Error of Estimated Position States with Different Stack Lengths in Measurement.

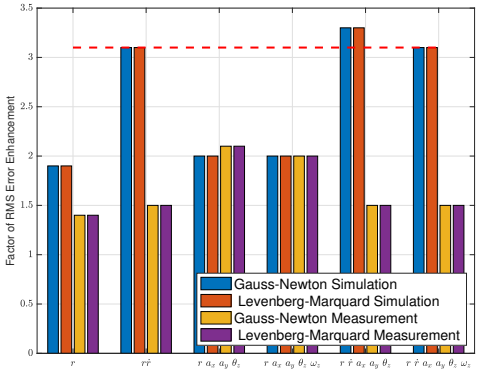
5.4 Factor of RMS Error Enhancement at Different Signal-to-Noise Ratios

It can be inferred from Figure 5.3a that even though RMS error of instantaneous trilateration increases exponentially, RMS error of estimated states increases linearly implying that 10 mm RMS error estimation is achievable for RF-based localization at various noise levels including highly noisy measurements. This trend is more obviously observed in Figure 5.3b, in which the blue line is exponentially increased RMS error of instantaneous trilateration and other lines are linearly increased RMS error of estimation with different sensor outputs based on NLE.

More specifically, as is shown in Figure 5.3a, RMS error before estimation is 135.9 mm and after estimation becomes 19.4 mm with sensor outputs of $\{r, \dot{r}\}$. Also, NLE

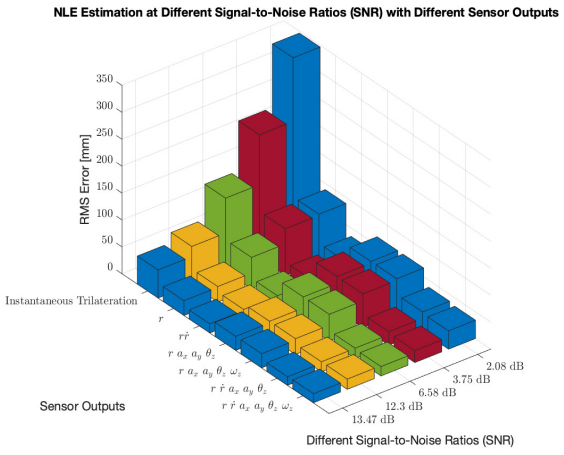


(a) Factor of Mean Error Enhancement with Different Sensor Outputs Based on Gauss-Newton and Levenberg-Marquardt Methods in Simulation and Measurement.

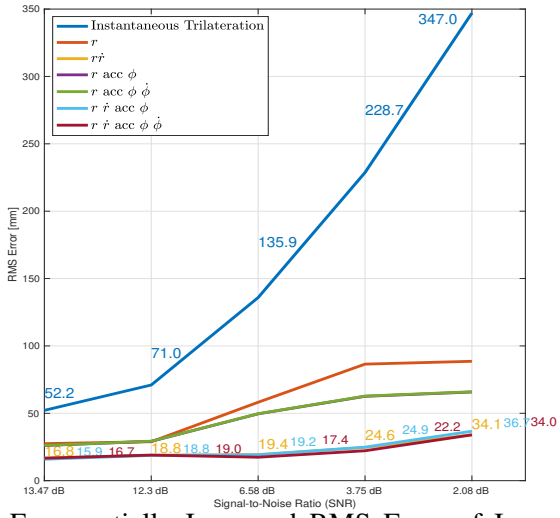


(b) Factor of RMS Error Enhancement with Different Sensor Outputs Based on Gauss-Newton and Levenberg-Marquardt Methods in Simulation and Measurement.

Figure 5.2: Factor of Mean Error and RMS Error Enhancement with Different Sensor Outputs Based on NLE.



(a) RMS Error at Different Signal-to-Noise Ratios (SNR) with Different Sensor Outputs Based on NLE in Simulation. Sensor Outputs of $\{r, \dot{r}\}$ is Always of Similar Estimation Error to Sensor Outputs of $\{r\ \dot{r}\ \theta_z\ a_x\ a_y\}$ and $\{r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y\}$ at Different Noise Levels, both of which Maintain 10mm Estimation RMS Error from 100mm Insantaneous trilateration.



(b) Exponentially Increased RMS Error of Instantaneous Trilateration and Linearly Increased RMS Error of Estimation based on NLE at Different Signal-to-Noise Ratios (SNR) with Different Sensor Outputs in Simulation, Indicating that the Factor of Enhancement for RMS in NLE Increases with the Increase of Noise Levels.

Figure 5.3: NLE Estimation Accuracy Maintains 10 mm on Simulated States at Different Noise Levels.

achieves 34.1mm RMS error estimation with sensor outputs of $\{r, \dot{r}\}$ for states of 347 mm RMS error. Therefore, estimation accuracy maintains for highly noisy states. Moreover, estimation with sensor outputs $\{r, \dot{r}\}$ is always of similar error to estimation with sensor outputs $\{r\ \dot{r}\ \theta_z\ a_x\ a_y\}$ and $\{r\ \dot{r}\ \theta_z\ \omega_z\ a_x\ a_y\}$ at different noise levels.

This increasing accuracy enhancement characteristic makes NLE maintain 10 mm RMS estimation error for highly noisy states. A possible reason of the increasing error elimination is error cancellation with stacked states, which is described in details in section 5.5.

5.5 Effects of Stack Length in NLE

A weighted gain of Kalman Filter based on noise covariances from measurement and state model makes Kalman Filter the best filter among all filters when the noise processes are Gaussian and also the best linear filter among all linear filters otherwise [62]. Discrete-time disturbance vector w accounts for the forces and will not be zero. Relatively, similar to Kalman Filter, if sensor noise v is high, then longer stack length that depends more on state-space model gets higher estimation accuracy. If sensor noise v is low, shorter stack length that depends more on instant measurement gets higher estimation accuracy as stated in Hypothsis 1. To verify the hypothesis, simulated states at different noise levels are estimated based on NLE.

Table 5.2: Optimal Stack Length for RMS Error Based on NLE on Simulated States at Different SNR

SNR (dB)	26.01	19.87	15.82	10.75	6.58	3.16	2.73	2.61	2.21	2.74
Stack Length	7	11	15	22	36	49	57	62	80	80

It can be derived from Table 5.2 that the lower sensor noise indicated by SNR is, the shorter optimal stack length for RMS error becomes. The higher sensor noise is, the longer optimal stack length for RMS error is, which verifies Hypothsis 1.

CHAPTER 6

RFID-BASED 3D LOCALIZATION SYSTEM SETUP

In this chapter, the RFID-based 3D localization system setup is described and displayed. Sensor signal collected both from RF and inertial sensors are illustrated in details. The 3D motion is described and shown. The setup of the 3D motion system is shown in Figure 6.1 and Figure 6.2, in which RFID Tag and IMU-embedded cellphone are combined together in the 3D motion and the motion profile is in Table 6.3.

1. RF Signal: Two pieces of information are sensed by the reader when the RF tag backscatters the data: the received signal amplitude $H_t(r)$ which is used to calculate the approximate position of the tag r relative to each reader and absolute phase $\phi(r)$ that is used to calculate the approximate tag velocity $\dot{r}$. Absolute phases from multiple frequencies are also used to derive position r relative to each reader.

2. Inertial Sensor Signal: In order to imitate an IMU sensor relayed on the RFID backscatter link, the system we are using applies RF tag and separate data-logging IMU sensor to meet the goals of our scientific study. Three pieces of inertial information are collected by the cellphone. Accelerations and angular velocities along x-, y-, z- axes of the cellphone are measured by the accelerometer and gyroscope of the embedded IMU from the cellphone along its body axes. Additionally, magnetic strengths along x-, y-, z- axes of the cellphone are measured by the embedded magnetometer.

3. 3D Motion Description: The combined unit is fixed to one arm of the 3D positioner. The combined unit moves along x- axis first, and then travels along y- axis, finally advances along z- axis. In the motion, four readers in 3D space sensed the RF signals back scattered from the tag. The locations of the four readers are listed in Table 6.1. In parallel, IMU sensor data including accelerations, angular velocities, and orientations are collected along

the body axes of the cellphone Galaxy S8.

In addition, Figure 6.1 presents the scheme of 3D motion profile, locations of the four readers, and the combination layout of antenna with cellphone. Figure 6.2 shows the picture of the setup, in which the 3D positioner, tag, cellphone, as well as four readers in the laboratory environment are presented.

The tag antenna is isotropic, and reader antenna is isotropic in both horizontal and vertical range. More specifications are in Table 6.1.

Table 6.1: Locations and Specifications of Four Readers in 3D Motion

Reader Index	Coordinates	Location [m]	Frequency	Power
Reader $\#1$	(x_1, y_1, z_1)	(0.00, 0.00, 0.865)	5.80 GHz	1 W
Reader $\#2$	(x_2, y_2, z_2)	(2.29, 0.00, 1.270)	5.83 GHz	1 W
Reader $\#3$	(x_3, y_3, z_3)	(2.29, 2.52, 0.865)	5.82 GHz	1 W
Reader $\#4$	(x_4, y_4, z_4)	(0.00, 2.52, 1.270)	5.85 GHz	1 W

The tag initially located at origin O in Table 6.2 and started from the rest, and then moved as described in (1) (2) (3) and (4) motion profiles.

Table 6.2: Location of the Origin

Origin	Coordinates	Location [m]
O	(x_0, y_0, z_0)	(1.03, 1.31, 1.03)

The details of two sets of moving profile are described in the three paragraphs:

(1) x Axis: Firstly, the tag accelerated at 0.128 m/s^2 to attain a velocity of 0.192 m/s and moved at this constant speed for 0.5 s before decelerated at a constant rate of -0.128 m/s^2 to static, travelling a total distance of 0.384 m away along x- axis.

(2) y Axis: Afterwards, the tag accelerated at 0.1472 m/s^2 to attain a velocity of 0.1472 m/s and moved at this constant speed for 1 s before decelerating at a constant rate of -0.1472 m/s^2 to static, travelling a total distance of 0.2944 m away along y- axis.

(3) z Axis: Hereafter, the tag accelerated at 0.0448 m/s^2 to attain a velocity of 0.0448 m/s and then decelerated at a constant rate of -0.0448 m/s^2 to static, travelling a total distance of 0.0448 m away along z- axis.

(4) xyz Axes Moving Together: After moving back to origin O, the tag moves along x-, y-, and z- axes simultaneously. The tag accelerated at 0.128 m/s^2 to attain a velocity of 0.128 m/s and moved at this constant speed for 2 s before decelerated at a constant rate of -0.128 m/s^2 to static along x- axis. At the same time, the tag accelerated at 0.1472 m/s^2 to attain a velocity of 0.1472 m/s before decelerating at a constant rate of -0.1472 m/s^2 to static along y- axis. Simultaneously, the tag accelerated at 0.0448 m/s^2 to attain a velocity of 0.0448 m/s and then decelerated at a constant rate of -0.0448 m/s^2 to static along z- axis. The total travel distances along x-, y-, and z- axes are the same as motion in (1) (2) (3).

The summary of the two sets of motion profiles including total distance travelled, maximal velocities, and accelerations are shown in Table 6.3.

Table 6.3: 3D Motion Profiles of Total Distances Travelled, Maximal Velocities, and Accelerations Attained Along x- y- z- Axes

Moving Along x- Axis, Then y- Axis, and Finally Along z- Axis

	x- Axis	y- Axis	z- Axis
Total Distance Travelled	0.384 m	0.2944 m	0.0448 m
Velocity	0.192 m/s	0.1472 m/s	0.0448 m/s
Acceleration	0.128 m/s^2	0.1472 m/s^2	0.0448 m/s^2
Time Duration of Acceleration - Constant Velocity -Deacceleration	1.5 s - 0.5 s - 1.5 s	1 s - 1 s - 1 s	1 s - 0 s - 1 s

Moving Together Along x- y- z- Axes Simultaneously

	x- Axis	y- Axis	z- Axis
Total Distance Travelled	0.384 m	0.2944 m	0.0448 m
Velocity	0.128 m/s	0.1472 m/s	0.0448 m/s
Acceleration	0.128 m/s^2	0.1472 m/s^2	0.0448 m/s^2
Time Duration of Acceleration - Constant Velocity -Deacceleration	1 s - 2 s - 1 s	1 s - 1 s - 1 s	1 s - 0 s - 1 s

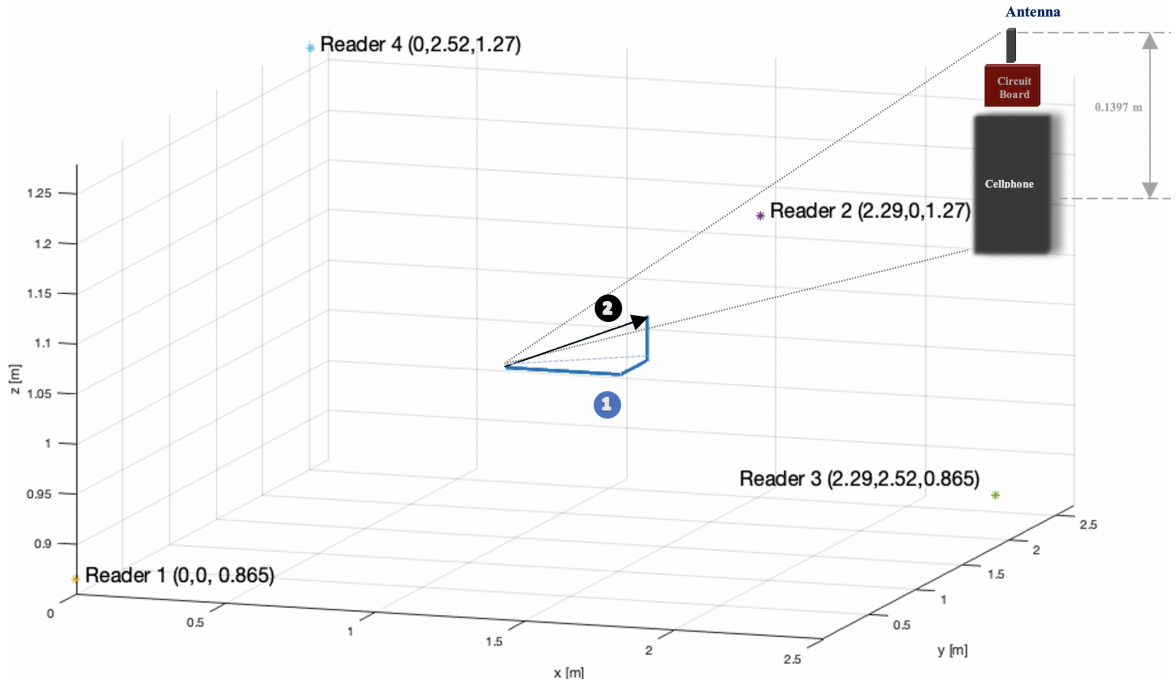


Figure 6.1: Scheme of 3D Motion Setup. Four Readers at Different Frequencies Sent and Sensed the Back-Scattered RF Signals From the Tag. IMU Collected Accelerations, Angular Velocities, and Orientations Along the Body Axes of the Cellphone, Which is Attached to the z Arm of the 3D Positioner.

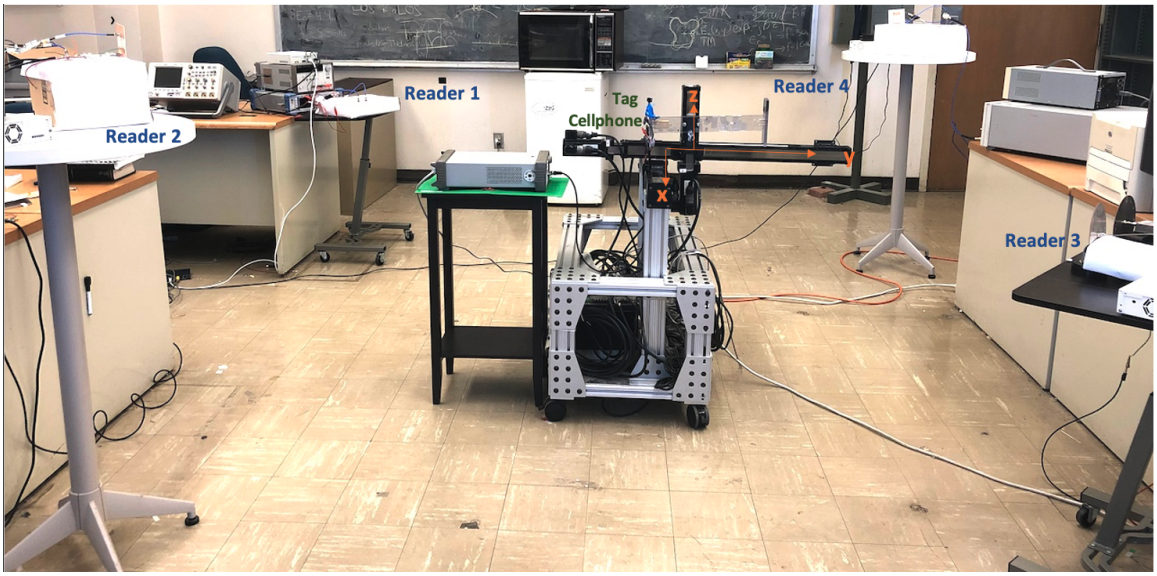


Figure 6.2: Picture of 3D Motion Setup. Four Readers Sent and Sensed the Back-Scattered RF Signals From the Tag. Accelerations, Angular Velocities, and Magnetic Strength of IMU Along Body Axes are Collected From the Accelerometer, Gyroscope, and Magnetometer of the Attached Cellphone Galaxy S8 in the Laboratory Environment.

CHAPTER 7

SIMULATION VERIFICATION ON 3D DATA IN INDOOR MULTIPATH

In this chapter, a simulation model of RFID-based 3D localization system is built. The simulation model includes multipath effects in the laboratory and indoor environment referring to the 3D localization system for semi-active tunnelling diode modulator tag in chapter 6. Based on model, noise distribution, and multipath model in chapter 2, RSS and RSP of tag from given reader are simulated in Figure 7.2 and Figure 7.3. Simulated magnitude and phase are very similar to measurement, which enables replacing exhaustive experiments in new environment with indoor multipath. [1]

7.1 Verification for RF Radial Distances and Radial Velocities in 3D Motion

Based on the motion profile in chapter 6, 3D coordinates along x- y- and z- axes in the motion are generated. Based on the four reader locations in Table 6.1 and origin coordinate in Table 6.2, radial distances from the four readers #1, #2, #3, #4 are derived from 3D coordinates as shown in Figure 7.1.

7.2 Verification for Multipath Effects on Magnitude and Phase

Parameters of the simulation based on tunnel diode modulator tag in Table 7.1 are input into the simulation model with multipath and noise distribution in subsection 7.2.1 - subsection 7.2.3. Similar to 1D and 2D simulation, with input motion profile, magnitude and phase measured from each reader are derived and compared to measured magnitude and phase as shown in Figure 7.2 and Figure 7.3.

[1] Simulation Model is in: `https://github.com/qq3575022/3DIndoorStateEstimationRFIDMotionCapture/tree/main/Chapter7`

Table 7.1: Simulation Template and Sample Parameters in 3D Motion

P_T	Reader's Transmitted Power	1W
G_T	Reader's Transmitter Antenna Gain	-16.15 dBi, -6.15 dBi,-14.60 dBi, -17.61 dBi
G_t	Tunnel Tag's Antenna Gain	-18.55 dBi
X	Polarization Mismatch Between Reader's and Tag's Antenna	0.85
R	Input impedance of the System	15 Ω
M	Load Modulation Factor of the Tag	1.0
f	Carrier Frequency	5.80 GHz, 5.83 GHz, 5.82 GHz, 5.85 GHz
K	K factor in Rician Distribution	40, 400, 11, 56
σ	Standard Deviation of Added Phase Noise	0.02

7.2.1 Multipath Reflections in the Laboratory

Power received by a reader from tag is in multipath propagation. Six reflections are included in the laboratory of dimension 7.32 m $\times$ 6.8 m $\times$ 3 m, i.e, one reflection from each wall.

$$P_R(r) = \frac{P_T G'_T G'_t \lambda^2 X' M}{(4\pi)^2} \left| \frac{1}{r_0} \exp(-j\frac{2\pi r_0}{\lambda}) + \sum_{i=1}^{6} \Gamma_i \frac{1}{r_i} \exp(-j\frac{2\pi r_i}{\lambda}) \right|^2 \tag{7.1}$$

where r_0 is the length of the direct path, r_i, $i = 1, ..., 6$ is the length of ith reflection path, $N = 6$ is the total number of reflection paths, and Γ_i is the complex reflection coefficient for the ith reflection path [61].

Measured magnitudes and phase measured from the four readers as well as simulated magnitudes and phase with multipath reflections from four walls, ceiling and floor are shown in Figure 7.2 and Figure 7.3. It can be inferred from Figure 7.2 that multipath reflections introduce magnitude shift and quasi-sinusoidal into the amplitude during motion phases of the tag, which is similar to measurement data. In addition, simulated phase in

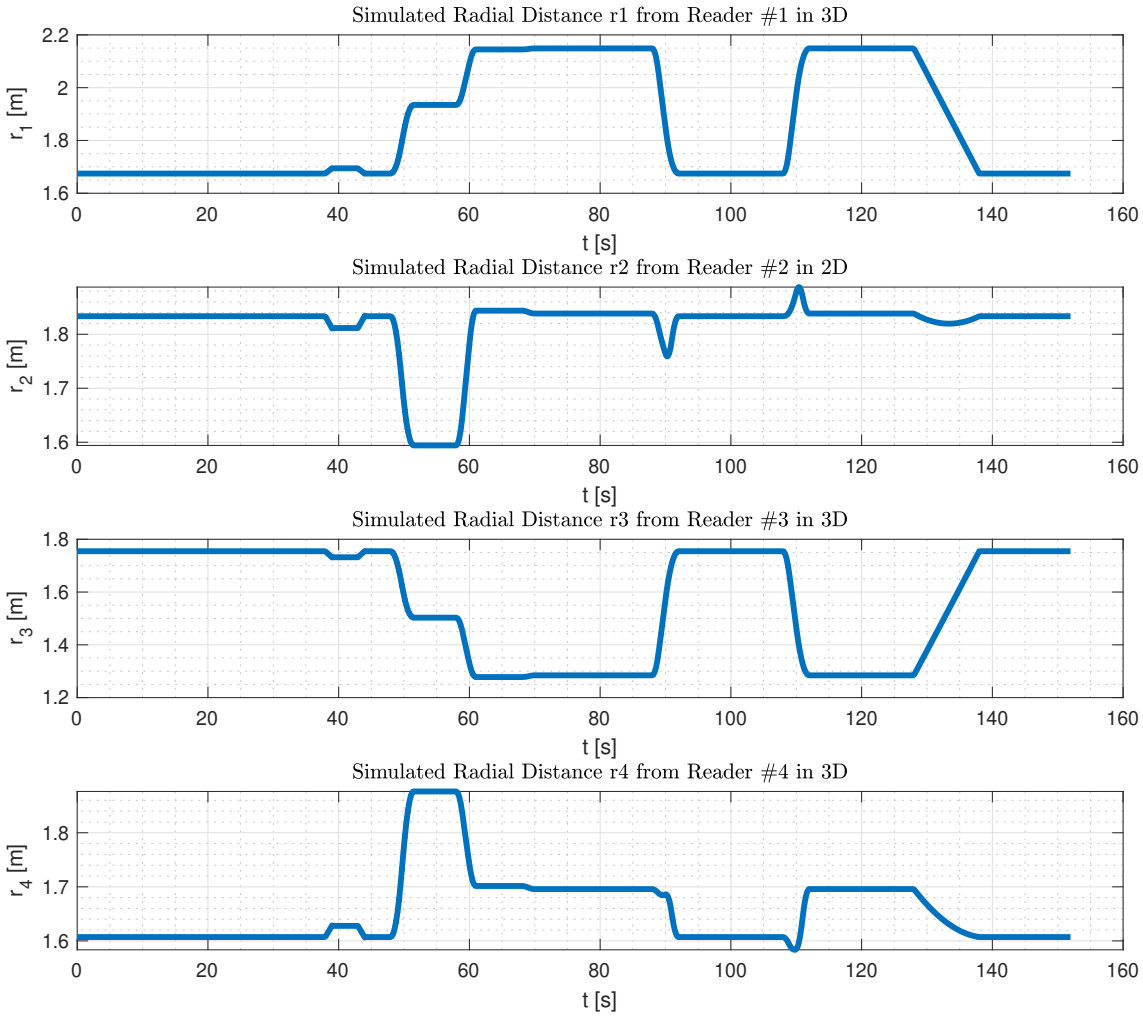


Figure 7.1: Ground Truth Radial Distances r_1 r_2 r_3 and r_4 from Four Readers Based on the 3D Motion Trajectory Described in chapter 6 and Reader Locations in Table 6.1 and Origin Coordinate in Table 6.2

multipath in Figure 7.3 is similar to measured phase as well.

7.2.2 Rician Distribution of Magnitude Noise

In addition to deterministic line-of-sight and multi-path reflections, added magnitude noise follows Rician distribution. Correspondingly, added noise in signal magnitude in simulation is generated from random generator `makedist('Rician')` in Matlab, which follows Rician distribution of mean value $\sqrt{H'}$ that H' is the average of $|H|$ and standard

deviation σ. The fluctuations of magnitude in Figure 7.2 are generated based on the Rician distribution, which is similar to fluctuations in measurement data in static.

7.2.3 Noise Distribution of Phase

Moreover, noise distribution for phase follows the distribution in Equation 2.23, which is added to simulated phase in the simulation model accounting for the thermal noise in the channel.

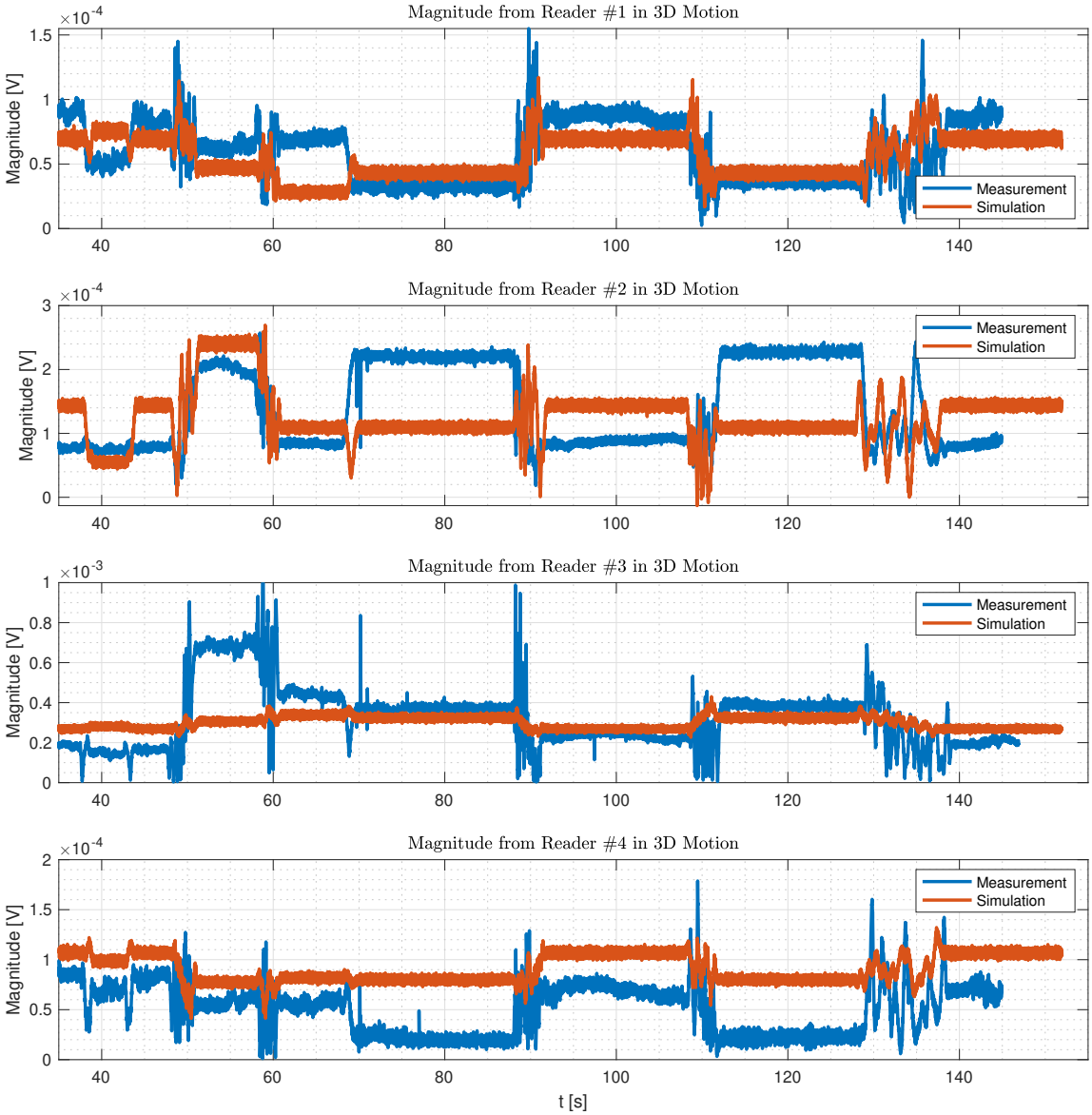


Figure 7.2: Measured and Simulated Magnitude $H(r)$ with 6 Reflections and Noise of Rician distribution from Four Readers with the Input 3D Trajectory in chapter 6.

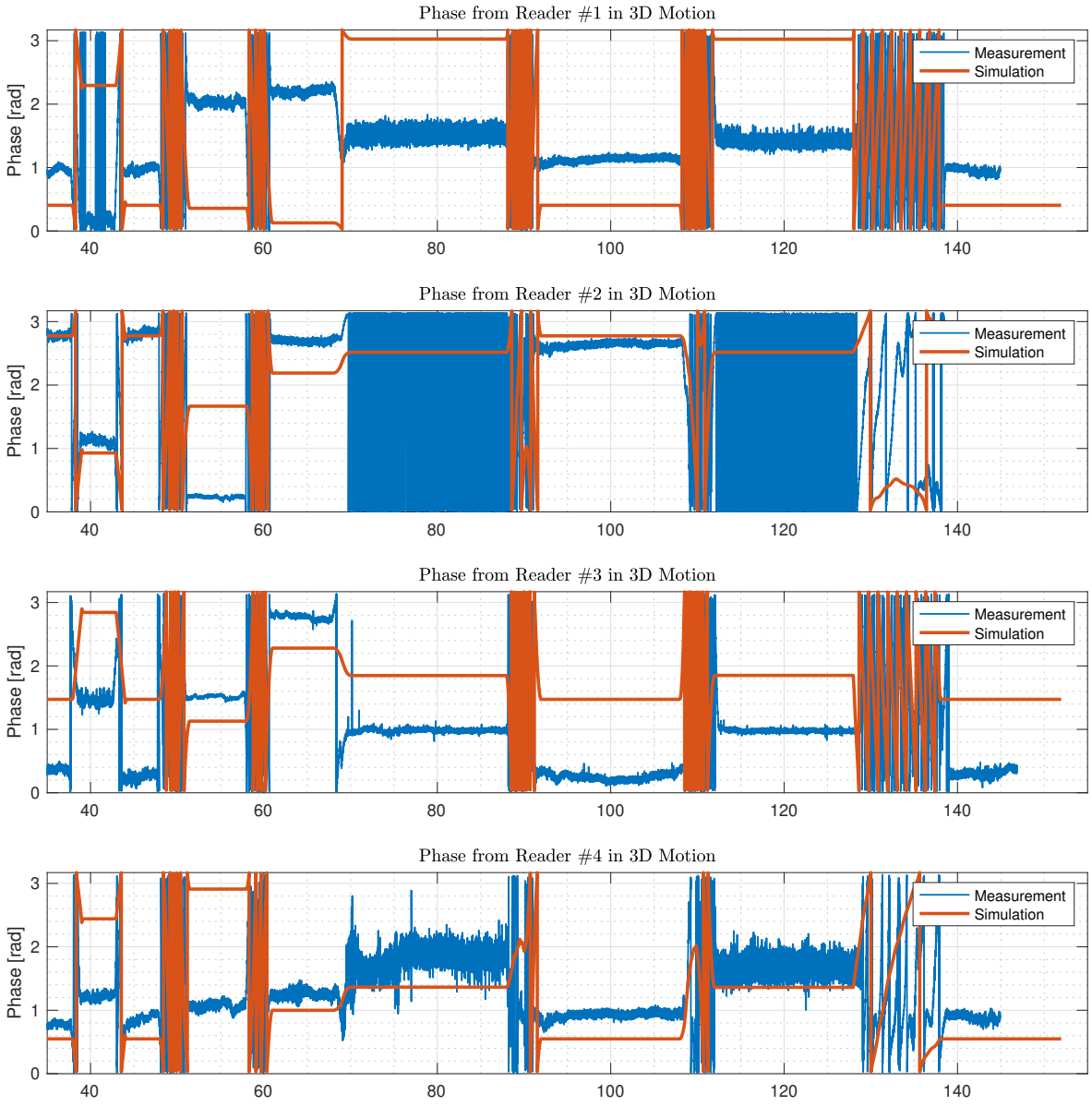


Figure 7.3: Measured Phase ϕ and Simulated Phase ϕ with 6 Reflections and Noise of Rician distribution from Four Readers with the Input Trajectory Along x- Then y- and Finally z- Axis. After Moving Back, Moving Along x- y- z- Axes All Together as Described in chapter 6.

CHAPTER 8

THREE KINDS OF POSITION SENSORS

In this chapter, three kinds of position sensors: magnitude based, ToF based, and ToF based in tracking along the 3D motion in chapter 6 are compared to each other in radial distances from four readers and 3D coordinates based on triangulation. Both LOS and multipath environment are involved. In addition, both static and tracking in motion are included in comparison. [1]

8.1 Magnitude Based Position Sensor

Lots of RFID applications use magnitude data for position sensor. In mutipath environment illustrated in section 7.2, radial distances of tunnel diode modulator tag can be derived from magnitude based on Equation 2.13 as

$$r = \frac{\sqrt{2P_T G'_T G'_t \lambda^2 X' M R}}{4\pi H_t(r)} \tag{8.1}$$

In which,

$$H_t(r) = \frac{\sqrt{2P_T G'_T G'_t \lambda^2 X' M R}}{4\pi} \left| \frac{1}{r_0} \exp(-j\frac{2\pi r_0}{\lambda}) + \sum_{i=1}^{6} \Gamma_i \frac{1}{r_i} \exp(-j\frac{2\pi r_i}{\lambda}) \right| \tag{8.2}$$

where r_0 is the length of the direct path, r_i, $i = 1, ..., 6$ is the length of $i-$th reflection path, $N = 6$ is the total number of reflection paths, and Γ_i is the complex reflection coefficient for the $i-$th reflection path [61].

Based on Equation 8.1 and Equation 8.2, derived radial distances from simulated magnitudes are shown in Figure 8.1, in which blue lines are radial distances derived from mag-

[1] https://github.com/qq3575022/3DIndoorStateEstimationRFIDMotionCapture/tree/main/Chapter8

nitude in case of multipath in section 7.2 and red lines ares ground truth radial distance in noise-free and multipath free environment in Figure 7.1.

It can be inferred from the figure that compared to ground truth radial distances, magnitude derived distance has offset in static states and is also of sinusoidal fluctuations in the motion.

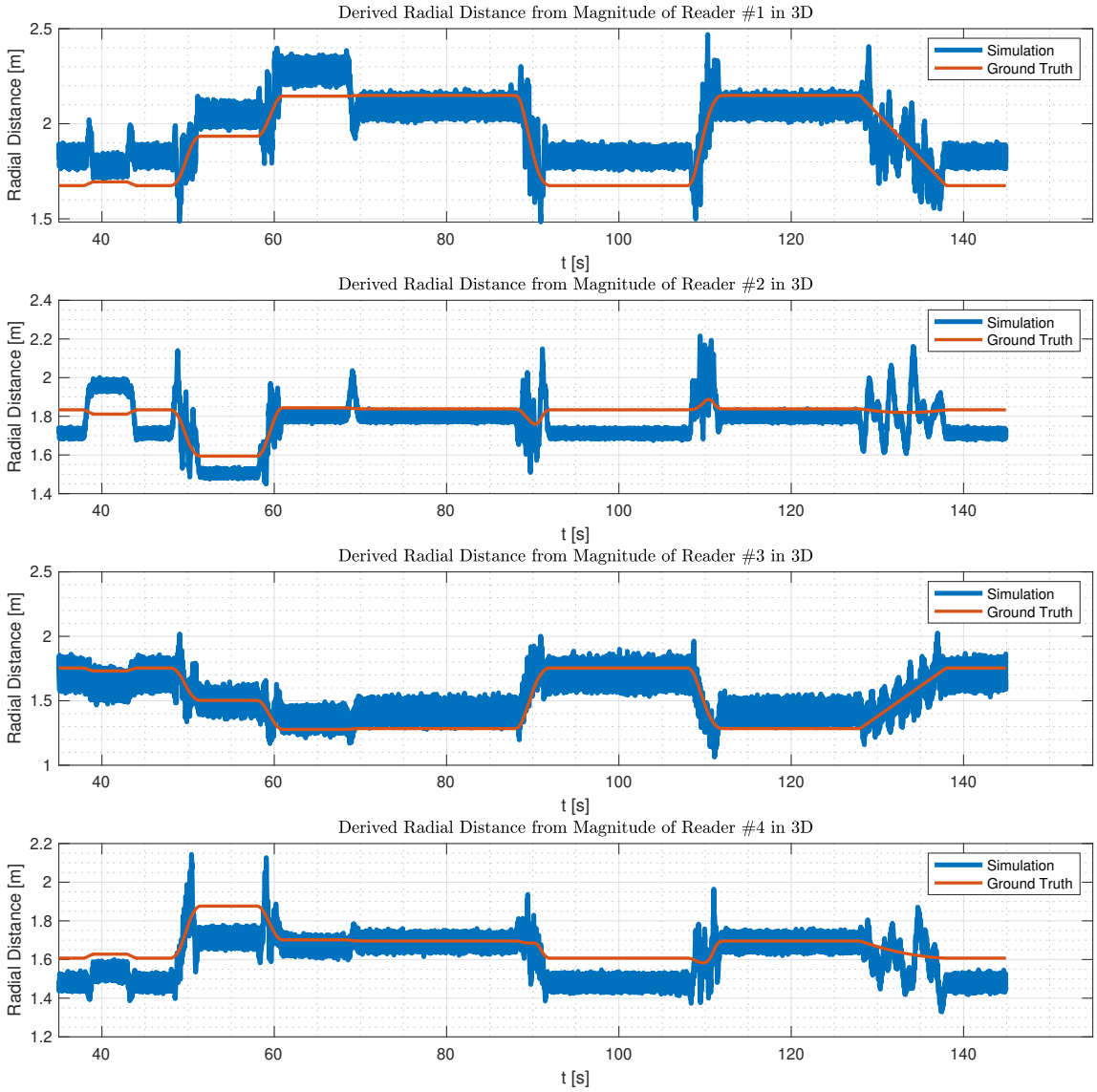


Figure 8.1: Simulated Derived Radial Distance from Magnitude in Multipath from Four Readers with the Input Trajectory Along x- Then y- and Finally z- Axis. After Moving Back, Moving Along x- y- z- Axes All Together as Described in chapter 6.

8.2 Time-of-Flight (ToF) Based Position Sensor

Based on two or more phases, the travelling distance can be obtained from two or more frequencies based on Time-of-Flight (ToF). As for two frequencies:

$$\begin{aligned} \phi_1 &= \frac{4\pi r_0}{\lambda_1} = \frac{4\pi f_1 r_0}{c} \\ \phi_2 &= \frac{4\pi r_0}{\lambda_2} = \frac{4\pi f_2 r_0}{c} \end{aligned} \tag{8.3}$$

Hence,

$$\begin{aligned} r_1 &= \frac{c\phi_1}{4\pi f_1} \\ r_2 &= \frac{c\phi_2}{4\pi f_2} \end{aligned} \tag{8.4}$$

Based on the derivation in [64]

$$r_1 = r_2 = \frac{c(\phi_1 - \phi_2)}{4\pi(f_1 - f_2)} \tag{8.5}$$

8.2.1 Ambiguity Removal of Duplicate Distances

Since phase is always within $[0, 2\pi)$. Phases ϕ_1, ϕ_2, ... , ϕ_M are of period 2π. To avoid ambiguity distance derived out of the 2π period in phases, the maximal distance that can be identified with frequency difference without duplicate distance is

$$r_{max} = \frac{2\pi c}{4\pi(f_1 - f_2)} \tag{8.6}$$

In this work, frequency difference $\Delta f = f_1 - f_2 = 1$ MHz, then $r_{max} = 150$ m, which is much larger than the dimension of the laboratory and is used in the simulation. Hence, ambiguity of duplicate distance is avoided.

In addition, for 5.8 GHz band, the bandwidth is 150 MHz making multiple frequency hopping of 1 MHz difference feasible.

8.2.2 Position Sensor from Multiple Frequencies

For multiple frequencies, a fitted line from multiple phases ϕ_1, ϕ_2, ϕ_3, ... , ϕ_M can be derived as shown in Figure 8.2. Four hopping frequencies are used in the fitting, in which the phase differences are $\Delta\phi_1$ corresponds to frequency difference Δf_1 and phase difference $\Delta\phi_2$ corresponds to frequency difference Δf_2. Then slope of phase difference α corresponds to the derived radial distance r.

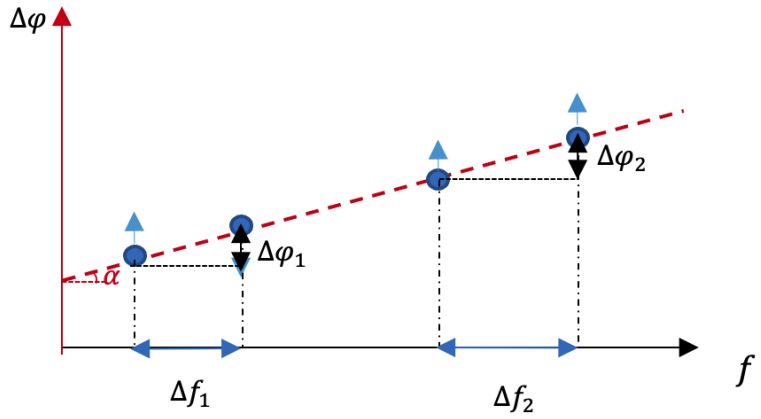


Figure 8.2: Scheme of Line Fitting of Four Hopping Frequencies. Phase difference $\Delta\phi_1$ corresponds to frequency difference Δf_1 and phase difference $\Delta\phi_2$ corresponds to frequency difference Δf_2. The slope α is the derived distance r.

Based on Equation 8.5, derived radial distances from simulated phases of two frequencies of difference $\Delta f = 1$ MHz using ToF are shown in Figure 8.3, in which blue lines are radial distances derived from ToF in case of multipath and red lines ares ground truth radial distance same as in Figure 7.1. Therefore, the ToF derived distance is also affected by multipath effect. It is also shown in the figure that ToF based position sensor also changes quasi-sinusoidal into the phase during motion of the tag.

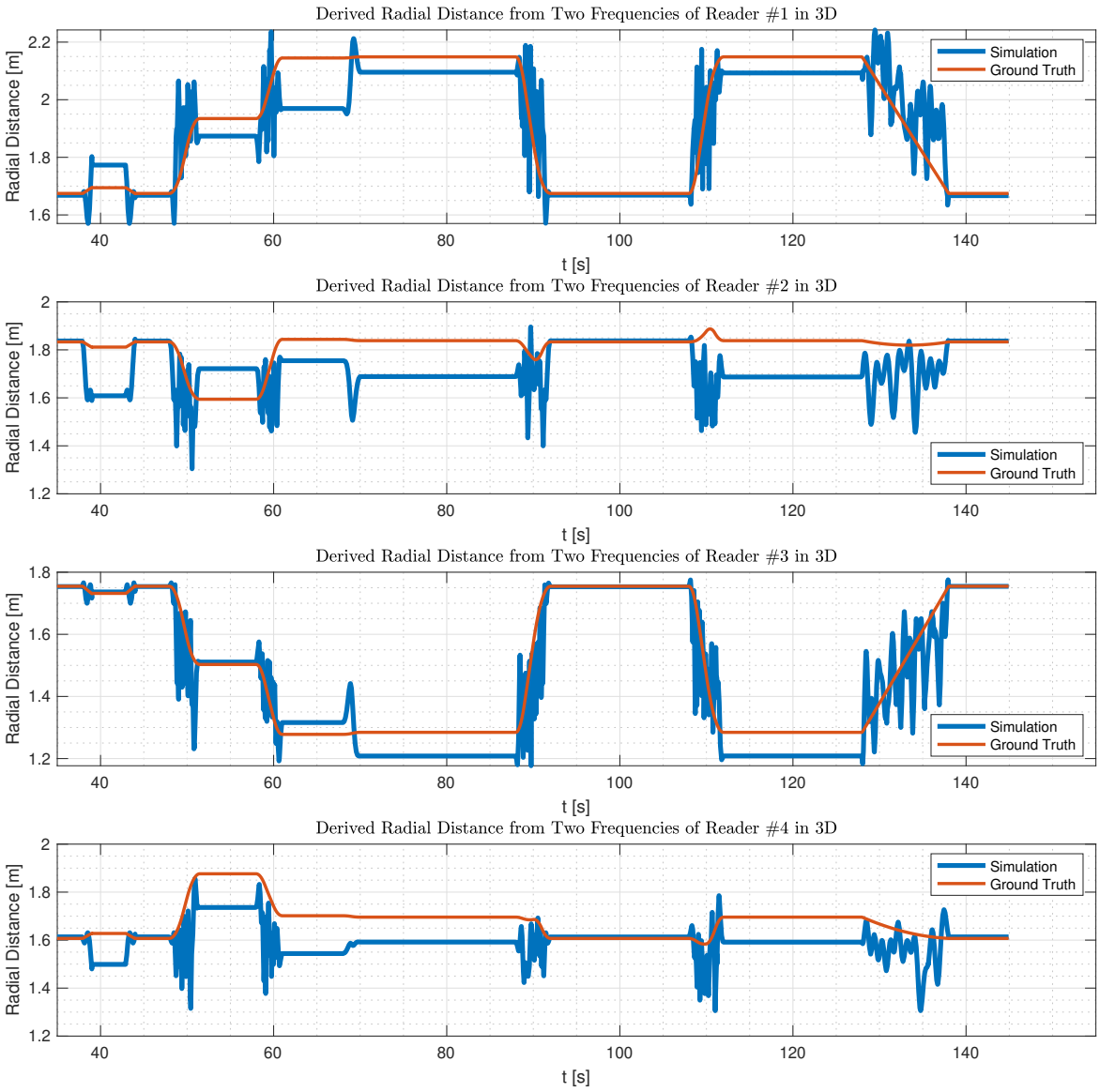


Figure 8.3: Simulated ToF of Phase Derived Radial Distance from Two Frequencies in Multipath from Four Readers at Sampling Frequency of 4847 Hz and Frequency Difference $\Delta f = 1$ MHz with the Input Trajectory Along x- Then y- and Finally z- Axis. After Moving Back, Moving Along x- y- z- Axes All Together as Described in chapter 6.

8.3 Time-of-Flight (ToF) Based Position Sensor in Tracking

Due to requirement of FCC, the maximal time duration for each frequency has a limit. As for 75 channels of 5.8 GHz, the average time of occupancy on any frequency must not

exceed 0.4 s in any 30 s period. In addition, two frequencies sent from the reader saturate the tag [65].

In order to realize real-time tracking for RFID-based motion-capture systems, at each time stamp only signal from one frequency is sent. The frequencies of adjacent sample are different. Hence, the distance from each reader can be derived as following. Similar to ToF based positioner sensor, the travelling distance can be obtained from two frequencies from samples at different time stamps as

$$\begin{aligned} \phi_1 &= \frac{4\pi r_1}{\lambda_1} = \frac{4\pi f_1 r_1}{c} \\ \phi_2 &= \frac{4\pi r_0}{\lambda_2} = \frac{4\pi f_2 r_0}{c} \end{aligned} \tag{8.7}$$

Hence,

$$\begin{aligned} r_1 &= \frac{c\phi_1}{4\pi f_1} \\ r_2 &= \frac{c\phi_2}{4\pi f_2} \end{aligned} \tag{8.8}$$

Based on the derivation in [64]

$$r_0' = \frac{c(\phi_1 - \phi_2)}{4\pi(f_1 - f_2)} \tag{8.9}$$

Where r_0' is between r_1 and r_2. If the sampling time t_s is small enough, then $r_1 - r_2 \ll 1$ m. Since the sampling frequency is usually $f_i \gg 1$ MHz $\forall i = 1, 2$ and moving speed is mostly < 50 m/s, the difference $r_1 - r_2 \ll 5 \times 10^{-5}$ m, which meets the centimeter-level localization accuracy requirement.

The derived radial distances based on phase samples at adjacent time stamps of different frequencies are shown in Figure 8.4. In the figure, blue lines are derived radial distances at adjacent samples of different frequencies and red lines are ground truth radial distances same as in Figure 7.1. It is shown in the figure that derived radial distances are similar to the noise-free ground-truth trajectory and also similar to radial distances derived from

ToF based position sensor in section 8.2. In addition, ToF based position sensor in tracking derived distance is also affected by multi-path effect.

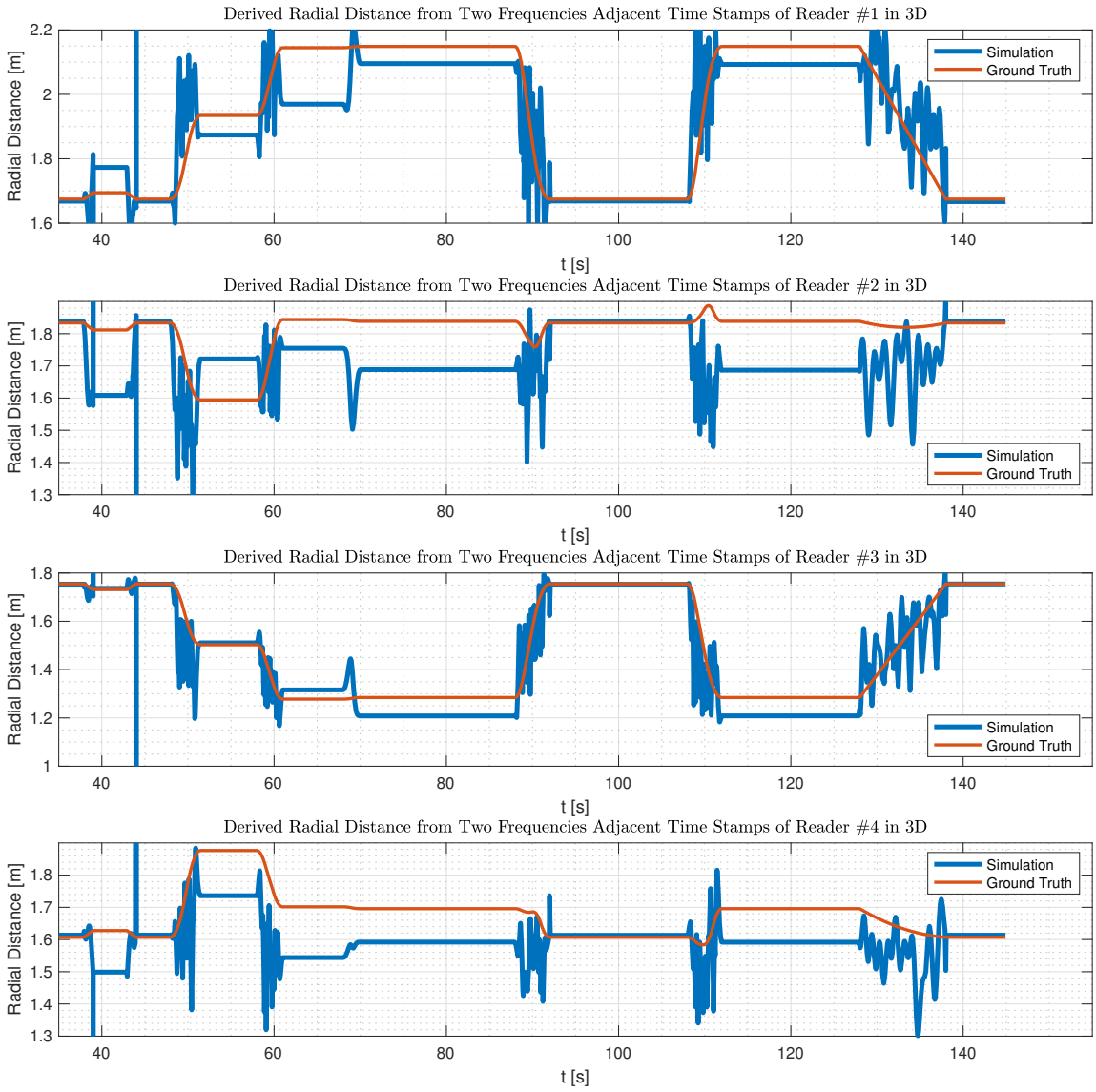


Figure 8.4: Simulated Phase Derived Radial Distance from Different Frequencies at Different time stamps in Multipath from Four Readers at Sampling Frequency of 4847 Hz and Frequency Difference $\Delta f = 1$ MHz with the Input Trajectory Along x- Then y- and Finally z- Axis. After Moving Back, Moving Along x- y- z- Axes All Together as Described in chapter 6.

Hence, ToF based position sensor in tracking using different frequencies at adjacent time stamps is able to achieve real-time localization and tracking for RFID based motion-

capture systems. Therefore, ToF based position sensor is able to be used in tracking as a replacement of magnitude based position sensor.

8.4 The Comparison of Three Kinds of Position Sensors

The comparisons of three kinds of position sensors: magnitude based position sensor in section 8.1, ToF based position sensor in section 8.2, and ToF based position sensor in tracking in section 8.3 is shown in Table 8.1 and Table 8.2. More specifically, RMS error of derived radial distance based on each position sensor compared to ground truth radial distance relative to four readers is shown in Table 8.1. Furthermore, RMS error of 3D coordinates (x, y, z) derived from radial distances relative to four readers based on triangulation in subsection 8.4.1 compared to ground truth coordinates for each position sensor is shown in Table 8.2.

8.4.1 Triangulation Derived 3D Coordinates from Four Radial Distances

The 3D coordinates (x, y, z) derived from the four radial distances based on triangulation is shown as following. From the four equations of deriving radial distances, we have three difference equations in matrix representation

$$\begin{bmatrix} r_1^2 - r_4^2 \\ r_2^2 - r_4^2 \\ r_3^2 - r_4^2 \end{bmatrix} = 2 \begin{bmatrix} x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \\ x_4 - x_2 & y_4 - y_2 & z_4 - z_2 \\ x_4 - x_3 & y_4 - y_3 & z_4 - z_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} x_1^2 - x_4^2 + y_1^2 - y_4^2 + z_1^2 - z_4^2 \\ x_2^2 - x_4^2 + y_2^2 - y_4^2 + z_2^2 - z_4^2 \\ x_3^2 - x_4^2 + y_3^2 - y_4^2 + z_3^2 - z_4^2 \end{bmatrix} \tag{8.10}$$

Therefore,

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \frac{1}{2} F^{-1}(g + u) \tag{8.11}$$

where

$$F = \begin{bmatrix} x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \\ x_4 - x_2 & y_4 - y_2 & z_4 - z_2 \\ x_4 - x_3 & y_4 - y_3 & z_4 - z_3 \end{bmatrix}, g = \begin{bmatrix} x_1^2 - x_4^2 + y_1^2 - y_4^2 + z_1^2 - z_4^2 \\ x_2^2 - x_4^2 + y_2^2 - y_4^2 + z_2^2 - z_4^2 \\ x_3^2 - x_4^2 + y_3^2 - y_4^2 + z_3^2 - z_4^2 \end{bmatrix}, u = \begin{bmatrix} r_1^2 - r_4^2 \\ r_2^2 - r_4^2 \\ r_3^2 - r_4^2 \end{bmatrix}$$

Then the 3D coordinates vector $[x, y, z]^T$ can be derived from the four radial distances r_1, r_2, r_3, r_4 based on Equation 8.11.

8.4.2 Comparison of Three Kinds of Position Sensors in Static/Tracking and LOS/Multipath

To compare the accuracy of three kinds of position sensors in case of line-of-sight (LOS) and multipath scenarios in static and tracking with motion, radial distances and 3D coordinates based on triangulation in subsection 8.4.1 based on the three kinds of position sensors are shown in Table 8.1 and Table 8.2.

In the comparison, all simulations are at 5.8 GHz with a sampling rate of 4847 Hz using tunnel diode modulator tag, which are the same as in measurement data. The multipath model employed was the statistical and 6-reflection model with parameters compared to measured magnitude. The MMSE model from triangulation in subsection 8.4.1 was used for the 3D coordinates for four-reader calculation.

1. Other assumptions in the simulation are as following, which holds true for all simulations

1. Average SNR used in all simulations is K = 40, 100, 11, 56 for the four readers respectively as shown in Table 7.1.

2. Standard deviation of noise for all phases is 0.02 as shown in Table 7.1.

3. The frequency difference of all ToF based position sensor in static/tracking and in LOS/multipath is Δf = 1 MHz. In addition, all ToF based position sensors use the same 4 frequencies.

4. The motion profile of all position sensors in static or tracking is the same as 3D motion trajectory described in chapter 6.

2. Radial Distances from Four Readers Respectively Compared to Ground truth

Table 8.1: Comparison of Three Position Sensors Based on Radial Distances from Four Readers Respectively Compared to Ground Truth 3D Trajectory in chapter 6

			RMS Error for Individual Reader			
			Reader 1 [mm]	Reader 2 [mm]	Reader3 [mm]	Reader 4 [mm]
Line-of-Sight (LOS)	Static	Simulated RSS	97.5	9.3	278.1	58.6
		Simulated ToF	23.8	25.2	26.2	23.2
	Tracking with Motion	Simulated RSS	97.5	9.3	278.1	58.6
		Simulated ToF	164.7	50.6	163.8	61.1
Multipath	Static	Simulated RSS	160.8	109.9	288.9	151.6
		Simulated ToF	40.6	18.2	31.1	25.7
	Tracking with Motion	Measured RSS	175.6	363.9	643.8	203.3
		Simulated RSS	160.8	109.9	288.9	151.6
		Simulated ToF	151.9	118.8	136.6	155.7

In the table:

1. Static means that the measurement is taken at each stationary point and the tag doesn't move in the simulation/measurement until data are taken done along the same motion profile as in tracking.

2. Tracking with motion, on the contrary, describes the tagged object moving along the motion profile regardless of whether the data are taken in the experiment and simulation.

3. Simulated RSS is the magnitude based position sensor in section 8.1 to get distance r from magnitude $H(r)$ due to the attenuation of power w.r.t. the increase of distance r as in Equation 8.1.

4. Simulated ToF describes ToF based position sensor in section 8.2 and in ToF based position sensor in tracking in section 8.3 using the phase difference to get distance r from the slope of $\Delta\phi$ w.r.t frequency difference Δf due to the phase change related to distance as shown in Equation 2.9.

It can be inferred from Table 8.1 that simulated RSS based position sensor is affected more by multipath than ToF based position sensor. In addition, ToF in tracking is of higher RMS error compared to ToF, but is still smaller than RSS based position sensor in multipath scenario.

3. 3D Coordinates Derived From Four Radial Distances Compared to Ground truth

Table 8.2: Comparison of Three Position Estimators Based on 3D Coordinates (x, y, z) Using Triangulation in subsection 8.4.1 Compared to Ground Truth Trajectory in chapter 6

			RMS Error for 4 Readers		
			3D [mm]	xy-plane [mm]	z-axis [mm]
Line-of-Sight (LOS)	Static	Simulated RSS	663.8	154.1	645.7
		Simulated ToF	105.8	24.6	102.9
	Tracking with Motion	Simulated RSS	663.8	154.1	645.7
		Simulated ToF	210.1	171.6	121.2
Multipath	Static	Simulated RSS	750.3	206.5	721.3
		Simulated ToF	82.0	40.6	71.2
	Tracking with Motion	Measured RSS	719.9	349.9	629.2
		Simulated RSS	758.2	207.8	729.1
		Simulated ToF	520.6	152.8	497.7

It can be inferred from Table 8.1 and Table 8.2 that:

1. In LOS and multipath scenarios, ToF based position sensors are of higher accuracy for radial distances from four readers compared to RSS based position sensors.

2. In tracking with motion, ToF based position sensor is of higher RMS error, which decreases as sampling frequency increases.

3. Similar to measurement, radial distances from Reader #1 and Reader #3 has larger RMS errors compared to Reader #2 and Reader #4, which may because the motion is along direction of Reader #1 to Reader #3 and the motion introduces error compared to groundtruth trajectory.

4. Derived 3D coordinates based on triangulation in subsection 8.4.1 has larger RMS error along z- axis compared to xy- plane for both simulation and measurement.

5. As shown in Table 8.2, 3D RMS error follows the same trend as radial distances. RMS error along z- axis is larger than RMS error in xy- plane.

CHAPTER 9

INERTIAL SENSOR MODEL

In this chapter, due to the low-cost cellphone embedded IMU, sensor error is larger than high-precision embedded IMU. Hence, sensor model for inertial sensors is built and verified with measured IMU data. Calibration of inertial sensors in section E.3 is also based on the built sensor model.

9.1 Rotation Convention Definition

The definition of rotation angles, i.e., roll rotation along x-axis, pitch rotation along y-axis, yaw rotation along z-axis is as shown in Figure 9.1.

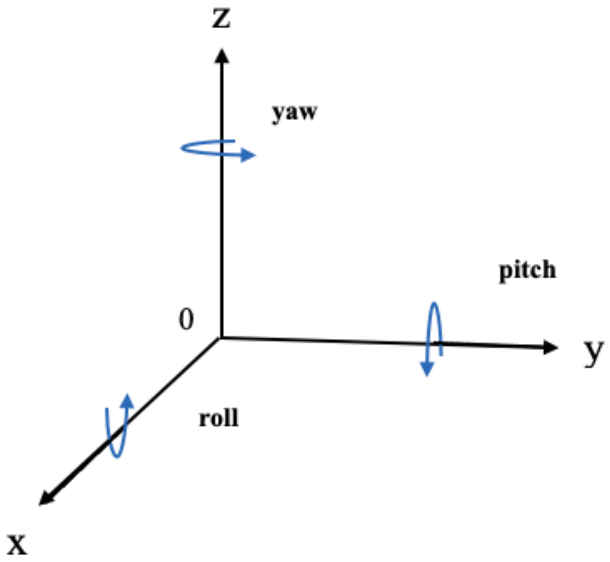


Figure 9.1: Scheme of Rotation Angles: Roll θ_x, Rotation Counterclockwise Around x-axis. Pitch θ_y, Rotation Counterclockwise Around y-axis and Yaw θ_z, Rotation Counterclockwise Around z-axis.

9.2 Sensor Error Model of Inertial Sensors

Sensor error models of inertial sensors: accelerometer, gyroscope, and magnetometer are built to calibrate measurement errors and thus enhancing state estimation accuracy of localization furthermore. Sensor error models provide more precise simulations for states that can be applied to algorithm design before taking experiment measurements. In this section, sensor error models of inertial sensors are built. In addition, related traid transformations among different sensors are also set up.

There are two kinds of errors for inertial sensors: deterministic error and stochastic error. Both types of errors contribute to output signal quality degradation but deterministic errors contribute more in practice [66, 67].

Deterministic error is attributed to manufacturing processes, component variation, and material differences. Deterministic error can be detected by monitoring outputs calibration which can be divided into three types: Bias, Scalar Factor, and Non-orthogonality Misalignment errors [68]:

(a) Bias Error: Bias is the deviation from the reference when no external stimuli or force is applied to the sensor.

(b) Scaling Factor Error: Scaling factor is the ratio of change in the output to the change of the input. The ratio between reference and actual scaling is termed as the scaling factor [69]. Manufacturer provides a default scalar factor but needs re-calibration in measurements.

(c) Misalignment Error: Each sensor has its own reference origin point and should be the same. In addition, reference axes of each sensor should be orthogonal to each other. Due to mounting point differences, manufacturing, and component variation, sensor axes are not orthogonal to each other and reference points are not aligned. Therefore, misalignment errors are introduced in measurements.

In contrast, stochastic error is electronic noise in the form of Additive White Gaussian Noise (AWGN) [70, 71, 72]. Stochastic error cannot be exactly measured based on output and can be approximated statistically.

Three axes of the accelerometers triad (AOF), three axes of the gyroscope triad (GOF), and three axes of the magnetometer triad (MOF) define a shared and orthogonal 3D body frame b. Nevertheless, due to assembly inaccuracy, misalignment and non-orthogonal, actual triads of accelerometer gyroscope and magnetometer are non-orthogonal and not overlapping, which are defined as AF, GF, and MF.

In addition, for each sensor, scaling factors are needed to convert sensor outputs to actual physical quantity. Moreover, sensor measurement outputs are always affected by non zero variable biases as well [73], corresponds to misalignment error, scaling factor error, and bias error that are introduced above.

We define the body frame b as following: the x- axis of frame b coincides with x- axis of AF; the y- axis of frame b lies in the plane spanned by the x- and y- axes of AF. In this way, AF, GF, and MF are related to b by a pure rotation. Additionally, AOF, GOF, and MOF overlap with frame b [73]. Due to this definition, transformation matrix of accelerometer from AF to body frame b, i.e., T^a is an upper triangle matrix as shown in Equation 9.3. In addition, the transformation matrix of gyroscope from GF to body frame b, i.e., T^g is a full square matrix as shown in Equation 9.6. Similarly, the transformation matrix of magnetometer from MF to body frame b, i.e., T^m is also a full square matrix in Equation 9.8.

9.2.1 Sensor Error Model of Accelerometer

There exists bias, scalar factor, and drift errors for accelerometer. Based on [73], the sensor error model of accelerometer is

$$a^b = T^a K^a (a^a + b^a + v^a) \tag{9.1}$$

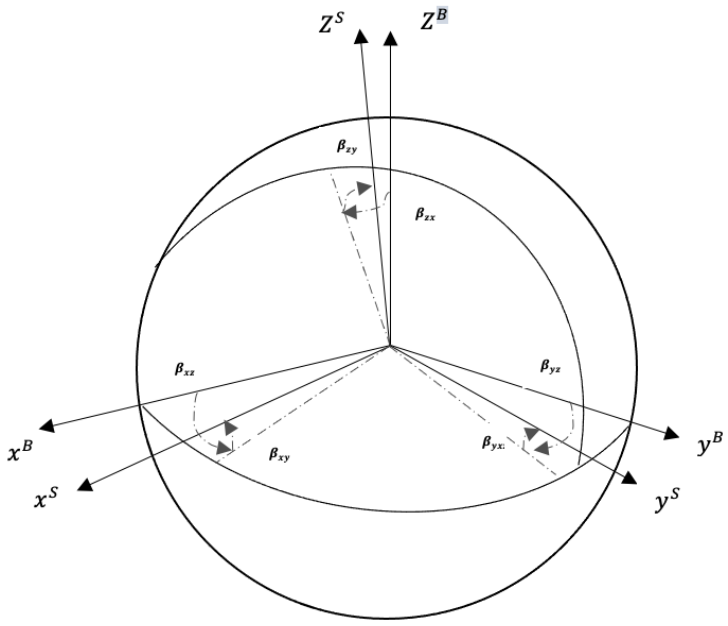


Figure 9.2: Scheme of Accelerometer Gyroscope and Magnetometer Manufacture Frames AF, GF, and MF (x^S, y^S, z^S) and Body Frame b (x^B, y^B, z^B), Which Overlaps with AOF, GOF, and MOF. Manufacture Frames AF, GF, and MF are Related to Body Frame b with Rotation Angles β_{ij}.

where $a^b \in R^3$ denotes the calibrated accelerations in body frame b of the tagged object. $a^a \in R^3$ is the raw measured accelerations in triad AF. Moreover, $T^a K^a$ is nonsingular for nearly orthogonal triad of similar sensors. $b^a \in R^3$ is the bias of accelerations [74]. v^a is stochastic noise, $v^a \sim N(0, \sigma)$ where σ is the standard deviation of the stochastic noise. The scaling matrix $K^a \in R^{3\times3}$ and bias vector $b^a \in R^3$ are composed of

$$K^a = \begin{bmatrix} s_x^a & 0 & 0 \\ 0 & s_y^a & 0 \\ 0 & 0 & s_z^a \end{bmatrix}, b^a = \begin{bmatrix} b_x^a \\ b_y^a \\ b_z^a \end{bmatrix} \tag{9.2}$$

The rotation or misalignment matrix $T^a \in R^{3\times3}$ is upper triangle as discussed

$$T^a = \begin{bmatrix} 1 & -\alpha_{yz} & \alpha_{zy} \\ 0 & 1 & -\alpha_{zx} \\ 0 & 0 & 1 \end{bmatrix} \tag{9.3}$$

Where α_{ij} is the rotation of the ith accelerometer around the jth b axis as shown in Figure 9.2 where β is replaced by α to avoid repeated parameters.

In the stationary period, accelerometer is expected to measure the gravity, the centrifugal acceleration and Coriolis acceleration, which are corrupted by noise. Measured accelerations are fitted to Gaussian distribution as shown in Figure 9.3 [75]. The non-zero mean denotes the bias b^a of accelerometer.

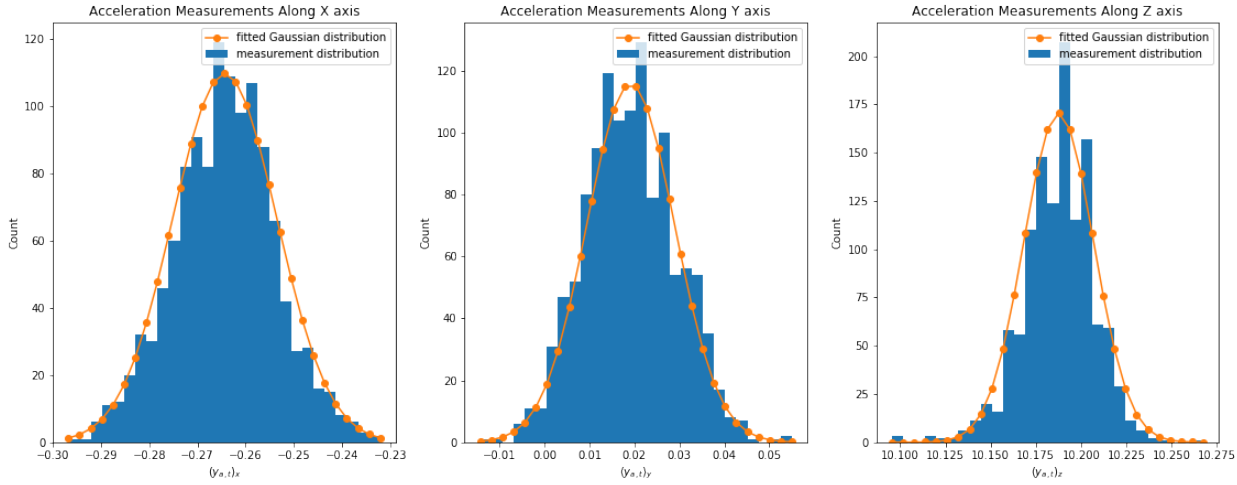


Figure 9.3: Blue: Histogram of Accelerometer Measurements. Orange: Fitted Gaussian Distribution of Measurements.

9.2.2 Sensor Error Model of Gyroscope

There exists bias, scalar factor, and drift errors in gyroscope measurements. Gyroscope measures the angular velocity with respect to the body frame GF in raw measurements.

Similarly, based on [73]

$$w^b = T^g K^g (w^g + b^g + v^g) \tag{9.4}$$

where $w^b \in R^3$ denotes the calibrated gyroscopes in body frame b of the tagged object and $w^g \in R^3$ is raw gyroscope measurements in triad GF. $b^g \in R^3$ is the bias of gyroscope [74]. v^g is gyroscope measurement noise, $v^g \sim N(0, \sigma)$ where σ is the standard deviation of the stochastic noise. The scaling matrix $K^g \in R^{3\times3}$ and bias vector $b^g \in R^3$ are

$$K^g = \begin{bmatrix} s_x^g & 0 & 0 \\ 0 & s_y^g & 0 \\ 0 & 0 & s_z^g \end{bmatrix}, b^g = \begin{bmatrix} b_x^g \\ b_y^g \\ b_z^g \end{bmatrix}, \tag{9.5}$$

The rotation matrix $T^g \in R^{3\times3}$ is as

$$T^g = \begin{bmatrix} 1 & -\gamma_{yz} & \gamma_{zy} \\ \gamma_{xz} & 1 & -\gamma_{zx} \\ -\gamma_{xy} & \gamma_{yx} & 1 \end{bmatrix}, \tag{9.6}$$

Similar to T^a, γ_{ij} is the rotation of the ith gyroscope around the jth b axis as shown in Figure 9.2 where β is replaced by γ.

In the stationary period, gyroscope is expected to measure the earth's angular velocity which is corrupted by noise. As is seen from Figure 9.4, the distribution of gyroscope measurements is close to Gaussian with a smaller standard deviation and there exists bias for gyroscopes along different axes as well.

9.2.3 Sensor Error Model of Magnetometer

The body frame of the tagged object rotates with respect to the earth frame. Therefore, rotating magnetic field vector back to completely aligned with gravity, i.e., the earth frame

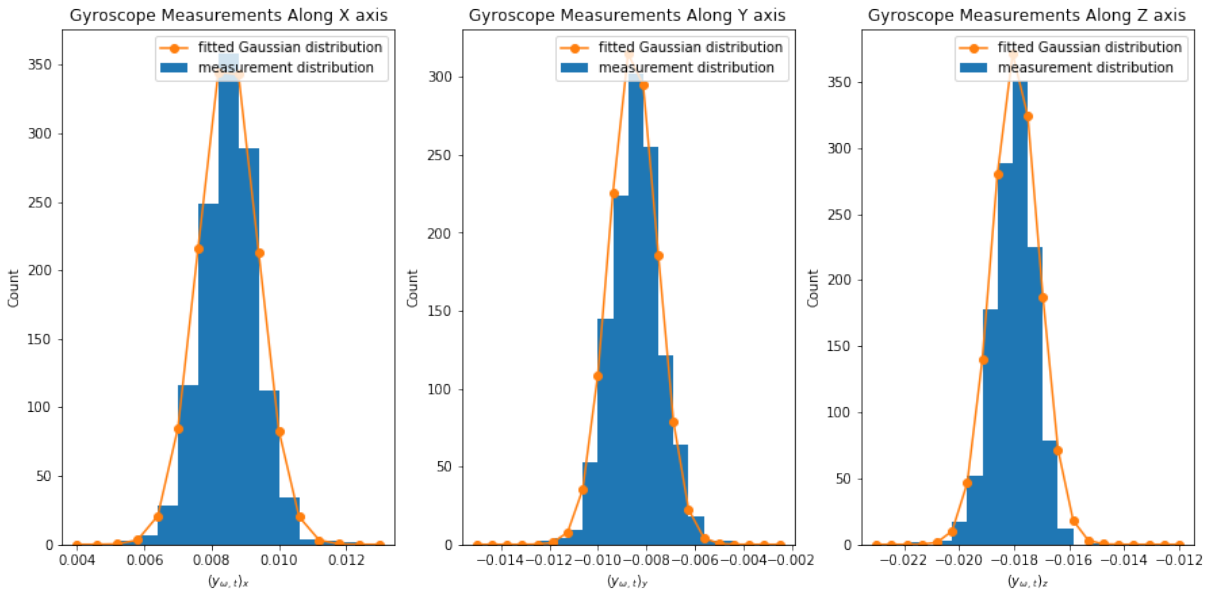


Figure 9.4: Blue: Histogram of Gyroscope Measurements. Orange: Fitted Gaussian Distribution of Measurements.

is needed, such that x and y components can be used to determine the orientation angle.

There exists soft iron and hard iron distortions for magnetometer as well as scalar factor and misalignment errors [76]. The error model of magnetometer is

$$m^b = T^m K^m S^m (m^m + b^m + v^m) \tag{9.7}$$

where $m^b \in R^3$ denotes the calibrated magnetometer in body frame b of the tagged object and $m^m \in R^3$ is raw magnetometer measurements in triad MF. v^m is magnetometer measurement noise, $v^m \sim N(0, \sigma)$ where σ is the standard deviation of the stochastic noise.

The rotation matrix or misalignment matrix $T^m \in R^{3\times3}$ and the scaling matrix $K^m \in R^{3\times3}$ are as

$$T^m = \begin{bmatrix} 1 & -\lambda_{yz} & \lambda_{zy} \\ \lambda_{xz} & 1 & -\lambda_{zx} \\ -\lambda_{xy} & \lambda_{yx} & 1 \end{bmatrix}, K^m = \begin{bmatrix} s_x^m & 0 & 0 \\ 0 & s_y^m & 0 \\ 0 & 0 & s_z^m \end{bmatrix}, \tag{9.8}$$

Similarly, λ_{ij} is the rotation of the ith gyroscope around the jth b axis as shown in Figure 9.2 where β is replaced by λ. S^m corresponds to soft iron distortion and $b^m \in R^3$ corresponds to hard iron distortion

$$S^m = \begin{bmatrix} C_1 & C_2 & C_3 \\ C_4 & C_5 & C_6 \\ C_7 & C_8 & C_9 \end{bmatrix}, b^m = \begin{bmatrix} C_{10} \\ C_{11} \\ C_{12} \end{bmatrix}, \tag{9.9}$$

Equation 9.9, C_1 - C_9 corresponds to soft iron distortion and C_{10} - C_{12} corresponds to hard iron distortion.

9.3 Simulation Verification

To verify the error model and the sensor data measured from the embedded IMU of MPU-9250 and of cellphone. Simulated and measurement IMU data from MPU-9250 and cellphone with the input trajectory are compared in this section.

9.3.1 Verification for IMU Sensor Data in 2D Motion

The IMU sensor data measured from the MPU-9250 and simulation data are presented in the subsection. For inertial measurements, random Gaussian noise is added to orientation ψ and angular velocity $\dot{\psi}$ as magnetometer and gyroscope noise. Simulated angular velocity is compared with measured angular velocity with an applied ratio of 0.85 (derived from scale between measurement and ground truth of angular velocity). Simulated orientation and angular velocity are of similar trend and shape to measured orientation and angular velocity, indicating that generated 2D orientation and angular velocity are close to the measurement.

Furthermore, accelerations along body axes of the tag x^B and y^B, in which the negative x^B-axis points towards the center of the circular motion trajectory and y^B-axis is tangent to its circular motion trajectory can be calculated from orientation ψ and accelerations $\ddot{x}$ and

$\ddot{y}$ along x and y axes [39] as

$$a_x = \ddot{x}\cos(\psi) + \ddot{y}\sin(\psi) \tag{9.10}$$

$$a_y = -\ddot{x}\sin(\psi) + \ddot{y}\cos(\psi) \tag{9.11}$$

where acceleration $\ddot{x}$ and $\ddot{y}$ along x and y axes, which can be calculated by taking on derivative of x and y with respect to ψ, where ψ is orientation. To verify simulated acceleration, simulated accelerations along tag body axes x^B and y^B based on Equation 9.10 and Equation 9.11 and corresponding measured acceleration with an applied ratio of 0.8 (derived from the scale between measurement and ground truth accelerations) are compared. Random Gaussian noise is added to acceleration data as inertial sensor noise. Simulated accelerations along axes x^B and y^B are of similar trend and shape to measured acceleration, indicating that described 2D angular acceleration with corresponding axes transformation are close to the measurement setup.

In summary, noise distribution of position states from the simulation model in Section section 2.3 is close to the measurement error in 1D and 2D motion. In addition, simulated states are of similar shape and noise level to measurement states. Motion descriptions in 1D and 2D motion fit measurement states.

9.3.2 Verification for IMU Sensor Data in 3D Motion

The IMU sensor data measured from the cellphone and simulated IMU sensor data based on the sensor models in section 9.2 are presented in the subsection with the motion described in chapter 6.

The cellphone embedded IMU sensor measurements and ground truth data based on the motion profile are show in Figure 7.1. It can be inferred from Figure 7.1 that accelerations in the motion fit the ground truth profiles. In addition, angular velocities from gyroscope and magnetic strength from magnetometer are close to constant zeros in the ratio of the

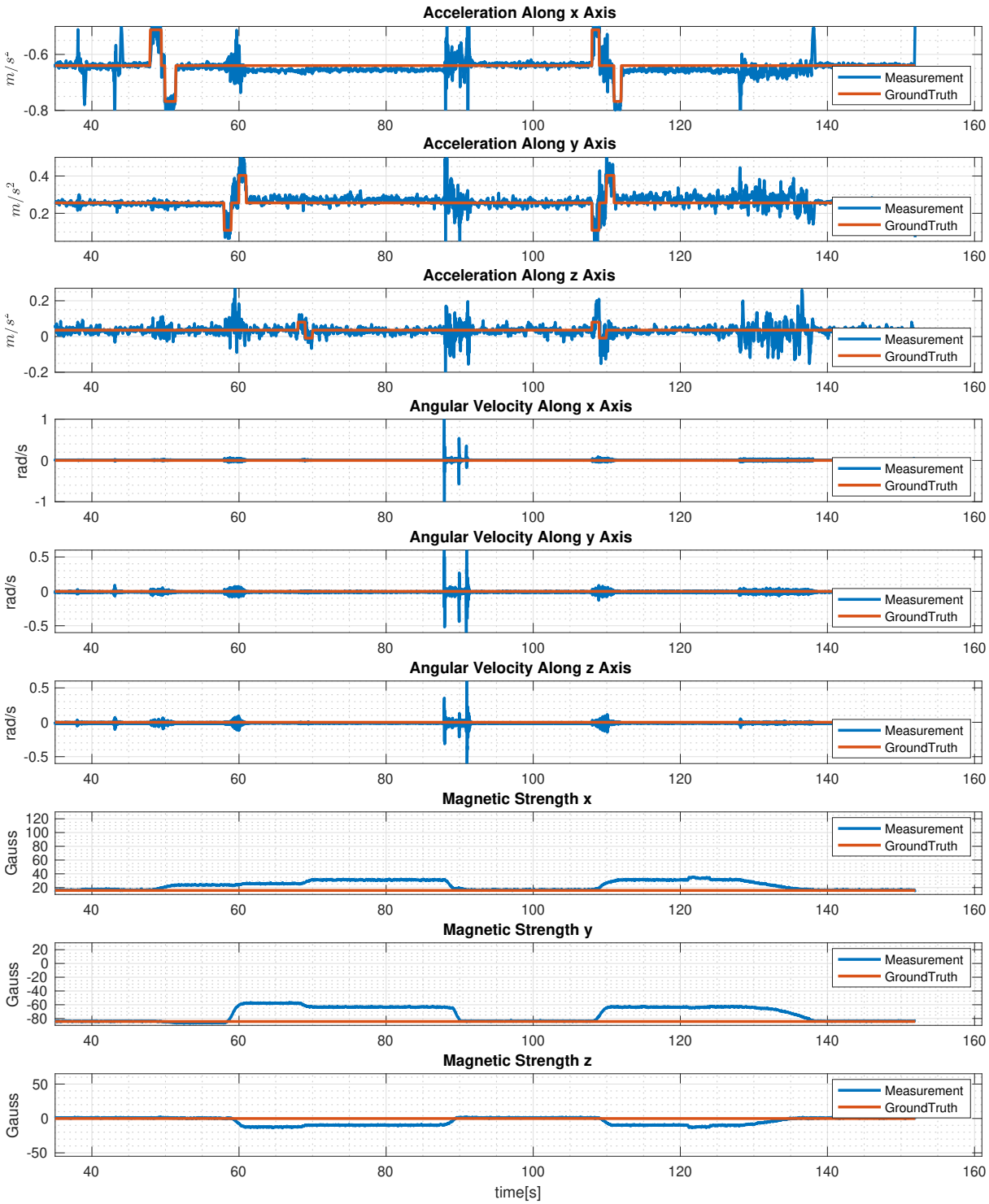


Figure 9.5: Measurement IMU Data Including Accelerations, Angular Velocities, and Magnetic Strength Along x- y- and z- Axes of the Cellphone Embedded IMU Compared to Simulated Ground Truth with the Input Trajectory Along x- Then y- and Finally z- Axis. After Moving Back, Moving Along x- y- z- Axes All Together in chapter 6.

measurement range.

In addition, the simulation of accelerations based on the sensor error model in section 9.2 is compared to measurement accelerations as show in Figure 9.6, Figure 9.7, Table 9.1, and Table 9.2 along the two motions described below. The simulated acceleration data are close to measurement data in the motion as shown in Table 9.1 and Table 9.2.

9.3.3 Verification for Accelerometer Model in 3D Motion

1. Moving Along x- Axis Then y- and Finally z- Axis

Accelerations with bias, skew factors, and scalar factors based on the error model in subsection 9.2.1 are simulated and compared with measurement data as shown in Figure 9.6.

As shown in Figure 9.6, blue lines are IMU measurement acceleration data. Red lines are simualted acceleration data. Yellow lines are ground truth acceleration profile with bias. It can be identified from Figure 9.6 that simulated IMU accelerations based on the sensor model, error model and noise distribution in subsection 9.2.1 are similar to the measurement accelerations. Therefore, there exists bias and drift in accelerations of the cellphone embedded IMU sensor.

The statistics of measurement and simulated accelerations comparison of the cellphone embedded IMU are shown in Table 9.1.

Table 9.1: Statistics of Measurement and Simulated Accelerations Compared to Ground Truth Profile When Moving Along x- Axis Then y- Axis and Finally Along z- Axis

Axis	Position Errors	Absolute Mean Error from Ground Truth [m]	Variance of Error [m^2]	Standard Deviation of Error [m]
x	Measurement	0.0081	0.0368	0.1919
	Simulation	0.0000	0.0129	0.1135
y	Measurement	0.0121	0.0364	0.1909
	Simulation	0.0380	0.0457	0.2137
z	Measurement	0.0190	0.0454	0.2130
	Simulation	0.0000	0.0128	0.1132

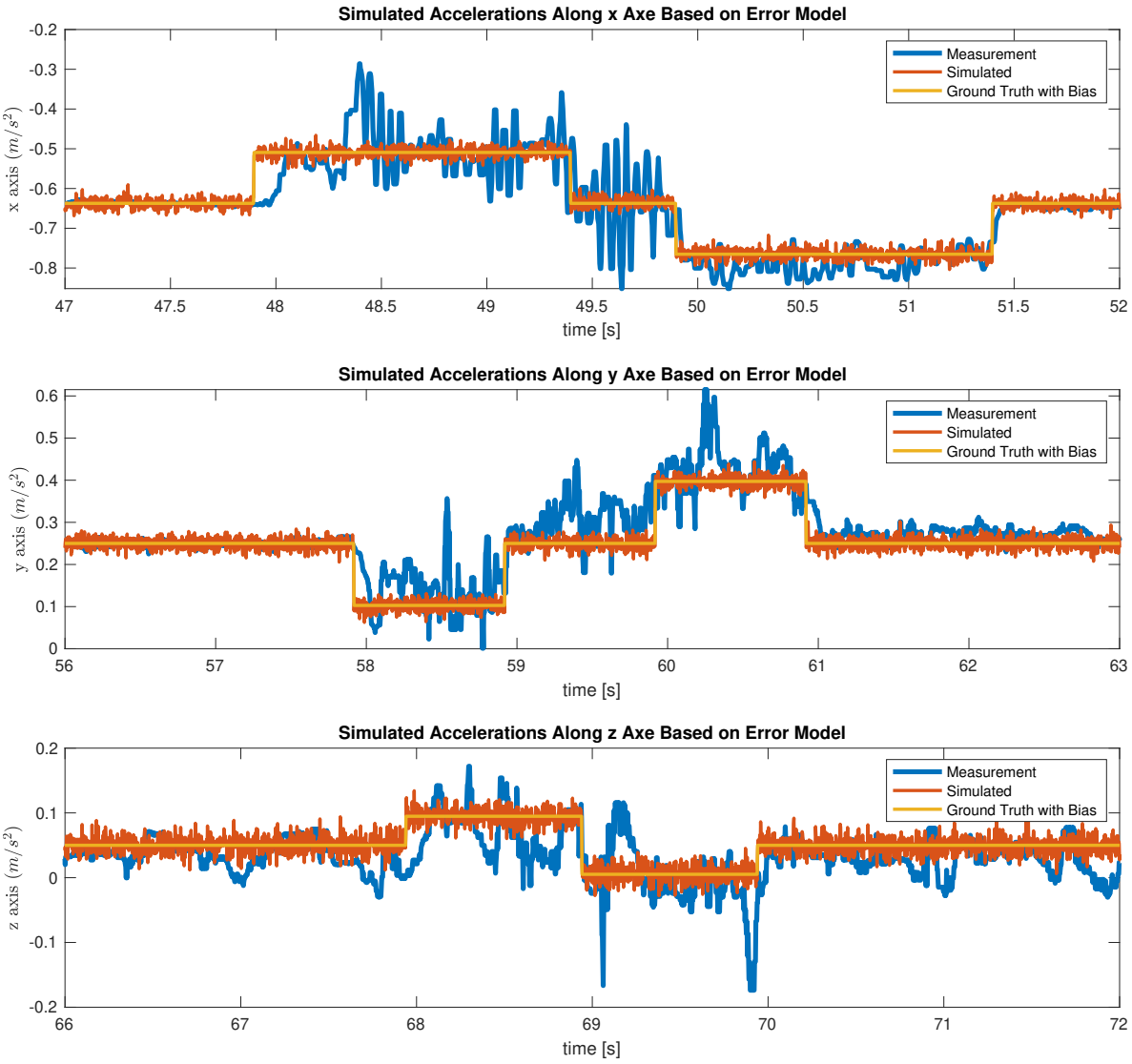


Figure 9.6: Simulated Accelerations of Cellphone Embedded IMU Sensor in Moving Along x- Then Along y- and Finally Along z- Axes With the Input Trajectory Described in chapter 6 Based on Sensor Models, Noise Distribution, As Well As Error Models in subsection 9.2.1.

It can be inferred from Table 9.1 that statistics of simulation and measurement are comparable to each other. Absolute value of position error $|e_t|$ in simulation and measurement at time t is plotted and compared in Figure 9.6, where $e_t = P_t^x - P_t^{gt}$. P_t^x in which $x \in \{\text{meas, sim}\}$ indicating measurement and simulation position states. P_t^{gt} is the ground truth

position state at time t.

Statistics of e, i.e. Mean= $E(e) = \bar{e}$, Variance= $\frac{\sum_{t=1}^{N}(e_t - E(e))^2}{N}$, and standard deviation, the root square of variance, are shown in Table 9.1. In addition, orientations and angular velocities in the motion are close to zero based on the error model in subsection 9.2.2 and subsection 9.2.3, similar to motion and measurement.

2. Moving Along x- y- z- Axes All Together Simultaneously

In addition, accelerations with bias, skew factors, and scalar factors based on the error model in subsection 9.2.1 when tag is moving along x-, y-, and z- axes together are simulated and compared with measurement data as shown in Figure 9.7.

The statistics of measurement and simulated accelerations comparison of the cellphone embedded IMU are shown in Table 9.2.

Table 9.2: Statistics of Measurement and Simulated Accelerations Compared to Ground Truth Profile When x- y- z- Axes Moving All Together Simultaneously

Axis	Position Error	Absolute Mean Error from Ground Truth [m]	Variance of Error [m^2]	Standard Deviation of Error [m]
x	Measurement	0.0081	0.0384	0.1959
	Simulation	0.0000	0.0125	0.1119
y	Measurement	0.0121	0.0372	0.1929
	Simulation	0.0000	0.0125	0.1120
z	Measurement	0.0190	0.0449	0.2119
	Simulation	0.0000	0.0125	0.1117

It can be inferred from Table 9.2 that statistics of simulation and measurement are comparable to each other. Absolute value of position error $|e_t|$ in simulation and measurement at time t is plotted and compared in Figure 9.7, where $e_t = P_t^x - P_t^{gt}$. P_t^x in which $x \in \{\text{meas, sim}\}$ indicating measurement and simulation position states. P_t^{gt} is the ground truth position state at time t.

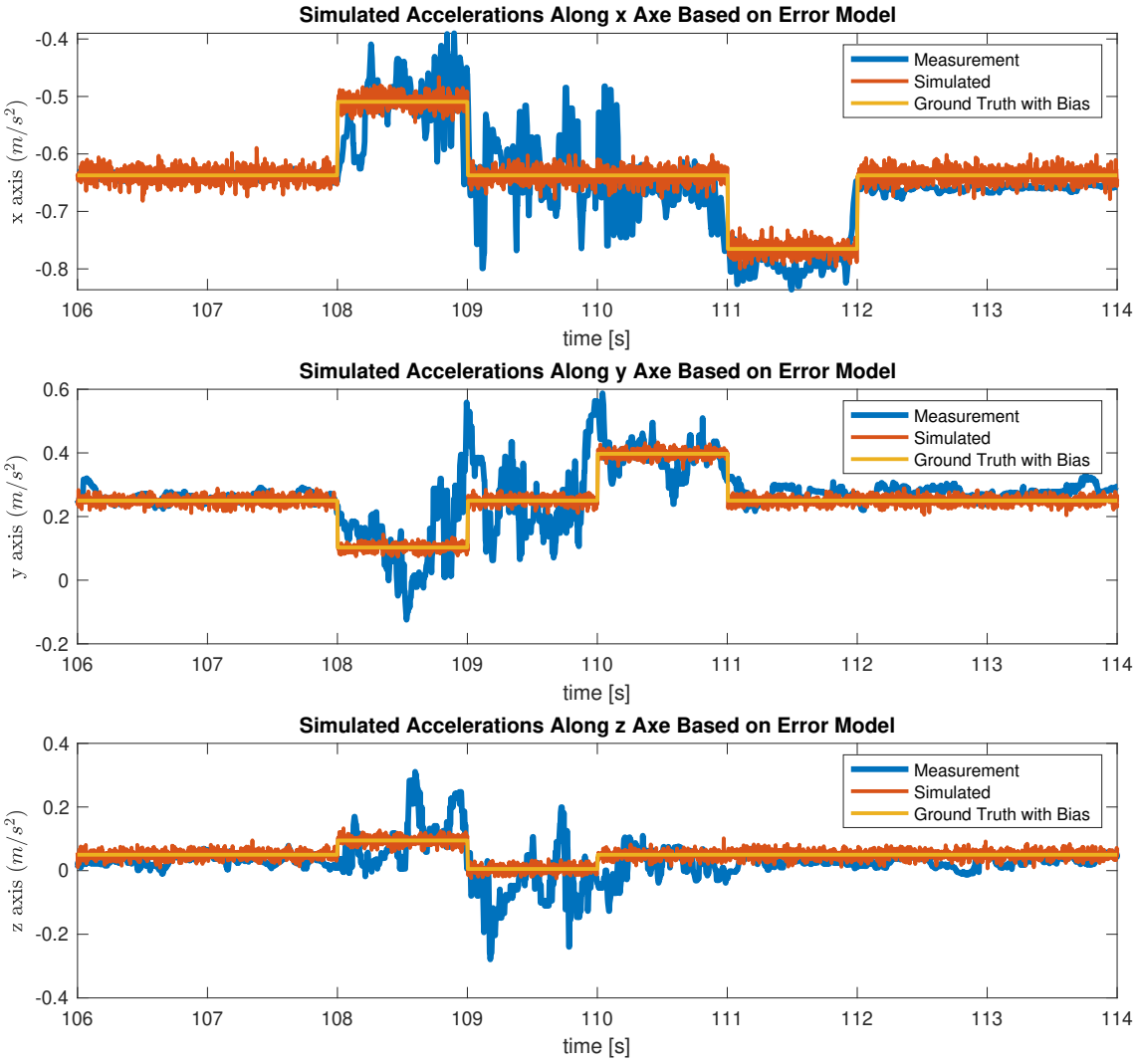


Figure 9.7: Simulated Accelerations Along x- y- and z- Axes of Cellphone embedded IMU Sensor With x- y- and z- Moving All Together With the 3D Input Trajectory Described in chapter 6 Based on Sensor Models, Noise Distribution, As Well As Error Models in subsection 9.2.1.

Statistics of e, i.e. Mean$= E(e) = \bar{e}$, Variance$= \frac{\sum_{t=1}^{N}(e_t - E(e))^2}{N}$, and standard deviation, the root square of variance, are shown in Table 9.2.

3. Results of Verification for IMU Sensor Data

The maximal sampling frequency of cellphone IMU data is 400 Hz. In addition, the sampling frequency and timestamp of accelerometer, gyroscope, and magnetometer are dif-

ferent. More specifically, the sampling frequencies of accelerometer and gyroscope are 4 times compared to magnetometer.

In all simulations, the input timestamp of the simulation model is the same as the timestamp of measurement from cellphone embedded IMU. It can be inferred from Figure 9.6, Figure 9.7, Table 9.1, and Table 9.2 that:

1. Simulation model of IMU sensor fits the measurement data in both separate 3D motion and moving along all axes all together simultaneously in 3D motion.

2. Accelerations of moving all together along x- y- and z- axes are separable and similar to moving along each axis as along x- axis first, then along y- axis and finally along z- axis. Therefore, simulated IMU sensor data with sensor model can be used in localization system design and implementation before taking measurements.

CHAPTER 10

DESIGN OF 3D NONLINEAR LEAST-SQUARES ESTIMATION

In this chapter, advancing from 2D motion, nonlinear state-space model and nonlinear state estimation for real-time RFID-based 3D localization are built. Nonlinear state-space model is designed with different state vectors and sensor outputs for both sensor fusion and RF estimation. Nonlinear Least-Squares Estimation with stacked measurement vector is applied to nonlinear state estimation in 3D motion with rotation matrix multiplication along x- y- and z- axes. For comparison, EKF is built for nonlinear state estimation in 3D motion as well. In order to handle different sampling rates of RF signal and IMU data as well as different sensors within IMU data, asynchronous and synchronous models are built for both NLE and EKF. Further, Quaternion is applied to orientation estimation.

10.1 Frame Systems of 3D Motion

In the 3D localization system, two frame systems in measurement and estimation are involved for inertial sensors [75]:

1. The body frame b: is the coordinate frame of the moving tagged object. Its origin is located at the center of the tagged object and is aligned to the triad of the inertial sensor.

2. The navigation frame n: is the geographic frame to navigate. The estimated position, velocity, acceleration, orientation, and angular velocity of the tagged object is with respect to this frame. We define this frame as stationary to the earth.

More specifically, x-, y-, and z- axes of the navigation frame n are along x-, y-, and z- axes of the 3D positioner. For the body frame b, x- axis overlaps with x- axis of the navi-

gation frame. y- axis overlaps with z- axis of the navigation frame. z- axis overlaps with y- axis of the navigation frame and points toward the negative direction. In the following 3D state estimation, the measurements are taken in the body frame b and position orientation states are estimated in the navigation frame n.

10.2 3D State-Space Model

The motion equation of the tag moving in 3D planar can be represented as [39]

$$m\ddot{x} = F_x \tag{10.1}$$

$$m\ddot{y} = F_y \tag{10.2}$$

$$m\ddot{z} = F_z \tag{10.3}$$

$$I_x\ddot{\alpha} = T_x \tag{10.4}$$

$$I_y\ddot{\beta} = T_y \tag{10.5}$$

$$I_z\ddot{\psi} = T_z \tag{10.6}$$

where m is the mass of the moving object, $\ddot{x}$ $\ddot{y}$ and $\ddot{z}$ are accelerations along x-, y-, and z-axes. $\ddot{\alpha}$ $\ddot{\beta}$ and $\ddot{\psi}$ are the angular accelerations around the x-axis y-axis and z-axis. F_x F_y and F_z are the resultant forces applied to the mass along x-, y-, and z- axes. I_x I_y I_z and T_x T_y T_z are the inertial moment and resultant torque around the x-axis y-axis and z-axis. The force/mass and torque/moment determines the three-dimensional motion of the tag. The motion profiles explained above are unknown in the experiment. In addition, unlike [39] initial conditions of the estimation are unknown in estimation.

The nonlinear state-space models of 3D motion with different sensor outputs are shown

in Equation 10.7 and Equation 10.8

$$\dot{X} = AX + Bw \tag{10.7}$$

$$Y = h(X) + v \tag{10.8}$$

where X shown in Equation 10.13 is the fixed-length state vector composed of position, velocity, and acceleration states along x-, y-, z- axes, and orientation, angular velocity, angular acceleration around the x-axis y-axis and z-axis. Y denotes the vector of noisy sensor outputs to estimate the state vector. Three sets of sensor outputs Y: $\{r\}$, $\{r, \dot{r}\}$; $\{r, \dot{r}, \theta, \omega, a_x, a_y, a_z\}$ are applied. Vector $h(X)$ describes the nonlinear transformation between state vector and the noise-free sensor outputs vector. w is the disturbance vector which accounts for the forces $(\frac{\dot{F}_x}{m}, \frac{\dot{F}_y}{m}, \frac{\dot{F}_z}{m}, \frac{\dot{T}_x}{I_x}, \frac{\dot{T}_y}{I_y}, \frac{\dot{T}_z}{I_z})$, and v is the vector of sensor noises.

$$w = \begin{bmatrix} \frac{\dot{F}_x}{m} & \frac{\dot{F}_y}{m} & \frac{\dot{F}_z}{m} & \frac{\dot{T}_x}{I_x} & \frac{\dot{T}_y}{I_y} & \frac{\dot{T}_z}{I_z} \end{bmatrix}' \tag{10.9}$$

For state vector without orientation or angular velocity, the state vector X and output vector Y are composed of

$$X = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} & z & \dot{z} & \ddot{z} \end{bmatrix}' \tag{10.10}$$

$$Y = \begin{bmatrix} r_1 & \dot{r}_1 & r_2 & \dot{r}_2 & r_3 & \dot{r}_3 & r_4 & \dot{r}_4 \end{bmatrix}' \tag{10.11}$$

Output vector Y in Equation 10.11 corresponds to sensor outputs of $\{r\}$, $\{r, \dot{r}\}$. In Equation 10.11, r_i, $\dot{r}_i$ are radial distances and radial velocities measured from reader i. If only radial distances are used, $\dot{r}_i, \forall i = 1, 2, 3, 4$ are removed from Equation 10.11. Correspond-

ing coefficient matrices A, B in the state-space model are

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{10.12}$$

For state vector with orientation and angular velocity, the state and output vectors X, Y are composed of

$$X = \begin{bmatrix} x & \dot{x} & \ddot{x} & y & \dot{y} & \ddot{y} & z & \dot{z} & \ddot{z} & \alpha & \dot{\alpha} & \beta & \dot{\beta} & \psi & \dot{\psi} \end{bmatrix}' \tag{10.13}$$

$$Y = \begin{bmatrix} r_1 & \dot{r}_1 & r_2 & \dot{r}_2 & r_3 & \dot{r}_3 & r_4 & \dot{r}_4 & \theta_x & \omega_x & \theta_y & \omega_y & \theta_z & \omega_z & a_x & a_y & a_z \end{bmatrix}' \tag{10.14}$$

r_i, $\dot{r}_i$ are radial distances and radial velocities measured from reference reader $i = 1, 2, 3, 4$. θ_x, θ_y, and θ_z are the orientation angles and ω_x, ω_y, ω_z are the angular velocities around the x-axis y-axis and z-axis. a_x, a_y, and a_z are acceleration along the x^B, y^B, and z^B axes.

Corresponding coefficient matrices A, B in the state-space models are

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{10.15}$$

Suppose that the values of the continuous signals are sampled at the sampling instants k with sampling period T. Discrete-time state-space models of the tag in 3D motion are [62]

$$X_{k+1} = FX_k + Gw_k \tag{10.16}$$

$$Y_k = h(X_k) + v_k \tag{10.17}$$

where X_k is discrete-time state vector of X composed of position, velocity, and acceleration states along x-, y-, and z- axes, as well as orientation, angular velocity, and angular acceleration around the x-axis y-axis and z-axis at time step k. Y_k denotes the vector of noisy sensor outputs at time step k. w_k and v_k are the discrete-time disturbance vector and the vector of sensor noises.

For state vector without orientation or angular velocity, discrete-time state vector and output vector corresponding to vectors in continuous time Equation 10.10 - Equation 10.11

are

$$X_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k & y_k & \dot{y}_k & \ddot{y}_k & z_k & \dot{z}_k & \ddot{z}_k \end{bmatrix}' \tag{10.18}$$

$$Y_k = \begin{bmatrix} r_{1k} & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} & r_{4k} & \dot{r}_{4k} \end{bmatrix}' \tag{10.19}$$

If only radial distances are used, $\dot{r}_{ik}$ where $i = 1, 2, 3, 4$ are removed from Equation 10.19. Coefficient matrices F and G are shown in Equation 10.20, where T is the sampling period.

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, G = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ T & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{10.20}$$

For state vector with orientation and angular velocity, discrete-time state and output vectors corresponding to vectors in Equation 10.13 - Equation 10.14 become

$$X_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k & y_k & \dot{y}_k & \ddot{y}_k & z_k & \dot{z}_k & \ddot{z}_k & \alpha_k & \dot{\alpha}_k & \beta_k & \dot{\beta}_k & \psi_k & \dot{\psi}_k \end{bmatrix}' \tag{10.21}$$

$$Y_k = \begin{bmatrix} r_{1k} & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} & r_{4k} & \dot{r}_{4k} & \theta_{xk} & \omega_{xk} & \theta_{yk} & \omega_{yk} & \theta_{zk} & \omega_{zk} & a_{xk} & a_{yk} & a_{zk} \end{bmatrix}' \tag{10.22}$$

Corresponding coefficient matrices F and G are

$$F = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, G = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ T & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & T & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{10.23}$$

where T is the sampling period. Nonlinear transformation between state vector and noise-free sensor outputs vector can be represented as following. For state vector without orientation or angular velocity,

$$h(X_k) = \begin{bmatrix} h_1 & \dot{h}_1 & h_2 & \dot{h}_2 & h_3 & \dot{h}_3 & h_4 & \dot{h}_4 \end{bmatrix}' \tag{10.24}$$

$\dot{h}_i$ where $i = 1, 2, 3, 4$ are removed from Equation 10.24 if only radial distances are used. For state vector with orientation and angular velocity,

$$h(X_k) = \begin{bmatrix} h_1 & \dot{h}_1 & h_2 & \dot{h}_2 & h_3 & \dot{h}_3 & h_4 & \dot{h}_4 & h_5 & h_6 & h_7 & h_8 & h_9 & h_{10} & h_{11} & h_{12} & h_{13} \end{bmatrix}' \tag{10.25}$$

In both situations,

$$h_u = \sqrt{(x_k - x_u)^2 + (y_k - y_u)^2 + (z_k - z_u)^2}, \quad u = 1, 2, 3, 4$$

$$\dot{h}_u = \frac{(x_k - x_u)\dot{x}_k + (y_k - y_u)\dot{y}_k + (z_k - z_u)\dot{z}_k}{\sqrt{(x_k - x_u)^2 + (y_k - y_u)^2 + (z_k - z_u)^2}}, \quad u = 1, 2, 3, 4$$

$$h_5 = \alpha_k$$

$$h_6 = \dot{\alpha}_k$$

$$h_7 = \beta_k$$

$$h_8 = \dot{\beta}_k$$

$$h_9 = \psi_k$$

$$h_{10} = \dot{\psi}_k$$

$$\begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \end{bmatrix} = \begin{bmatrix} \cos\theta_{zk} & -\sin\theta_{zk} & 0 \\ \sin\theta_{zk} & \cos\theta_{zk} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_{yk} & 0 & \sin\theta_{yk} \\ 0 & 1 & 0 \\ -\sin\theta_{yk} & 0 & \cos\theta_{yk} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{xk} & -\sin\theta_{xk} \\ 0 & \sin\theta_{xk} & \cos\theta_{xk} \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}$$

in which x_u, y_u, $u = 1, 2, 3, 4$ are the locations of each reader that can be inferred in the system setup in Table 6.1.

10.3 3D Nonlinear State Estimation Based on NLE

Similarly, Nonlinear Least-Squares Estimation (NLE) proposed in [49] is applied for nonlinear state estimation in 3D motion. NLE is developed from the perspective of nonlinear least-squares optimization and the map-inversion philosophy based on discrete-time dynamics of the system [49].

More specifically, NLE determines an estimate $\hat{x}_k$ of x_k, using the l-length measurement set $\{y_k, y_{k-1}, ..., y_{k-l+1}\}$ to reduce the impact of sensor noise. Therefore, NLE includes a two-step procedure, wherein first $\hat{x}_{k-l+1}$ is determined from the measurement sets and second $\hat{x}_k$ is determined through forward propagation [49].

The stacked measurement vector of the tag in 3D motion based on section 10.2 is

$$Z_k = \begin{bmatrix} Y_{k-l+1}, & Y_{k-l+2}, & \dots & , Y_k \end{bmatrix}^T \tag{10.26}$$

where Y_k is the noisy measurement output vector from Equation 10.22 that is composed of the whole or part of

$$Y_k = \begin{bmatrix} r_{1k} & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} & r_{4k} & \dot{r}_{4k} & \theta_{xk} & \omega_{xk} & \theta_{yk} & \omega_{yk} & \theta_{zk} & \omega_{zk} & a_{xk} & a_{yk} & a_{zk} \end{bmatrix} \tag{10.27}$$

where r_{1k}, r_{2k}, r_{3k}, r_{4k}, $\dot{r}_{1k}$, $\dot{r}_{2k}$, $\dot{r}_{3k}$, $\dot{r}_{4k}$ are simulated/measurement radial distances and radial velocities with added noise at time stamp k of sampling period T. θ_{xk}, θ_{yk}, θ_{zk} and ω_{xk}, ω_{yk}, ω_{zk} are simulated/measurement orientations and angular velocities around the x-axis y-axis and z-axis.

A mapping from state and input values to output values is denoted as H

$$H_k(X_{k-l+1}) = \begin{bmatrix} h(X_{k-l+1}) \\ h(FX_{k-l+1}) \\ .. \\ h(F^{l-1}X_{k-l+1}) \end{bmatrix} \tag{10.28}$$

where F is shown in Equation 10.23. In the absence of modeling error and sensor noise,

$$Z_k = H_k(X_{k-l+1}) \tag{10.29}$$

holds true [49]. In case of noise, NLE minimizes the sum of the squares of the residual errors

$$\hat{X}_{k-l+1} = \text{argmin}||Z_k - H_k(X_{k-l+1})||^2 \tag{10.30}$$

This least-squares state estimate is l samples delayed in time, so it is propagated forward in time using the nominal system dynamics to get [49]

$$\hat{X}_k = F^{l-1} X_{k-l+1} \tag{10.31}$$

Suppose that the current iterate in the search for the minimizer of $||E_k(X_{k-l+1})||^2$ is denoted by $X^{(j)}_{k-l+1}$. Using Taylor series approximation, a corresponding local model for the iteration process would be [49]

$$\hat{E}_k(X_{k-l+1}) = E_k(X^{(j)}_{k-l+1}) + J_k(X^{(j)}_{k-l+1})(X_{k-l+1} - X^{(j)}_{k-l+1}) \tag{10.32}$$

where $J_k(X_{k-l+1})$ denotes the Jacobian matrix of $E_k(X_{k-l+1})$ with respect to X_{k-l+1}.

NLE estimation can be implemented in both causal and uncausal way. In the estimation for RFID states, we implement the estimation causally, using states before and at the time stamp.

In order to minimize $||E_k(X_{k-l+1})||^2 = ||Z_k - H_k(X_{k-l+1})||^2$, an iterative search of damped Gauss-Newton method is conducted to minimize the error of nonlinear least-squares problem [49]. In parallel, Levenberg-Marquardt method is also applied to minimize the error of nonlinear least-squares problem.

Implementation: Built-in function `lsqnonlin` in Matlab have been applied in the estimation. The two implementations give close enough results. Built-in function in Matlab `lsqnonlin`$(@(X)E_k(Y, X, l), X_k)$ is implemented in the estimation. Input $E_k(X)$ is the residual error in Equation 10.30, state vector X_k, $k = 1, 2, ...,$ *len* is initialized the same.

Minimal Stack Length: With orientation and angular velocity, the length of state vector is 15 and the length of measurement vector is 17. Hence, the minimal stack length needed is *len* = 1 in the 3D estimation.

10.4 3D Nonlinear State Estimation Based on EKF

EKF is also applied to nonlinear state estimation in 3D motion. There are two steps in EKF: prediction step in which state vector and prediction error matrix is propagating to next time stamp and update step in which gain matrix, state vector and prediction matrix are updated based on measurements.

In this section, we consider with orientation and angular velocity. State-space model without inertial sensors can be referred to section 10.2. State vector X_k of dimension 1×15 as shown in Equation 10.21, measurement vector Y_k of dimension of 1×17 as shown in Equation 10.22, and F_k of dimension 15×15 as shown in Equation 10.23 in the following steps. Moving from static, state vector X_0 is initialized as a zero vector.

1. Predict Step

$$\hat{X}_k = F_{k-1}\hat{X}_{k-1} \tag{10.33}$$

$$P_k = F_{k-1}P_{k-1}F_{k-1}^T + Q \tag{10.34}$$

2. Update Step

$$G_k = P_k L_k^T (H_k P_k L_k^T + R)^{-1} \tag{10.35}$$

$$\hat{X}_k = \hat{X}_k + G_k(Y_k - h(\hat{X}_k)) \tag{10.36}$$

$$P_k = (I - G_k L_k)P_k \tag{10.37}$$

where $L_k(X_k)$ is the Jacobian matrix of error $E_k(X_k) = Y_k - h(\hat{X}_k)$ with respect to X_k. P_k is the prediction error matrix, Q is the covariance matrix of disturbance vector ω_k, and R is the covariance matrix of v_k.

Prediction error matrix P_0 is initialized as

$$P_0 = \begin{bmatrix} 0.8 & & \\ & \ddots & \\ & & 0.8 \end{bmatrix}_{15\times 15} \tag{10.38}$$

Disturbance vector accounts for the forces $(\frac{\dot{F}_x}{m}, \frac{\dot{F}_y}{m}, \frac{\dot{F}_z}{m}, \frac{\dot{T}_x}{I_x}, \frac{\dot{T}_y}{I_y}, \frac{\dot{T}_z}{I_z})$, and v is the vector of sensor noises. Hence, Q is set as

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\dot{F}_x}{m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\dot{F}_y}{m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\dot{F}_z}{m} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\dot{T}_x}{I_x} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\dot{T}_y}{I_y} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\dot{T}_z}{I_z} \end{bmatrix}_{15\times 15} \tag{10.39}$$

Covariance matrix R of sensor noise vector is set with diagonal elements that is the variance of difference between measurement and ground truth states.

$$R = \begin{bmatrix} v_1 & & & \\ & v_2 & & \\ & & \ddots & \\ & & & v_{17} \end{bmatrix}_{17\times 17} \tag{10.40}$$

where:

$v_{2i-1} = ||r_i - r_{igt}||^2, i = 1, 2, 3, 4$

$v_{2i} = ||\dot{r}_i - \dot{r}_{igt}||^2, i = 1, 2, 3, 4$

$v_j = ||\theta_k - \theta_{kgt}||^2, v_{j+1} = ||\omega_k - \omega_{kgt}||^2, v_{j+6} = ||a_k - a_{kgt}||^2, j = 9, 11, 13\ k = x, y, z$

where r_{igt}, $\dot{r}_{igt}$, $\forall i = 1, 2, 3, 4$ are the ground truth radial distances and radial velocities from reader #1 #2 #3 #4 derived in the motion in chapter 6. θ_{kgt}, ω_{kgt}, a_{kgt}, $\forall k = x, y, z$ are orientations, angular velocities, and accelerations along x- y- z- axes.

Based on the fact that measurements from different readers are of different frequencies and thus are mutually independent. In addition, accelerometer gyroscope magnetometer measured from the cellphone embedded IMU are also independent.

10.5 Asynchronous Estimation

States in nonlinear transformation are dependent on each other, e.g. accelerations are dependent on orientations. Therefore, synchronous estimation are of reduced accuracy out of the time stamp difference among states, which is verified by the estimation results.

To avoid the time difference, asynchronous estimation is introduced in this section, in which the state vector is estimated whenever a measurement sensor output arrives. Hence, state vector gets updated at every measurement arrival. Since it only relies on single sensor output, time stamp is the same in the estimation.

10.5.1 3D State-Space Model in Asynchronous Estimation

We only consider with orientation and angular velocity. State-space model without inertial sensors can be referred to section 10.2. To match number of sensor inputs, coefficient matrix C is applied to both Y_{ki} and $h(X_k)$ to truncate corresponding arrived measurement sensor outputs from the whole measurement vector. State vector X_k is of the fixed length 15 as shown in Equation 10.41. Measurement vector Y_{ki} is sensor output of different length, i.e., 4, 8, 17 depends on number of arrival sensor measurements at the same time stamp k_i.

$$X_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k & y_k & \dot{y}_k & \ddot{y}_k & z_k & \dot{z}_k & \ddot{z}_k & \alpha_k & \dot{\alpha}_k & \beta_k & \dot{\beta}_k & \psi_k & \dot{\psi}_k \end{bmatrix}' \tag{10.41}$$

$$Y_k = \begin{bmatrix} r_{1k} & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} & r_{4k} & \dot{r}_{4k} & \theta_{xk} & \omega_{xk} & \theta_{yk} & \omega_{yk} & \theta_{zk} & \omega_{zk} & a_{xk} & a_{yk} & a_{zk} \end{bmatrix} \tag{10.42}$$

Coefficient matrix F of propagating state vector X_k at time stamp k to measurements Y_{k_i} at time stamp k_i becomes

$$F = \begin{bmatrix} 1 & T_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & T_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & T_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_i & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_i & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_i & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_i \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{10.43}$$

where $T_i = T_{k_i} - T_k$ is the time difference to propagate state vector X_k at time k to received measurement time stamp k_i. Since there is only one set of measurement, all T_i in F are the same.

Then stacked measurement vector Y_{k_i} is composed of sensor measurements of different lengths, e.g., 4, 8, or 17, from different sensors in 3D motion is able to provide sensor measurements from multiple sources.

For example, if three accelerations arrive, Y_{k_i} is of length 3. Jacobian matrix J_{ki} in NLE and Jacobian matrix L_k in Kalman filter are of size 3×15. C is of size 3×17, which is composed of an identity matrix of 3 with a 3 by 14 zero matrix. In addition, R matrix is also of size 3×3.

10.5.2 Estimation Using NLE and EKF in Asynchronous Estimation

1. Asynchronous Estimation Using NLE

$$Z_k = \begin{bmatrix} Y_{k-l+1}, & Y_{k-l+2}, & \dots & , Y_k \end{bmatrix}^T \tag{10.44}$$

where Y_{k-l+1}, Y_{k-l+2}, ... , Y_k is corresponding sensor measurements arrives at time stamp $k-l+1$, $k-l+2$, ... , k, which may be outputs from RF of length 8, accelerometer of length 3, gyroscope of length 3, or magnetometer of length 3.

Similarly, NLE estimates the state vector from the stacked measurement vector as in section 10.3 afterwards.

2. Asynchronous Estimation Using EKF

In addition, F_k is of fixed size 15×15, G_k is of size $17 \times len$ where len is the length of input measurement vector. R is of size $len \times len$. $h(X_k)$ is of length $len \times 1$.

If only RF signal comes, $h(X_k) = \begin{bmatrix} h_1 & \dot{h}_1 & h_2 & \dot{h}_2 & h_3 & \dot{h}_3 & h_4 & \dot{h}_4 \end{bmatrix}$; if only magnetometer measurement comes, $h(X_k) = \begin{bmatrix} h_5 & h_7 & h_9 \end{bmatrix}$; if only gyroscope measurement comes, $h(X_k) = \begin{bmatrix} h_6 & h_8 & h_{10} \end{bmatrix}$; if only accelerations comes, $h(X_k) = \begin{bmatrix} h_{11} & h_{12} & h_{13} \end{bmatrix}$. The rest is the union of different combinations of $h(X_k)$.

Correspondingly, Jacobian matrix L_k^T is of size $len \times 15$, where len is the number of arrival sensor measurements. Afterwards, EKF estimates the state vector from the varied length measurement vector following steps in section 10.4.

10.6 Synchronous Estimation

Synchronous estimation is applied to the fixed-length measurement vector Y_k. It waits until every sensor output arrives to build one set of the measurement vector. Therefore, dimensions of X_k, Y_k, F, G, Z_k, H_k in NLE and F_{k-1}, P_k, Q, R, L_k, G_k stay the same. Since all measurements may not come at the same time, measurement state vector as shown in Equation 10.46 may be of different arrival time stamps.

10.6.1 3D State-Space Model in Synchronous Estimation

Similarly, we only consider with orientation and angular velocity. State-space model without inertial sensors can be referred to section 10.2. Specifically, let k_r denote the arrival time stamp for RF signals where $r = 1, 2, 3, 4$. k_{ax}, k_{ay}, k_{az} denote time stamp for accelerations along x-, y-, and z- axes, $k_{\alpha'}$, $k_{\beta'}$, $k_{\psi'}$ denote time stamp for orientation around x-, y-, and z- axes, and $k_{\alpha x}$, $k_{\beta y}$, $k_{\psi z}$ denote time stamp for angular velocity around x-, y-, and z- axes.

Discrete-time state vector at time stamp k of fixed length is shown in Equation 10.45

$$X_k = \begin{bmatrix} x_k & \dot{x}_k & \ddot{x}_k & y_k & \dot{y}_k & \ddot{y}_k & z_k & \dot{z}_k & \ddot{z}_k & \alpha_k & \dot{\alpha}_k & \beta_k & \dot{\beta}_k & \psi_k & \dot{\psi}_k \end{bmatrix}' \tag{10.45}$$

Discrete-time measurement vector at different time stamps k_r, k_{ax}, k_{ay}, k_{az} $k_{\alpha'}$ $k_{\beta'}$ $k_{\psi'}$, $k_{\alpha x}$ $k_{\beta y}$ $k_{\psi z}$ as described above is shown in Equation 10.46

$$Y_k = \begin{bmatrix} r_{1k} & \dot{r}_{1k} & r_{2k} & \dot{r}_{2k} & r_{3k} & \dot{r}_{3k} & r_{4k} & \dot{r}_{4k} & \theta_{xk} & \omega_{xk} & \theta_{yk} & \omega_{yk} & \theta_{zk} & \omega_{zk} & a_{xk} & a_{yk} & a_{zk} \end{bmatrix} \tag{10.46}$$

Corresponding coefficient matrix F of propagating state vector X_k to corresponding measurements in Y_k at each time stamp

$$F = \begin{bmatrix}
1 & T_{ax} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & T_{ax} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & T_{ay} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & T_{ay} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & T_{az} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_{az} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_{\alpha_1} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_{\beta_1} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & T_{\psi_1} \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \tag{10.47}$$

where T_{ai}, $i = x, y, z$ is the time difference to propagate acceleration at time stamp k to position and velocity calculated from RF signal at time stamp k_r along x, y, z axes, i.e. $T_{ai} = k_r - k$, where $i = x, y, z$, $r = 1, 2, 3, 4$. RF signals from readers $r = 1, 2, 3, 4$ are of the same time stamp.

$T_{\alpha 1}$, $T_{\beta 1}$ $T_{\psi 1}$ are the time difference to propagate angular velocity to orientation measured from magnetometer at time stamp α', β', ψ' around x-axis y-axis and z-axis, i.e. $T_{\alpha 1} = \alpha' - k$, $T_{\beta 1} = \beta' - k$, and $T_{\psi 1} = \psi' - k$.

Therefore, state vector is propagated to each time stamp in the measurement vector correspondingly. Then each time stamp of X_k and Y_k are the same to each other.

10.6.2 Estimation Using NLE and EKF in Synchronous Estimation

1. Synchronous Estimation Using NLE

In NLE, once we have a full vector of measurements Y_k, NLE described in section 10.3 is then applied to estimate the state vector X_k.

More specifically, the stacked measurement vector in NLE is

$$Z_k = \begin{bmatrix} Y_{k-l+1}, & Y_{k-l+2}, & \dots & , Y_k \end{bmatrix}^T = \begin{bmatrix} h(F^{l-1}X_{k-l+1}), & h(F^{l-2}X_{k-l+2}), & \dots & , h(X_k) \end{bmatrix}^T \tag{10.48}$$

Z_k of length l contains each measurement Y_i of full length 17, which is shown in Equation 10.46 and F is the same as shown in Equation 10.47, and X_k is shown in Equation 10.45.

2. Synchronous Estimation Using EKF

In EKF, once we have a full vector of measurements Y_k, which is the same as in Equation 10.46. F is as shown in Equation 10.43. EKF in section 10.4 is applied to estimate the state vector X_k. In addition R matrix is the variance of measurement data.

10.7 Orientation Estimation Based on Quaternion

To alleviate the singularity problem caused by the Euler angles representation, quaternions are used to represent the attitude of rigid body. The singularity is in general a point at which a given mathematical object is not defined and it causes the gimbal lock. The singularity occurs when the pitch angles rotation is $\theta = \pm\pi/2$ [77].

The quaternion vector q that represents the attitude is

$$q = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix} \tag{10.49}$$

Based on the rotation principal, rotation matrices based on Euler angles and quaternion are [77]

$$R = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix} \tag{10.50}$$

$$= \begin{bmatrix} \cos\theta_y\cos\theta_z & -\cos\theta_x\sin\theta_z + \sin\theta_x\sin\theta_y\cos\theta_z & \sin\theta_x\sin\theta_z + \cos\theta_x\sin\theta_y\cos\theta_z \\ \cos\theta_y\sin\theta_z & \cos\theta_x\cos\theta_z + \sin\theta_x\sin\theta_y\sin\theta_z & -\sin\theta_x\cos\theta_z + \cos\theta_x\sin\theta_y\sin\theta_z \\ -\sin\theta_y & \sin\theta_x\cos\theta_y & \cos\theta_x\cos\theta_y \end{bmatrix} \tag{10.51}$$

$$= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{10.52}$$

Therefore, the pitch θ_x roll θ_y and yaw θ_z angles derived from quaternion are

$$\theta_x = \arctan\left(\frac{R_{23}}{R_{33}}\right) = \arctan\left(\frac{2(q_2q_3 - q_0q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\right) \tag{10.53}$$

$$\theta_y = -\arcsin(R_{13}) = -\arcsin(2(q_1q_3 + q_0q_2)) \tag{10.54}$$

$$\theta_z = \arctan\left(\frac{R_{12}}{R_{11}}\right) = \arctan\left(\frac{2(q_1q_2 - q_0q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2}\right) \tag{10.55}$$

Where θ_x is pitch angle in the range $[-\pi, \pi]$, θ_y is roll angle in the range $[-\pi/2, \pi/2]$, and θ_z is yaw angle in the range $[-\pi, \pi]$.

Measurement accelerations $[a_x, a_y, a_z]$ after rotating Euler angles of $[\theta_x, \theta_y, \theta_z]$ or quaternion representation angles of $[q_0, q_1, q_2, q_3]$ are

$$\begin{bmatrix} a'_x \\ a'_y \\ a'_z \end{bmatrix} = R \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \tag{10.56}$$

In static states, there only exists gravity in acceleration. In this case, measured accelerations are

$$\begin{bmatrix} a'_x \\ a'_y \\ a'_z \end{bmatrix} = R \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{10.57}$$

$$= \begin{bmatrix} 2(q_1q_3 + q_0q_2)g \\ 2(q_2q_3 + q_0q_1)g \\ (q_0^2 - q_1^2 - q_2^2 + q_3^2)g \end{bmatrix} \tag{10.58}$$

subsection 10.7.1 and subsection 10.7.3 describe orientation estimation using angular velocities from gyroscope and accelerations from accelerometer in quaternion [78].

10.7.1 State Space Model for Quaternion Based Orientation Estimation

State space model to estimate orientation based on quaternion is as shown below [77, 78]

$$\dot{q} = \frac{1}{2} W q \tag{10.59}$$

$$y = h(q) + v_k \tag{10.60}$$

where W is composed of angular velocities ω_x ω_y ω_z along x- y- z- axes

$$W = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \tag{10.61}$$

and the nonlinear transformation from quaternion to accelerations is the same as in Equation 10.58

$$h(q) = \begin{bmatrix} 2(q_1q_3 + q_0q_2)g \\ 2(q_2q_3 + q_0q_1)g \\ (q_0^2 - q_1^2 - q_2^2 + q_3^2)g \end{bmatrix} \tag{10.62}$$

Discrete-time model of Equation 10.59 and Equation 10.60 are as shown in the following

$$q_k = F_{k-1}q_{k-1} + \omega_k \tag{10.63}$$

$$Y_k = h(q_k) + v_k \tag{10.64}$$

where

$$F_k = \frac{1}{2}W_kT + I = \begin{bmatrix} 1 & -\frac{1}{2}\omega_xT & -\frac{1}{2}\omega_yT & -\frac{1}{2}\omega_zT \\ \frac{1}{2}\omega_xT & 1 & \frac{1}{2}\omega_zT & -\frac{1}{2}\omega_yT \\ \frac{1}{2}\omega_yT & -\frac{1}{2}\omega_zT & 1 & \frac{1}{2}\omega_xT \\ \frac{1}{2}\omega_zT & \frac{1}{2}\omega_yT & -\frac{1}{2}\omega_xT & 1 \end{bmatrix} \tag{10.65}$$

where ω_x, ω_x, and ω_x are angular velocities along x-, y-, and z- axes at time stamp k. T is the time difference between time stamp k and time stamp $k-1$.

10.7.2 Quaternion Based Orientation Estimation Using NLE

The stacked measurement vector of the quaternion based orientation estimation

$$Z_k = \begin{bmatrix} Y_{k-l+1}, & Y_{k-l+2}, & \dots & , Y_k \end{bmatrix}^T \tag{10.66}$$

where Y_k is the noisy measurement acceleration vector

$$Y_k = \begin{bmatrix} a_{xk} & a_{yk} & a_{zk} \end{bmatrix} \tag{10.67}$$

where a_{xk}, a_{yk}, a_{zk} are simulated/measurement accelerations with added noise at time stamp k of sampling period T along x-axis y-axis and z-axis.

A mapping from state and input values to output values is denoted as H

$$H_k(X_{k-l+1}) = \begin{bmatrix} h(X_{k-l+1}) \\ h(FX_{k-l+1}) \\ .. \\ h(F^{l-1}X_{k-l+1}) \end{bmatrix} \tag{10.68}$$

where F is shown in Equation 10.23. In the absence of modeling error and sensor noise,

$$Z_k = H_k(X_{k-l+1}) \tag{10.69}$$

holds true [49]. In case of noise, NLE minimizes the sum of the squares of the residual errors

$$\hat{X}_{k-l+1} = \text{argmin}||Z_k - H_k(X_{k-l+1})||^2 \tag{10.70}$$

This least-squares state estimate is l samples delayed in time, so it is propagated forward in time using the nominal system dynamics to get [49]

$$\hat{X}_k = F^{l-1}X_{k-l+1} \tag{10.71}$$

The implementation of estimation is the same as in section 10.3.

10.7.3 Quaternion Based Orientation Estimation Using EKF

Predict and update stages of EKF to estimate the attitude using quaternion are

Predict:

$$\hat{q}_k = F_{k-1}\hat{q}_{k-1} \tag{10.72}$$

$$P_k = F_{k-1}P_{k-1}F_{k-1}^T + Q \tag{10.73}$$

Update:

$$G_k = P_k H_k^T (H_k P_k H_k^T + R)^{-1} \tag{10.74}$$

$$\hat{X}_k = \hat{X}_k + G_k(z_k - h(\hat{X}_k)) \tag{10.75}$$

$$P_k = (I - G_k H_k)P_k \tag{10.76}$$

In addition, P_k is the prediction error matrix, Q is the covariance matrix of ω_k, and R is the covariance matrix of v_k.

Estimation results of attitude based on the models above are shown in Figure 10.1. The cellphone firstly rotates counter-clockwisely along x- axis for 2π, i.e., θ_x rotating counter-clockwise for 2π rad. Then rotating counter-clockwisely along y- axis for 2π rad. Finally rotating along z- axis for 2π rad.

Rotation angles are defined counterclockwise as shown in Figure 9.1. In Figure 10.1, roll angle θ_x increases to π and decreases to $-\pi$, travelling a total orientation of 2π. The yaw angle θ_z is of the same motion profile. Pitch angle θ_y increases to $\pi/2$ and decreases to $-\pi/2$, travelling a total orientation of π.

As shown in Figure 10.1, there exists jitters in estimated roll θ_x and yaw θ_z due to the gimbol lock in transforming quaternion to Euler angles based on Equation 10.55 when pitch θ_y is close to $\pm\pi/2$.

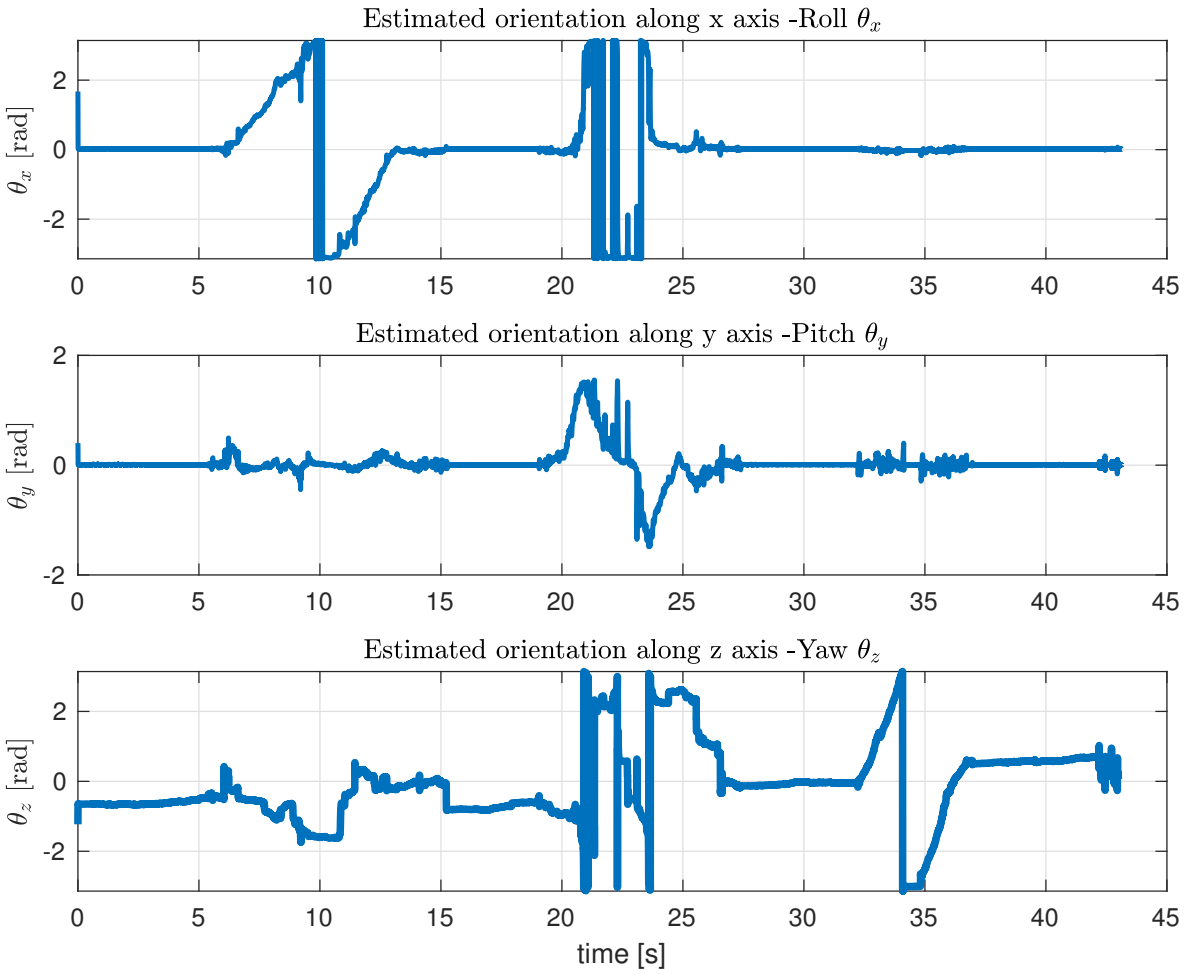


Figure 10.1: Estimated Orientation Along x- y- and z- Axes, i.e., Roll θ_x Pitch θ_y Yaw θ_z Using Quaternion and EKF.

CHAPTER 11

RESULTS OF 3D NONLINEAR LEAST-SQUARES ESTIMATION

In this chapter, estimation based on NLE and EKF is implemented on measurement data including RF signal and IMU sensors. Estimation results using both RF and IMU and RF signal only are compared and evaluated. From estimation results, RF-signal achieves similar estimation results to RF-signal with added inertial information. In addition, stacked states in NLE maintain the motion-capture grade, i.e., 30 cm estimation accuracy for states at various noise levels in indoor multipath environment. Estimation results also verify the similar estimation results between NLE and EKF. In addition, NLE is of 2 cm smaller RMS error compared to EKF in both estimations.

This is the first attempt in the literature of a true 3D RFID-based motion capture system. The collection includes amplitude, phase, and IMU data to provide a comprehensive data set that can be down-selected to explore any number of sensor fusion estimation techniques. Selectively using some or part of the data set will allow us to judge the relative contributions of each piece of data to the overall location accduracy. NLE and EKF using different sensor information $\{r\}$, $\{r, \dot{r}\}$, $\{r, \dot{r}, \theta_x, \theta_y, \theta_z, \omega_x, \omega_y, \omega_z, a_x, a_y, a_z\}$ corresponding to output vectors in Equation 10.19 or Equation 10.22 are implemented following estimation steps in section 10.3 and section 10.4. [1]

11.1 Bias Jump After Motion

In cellphone based IMU data, there is a bias jump after each motion as shown in Figure 11.1. For the low-cost cellphone embedded IMU, there exists a threshold for sensor measurements to be identified. Accelerations > 0.0447 m/s^2 can be identified.

[1] https://github.com/qq3575022/3DIndoorStateEstimationRFIDMotionCapture/tree/main/Chapter11

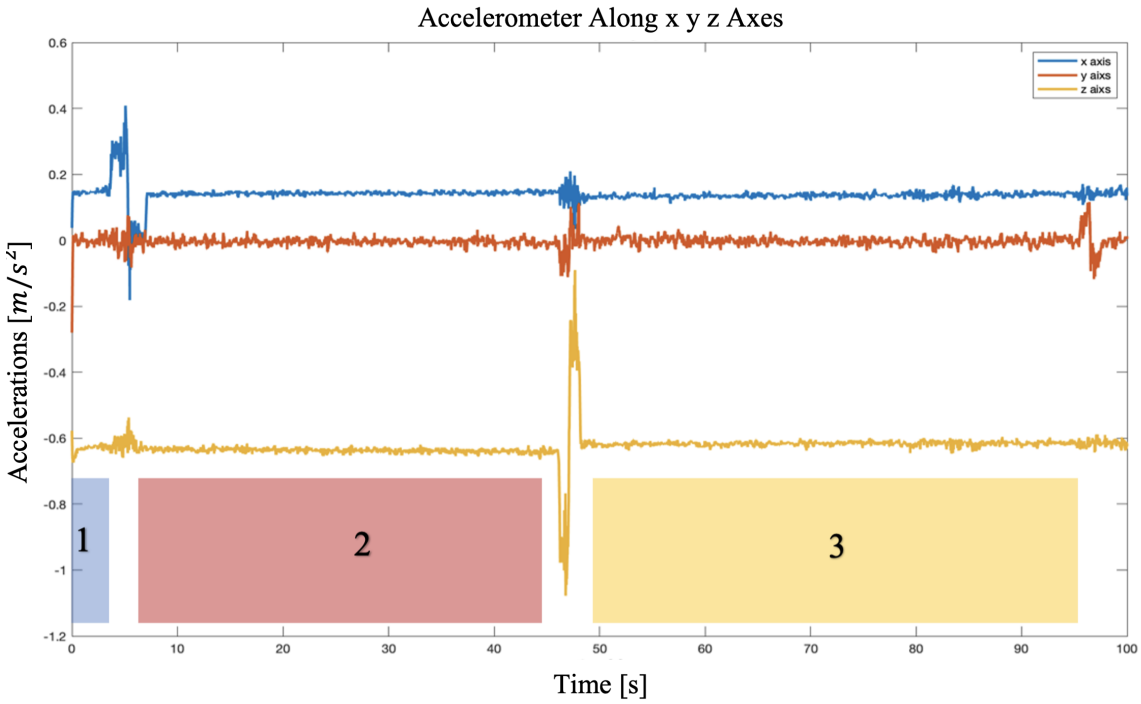


Figure 11.1: Acceleration of 0.128 m/s^2 Along x- Axis and Then 0.2943 m/s^2 Along y- Axis and Finally 0.0447 m/s^2 Along z- Axis, Dividing the Static Regions into 1, 2, 3.

The mean values of bias for accelerations along x- y- and z- axes in static regions 1, 2, 3 are shown in Table 11.1.

Table 11.1: Mean Values of Accelerations in Regions 1, 2, 3

	Region 1	Region 2	Region 3
Mean Error Along x Axes [m/s^2]	0.1453	0.1427	0.1364
Mean Error Along y Axes [m/s^2]	-0.0063	-0.0043	-0.0023
Mean Error Along z Axes [m/s^2]	-0.6318	-0.6371	-0.6162

As shown in Table 11.1, the bias in regions 1, 2, 3 are different for all accelerations along different axes. In addition, the maximal difference among different regions is 0.0209 m/s^2. Therefore, updates on parameters of error model need to be updated in each static region.

11.2 Static Detection on IMU Measurement

Zero velocity detection in subsection E.2.1 is applied to detect the stationary states for calibration. Hard thresholds on accelerations and angular velocities are applied for detection.

The result is shown in Figure 11.2.

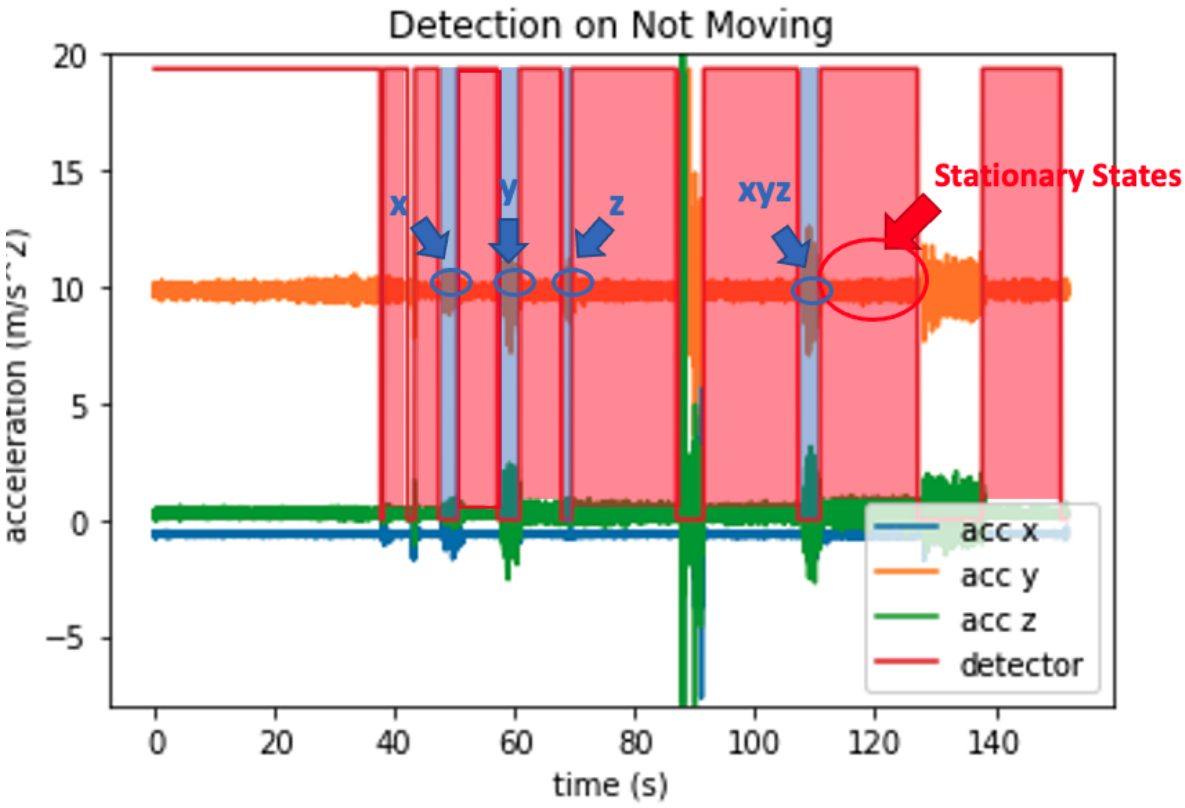


Figure 11.2: Zero Velocity Detection For Accelerations and Angular Velocities Along x- y- and z- Axes Using Hard Set Thresholds: 0.37 m/s^2 for Accelerations and 0.018 rad/s for Angular Velocities.

In Figure 11.2, blue lines are acceleration along x- axis, orange lines are acceleration along y- axis, and green lines are acceleration along z- axis. The red rectangle are states of detected static regions. It can be inferred from Figure 11.2 that motion along x- y- and z- axes separately and x- y- z- axes all together are identified. In addition, static states in between are detected as stationary states as shown in the red rectangles.

More specifically, the start time stamps in the motion along x- axis first, then along y- axis, and finally along z- axis as well as moving along x- y- z- axes all together are shown in Table 11.2.

In addition, the small time difference, like in the detected method < 0.1 s, the estimation

Table 11.2: Static Detection on IMU Accelerometer and Gyroscope Measurements

Start TimeStamp in Motion	Along x-	Along y-	Along z-	Along xyz-
Detected Start TimeStamp	47.32 s	57.09 s	67.72 s	107.153 s

results are not affected. Moreover, the start times stamps are very close to the ground truth start time based on the motion profile.

11.3 Orientation Estimation in the Motion

Rotation angles are defined counterclockwise as shown in Figure 11.3. In Figure 11.3, roll angle θ_x, yaw angle θ_z, and pitch angle θ_y decreases to a constant around 0, indicating the orientation in the motion does not change. The input are acceleration and angular velocity as described in subsection 10.7.3. More specifically, the accelerations are calibrated with parameters derived in section E.6.

11.4 Sensor Fusion Based Estimation Using Both RF and IMU Sensor Data

The IMU accelerometer measurement data are calibrated using the scalar and skew matrices from Table E.3 and the bias estimated before input into the NLE estimation. The values are shown in Table 11.3 as well.

In addition, the bias of accelerometer change in different setup and also have time drift. Bias value of accelerometer are estimated with Kalman filer. For easier implementation, mean values are applied instead. It is tested that mean value of time difference larger than a few minutes are still able to get reasonable estimation results.

Moreover, since the orientation and angular velocity do not change much in the motion, bias of gyroscope as well as orientation derived in Figure 11.3 are compensated for calibration.

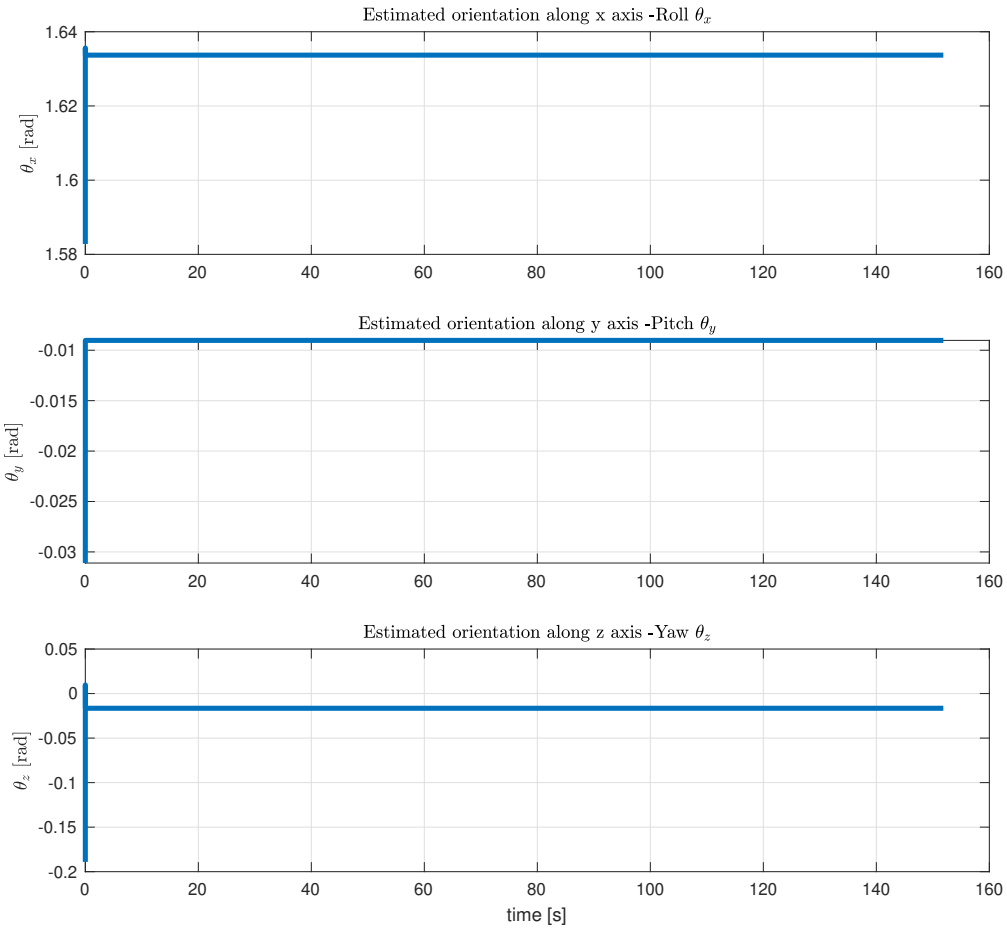


Figure 11.3: Estimated Orientation Along x- y- and z- Axes, i.e., Roll θ_x Pitch θ_y Yaw θ_z Using Quaternion and EKF.

Table 11.3: Derived Scalar Factor Bias and Skew Factor in Measurement

Skew Factor		Scalar Factor		Bias	
$-\alpha_{yz}$	0.0623	s_x^a	0.9944	b_x^a	0.1739
α_{zy}	0.0055	s_y^a	0.9999	b_y^a	0.0071
$-\alpha_{zx}$	-0.0041	s_z^a	0.9880	b_z^a	-0.2999

In the estimation below, the tagged object is moving along x- y- z- axes all together simultaneously.

11.4.1 NLE Based State Estimation on RF and IMU Data

Following steps in the asynchronous estimation in section 10.5 that is based on NLE in section 10.3, sensor fusion based estimation including RF and IMU sensor data using NLE is implemented. Input are radial distance r and radial velocity $\dot{r}$ as well as IMU sensor data: accelerations, angular velocities from gyroscope, and orientation from magnetometer. The sampling rate of RF signal is 4847 Hz and the sampling rate of IMU is 813 Hz. Estimation results of using both RF signal and IMU $\{r, \dot{r}, \theta_x, \theta_y, \theta_z, \omega_x, \omega_y, \omega_z, a_x, a_y, a_z\}$ based on NLE of stack length 5 are shown in Table 11.4 and Figure 11.4.

Table 11.4: RMS Error for Estimated Position Along x- y- and z- Axes All Together Based on NLE in Asynchronous Model Using RF Signal and Cellphone Embedded Measurement Data with Stack Length of 5.

	RMS Error for 4 Readers		
	3D [mm]	xy-plane [mm]	z-axis [mm]
Before Estimation	719.9	349.9	629.2
r $\dot{r}$ θ_x, θ_y, θ_z, ω_x, ω_y, ω_z, a_x, a_y, a_z	319.0	225.0	226.1

In addition, the estimated positions, velocities, and accelertaions along x- y- and z- axes of NLE with stack length of 5 on RF and IMU sensors are show in Figure 11.4.

It can be inferred from Table 11.4 and Figure 11.4 that estimated position and velocity states follow the ground truth position and velocity. In addition, estimated RMS error along z- axis is larger than RMS error in xy- plane similar to measurement data before estimation. In the xy- plane, NLE based estimation achieves 22.50 cm level accuracy in multipath. Therefore, NLE based sensor fusion estimation achieves fine-scale localization of 31.90 cm from 71.99 cm in 3D motion in real-time.

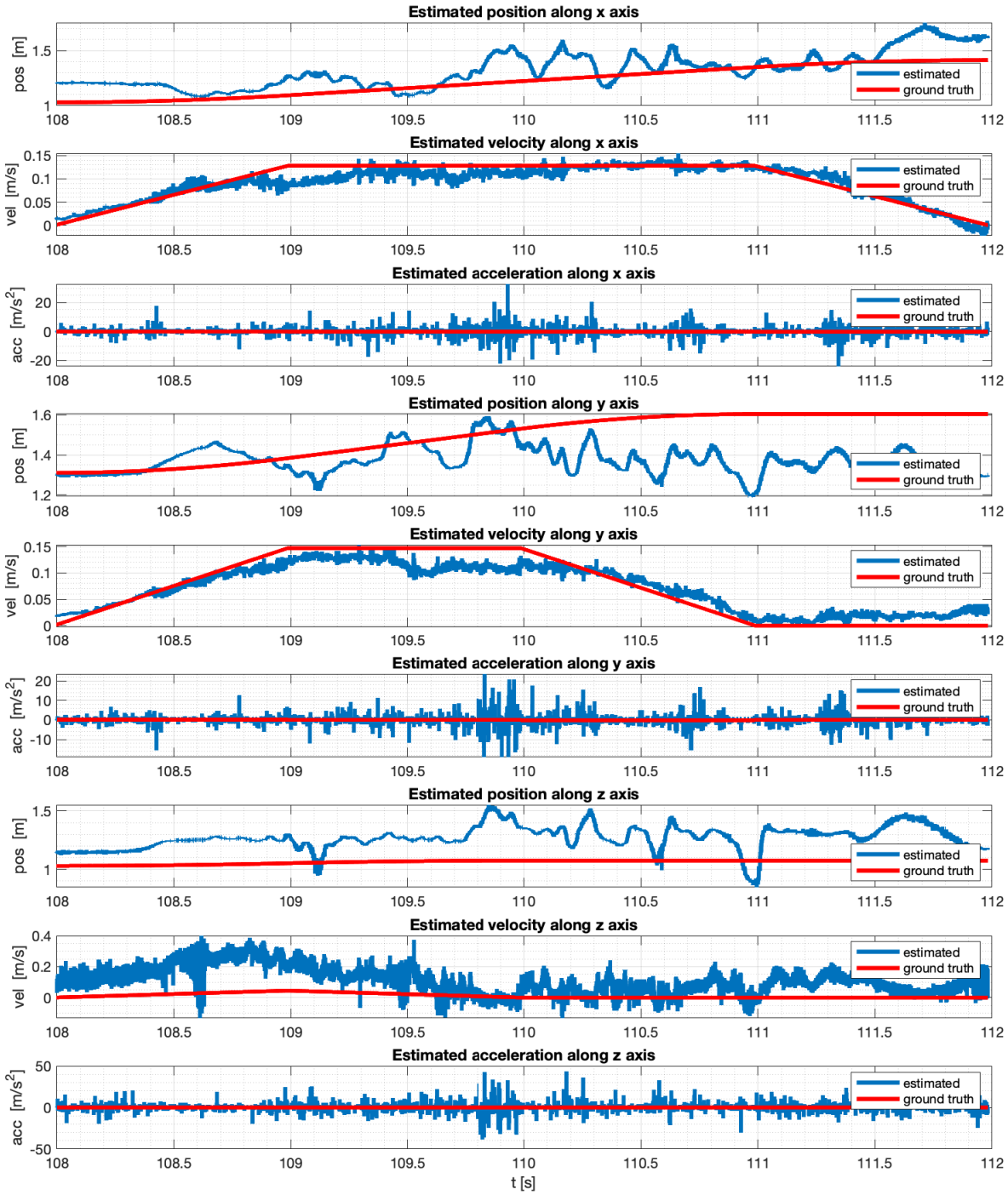


Figure 11.4: Estimation Results of the 3D Motion Along x- y- and z- Axes Using RF Signal and Cellphone Embedded Measurement Data Based on NLE in Asynchronous Model When Moving x- y- and z- Axes All Together Simultaneously with Stack Length of 5.

11.4.2 EKF Based State Estimation on RF and IMU Data

Following steps in the asynchronous estimation in section 10.5 that is based on EKF in section 10.4, sensor fusion based estimation including RF and IMU sensor data using EKF is implemented. Input are radial distance r and radial velocity $\dot{r}$ as well as IMU sensor data: accelerations, angular velocities from gyroscope, and orientation from magnetometer. The sampling rate of RF signal is 4847 Hz and the sampling rate of IMU is 813 Hz. Estimation results of using both RF signal and IMU $\{r, \dot{r}, \theta_x, \theta_y, \theta_z, \omega_x, \omega_y, \omega_z, a_x, a_y, a_z\}$ based on EKF are shown in Table 11.5 and Figure 11.5.

Table 11.5: RMS Error for Estimated Position Along x- y- and z- Axes All Together Based on EKF in Asynchronous Model Using RF Signal and Cellphone Embedded Measurement Data.

	RMS Error for 4 Readers		
	3D [mm]	xy-plane [mm]	z-axis [mm]
Before Estimation	719.9	349.9	629.2
r $\dot{r}$ θ_x, θ_y, θ_z, ω_x, ω_y, ω_z, a_x, a_y, a_z	335.4	230.5	234.7

It can be inferred from Table 11.5 and Figure 11.5 that estimates position and velocity states based on EKF are similar to ground truth. In addition, estimation results of EKF is similar to NLE, which validate both estimation methods. Hence, sensor fusion based estimation in 3D motion achieves motion capture level accuracy, i.e., 33.54 cm, which holds the similar accuracy level in 3D motion in NLE of 31.90 cm.

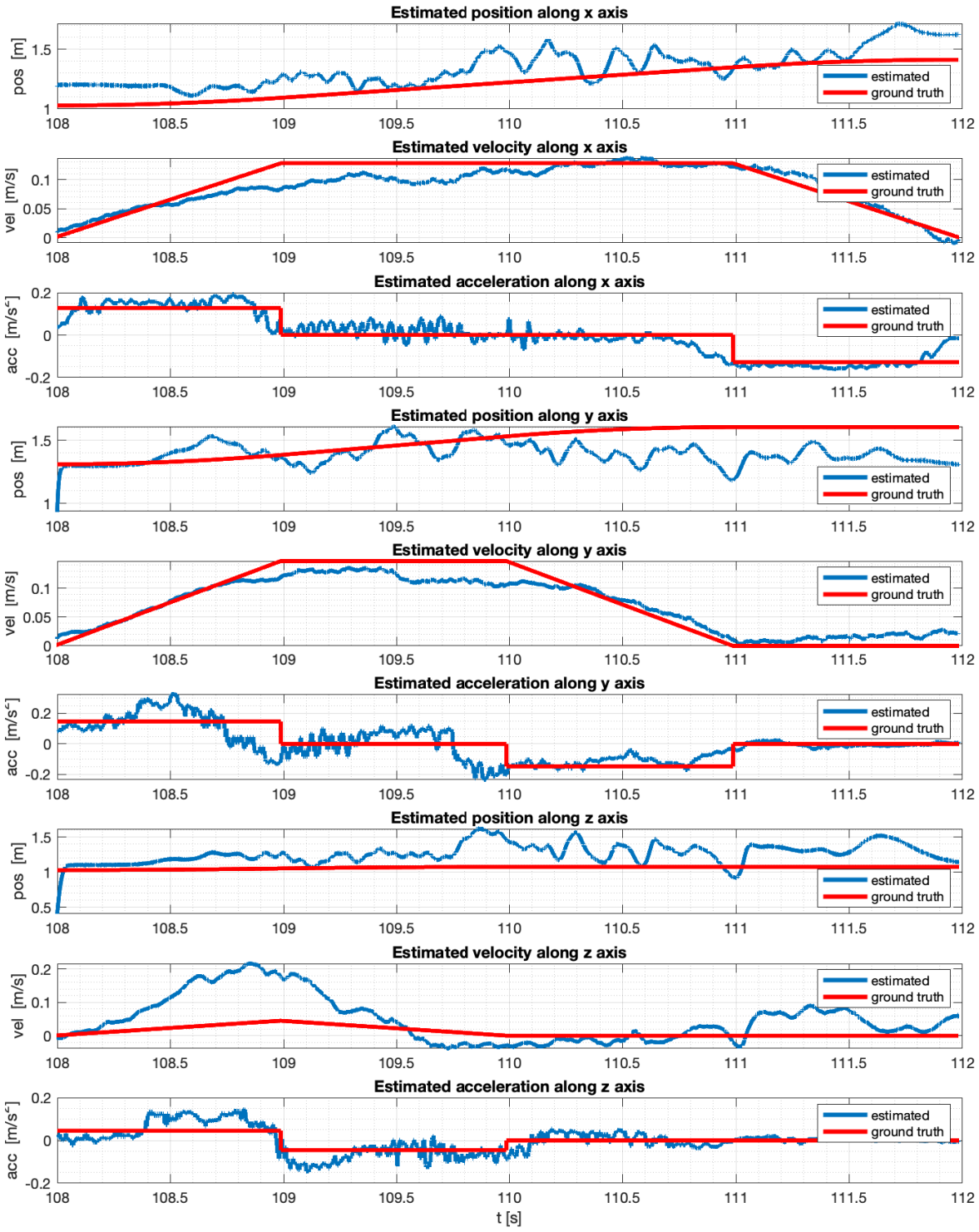


Figure 11.5: Estimation Results of the 3D Motion Along x- y- and z- Axes Using RF Signal and Cellphone Embedded Measurement Data Based on EKF in Asynchronous Model When Moving x- y- and z- Axes All Together Simultaneously.

11.5 RF Signal Only Based Estimation

In this section, we show that RF signal only, i.e., sensor combination $\{r\}$, $\{r, \dot{r}\}$ achieve the same level accuracy as sensor fusion based estimation. Thus, RF signal only is able to achieve motion capture level accuracy estimation. In this way, RFID based motion capture systems can be greatly simplified from embedding IMU.

11.5.1 NLE Based State Estimation on RF Signal Only

Asynchronous estimation including RF sensor data using NLE is implemented. Input are radial distance r and radial velocity $\dot{r}$. The sampling rate of RF signal is 4847 Hz. Estimation results of using RF signal only $\{r\}$, $\{r, \dot{r}\}$ based on NLE of stack length 5 are shown in Table 11.6 and Figure 11.6.

Table 11.6: RMS Error for Estimated Position Along x- y- and z- Axes All Together Based on NLE in Asynchronous Model Using RF Signal Only with Stack Length of 5.

	RMS Error for 4 Readers		
	3D [mm]	xy-plane [mm]	z-axis [mm]
Before Estimation	719.9	349.9	629.2
r	319.6	225.7	226.2
$r\ \dot{r}$	319.0	224.8	226.2

It can be inferred from Table 11.6 and Figure 11.6 that using RF signal only achieves similar position accuracy to using both RF signal and IMU data together. More specifically, using RF signal only is of 31.96 cm and 31.90 cm accuracies. The estimated position and velocity states are close to the shape of ground truth position and velocity states. Hence, RF signal only based estimation achieves motion capture level accuracy, i.e., 31.90 cm, using only RF signal, which is similar to using both RF signal and IMU data of 31.90 cm.

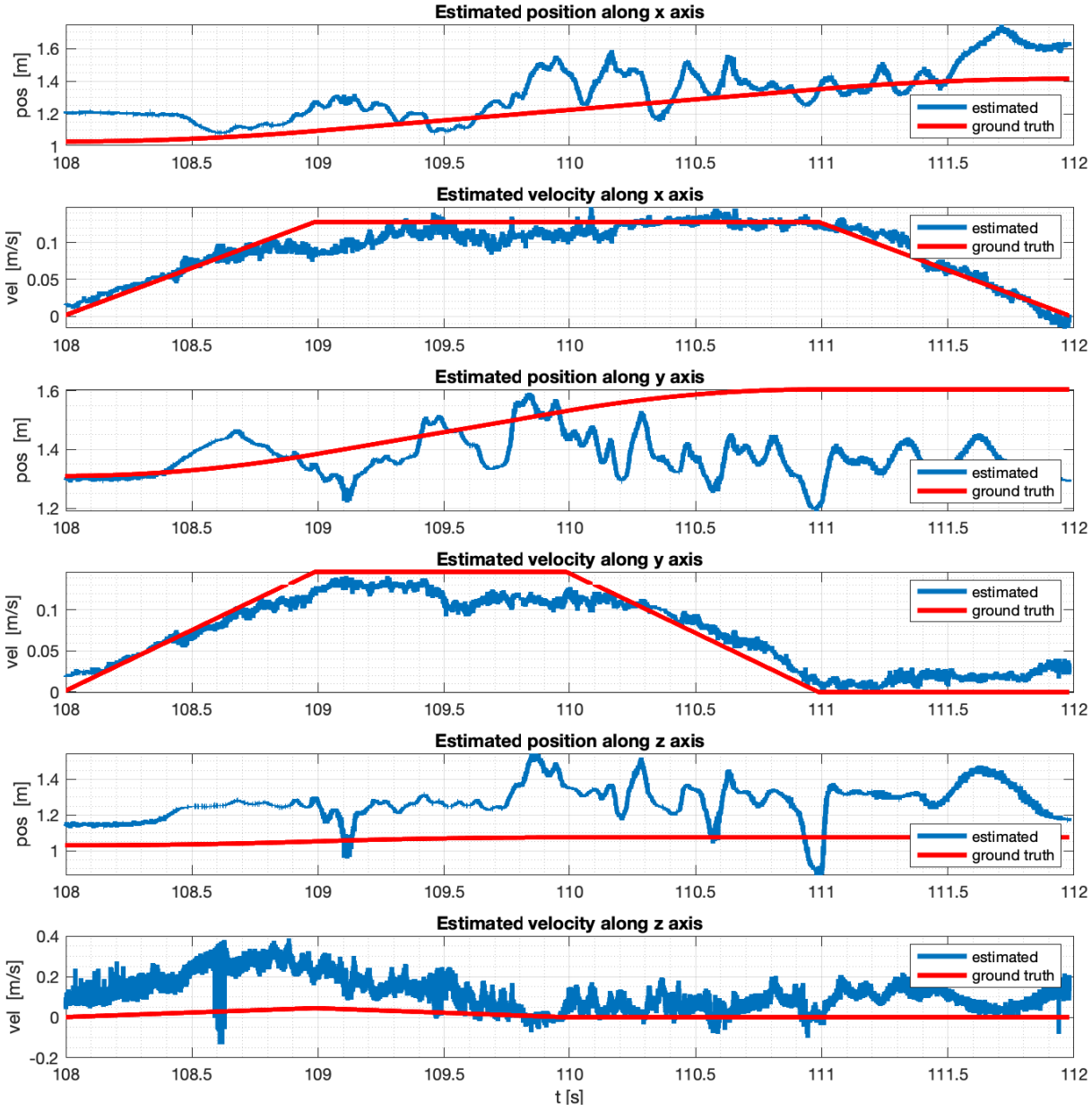


Figure 11.6: Estimation Results of the 3D Motion Along x- y- and z- Axes Using RF Signal Only Based on NLE in Asynchronous Model When Moving x- y- and z- Axes All Together Simultaneously with Stack Length of 5.

11.5.2 EKF Based State Estimation on RF Signal Only

Asynchronous estimation including RF sensor data using EKF is implemented. Input are radial distance r and radial velocity $\dot{r}$. The sampling rate of RF signal is 4847 Hz. Estimation results of using RF signal only $\{r\}$, $\{r, \dot{r}\}$ based on EKF are shown in Table 11.7 and Figure 11.7.

Table 11.7: RMS Error for Estimated Position Along x- y- and z- Axes All Together Based on EKF in Asynchronous Model Using RF Signal Only.

	RMS Error for 4 Readers		
	3D [mm]	xy-plane [mm]	z-axis [mm]
Before Estimation	719.9	349.9	629.2
r	339.3	233.0	246.7
$r\ \dot{r}$	335.9	230.8	244.0

It can be inferred from Table 11.7 and Figure 11.7 that EKF estimation on RF signal only is of similar results to EKF state estimation using sensor fusion of RF signal and IMU together as well as NLE estimation on RF signal only, i.e., 33.59 cm compared to 33.54 cm and 31.90 cm.

In summary, RF signal only is able to achieve motion capture level accuracy of 31.90 cm in 3D state estimation in indoor multipath environment. Therefore, RF signal only based state estimation is of similar accuracy to sensor fusion based localization using both RF signal and IMU data in 3D motion, which is similar to 2D scenario and thus simplifying the RFID based motion capture system from embedding IMU.

11.6 Relation Between Newton Observer and EKF

It has shown that the measurement update equations for $\hat{x}_k$ in the iterated EKF are exactly as the minimization problem when a Gauss-Newton method is used with $\hat{x}_k^-$ as initial guess.

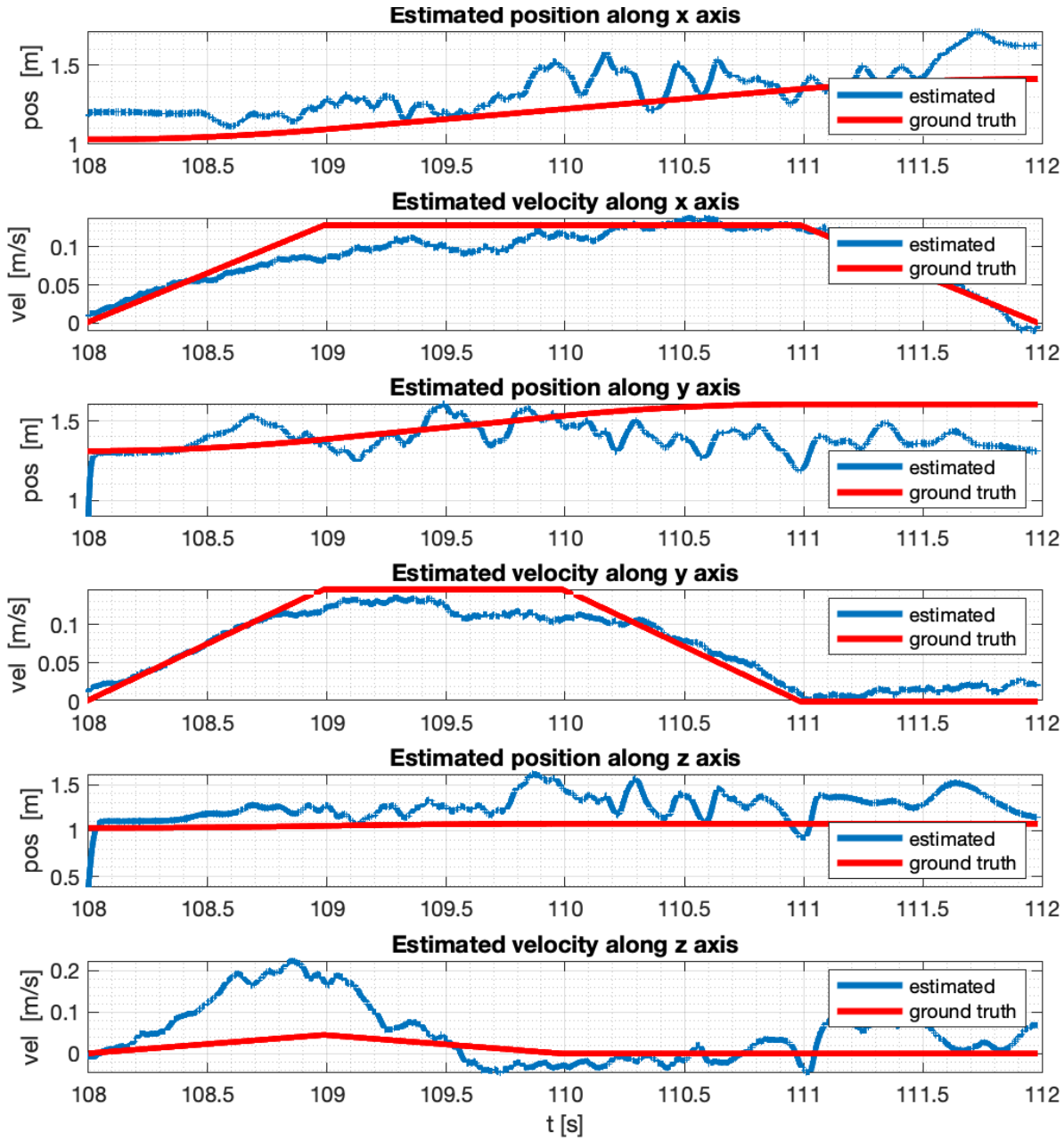


Figure 11.7: Estimation Results of the 3D Motion Along x- y- and z- Axes Using RF Signal Only Based on EKF in Asynchronous Model When Moving x- y- and z- Axes All Together Simultaneously.

Hence, the covariance matrices R and Q_k^- may be interpreted as weights on the norms in the ouput space and state space [79]. In the derivation, the number of measurements is equal to the number of state variables, i.e., the output map H is $n \times n$.

CHAPTER 12

COMPARISON OF STATE ESTIMATION ON THREE KINDS OF POSITION SENSORS

In this chapter, not only the magnitude based position sensor in measurement in Figure 10.7.3 but also all the three kinds of position sensors: magnitude position sensor in section 8.1, ToF based position sensor in section 8.2, and ToF based position sensor in tracking in section 8.3 are used for 3D state estimation based on NLE and EKF. [1]

In the simulation:

1. The sampling rate of RF is 4847 Hz, sampling rate of IMU is 813 Hz, time stamps in simulation are the same as in measurement.

2. All ToF in static or in tracking uses the same 4 frequencies.

3. All frequency separations in ToF is 1 MHz.

4. Both statistical noise and 6 deterministic reflection are added in the simulation model compared to RMS error of measured magnitude. In all simulations for RSS and ToF, the K values are 40, 400, 11, 56 for each reader and the standard deviation for phase noise is all 0.02 as shown in Table 7.1.

In addition, definition of static, tracking with motion, simulated RSS and simualted ToF can be inferred from subsection 8.4.2.

[1] `https://github.com/qq3575022/3DIndoorStateEstimationRFIDMotionCapture/tree/main/Chapter12`

12.1 Using Both RF and IMU All Together Based on NLE and EKF

In this section, comparisons of estimated 3D coordinates (x, y, z) based on three position sensors in LOS and multipath in static and tracking integrated with simulated IMU data based on sensor model in section 9.2 along motion trajectory described in chapter 6 are shown in Table 12.1 and Table 12.2.

Table 12.1: RMS Error Based on 3D Coordinates (x, y, z) Using Triangulation in subsection 8.4.1 Based on NLE in Asynchronous Model Using Three Position Sensors and Simulated Cellphone IMU Data Compared to Ground Truth Trajectory in chapter 6.

			RMS Error for 4 Readers		
			3D [mm]	xy-plane [mm]	z-axis [mm]
Line-of-Sight (LOS)	Static	Simulated RSS	359.5	95.4	346.6
		Simulated ToF	38.8	19.4	33.6
	Tracking with Motion	Simulated RSS	359.5	95.4	346.6
		Simulated ToF	173.3	170.1	33.2
Multipath	Static	Simulated RSS	386.8	161.9	351.3
		Simulated ToF	82.2	41.8	70.8
	Tracking with Motion	Measured RSS	319.0	225.0	226.1
		Simulated RSS	386.8	161.9	351.3
		Simulated ToF	395.6	157.5	362.9

Table 12.2: RMS Error Based on 3D Coordinates (x, y, z) Using Triangulation in subsection 8.4.1 Based on EKF in Asynchronous Model Using Three Position Sensors and Simulated Cellphone IMU Data Compared to Ground Truth Trajectory in chapter 6.

			RMS Error for 4 Readers		
			3D [mm]	xy-plane [mm]	z-axis [mm]
Line-of-Sight (LOS)	Static	Simulated RSS	109.8	55.4	94.8
		Simulated ToF	89.0	52.2	72.1
	Tracking with Motion	Simulated RSS	109.8	55.4	94.8
		Simulated ToF	190.4	175.5	73.8
Multipath	Static	Simulated RSS	265.0	147.9	219.9
		Simulated ToF	111.3	65.1	90.3
	Tracking with Motion	Measured RSS	335.4	230.5	234.7
		Simulated RSS	265.0	147.9	219.9
		Simulated ToF	380.9	162.5	343.5

It can be inferred from Table 12.1 and Table 12.2 that:

1. Similar to results before estimation, RMS error in 3D for ToF based position sensors is smaller than RSS based position sensor in LOS and multipath with static or tracking with motion for NLE out of better non-Gaussian noise cancellation.

2. EKF has better noise cancellation for RSS based position sensor compared to ToF based position sensors out of better Gaussian noise cancellation.

3. More specifically, for RSS based position sensor, after estimation RMS errors are 10.98 cm and 26.50 cm in EKF compared to 35.95 cm and 38.68 cm in NLE. In contrast, for ToF based position sensors, RMS errors are 8.90 cm, 19.04 cm, 11.13 cm, and 38.09 cm in EKF compared to 3.88 cm, 17.33 cm, 8.22 cm, and 39.56 cm in NLE.

4. After estimation, RMS in 3D is reduced by half compared to Table 8.2. In addition, most reduction in along z-axis for NLE based estimation in asynchronous model.

5. Estimated RMS error in xy-plane based on simulated position sensors is smaller than RMS error along z-axis, which is similar to the measurement estimation results in Table 11.4 - Table 11.7.

6. NLE outperforms EKF in estimation for ToF based position sensors, in tracking and multipath scenarios for non-Gaussian noise.

7. EKF is of higher estimation accuracy for RSS based position sensors, especially in LOS in which Gaussian noise is dominant.

8. NLE has better cancellation for large error i.e., error along z- axis in the motion.

9. In multipath simulated RSS based estimation, accuracy in xy-plane is 26.50 cm, which is similar to estimation results in Table 11.6 and Table 11.7.

In LOS, the main noise source of RSS is Gaussian noise. EKF like KF is designed based on Gaussian assumptions. On the other side, the larger the stack length is, the more accurate NLE gives. The bias also increases if the stack length is too large. In multipath and tracking scenarios, NLE comparably enhances non-Gaussian noise more compared to EKF especially for ToF.

12.2 Using RF Signal Only based on NLE and EKF

In this section, comparisons of estimated 3D coordinates (x, y, z) based on three position sensors of RF signal only in LOS and multipath in static and tracking along motion trajectory described in chapter 6 are shown in Table 12.3 and Table 12.4 respectively.

Table 12.3: RMS Error Based on 3D Coordinates (x, y, z) Using Triangulation in subsection 8.4.1 Based on NLE in Asynchronous Model Using Three Position Sensors Only Compared to Ground Truth Trajectory in chapter 6.

			RMS Error for 4 Readers		
			3D [mm]	xy-plane [mm]	z-axis [mm]
Line-of-Sight (LOS)	Static	Simulated RSS	354.0	101.5	339.2
		Simulated ToF	42.3	18.2	38.2
	Tracking with Motion	Simulated RSS	354.0	101.5	339.2
		Simulated ToF	173.4	169.2	37.5
Multipath	Static	Simulated RSS	382.8	165.5	345.2
		Simulated ToF	87.0	42.9	75.7
	Tracking with Motion	Measured RSS	319.0	224.8	226.2
		Simulated RSS	382.8	165.5	345.2
		Simulated ToF	395.4	156.8	363.0

It can be inferred from Table 12.3 and Table 12.4 that:

1. Using simulated RF signal only gets very similar estimation results to using both simulated RF signal and IMU data based on NLE and EKF in asynchronous model, the difference for both NLE and EKF $<$ 5 mm.

2. Similar to using both RF signal and IMU data together, estimated RMS error for simulated ToF with multipath in static is smaller than estimation error of simulated RSS in LOS and multipath.

3. In addition, ToF based position sensor in tracking with phases of adjacent time stamps still keeps high localization accuracy.

4. The estimation error along z-axis is larger than RMS error in the xy-plane like in estimation with simulated RF signal and IMU data together.

Table 12.4: RMS Error Based on 3D Coordinates (x, y, z) Using Triangulation in subsection 8.4.1 Based on EKF in Asynchronous Model Using Three Position Sensors Only Compared to Ground Truth Trajectory in chapter 6.

			RMS Error for 4 Readers		
			3D [mm]	xy-plane [mm]	z-axis [mm]
Line-of-Sight (LOS)	Static	Simulated RSS	111.9	57.0	96.3
		Simulated ToF	92.3	54.1	74.8
	Tracking with Motion	Simulated RSS	111.9	57.0	96.3
		Simulated ToF	191.9	176.1	76.3
Multipath	Static	Simulated RSS	264.3	148.5	218.6
		Simulated ToF	113.8	66.6	92.2
	Tracking with Motion	Measured RSS	335.9	230.8	244.0
		Simulated RSS	264.3	148.5	218.6
		Simulated ToF	379.8	163.0	343.1

In summary, for both NLE and EKF, estimation results using RF signal only is similar to using both RF signal and IMU data in simulation, which are similar to measurement estimation results. Before and after estimation, RMS error along z-axis is larger than RMS error in xy-plane. NLE and EKF Estimation reduces raw RMS error by half. Before and after estimation, ToF based position sensor gains higher localization accuracy in 3D compared to RSS based position sensor.

CHAPTER 13

CONCLUSIONS

This dissertation presented the competitive fine-scale and computational efficient wireless position estimation with low infrastructure using both real measurements and realistic simulation models, both of which included multipath effects. 3D indoor state estimation for RFID based motion capture systems achieves 31.90 cm in the 3D motion in the indoor multipath environment. More specifically, 22.50 cm in xy-plane and 22.61 cm along z-axis. In addition, RF signal only estimation achieves 31.90 cm in 3D motion, 22.48 cm in xy-plane and 22.62 cm along z-axis, enabling RF signal only based motion capture and localization systems, which simplify fine-scale 3D localization systems to a great deal from embedding inertial sensors in the indoor environment.

In addition, multipath has been simulated and added. Simulated magnitudes and phases are very similar to measurements. Three kinds of position sensors are built in the simulation model in both LOS and multipath scenarios in static and motion with tracking. It is shown that RSS based position sensor is more sensitive to multipath than ToF. In addition, ToF in tracking is able to achieve comparable localization accuracy compared to static, making ToF based position sensor an alternative of RSS in real-time RFID based tracking and motion capture. Simulation results before and after estimation are very similar to measurements. Additionally, RMS error along z- axis is larger than that in xy- plane.

Both NLE and EKF have been applied to 3D state estimation in different combinations of RF signal and IMU data. EKF on Gaussian noise assumptions has better noise cancellation on RSS based position sensors. NLE, developed from the perspective of nonlinear least-squares optimization and the map-inversion philosophy on discrete-time dynamics, has better noise cancellation for non-Gaussian noise in ToF and multipath. NLE based estimation in 3D motion outperforms EKF by 2 cm.

13.1 Major Contributions

1. First-of-its-kind 3D state-space model for state estimation in 3D space for sensors of different sampling rates:

One major contribution of this dissertation was a first-of-its-kind 3D state-space model with state vector including position and orientation information. Nonlinear state estimation is extended to 3D space correspondingly with both asynchronous and synchronous models to handle different sensor sampling rates. NLE in this research has better noise cancellation and convergence for non-Gaussian noise in tracking and multipath as well as higher localization accuracy compared to existing EKF and UKF with different coefficient matrices that experience divergence in RF localization [80].

In this work, a 3D localization scheme that works with an accuracy of 31.9 cm RMS error was achieved indoor in a high multipath environment. Few techniques in the literature have even been tested in such a realistic multi path environment. The best results achieved prior to the study in this dissertation have been 21.5 - 76.7 cm [17] or 10 to 13 cm in the x- and y- dimensions, and 21 cm in the z- dimension [18]. The 3D state estimation in this work avoids the complex infrastructure of phased array as well as specific design and layout of FMCW transmit receive antennas.

2.Sensor fusion estimation in linear and nonlinear motions deriving RF signal only achieves fine-scale localization which simplifies RFID-based motion capture systems:

Another major contribution of this work was first sensor fusion estimation to explore different sensor combinations. Estimation results of sensor combinations are compared, unlike integration all signals from different sources together in the existing hybrid RF localization

systems [81, 82, 83, 84]. It is discovered from sensor combination comparison that RF signal only achieves similar accuracy to using both RF signal and IMU data in both linear and nonlinear motions. Hence, RF signal only is able to achieve fine-scale localization in LOS and multipath environment, which greatly simplified RFID-based motion capture systems from embedding inertial sensors as in [85, 86, 87].

Additionally, ToF based position sensor in tracking is of comparable accuracy to RSS based position sensor on the basis of results from the built multipath simulation model, enabling using ToF in tracking as a replacement of RSS for RFID real-time localization and motion capture systems.

CHAPTER 14

FUTURE WORK

The following tasks may be helpful for enhancing 3D indoor state estimation and localization for RFID-based motion capture systems in real-time tracking with multipath:

- More complex motion and longer motion time may be needed: Rotation motion around x- y- z- axes together with linear motion along x- y- z- axes may need to be combined together for more complex motion in real-time 3D motion capture and localization. In addition, motion of longer time may also be needed for verification of stability .

- Using ToF based position sensor in replacement of RSS: In replacement of RSS, ToF in tracking may be used as an alternative position sensor in 3D localization and tracking using samples from adjacent time stamps.

- Real-time multipath removal in the indoor environment in case of dominant reflections along the motion: More study of removing multipath and dominant reflections in door localization and tracking in real-time may need to be investigated and developed.

- Measurement taken behind walls in tracking: Unlike other schemes, EMS is able to transmit in blockage. Hence, measurements behind walls with obstacles for estimation and motion capture may also enhance the 3D indoor state estimation in tracking.

- Combining multiple position sensors in the estimation: Instead of using one position sensor in each estimation, multiple position sensors may be fused to input into the estimation model.

- Using RF signal only for 3D estimation: Since RF singal only is able to achieve fine-scale motion capture as combining RF with inertial sensors together. Future systems may focus on enhancement of RF signal either in processing or state estimation without embedding IMU.

- Enhancement on estimation accuracy with coefficient matrix: Investigating coefficient matrices in estimation for different sensor inputs and more than one sensor inputs may be helpful to enhance the estimation accuracy.

- Integrating static detection, tracking, and calibration together for real-time in door localization: Real-time estimation instead of offline processing combining all modules together to achieve real-time indoor localization may extend to many applications and achieve RFID-based motion capture systems.

- Selecting frequency sequence to enhance multipath mitigation: selecting frequency sequence in the hopping may also be another way for mitigating multipath effects.

- Obtaining orientation of the object from RF signal: Orientation of the object may also be obtained from RF signal without using IMU.

Accuracy enhancement and sensor source extension are helpful for the real-time RFID-based motion capture system.

Appendices

APPENDIX A

K FACTOR OF FOUR READERS IN MEASUREMENT

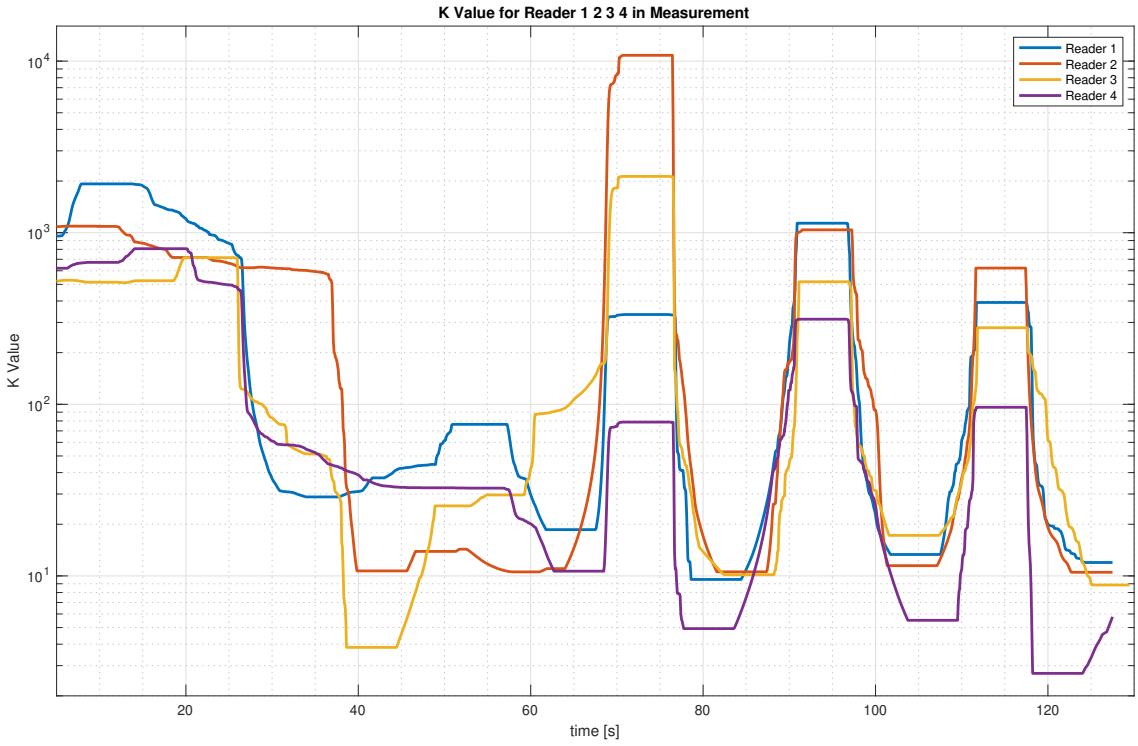


Figure A.1: K Values Calculated Based on $\hat{K}_{2,4}$ in Equation 2.29 From Square Root of Magnitude $H_t(r) = \sqrt{H(r)}$ of Four Readers.

APPENDIX B

DERIVATION OF PHASE IN MULTIPATH SCENARIO

$$
\begin{aligned}
&\frac{1}{r_0}\exp(-j\frac{2\pi r_0}{\lambda}) + \Gamma_1\frac{1}{r_1}\exp(-j\frac{2\pi r_1}{\lambda}) \\
&= \frac{1}{r_0}\cos(\frac{2\pi r_0}{\lambda}) - j\frac{1}{r_0}\sin(\frac{2\pi r_0}{\lambda}) + \frac{\Gamma_1}{r_1}\cos(\frac{2\pi r_1}{\lambda}) - j\frac{\Gamma_1}{r_1}\sin(\frac{2\pi r_1}{\lambda}) \\
&= \frac{1}{r_0}\cos(\frac{2\pi r_0}{\lambda}) - j\frac{1}{r_0}\sin(\frac{2\pi r_0}{\lambda}) + \frac{\Gamma_1}{r_1}\cos(\frac{2\pi(r_0+\Delta r)}{\lambda}) - j\frac{\Gamma_1}{r_1}\sin(\frac{2\pi(r_0+\Delta r)}{\lambda}) \\
&= \frac{1}{r_0}\cos(\frac{2\pi r_0}{\lambda}) - j\frac{1}{r_0}\sin(\frac{2\pi r_0}{\lambda}) + \frac{\Gamma_1}{r_1}\cos(\frac{2\pi r_0}{\lambda})\cos(\frac{2\pi\Delta r}{\lambda}) - \frac{\Gamma_1}{r_1}\sin(\frac{2\pi r_0}{\lambda})\sin(\frac{2\pi\Delta r}{\lambda}) \\
&- j\frac{\Gamma_1}{r_1}\sin(\frac{2\pi r_0}{\lambda})\cos(\frac{2\pi\Delta r}{\lambda}) - j\frac{\Gamma_1}{r_1}\cos(\frac{2\pi r_0}{\lambda})\sin(\frac{2\pi\Delta r}{\lambda}) \\
&= -\frac{\Gamma_1}{r_1}\sin(\frac{2\pi\Delta r}{\lambda})\sin(\frac{2\pi r_0}{\lambda}) + (\frac{1}{r_0} + \frac{\Gamma_1}{r_1}\cos(\frac{2\pi\Delta r}{\lambda}))\cos(\frac{2\pi r_0}{\lambda}) - j[(\frac{1}{r_0} + \frac{\Gamma_1}{r_1}\cos(\frac{2\pi\Delta r}{\lambda})) \\
&\sin(\frac{2\pi r_0}{\lambda}) + \frac{\Gamma_1}{r_1}\sin(\frac{2\pi\Delta r}{\lambda})\cos(\frac{2\pi r_0}{\lambda})] \\
&= \sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi) - j\sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi') \\
&= \sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi'-\frac{\pi}{2}) - j\sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi')
\end{aligned}
\tag{B.1}
$$

Set $A = -\frac{\Gamma_1}{r_1}\sin(\frac{2\pi\Delta r}{\lambda})$ and $B = \frac{1}{r_0} + \frac{\Gamma_1}{r_1}\cos(\frac{2\pi\Delta r}{\lambda}))$. $\phi = \arctan -\frac{B}{A}$ and $\phi' = \arctan\frac{A}{B}$. Then $\phi = -(\frac{\pi}{2} - \phi')$. Check: $\tan\phi = \tan(\phi' - \frac{\pi}{2}) = -\tan(\frac{\pi}{2} - \phi') = -\frac{1}{\tan\phi'} = -\frac{B}{A}$, which verifies the derivation.

$$
\begin{aligned}
&= \sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi'-\frac{\pi}{2}) - j\sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi') \\
&= \pm\sqrt{A^2+B^2}\cos(\frac{2\pi r_0}{\lambda}+\phi') - j\sqrt{A^2+B^2}\sin(\frac{2\pi r_0}{\lambda}+\phi') \\
&= \pm\sqrt{A^2+B^2}\exp(\mp j(\frac{2\pi r_0}{\lambda}+\phi')) = \pm\sqrt{A^2+B^2}\exp(\mp j(\frac{2\pi r_0}{\lambda} + \arctan\frac{\sin\frac{2\pi\Delta r}{\lambda}}{\Gamma_1\frac{r_1}{r_0} + \cos\frac{2\pi\Delta r}{\lambda}}))
\end{aligned}
\tag{B.2}
$$

Therefore, $\phi' = \arctan\frac{A}{B} = \arctan\frac{\frac{\Gamma_1}{r_1}\sin\frac{2\pi\Delta r}{\lambda}}{\frac{1}{r_0}+\frac{\Gamma_1}{r_1}\cos\frac{2\pi\Delta r}{\lambda}} = \arctan\frac{A}{B} = \arctan\frac{\sin\frac{2\pi\Delta r}{\lambda}}{\frac{r_1}{\Gamma_1 r_0}+\cos\frac{2\pi\Delta r}{\lambda}}$.

B.0.1 Removal of Multi-Path in Time-of-Flight Based Distance Estimator

Assume that $\phi = \frac{4\pi r_0}{\lambda} + 2\arctan(\frac{\sin\frac{2\pi\Delta r}{\lambda}}{\frac{r_1}{r_0}\Gamma_1+\cos\frac{2\pi\Delta r}{\lambda}})$. Assume that reflection coefficient Γ_1 is known, Multi-path length can be solved from the nonlinear equation. Multiple roots are solved from the equation out of the period of π of $\arctan$ angle. After removing the multi-path, radial distances based on time-of-flight are shown in Figure B.1.

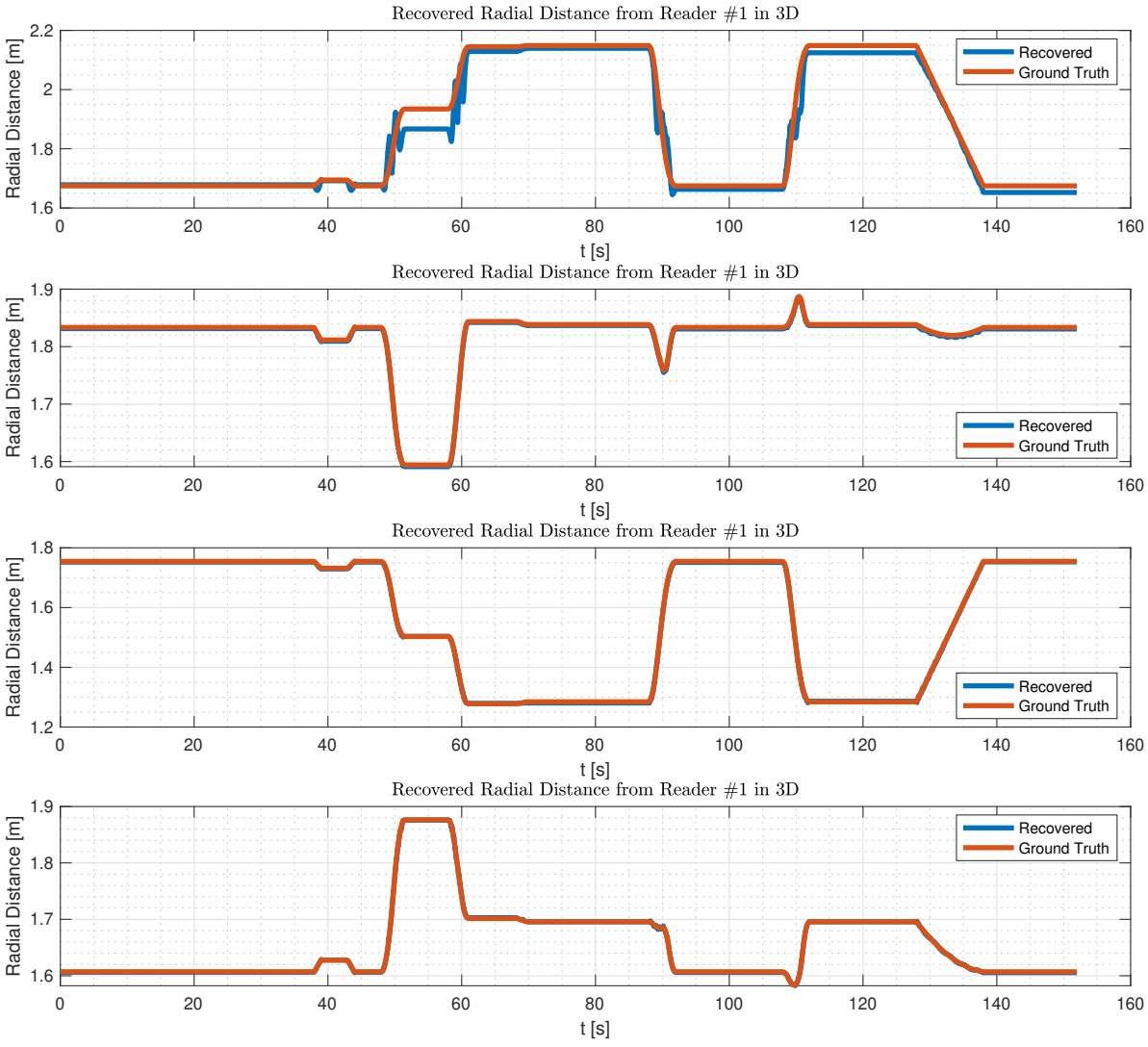


Figure B.1: Recovered Phase Derived Radial Distance from Two Frequencies in Multipath from Four Readers with the Input Trajectory Along x- Then y- and Finally z- Axis. After Moving Back, Moving Along x- y- z- Axes All Together as Described in chapter 6.

APPENDIX C

COMPARISON OF RADIAL DISTANCE AND RADIAL VELOCITY IN SIMULATION AND MEASUREMENT

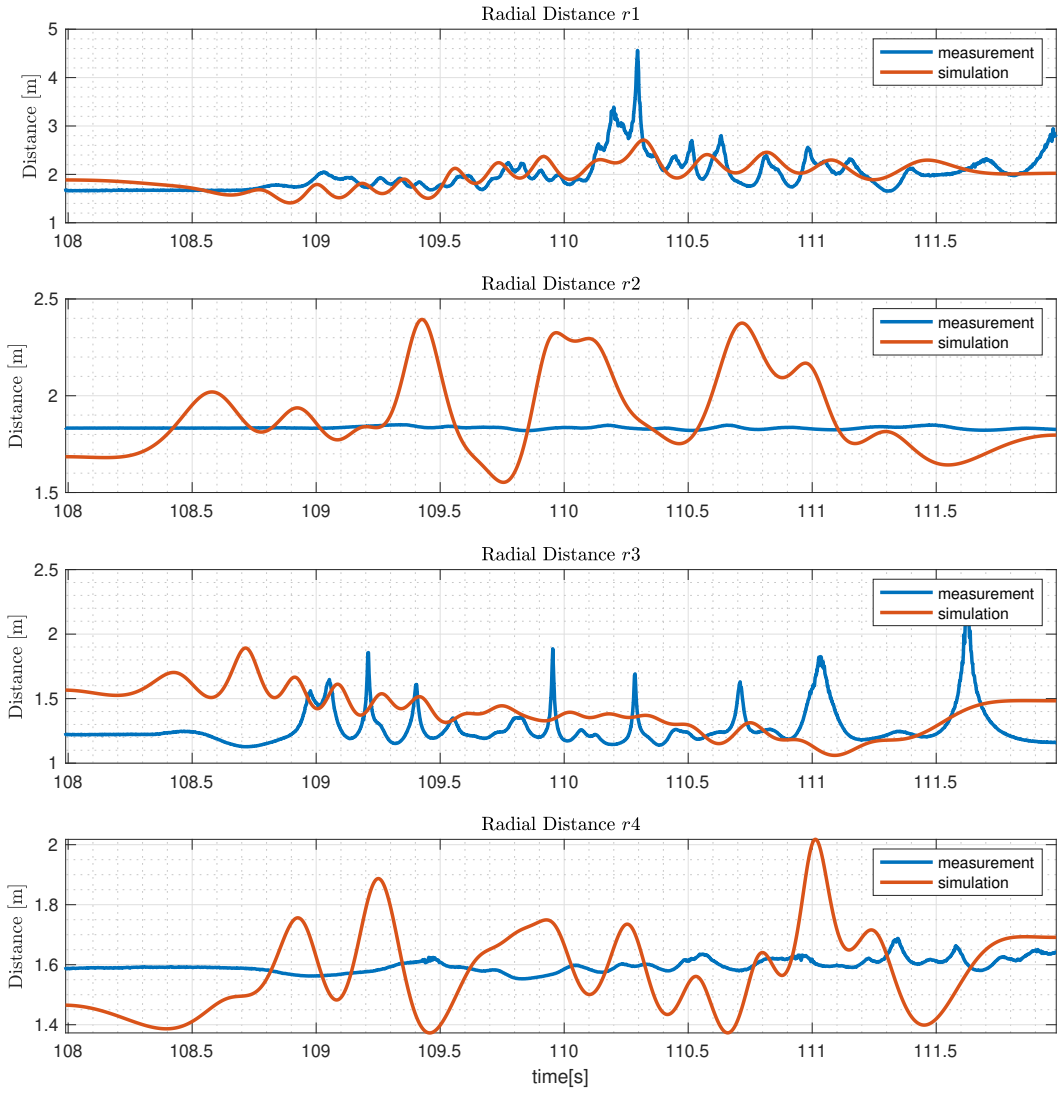


Figure C.1: Radial Distance r_1 r_2 r_3 r_4 Derived from Magnitude $H(r)$ of Four Readers. Blue Lines are Radial Distances from Measurement. Red Lines are Ground Truth Radial Distances Based on Motion Profile.

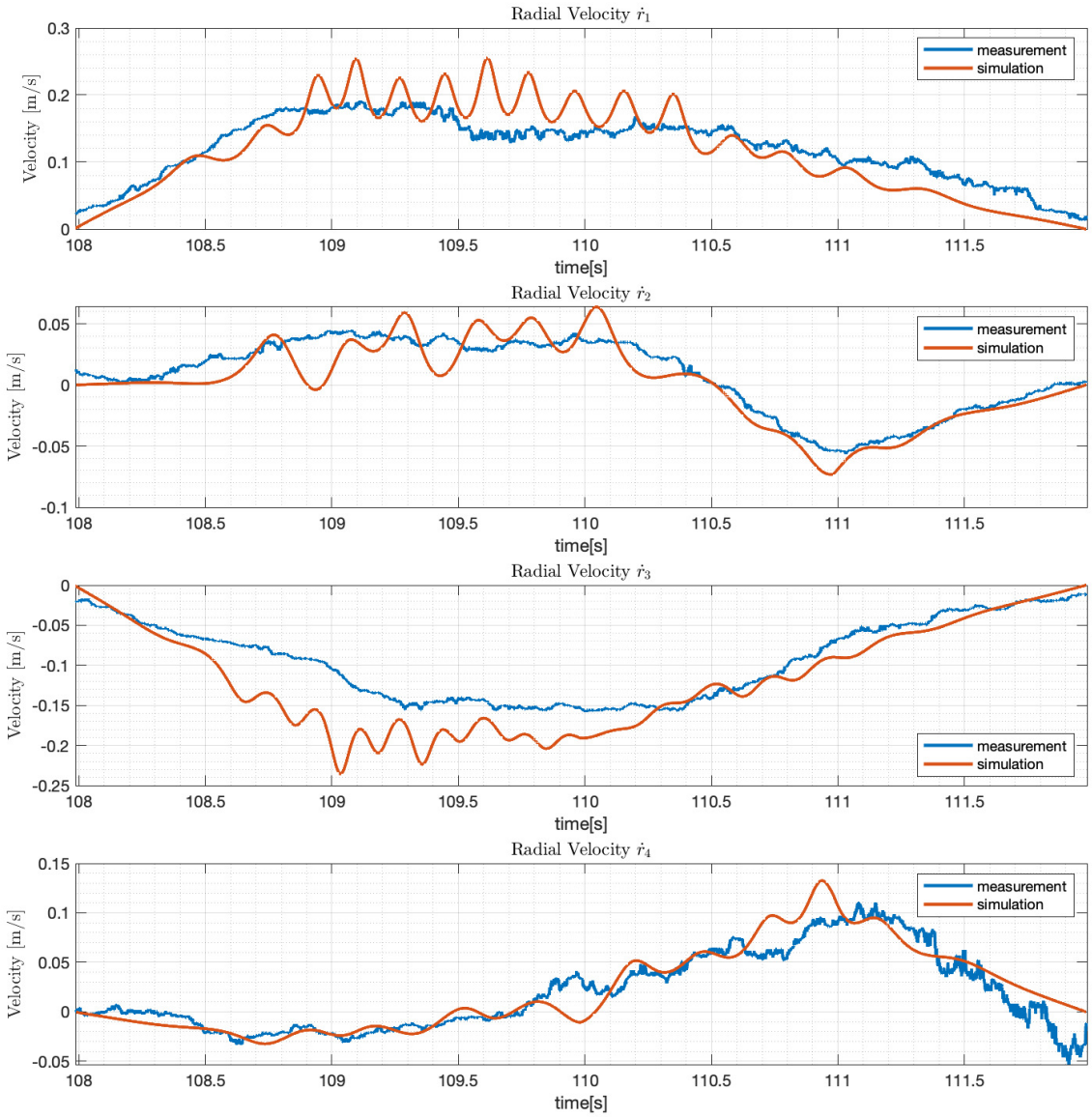


Figure C.2: Radial Velocities $\dot{r}_1$ $\dot{r}_2$ $\dot{r}_3$ $\dot{r}_4$ Derived from Phase Difference $\phi(r)$ with Applied Mean Filter of Length 4000 from Four Readers. Blue Lines are Radial Velocities from Measurement. Red Lines are Simulated Radial Velocities Based on 3D Motion Profile.

APPENDIX D

MAGNETOMETER MEASUREMENTS RESEMBLES TO RADIAL DISTANCES

When zoom into the magnetic strength, it can be inferred from Figure D.1 that magnetic strength along x- and z- axes are similar to radial distance r_1 and r_3 from reader 1 and 3. It may be from the magnetometer sensed from a certain point so correspondingly change along the motion, which also verifies the correctness of the simulated radial distances.

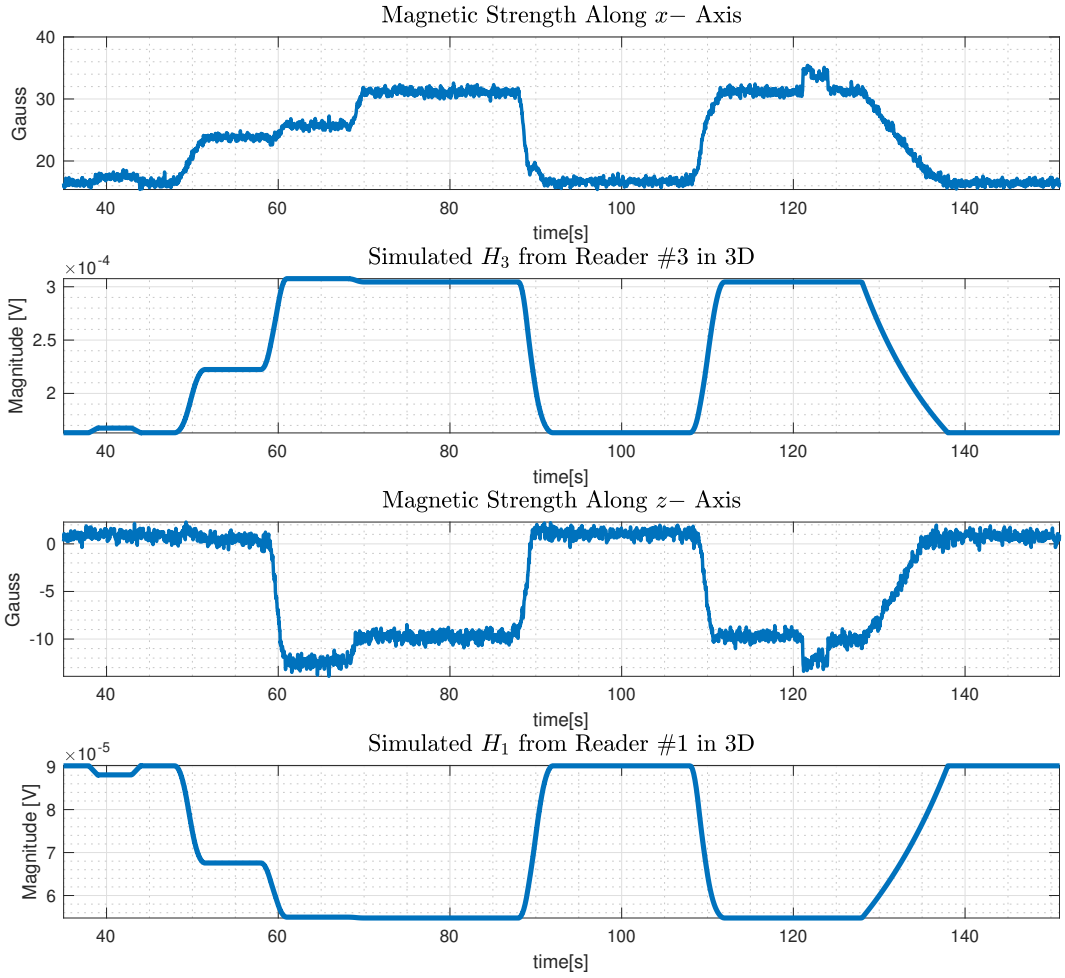


Figure D.1: Magnetic Strength Along x- and z- Axes Compared to Simulated RF Magnitude H_3 and H_1 From Reader #3 and Reader #1. The Resemblance May be From That Magnetometer Magnitude is Based on Certain Point in Environment So Correspondingly Changes Along the Motion.

APPENDIX E

PLATFORM-FREE CALIBRATION METHODS FOR INERTIAL SENSORS

The embedded IMU sensor of the cellphone is low-cost and of imperfections in manufacturing and installation as well as high noise levels. Zero-velocity update (ZUPT) for inertial sensor calibration to compensate drift in long time measurement has been developed in this chapter to enhance the localization accuracy. In this section, we proposed an on-line calibration method enables platform free calibration for real-time localization.

E.1 Allan Variance

Allan variance [88, 89] characterizes gyroscope bias drifts, which measures the variance of difference between consecutive interval averages. Allan variance is defined as

$$\sigma_a^2 = \frac{1}{2} < (x(t,k) - x(t,k-1))^2 >= \frac{1}{2K} \sum_{k=1}^{K} (x(t,k) - x(t,k-1))^2 \tag{E.1}$$

where $x(t,k)$ is the $k\text{-}th$ interval average which spans t seconds, and K is the total number of intervals. The time in which the Allan variances of the three axis converge to a small values represents the initialization period T_{init}.

At the beginning of measurements, calculate the time taken for Allan variance to reach small. It is the initial time. $T_{init} \approx 66$ s for the IMU embedded in Galaxy S8.

Zero velocity detection needs to start after T_{init} since the measured states before convergence may introduce false zero-velocity detections. Figure E.1 shows the Allan variance with respect to time intervals from 10 s to 300 s in human walking.

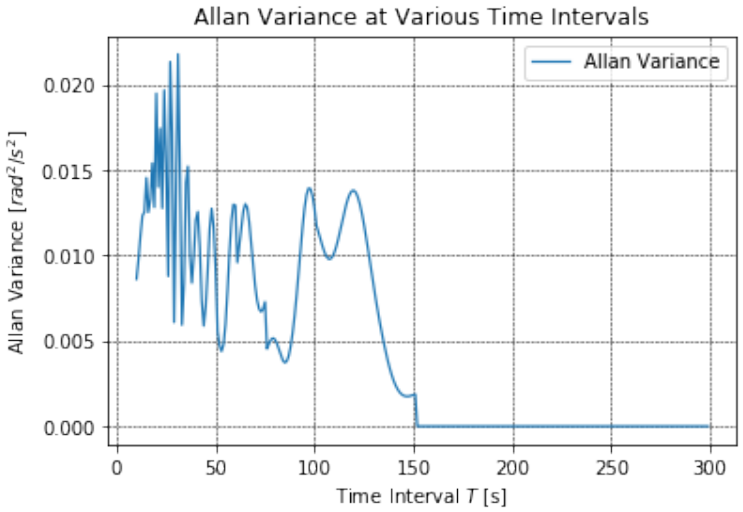


Figure E.1: Allan Variance σ_a^2 w.r.t. Different Time Intervals T [s] From 10 s to 300 s for Galaxy S8. Based on the Trend, T_{init} is Set at 66 s.

E.2 Zero Velocity Detection

There exists two kinds of static detection methods: dynamic and with set thresholds. If multiple sensors of similar parameters are used, set thresholds as in [90, 91, 92] are applicable. If factors among different motions are stable, dynamic static detection as described below is applicable [68].

A dynamic way is applied to detect zero velocity in order to get stationary states for calibration.

Firstly, standard deviations of acceleration and gyroscope measurements in a set window size of min are calculated. If both values are smaller than the set threshold times, global minimal values of acceleration and angular velocity, standard deviations of acceleration and gyroscope measurements in a larger window size of max are calculated. If both values are still smaller than the set thresholds, the detected periods are identified as stationary states.

In addition, global minimal values of standard deviations for accelerometers and gyroscopes are updated if the calculated values in the larger window size are smaller than the original global minimal values [68].

Two different sizes of windows are set. Window size of min is applied to enhance computational efficiency. Smaller window size enables faster standard deviation calculation as the initial selection. On the other hand, window size of max is applied to reduce false positive detection. More specifically, the larger window size of max is at least $1s$ to avoid detecting transient static outliers.

The threshold coefficients for acceleromter and gyroscope minimal values c_1 c_2 and window size min max can be learned from pre-collected labelled measurement data using SVM and/or LSTM.

In the Pseudocode, c_1 is set as 2 and c_2 is set as 3. In order to classify quasi-static and semi-static also as static states, higher threshold about 15 - 35 should be used. The pseudocode for stationary state detection algorithm is shown as following

- Global minimal value of standard deviations for acceleration $\sigma_{a_{min}}$ is initialized around g and global minimal value of standard deviations for gyroscope $\sigma_{g_{min}}$ is initialized around 0.

- Sizes for smaller and larger windows, i.e., w_{min} and w_{max} are set.

- At time stamp t, standard deviations of accelerations σ_{cw_a} and gyroscopes σ_{cw_g} in window size of w_{min} are calculated.

- If the calculated standard deviation σ_{cw_a} is less than set threshold $c_1\sigma_{a_{min}} = 2\sigma_{a_{min}}$ and the calculated standard deviation of gyroscope σ_{cw_g} is less than the set threshold $c_2\sigma_{g_{min}} = 3\sigma_{g_{min}}$, then both values are re-calculated in the larger window size of w_{max}.

- If both re-calculated standard deviations are also within set thresholds, states in the larger window size are identified as static.

- If calculated standard deviations σ_{cw_a} and σ_{cw_g} are smaller than global minimal values $\sigma_{a_{min}}$ and $\sigma_{g_{min}}$, then global minimal standards deviations of acceleration or gy-

roscope are updated.

```
1: procedure STATIONARY STATE DETECTION PSEUDOCODE
2: σ_{a_min} ← g ± v
3: σ_{g_min} ← 0 ± v'
4: w_min ← min_size
5: w_max ← max_size
6: repeat
7:     cw_a, cw_g ← gather w_min number of IMU measurements
8:     σ_{cw_a} ← calc_std(L2(cw_a))
9:     σ_{cw_g} ← calc_std(L2(cw_g))
10:    if σ_{cw_a} < c_1 σ_{a_min} and σ_{cw_gmin} < c_2 σ_g then
11:        cw_a, cw_g ← gather w_max number of IMU measurements
12:        σ_{cw_a} ← calc_std(L2(cw_a))
13:        σ_{cw_g} ← calc_std(L2(cw_g))
14:        if σ_{cw_a} < c_1 σ_{amin} and σ_{cw_gmin} < c_2 σ_{gmin} then
15:            if σ_{cw_a} < σ_{a_min} then
16:                σ_{a_min} ← σ_{cw_a}
17:            if σ_{cw_g} < σ_{g_min} then
18:                σ_{g_min} ← σ_{cw_g}
               states ← static states
19:
20:        close
21:
       close
22:
=0
```

E.2.1 Measurement Results for Zero Velocity Detection

As presented in section E.1, Allan variance is used to derive T_{init}. Then static states are automatically detected using the method of Stationary State Detection Algorithm in section E.2. Three categories of measurements are taken for verification:

1. Measurement 1

Rotations with small angles along different axes are presented in Figure E.2, in which red rectangles are the intervals in measurements detected as static intervals. As shown in the two figures, the resolution of this automatic detection method is within seconds-level. In addition, the automatic detection algorithm is able to detect sharp changes in the continu-

ous motion.

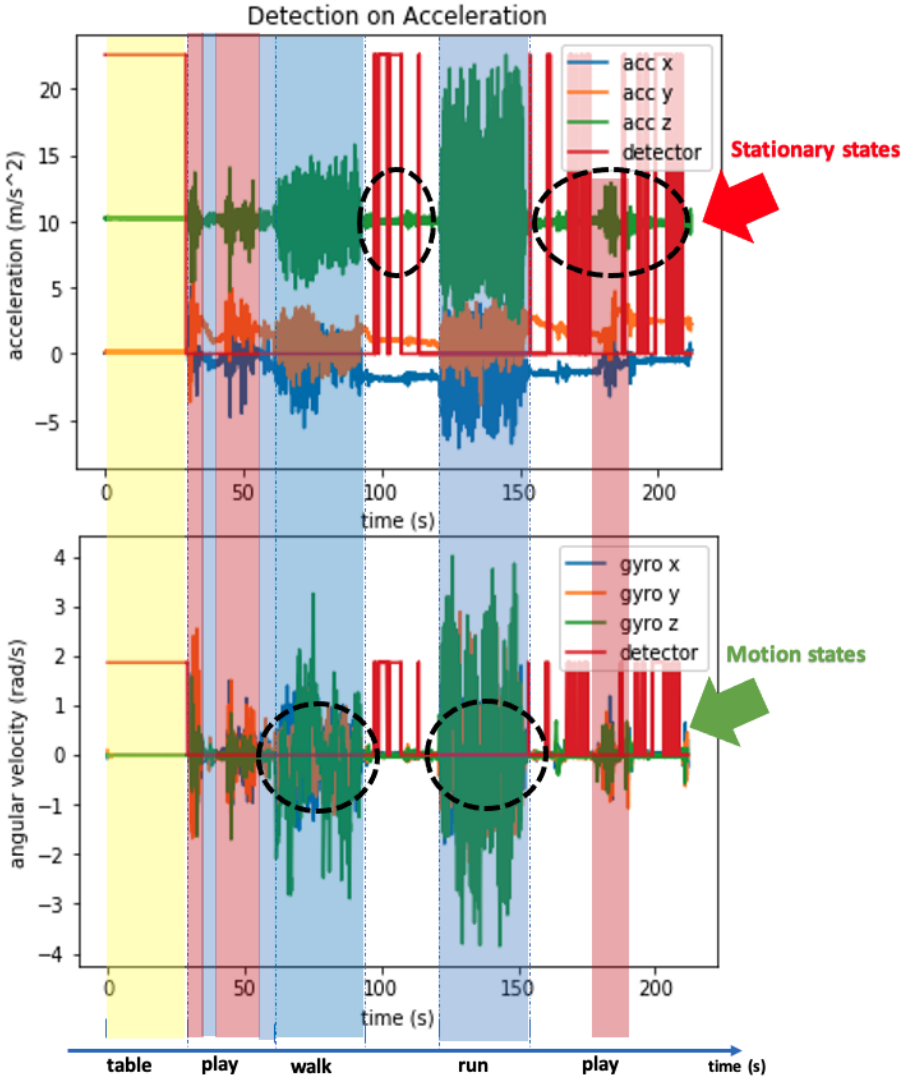


Figure E.2: Detection Result on Accelerations and Gyroscopes. Rotations with Small Angles Along Different Axes Using Dynamic Zero Velocity Detection. There are 51 detected static intervals. Red Rectangles: Intervals Detected as Static Intervals. Blue, Orange, Green Curves: Raw Angular Velocity Measurements from Gyroscope.

2. Measurement 2

Stationary with standard deviations of acceleration and angular velocity close to 0 when putting onto the table and routine activities like using apps, searching online etc. are shown in Figure E.3. As shown in both figures, standard deviation in certain interval close enough to 0 doesn't turn off static detections afterwards.

3. Measurement 3

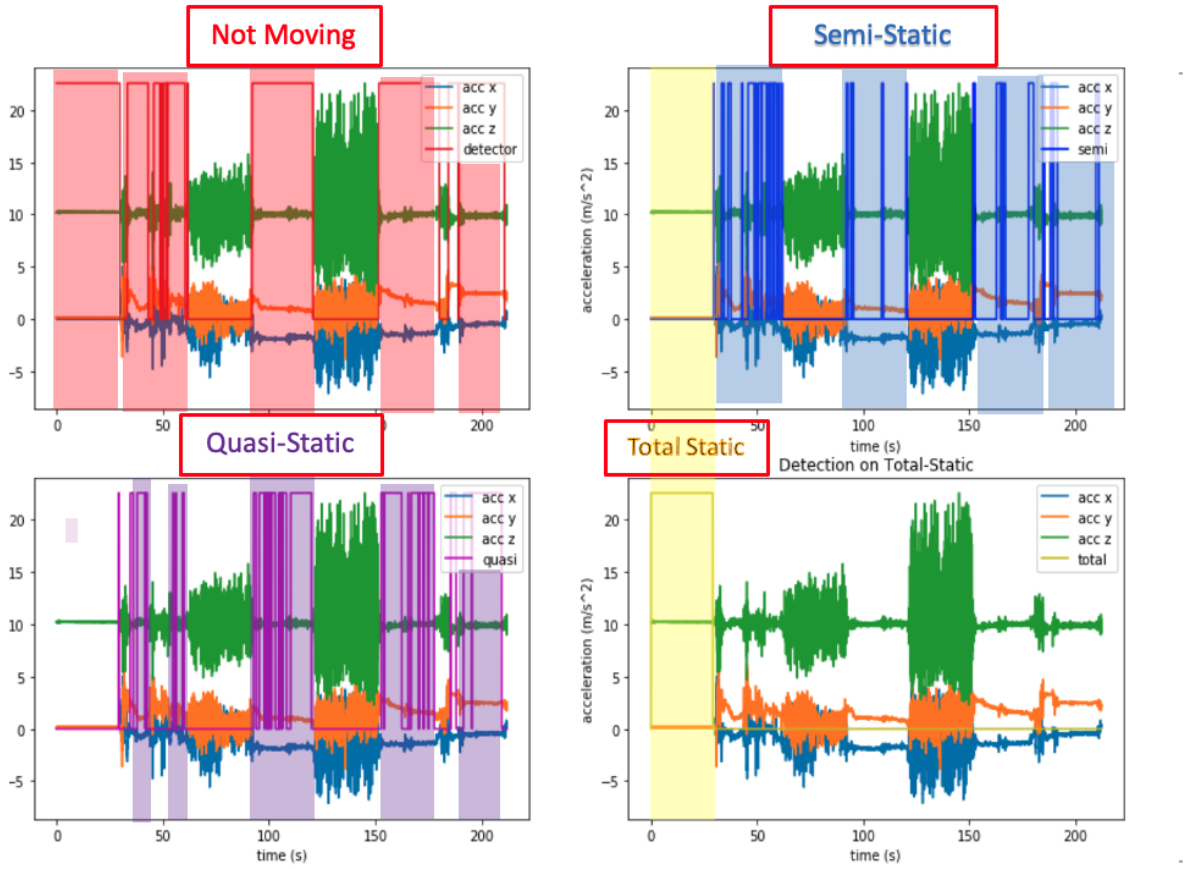


Figure E.3: Detection Result on Accelerations. Stationary When Putting onto the Table and Routine Activities Like Using Apps, Searching Online etc. Using Dynamic Zero Velocity Detection. There are 6 Detected Static Intervals. Red Rectangles: Intervals Detected as Static Intervals. Blue, Orange, Green Curves: Raw Acceleration Measurements From Accelerometer.

Measurement 3 in Figure E.4 is taken in human walking and standing activities. It can be inferred from both figures that detected static intervals of human activities are not as continuous as putting static onto the table. In addition, with accelerations not converge to 0 (the acceleration along y axis is positive and acceleration along x axis is negative) when standing still, static states are still be able to be detected.

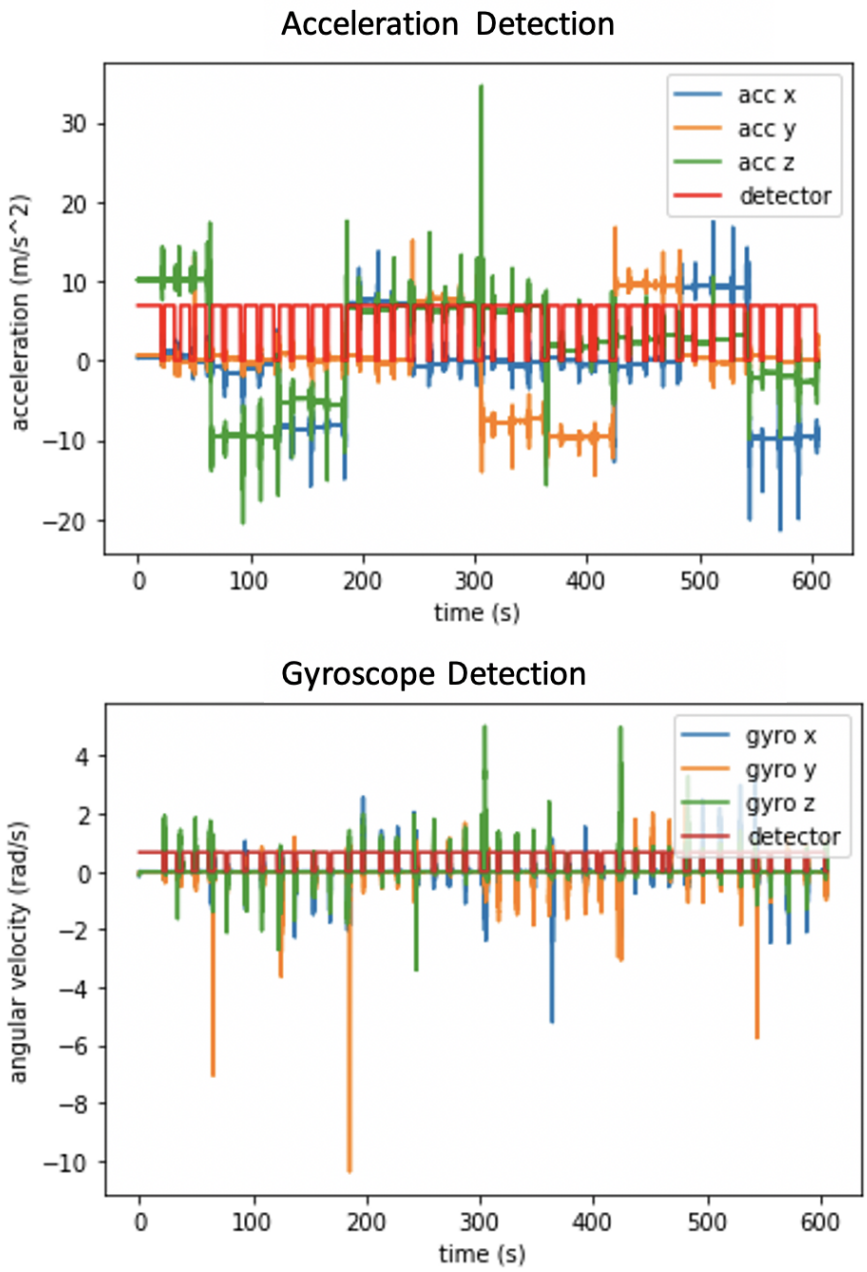


Figure E.4: Detection Result on Gyroscopes. Stationary When Putting onto the Table and Routine Activities Like Using Apps, Searching Online etc. Using Dynamic Zero Velocity Detection. There are 6 Detected Static Intervals. Red Rectangles: Intervals Detected as Static Intervals. Blue, Orange, Green Curves: Raw Angular Velocity Measurements From Gyroscope.

E.3 Platform-Free Calibration Methods for Inertial Sensors

Based on whether or not there's a requirement in using any additional equipment, calibration methods for inertial sensors can be divided into two groups [68]: Traditionally, calibration is performed by using high-end measurement equipment to get parameters hard-coded into sensor. This method requires an attitude controlled turntable with each axis perfectly aligned with the instrument. A major drawback is the cost out of using the specialized and expensive equipment. In addition, it's also labor intensive and can only be performed in a controlled environment [93].

In applications of mobile phone, activity trackers, and wearables etc., in-field calibrations without using additional equipment are more and more widely used. The principal idea is to use known reference values in certain positions. When an object is stationary, the only force applied to it is gravity. By using the value of gravity acceleration, g, optimization methods enable reliable calibration of coefficient matrices. The first calibration required placing the IMU in six-different positionss [94]. Methods in [95, 96] focus on calibrating accelerometer only. Calibrations methods that calibrate both accelerometer and gyroscope are documented in [97, 98]. In [97, 99], calibrated accelerations are used to calibrate gyroscope.

To improve current one-shot procedures, an automatic re-calibration method combining static detection and calibration for accelerometer and gyroscope together are designed.

E.4 Platform-Free Calibration for Accelerometer

Accelerometer is calibrated based on the principal idea that the only force applied to is gravity acceleration, g, when the object is stationary.

E.4.1 Removal of Random Gaussian Noise

Kalman filter is applied to remove stochastic noise v^a. After filtering, the error model of acceleration becomes

$$a^b = T^a K^a (a^a + b^a) \tag{E.2}$$

where T^a, K^a, and b^a are rotation matrix, scalar matrix, and bias vector to be calibrated.

E.4.2 Estimate Acceleration Parameter Vector

In the multi-position scheme, a set of N distinct, temporarily stable states at different orientations are detected based on detection method in section E.2. Afterwards, N raw acceleration measurements a^b are extracted from the static states. After calibration, the norm of the acceleration vector should be equal to the ground truth gravity at this location.

Therefore, parameter vector θ^{acc} can be obtained by minimizing the difference between the two norms based on Levenberg-Marquardt (LM) method as

$$\theta^{acc} = arg \min_{\theta^{acc}} \sum_{k=1}^{N} (||g||^2 - ||a^b||^2)^2 \tag{E.3}$$

where a^b is the raw acceleration measurement vector as shown in Equation E.2 that contains the parameter vector θ^{acc}.

$||g||$ is the actual magnitude of the local gravity vector that can be inferred from the three methods described as following.

Method 1 [68]

International Gravity Formula (IGF) with Free Air Correction (FAC) are used to calculate

ground truth vaue of gravity [68]

$$g = IGF + FAC \tag{E.4}$$

$$IGF = 9.780327(1 + e_1 \sin\Phi^2 - e_2 \sin 2\Phi^2) \tag{E.5}$$

$$FAC = -3.08610^{-6} \times h \tag{E.6}$$

where Φ is the latitude in radians, $e_1 = 0.0053024$, $e_2 = 0.0000058$, h is the height relative to sea level.

Method 2

Ground truth value of gravity g can also be calculated as [100]

$$g = 9.7803253359\frac{1 + 0.001931853 \sin\Phi^2}{\sqrt{1 - e^2 \sin\Phi^2}} \tag{E.7}$$

$$g = g(\frac{R_{earth}}{R_{earth} + h})^2 \tag{E.8}$$

where Φ is the latitude in radians, h is the height relative to sea level, $e^2 = 0.00669437999014$ is the square of the first eccentricity, $R_{earth} = 6371000$ m is the radius of the earth.

Method 3

The value of gravity can also be inferred from websites, which includes getting latitude, longitude, and height above the sea level as well as getting gravity value from obtained latitude, longitude, and height [101, 102].

E.5 Platform-Free Calibration for Gyroscope

Gyroscope is calibrated using the calibrated acceleration from section E.4 unit vector which contains the orientation information and is related to angular velocity measured from gyroscope.

E.5.1 Removal of Random Gaussian Noise

Kalman filter is applied to remove stochastic noise v^g. After filtering, we have the error model of gyroscope

$$w^b = T^g K^g (w^g + b^g) \tag{E.9}$$

where T^g, K^g, and b^g are rotation matrix, scalar matrix, and bias vector to be calibrated.

E.5.2 Estimate Gyroscope Bias

At the beginning of measurements, when becomes small enough, the biases of gyroscopes can be correctly determined by averaging the values along each axis. The time taken for Allan variance to reach small is around 66 s. Mean values of gyroscopes in static states approximate bias vector b^g as

$$b^g = \begin{bmatrix} b_x^g \\ b_y^g \\ b_z^g \end{bmatrix} = \begin{bmatrix} avg(w_x^b) \\ avg(w_y^b) \\ avg(w_z^b) \end{bmatrix} \tag{E.10}$$

Then the error model for gyroscope becomes

$$w^b = T^g K^g w^g \tag{E.11}$$

where T^g, K^g, and b^g are rotation matrix, scalar matrix, and bias vector to be calibrated. Unknown parameters in T^g, K^g, and b^g compose the gyroscope parameter vector

$$\theta^{gyro} = \begin{bmatrix} \gamma_{yz} & \gamma_{zy} & \gamma_{xz} & \gamma_{zx} & \gamma_{xy} & \gamma_{yx} & s_x^g & s_y^g & s_z^g \end{bmatrix}' \tag{E.12}$$

E.5.3 Estimate Gyroscope Parameter Vector

In order to relate gyroscope measurements to the unit vector from calibrated acceleration vector, integration on angular velocity of gyroscope to derive orientation angle change is applied.

Integrate gyroscope measurements between static states t_{k-1} and t_k with corresponding measurement time-stamp to get orientation angle change between the two static states t_{k-1} and t_k.

$$\theta_{xk} = \sum_{t_d \in (k-1,k)} w_x^b t_d \tag{E.13}$$

$$\theta_{yk} = \sum_{t_d \in (k-1,k)} w_y^b t_d \tag{E.14}$$

$$\theta_{zk} = \sum_{t_d \in (k-1,k)} w_z^b t_d \tag{E.15}$$

Then we can transform the unit vector representing gravity from $u_{g,k-1}$ to $u_{g,k}$ using integrated orientation angle change α_k β_k, and ψ_k

$$u_{g,k} = \begin{bmatrix} \cos\theta_{zk} & -\sin\theta_{zk} & 0 \\ \sin\theta_{zk} & \cos\theta_{zk} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_{yk} & 0 & \sin\theta_{yk} \\ 0 & 1 & 0 \\ -\sin\theta_{yk} & 0 & \cos\theta_{yk} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{xk} & -\sin\theta_{xk} \\ 0 & \sin\theta_{xk} & \cos\theta_{xk} \end{bmatrix} u_{g,k-1} \tag{E.16}$$

where θ_{xk}, θ_{yk}, and θ_{zk} are roll, pitch, and yaw rotation angles along x-axis, y-axis, and z-axis, respectively.

Afterwards, $u_{g,k}$ at time stamp t_k is calculated based on Equation E.16. In addition, unit vector $u_{a,k}$ can also be derived from measured acceleration vector a^a that is calibrated based on Equation E.2 with known θ^{acc}. The calibrated vector then normalized as unit vector $u_{a,k}$. By minimizing the difference between $u_{a,k}$ and $u_{g,k}$ from $N-1$ intervals between static

periods, θ^{gyro} can be estimated using Levenberg-Marquardt(LM) method as

$$\theta^{gyro} = arg \min_{\theta^{gyro}} \sum_{k=1}^{N-1} ||u_{a,k} - u_{g,k}||^2 \tag{E.17}$$

In summary, the gyroscope calibration steps are as following:

- Integrate angular velocity from gyroscope measurements in intervals between static states as shown in Equation E.13 - Equation E.15 to get the changed orientation angles θ_{xk}, θ_{yk}, and θ_{zk}.
- Transform unit vector $u_{g,k-1}$ to $u_{g,k}$ representing gravity from t_{k-1} to t_k based on integrated angles θ_{xk}, θ_{yk}, and θ_{zk} from Equation E.16.
- At the same time, with known acceleration parameter vector θ^{acc}, derive the unit vector $u_{a,k}$ at t_k from measurement acceleration a^a by using Equation E.11.
- Then θ^{gyro} can be estimated by minimizing the difference between $u_{a,k}$ and $u_{g,k}$ on formula Equation E.17 using LM optimization.
- Then we have the gyroscope parameter vector θ^{gyro}.

E.6 Platform-Free Calibration Results for Accelerometer

E.6.1 Ground Truth Value of Gravity Based on Different Methods

Based on measurement location of Atlanta GA, the latitude is 33.7490° N, 84.3880° W. The Barometer reading is 985.100100 m above the sea level.

Results of ground truth values of gravity calculated from Method 1, Method 2, and Method 3 in section E.4 are compared and shown in Table E.1. It can be inferred from Table E.1 that the three methods give very close results, which is up to 4-digit accuracy. Ground truth gravity can reach difference < 0.0001 m/s^2.

Table E.1: Derived Ground Truth Value of Gravity from Method 1, 2, 3

	Method 1	Method 2	Method 3
Gravity Value [m/s^2]	9.79324452	9.793253733	9.7925 – 9.7951

E.6.2 Calibration Results

24 positions scheme for calibration: rotation counterclockwise for 90 degrees around x- y- and z- axes which accumulates to 12 positions and flip upside down to get total 24 positions as shown in [103].

From two kinds detected stationary intervals in subsection E.2.1, corresponding acceleration vector is extracted from each interval by averaging or Kalman filtering. Afterwards, calibrated coefficient matrix in Equation 9.1 are derived based on the accelerometer calibration method in section E.4.

1. Same Sign Among Different DataSets and Detection Methods

In all 7 sets of measurements using dynamic static detection or set thresholds static detection, all 14 derived calibrated coefficient matrices have the same sign for each parameter as shown in Table E.2.

Table E.2: Sign of Parameters in Derived Calibrated Coefficient Matrices Among All Measurements

Skew Factor		Scalar Factor		Bias	
$-\alpha_{yz}$	$+$	s_x^a	$+$	b_x^a	$+$
α_{zy}	$+$	s_y^a	$+$	b_y^a	$+$
$-\alpha_{zx}$	$-$	s_z^a	$+$	b_z^a	$-$

2. Small Differences in Calibrated Coefficient Matrices with Different Detection Methods

In set thresholds and dynamic static detection methods, the differences between derived calibrated coefficient matrices is < 0.03 for each parameter.

3. Range of Parameters in Calibrated Coefficient Matrices in Different Measurements

The calibrated coefficient matrix result are shown in Table E.3. The range of each parameter is shown in Table E.4.

Table E.3: Derived Scalar Factor Bias and Skew Factor in Measurement

Skew Factor		Scalar Factor		Bias	
$-\alpha_{yz}$	0.0623	s_x^a	0.9944	b_x^a	0.1739
α_{zy}	0.0055	s_y^a	0.9999	b_y^a	0.0071
$-\alpha_{zx}$	-0.0041	s_z^a	0.9880	b_z^a	-0.2999

Table E.4: Derived Scalar Factor Bias and Skew Factor Difference Ranges in All Measurements

Skew	Factor	Scalar	Factor	Bias	
$-\alpha_{yz}$	$0.0579 - 0.0779$	s_x^a	$0.9914 - 0.9953$	b_x^a	$0.1719 - 0.1887$
α_{zy}	$-0.0013 - 0.0075$	s_y^a	$0.9978 - 0.9999$	b_y^a	$0.0058 - 0.0394$
$-\alpha_{zx}$	$-0.0125 - -0.0041$	s_z^a	$0.9867 - 0.9892$	b_z^a	$-0.3036 - -0.2761$

It can be inferred from Table E.3 and Table E.4 that automatic static interval detections and calibration methods derive similar calibration results from different measurements and motions.

E.6.3 Calibration Verification

Table E.5 show mean and standard deviation of error deviated from the ground truth value of gravity on measurement 2 - 7 with parameters calibrated from measurement 1. Table E.6 and Table E.7 are verification on different measurements as well.

It can be inferred from Table E.5, Table E.6, Table E.7, Table E.8, Table E.9, and Table E.7 that calibrated parameters achieve acceptable calibration accuracy when applied

to different motions and measurements. The calibration method for accelerometer can be applied to real-time calibration in measurements.

Table E.5: Calibration Verification for Error Mean and Standard Deviation Among Measurement 2 - 6 Using Coefficient Matrix Derived from Measurement 1

	Error (m/s^2): Measurement 1 $\rightarrow$ Measurement 2-6				
Mean	-0.000095807	0.0051	-0.00092365	-0.0022	0.0017
Standard Deviation	0.0395	0.0447	0.0437	0.0434	0.0416

Table E.6: Calibration Verification for Error Mean and Standard Deviation Among Measurement 1, 3 - 6 Using Coefficient Matrix Derived from Measurement 2

	Error (m/s^2): Measurement 2 $\rightarrow$ Measurement 1, 3-6				
Mean	-0.0035	0.0029	-0.0012	-0.0060	-0.0020
Standard Deviation	0.0477	0.0470	0.0523	0.0505	0.0480

Table E.7: Calibration Verification for Error Mean and Standard Deviation Among Measurement 1,2, 4 - 6 Using Coefficient Matrix Derived from Measurement 3

	Error (m/s^2): Measurement 3 $\rightarrow$ Measurement 1,2, 4 - 6				
Mean	-0.0075	-0.0093	-0.0084	-0.0141	-0.0110
Standard Deviation	0.0448	0.0409	0.0487	0.0451	0.0432

Table E.8: Calibration Verification for Error Mean and Standard Deviation Among Measurement 1 - 3, 5, 6 Using Coefficient Matrix Derived from Measurement 4

	Error (m/s^2): Measurement 4 $\rightarrow$ Measurement 1 - 3, 5, 6				
Mean	-0.0035	-0.0010	0.0042	-0.0038	-0.00045571
Standard Deviation	0.0393	0.0381	0.0439	0.0393	0.0373

E.6.4 Parameter Range with Different Number of Positions

In addition, the number of positions and distributions of positions placed in the calibration affect the range of parameters. It can be inferred from Figure E.5 that using 24, 48, or mod-

Table E.9: Calibration Verification for Error Mean and Standard Deviation Among Measurement 1 - 4, 6 Using Coefficient Matrix Derived from Measurement 5

	Error (m/s^2): Measurement 5 $\rightarrow$ Measurement 1 - 4, 6				
Mean	0.0013	0.00090573	0.0042	-0.0012	-0.0018
Standard Deviation	0.0422	0.0380	0.0420	0.0432	0.0376

Table E.10: Calibration Verification for Error Mean and Standard Deviation Among Measurement 1 - 5 Using Coefficient Matrix Derived from Measurement 6

	Error (m/s^2): Measurement 6 $\rightarrow$ Measurement 1 - 5				
Mean	-0.0011	-0.00089762	0.0034	-0.0022	-0.0060
Standard Deviation	0.0447	0.0403	0.0438	0.0455	0.0425

ified 40 positions all within same mean and region. Parameters calibrated by 48 positions have smaller range compared to 24 positions.

Moreover, modified 40 positions are taken at evenly distributed angles along x- y- and z- axes. Due to the distributed attitudes, the range of parameters are much smaller than 48 positions with fewer number of positions. Hence, the distributions of attitudes also affects the variance of calibrated parameters.

E.6.5 Comparison with Platform-Based Calibration

The platform-free calibration gets rid of the high precision rotation table or reference instrument, maintaining high-precision calibration accuracy. In this way, platform-free and automatic calibration is feasible for cellphone-based realtime 3D localization.

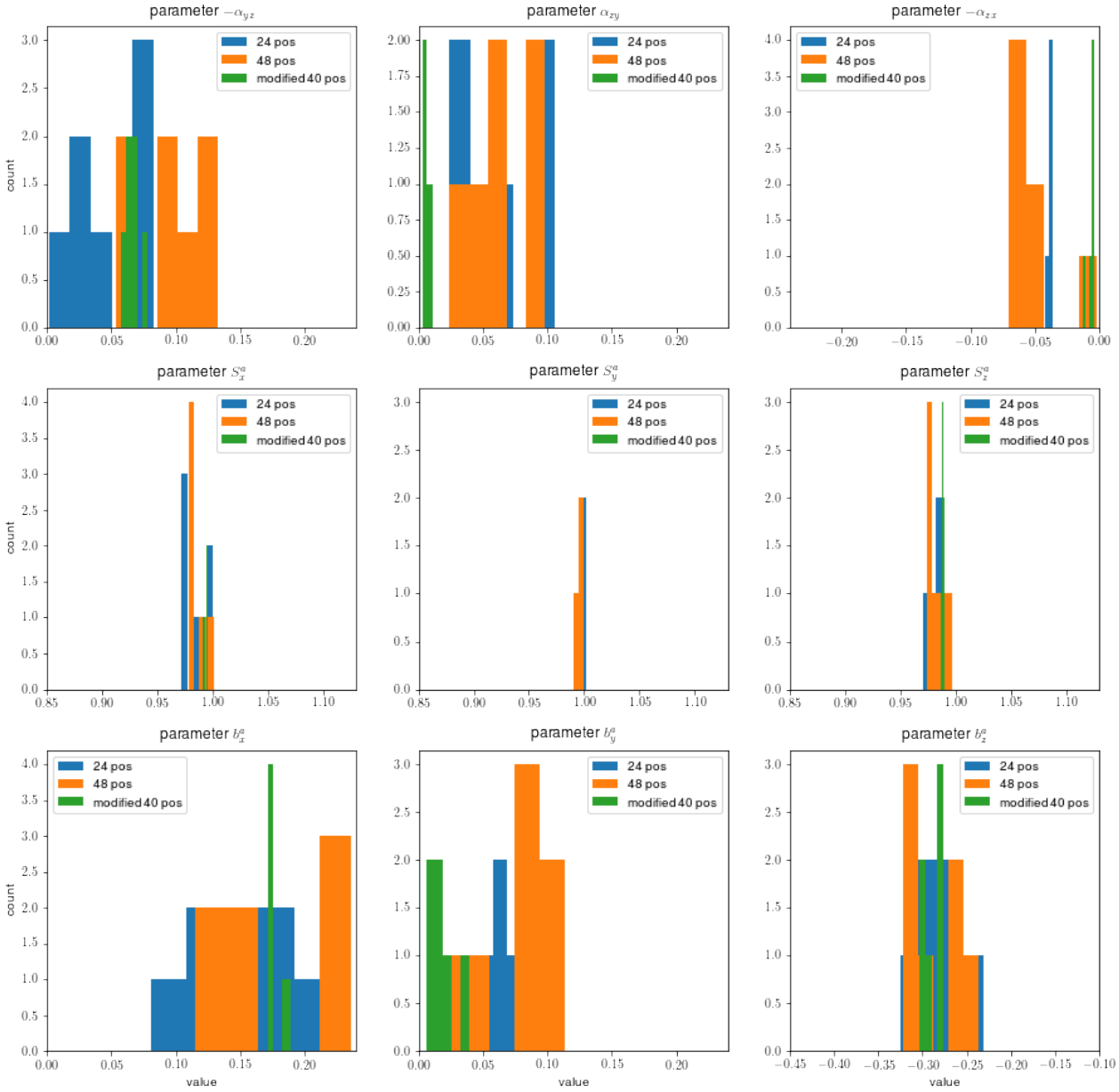


Figure E.5: Histogram of Distributions for Acceleration Calibration Parameters α_{yz} $-\alpha_{yz}$, α_{xy}, $-\alpha_{zx}$, S_x^a, S_y^a, S_z^a, b_x^a, b_y^a, and b_z^a with 24 Positions 48 Positions and Modified 40 Positions.

APPENDIX F

RF ONLY BASED LOCALIZATION MOVING ALONG $X-$ AXIS

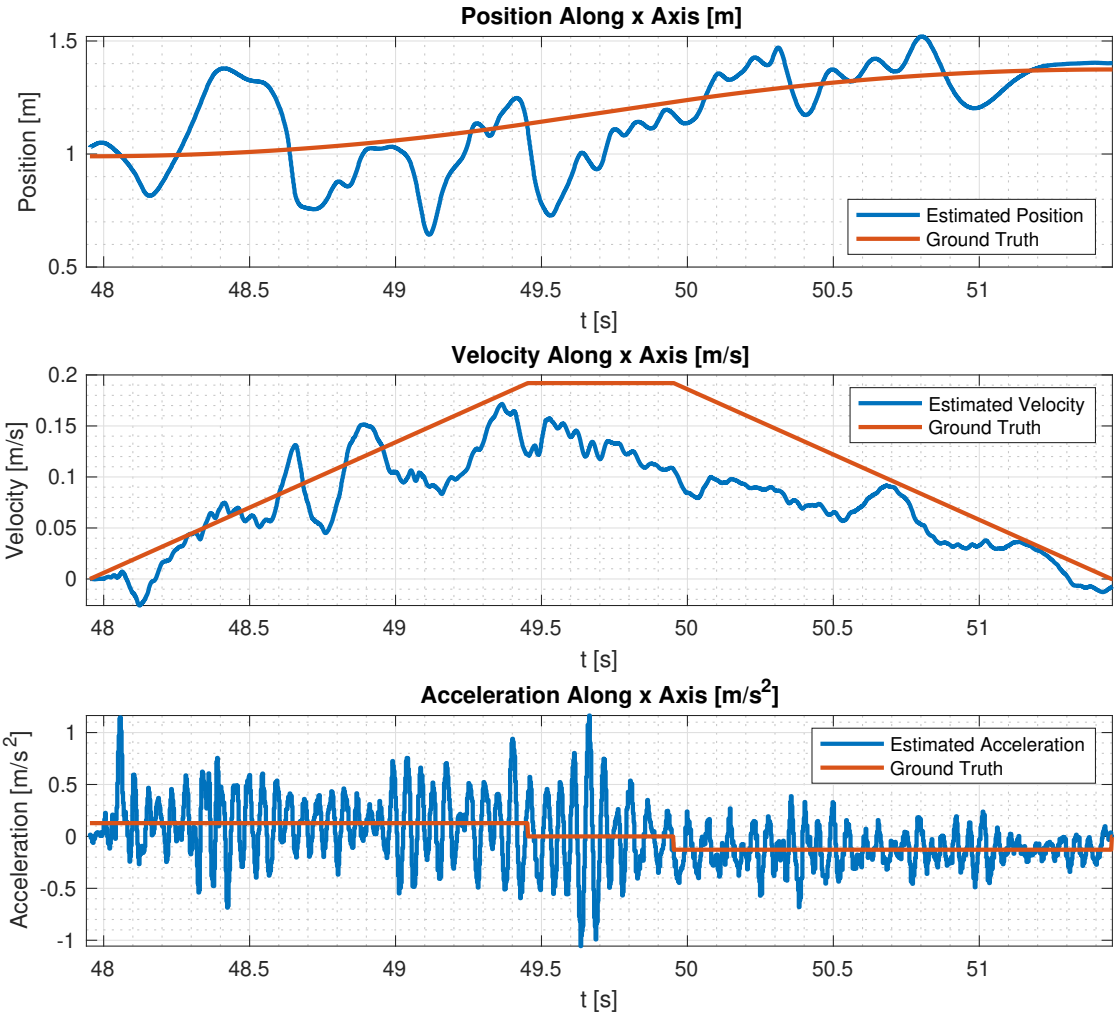


Figure F.1: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along x- Axis Using RF Signal Only When Moving x- Axis All Based on EKF in Asynchronous Model.

APPENDIX G

INERTIAL SENSOR ONLY BASED 3D LOCALIZATION

In this section, we propose that using cellphone embedded IMU is able to achieve centimeter-accuracy 3D localization. In addition, the percentage accuracy is improved when travelling distance increases, which enables 3D localization within 3% accuracy. The IMU accelerometer measurement data are calibrated using the scalar and skew matrices from Table E.3 and the bias estimated before input into the NLE estimation. The values are shown in Table E.3 as well.

G.1 Moving Along x- Axis Then y- Axis and Finally z- Axis

(1) Asynchronous NLE Estimation

Following steps in the Asynchronous estimation in section 10.5 which is based on NLE in **??**. Input are IMU sensor data: accelerations, angular velocities from gyroscope, and orientation from magnetometer.

Table G.1: RMS Error and Percentage of Relative and Maximal Distance for Estimated Position Along x- Then y- and Finally z- Axes Based on NLE in Asynchronous Model with Stack Length of 5

	X Axis	Y Axis	Z Axis
Estimated RMS Error [m]	0.0038	0.0187	0.0027
Estimated RMS Error After Calibration[m]	0.0032	0.0163	0.0027
Moving Range [m]	0.3840	0.2944	0.0448
Percentage of Relative Distance	0.83 %	5.54 %	6.03 %
Percentage of Maximal Distance [$\%$]		4.87 %	

As shown in Table G.1, estimated RMS error of position along x-axis is 0.0032 m, along y- axis is 0.013 m, along z- axis is 0.0027 m in the motion of 0.3840 m, 0.2944 m, and 0.0448 m along x- y- z- axes. The percentage of relative distance is 0.83 $\%$ - 6.03 $\%$.

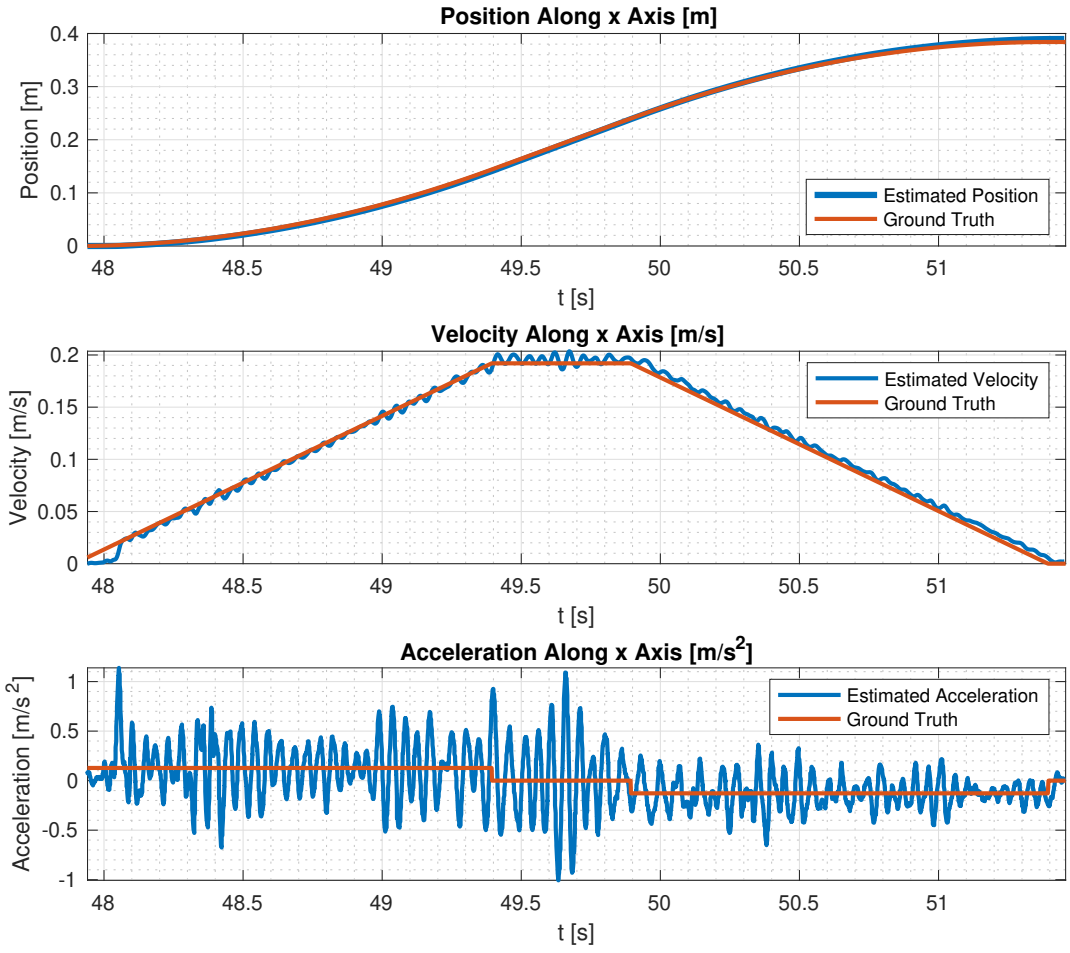


Figure G.1: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along x Axis Using Cellphone Embedded IMU Measurement Data Based on NLE in Asynchronous Model with Stack Length of 5 in Motion Along x- Axis Alone.

The percentage of maximal distance is 4.87 $\%$.

More specifically, estimated position, velocity, and acceleration states along x- y- z- axes as well as ground truth states are show in Figure G.1, Figure G.2, and Figure G.3. It can be inferred from the three figures that estimated position and velocity based on the cellphone embedded IMU measurements based on NLE of stack length of 5 in asynchronous model fit the ground truth states.

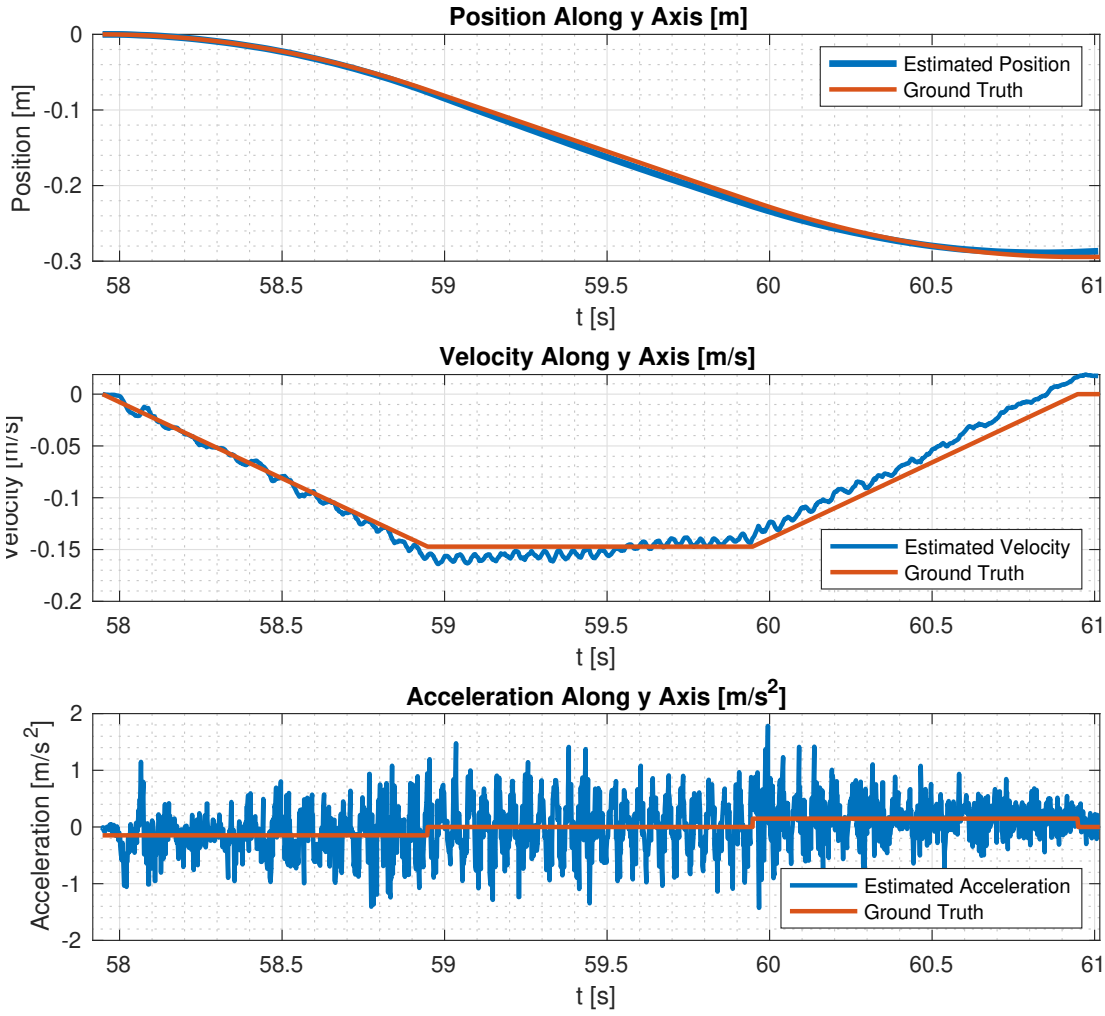


Figure G.2: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along y Axis Using Cellphone Embedded IMU Measurement Data Based on NLE in Asynchronous Model with Stack Length of 5 in Motion Along y- Axis Alone.

In addition, it can be inferred from Table G.1 that:

- As long as acceleration is > 0.0447 m/s^2, NLE based estimation achieves high accuracy position estimation. The percentage errors are all < 7 %.
- Estimation position error doesn't increase with the increase of travelling distance. The percentage error is decreasing from 6.03 % - 5.54 % - 0.83 %.

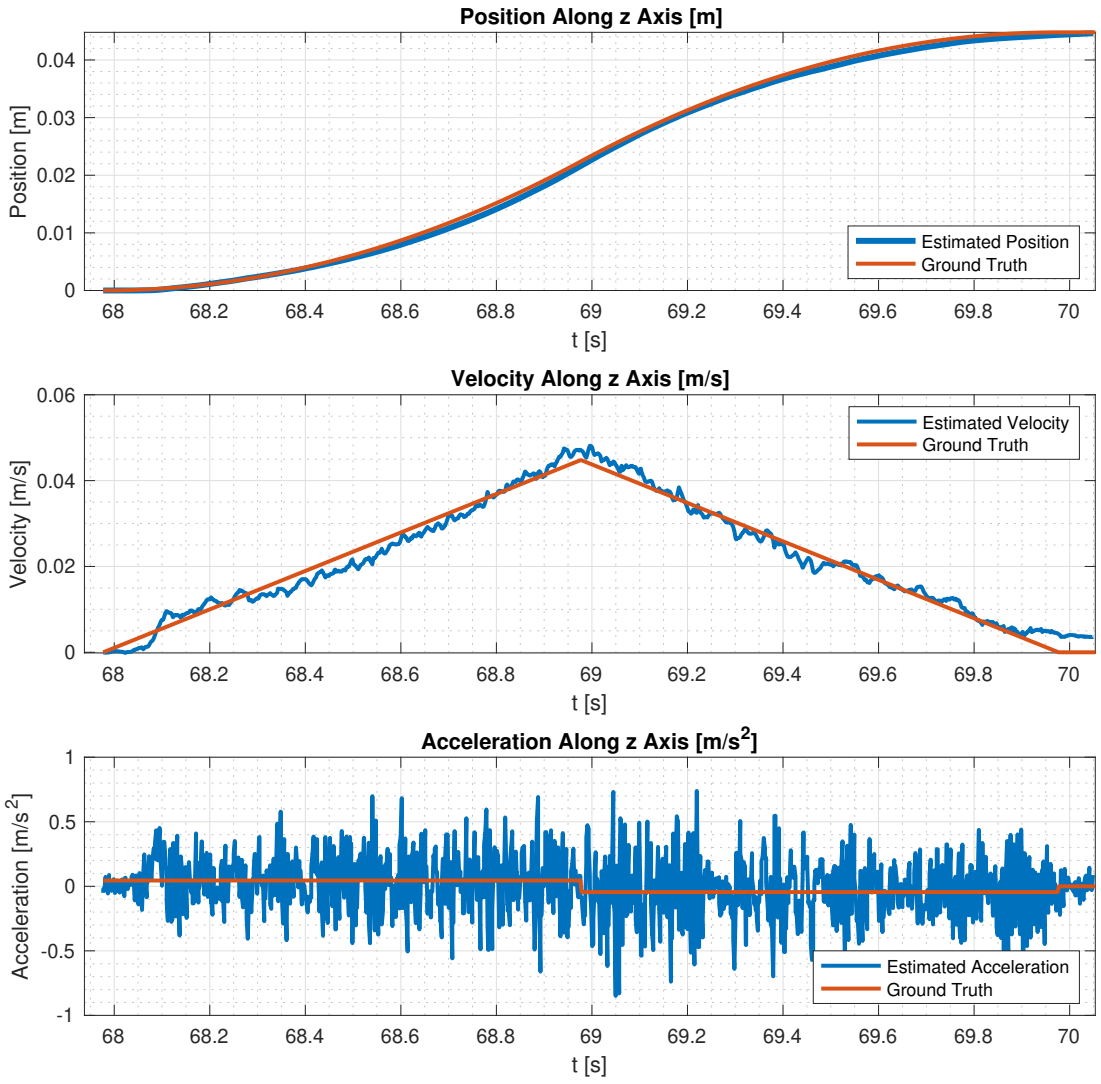


Figure G.3: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along z Axis Using Cellphone Embedded IMU Measurement Data Based on NLE in Asynchronous Model with Stack Length of 5 in Motion Along z- Axis Alone.

- In longer travelling distance, improved percentage error is achievable. Hence, low cost cellphone embedded IMU is able to achieve 3D high accuracy localization.

(2) Synchronous NLE Estimation

Table G.2: RMS Error and Percentage of Relative and Maximal Distance for Estimated Position Along x- Then y- and Finally z- Axes Based on NLE in Synchronous Model with Stack Length of 5

	X Axis	Y Axis	Z Axis
Estimated RMS Error [m]	0.1410	0.1464	0.0481
Moving Range [m]	0.3840	0.2944	0.0448

It can be inferred from Table G.2 that synchronous model is of much less accuracy compared to asychronized model due to time difference among different sensors.

(3) Asynchronous EKF Estimation

Following steps in asynchronous estimation in section 10.5 based on EKF in **??**. Input are also the IMU sensor data from accelerometer, gyroscope, and magnetometer using the scalar and skew matrices from section E.3 and the bias estimated before input the estimation.

Table G.3: RMS Error and Percentage of Relative and Maximal Distance for Estimated Position Along x- Then y- and Finally z- Axes Based on EKF in Asynchronous Model

	X Axis	Y Axis	Z Axis
Estimated RMS Error [m]	0.0060	0.0099	0.00066
Estimated RMS Error After Calibrationn [m]	0.0054	0.0077	0.00066
Moving Range [m]	0.3840	0.2944	0.0448
Percentage of Relative Distance	1.41 %	2.62 %	1.47 %
Percentage of Maximal Distance [$\%$]		2.01 %	

As shown in Table G.3, estimated RMS error of position along x- axis is 0.0060 m, along y- axis is 0.0099 m, along z- axis is 0.00066 m in the motion of 0.3840 m, 0.2944 m, and 0.0448 m along x- y- z- axes. The percentage of relative distance is 1.41 $\%$ - 2.62 $\%$. The percentage of maximal distance is 2.01 $\%$. Therefore, EKF is of higher estimation accuracy compared to NLE for separate motion along each axis.

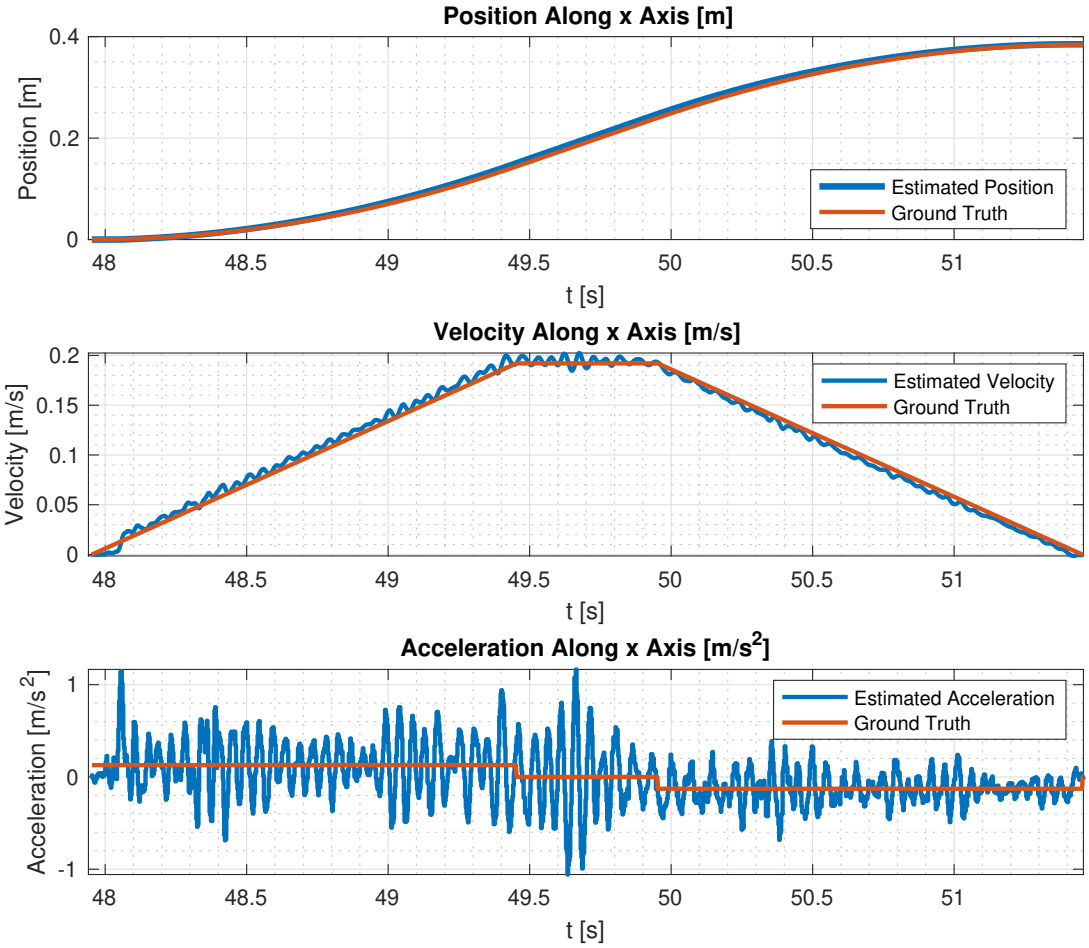


Figure G.4: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along x Axis Using Cellphone Embedded IMU Measurement Data Based on EKF in Asynchronous Model in Motion Along x- Axis Alone.

More specifically, estimated position, velocity, and acceleration states along x- y- z-axes as well as ground truth states are show in Figure G.4, Figure G.5, and Figure G.6. It can be inferred from the three figures that estimated position and velocity based on the cellphone embedded IMU measurements based on EKF in asynchronous model fit the ground truth states.

In addition, it can be inferred from Table G.3 that:

- EKF based estimation achieves high accuracy position estimation. The percentage

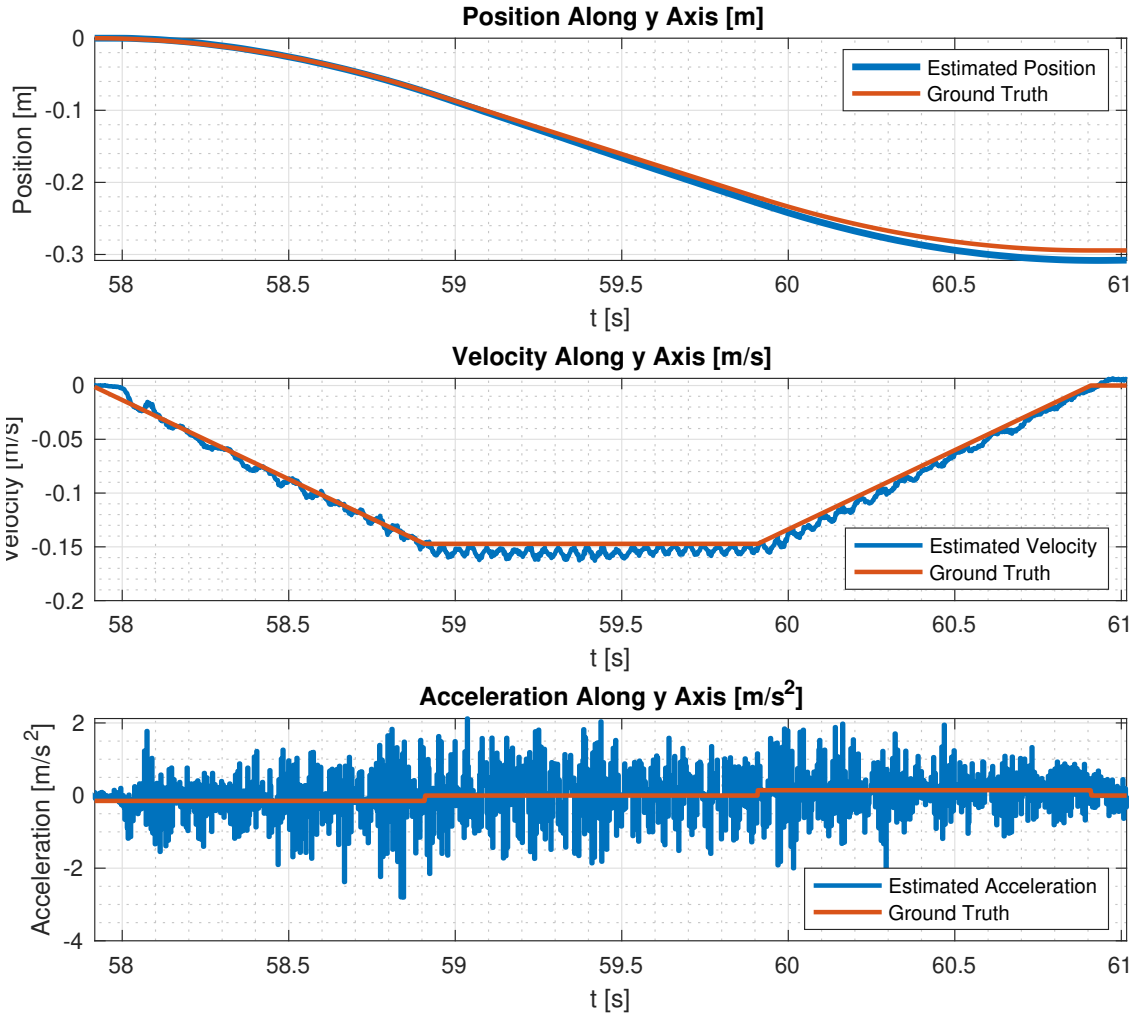


Figure G.5: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along y Axis Using Cellphone Embedded IMU Measurement Data Based on EKF in Asynchronous Model in Motion Along y- Axis Alone.

errors are all $< 3\ \%$.

- Estimation position error doesn't increase with the increase of travelling distance. The percentage error is decreasing from $1.47\ \%$ - $2.62\ \%$ - $1.41\ \%$.

- In longer travelling distance, improved percentage error is achievable. Hence, low cost cellphone embedded IMU is able to achieve 3D high accuracy localization.

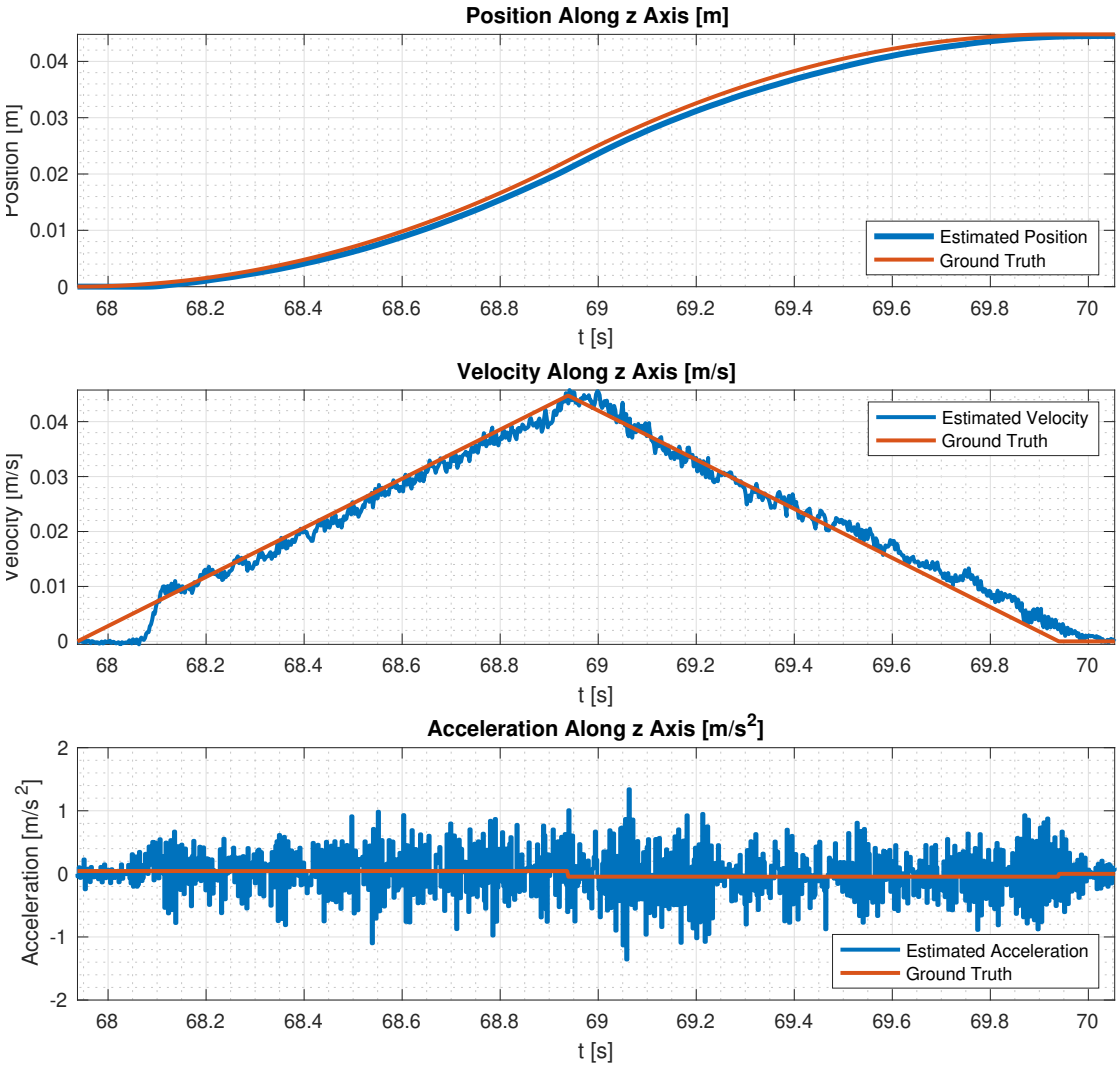


Figure G.6: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along z Axis Using Cellphone Embedded IMU Measurement Data Based on EKF in Asynchronous Model in Motion Along z- Axis Alone.

- NLE based estimation achieves higher estimation accuracy compared to EKF based estimation.

(4) Synchronous EKF Estimation

In addition, estimation results based on EKF in synchronous model are shown in Table G.3.

Table G.4: RMS Error and Percentage of Relative and Maximal Distance for Estimated Position Along x- Then y- and Finally z- Axes Based on EKF in Synchronous Model

	X Axis	Y Axis	Z Axis
Estimated RMS Error [m]	0.1448	0.1393	0.04141
Moving Range [m]	0.3840	0.2944	0.0448

G.2 Moving Along x- y- z- Axes All Together Simultaneously

(1) Asynchronous NLE Estimation

After moving back to origin O, the tag moves along x- y- z- axes all together. The estimation result based on NLE in asynchronous model with stack length of 5 is shown in Table G.5.

Table G.5: RMS Error and Percentage of Relative and Maximal Distance for Estimated Position Along x- y- and z- Axes All Together Based on NLE in Asynchronous Model with Stack Length of 5

	X Axis	Y Axis	Z Axis
Estimated RMS Error [m]	0.0170	0.0021	0.0432
Estimated RMS Error After Calibration [m]	0.0216	0.0036	0.0439
Moving Range [m]	0.3840	0.2944	0.0448
Percentage of Relative Distance	4.43 %	0.71 %	-
Percentage of Maximal Distance [%]		4.43 %	

As shown in Table G.5, estimated RMS error of position along x- axis is 0.0170 m, along y- axis is 0.0021 m, along z- axis is 0.0432 m in the motion. The percentage of relative distance is 4.43 % - 0.71 %. The percentage of maximal distance is 4.43 %. More specifically, estimated position, velocity, and acceleration states along x- y- z- axes as well as ground truth states are show in Figure G.7 - Figure G.9.

It can be inferred from Table G.5, Figure G.7, Figure G.8, and Figure G.9 that:

- Estimated position and velocity based on the cellphone embedded IMU measurements based on NLE in asynchronous model with stack length of 5 fit the ground truth states.
- NLE based estimation in moving along x- y- z- axes all together simultaneously achieves higher estimation accuracy compared to moving along x- then y- and finally along z- axis.

(2) Asynchronous EKF Estimation

In addition, estimation result based on EKF in Asynchronous model is shown in Table G.6.

Table G.6: RMS Error and Percentage of Relative and Maximal Distance for Estimated Position Along x- y- and z- Axes All Together Based on EKF in Asynchronous Model

	X Axis	Y Axis	Z Axis
Estimated RMS Error [m]	0.0196	0.0106	0.0025
Estimated RMS Error After Calibration [m]	0.0193	0.0079	0.0033
Moving Range [m]	0.3840	0.2944	0.0448
Percentage of Relative Distance	5.10 %	3.60 %	5.58 %
Percentage of Maximal Distance [%]		4.56 %	

It can be inferred from Table G.6, Figure G.10, Figure G.11, and Figure G.12 that:

- Estimated position and velocity based on the cellphone embedded IMU measurements based on EKF in asynchronous model fit the ground truth states.
- EKF based estimation in moving along x- y- z- axes all together simultaneously achieves lower estimation accuracy compared to moving along x- then y- and finally along z- axis.
- NLE based estimation in moving along x- y- z- axes all together simultaneously achieves higher estimation accuracy compared to EKF based estimation.

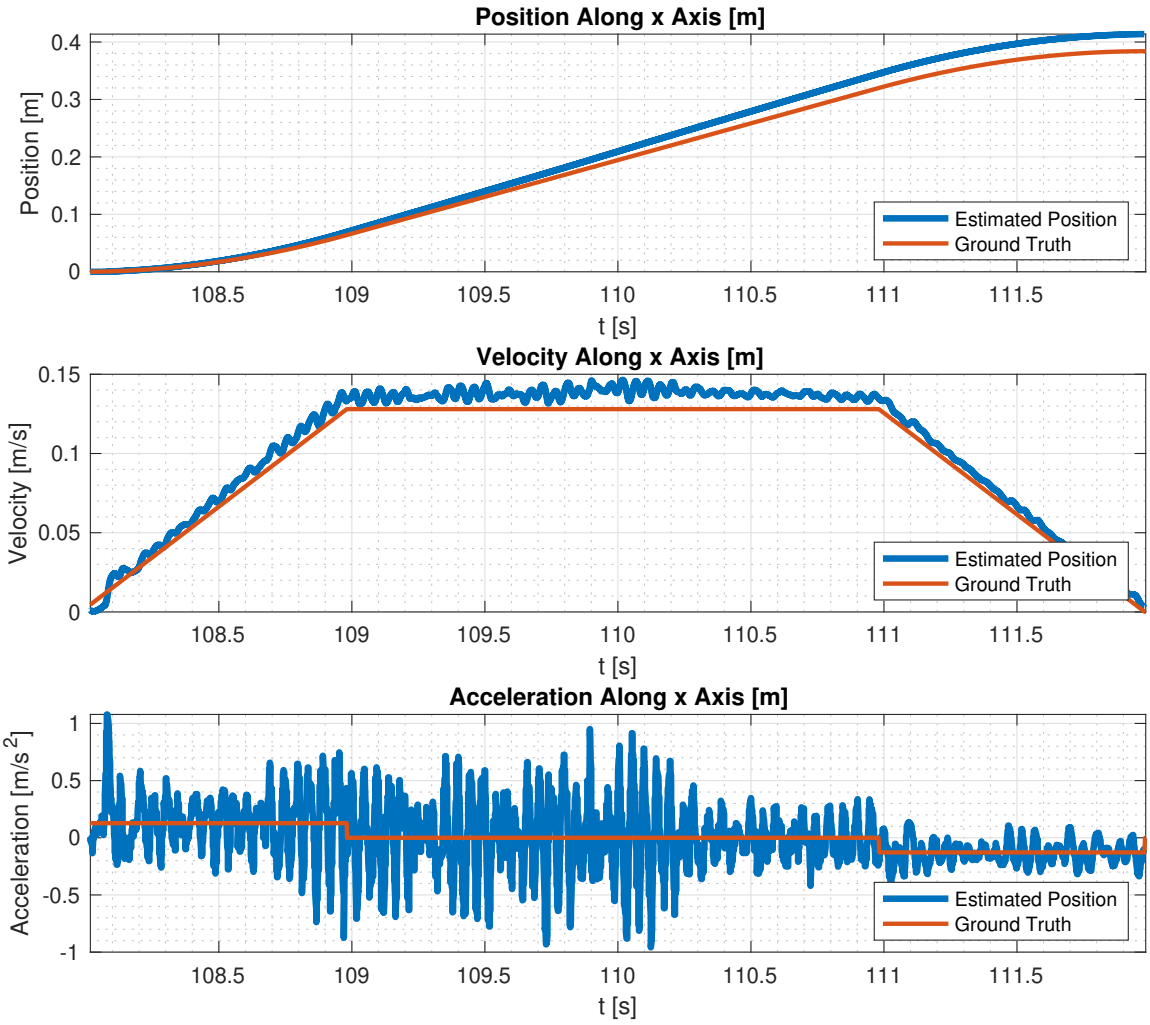


Figure G.7: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along x Axis Using Cellphone Embedded IMU Measurement Data Based on NLE in Asynchronous Model with Stack Length of 5 When Moving x- y- z- Axes All Together Simultaneously.

In IMU based estimation, the initial position needs to be known. Estimation based on accelerations, orientations, and angular velocities derives relative distance, velocity, and acceleration travelled from origin O.

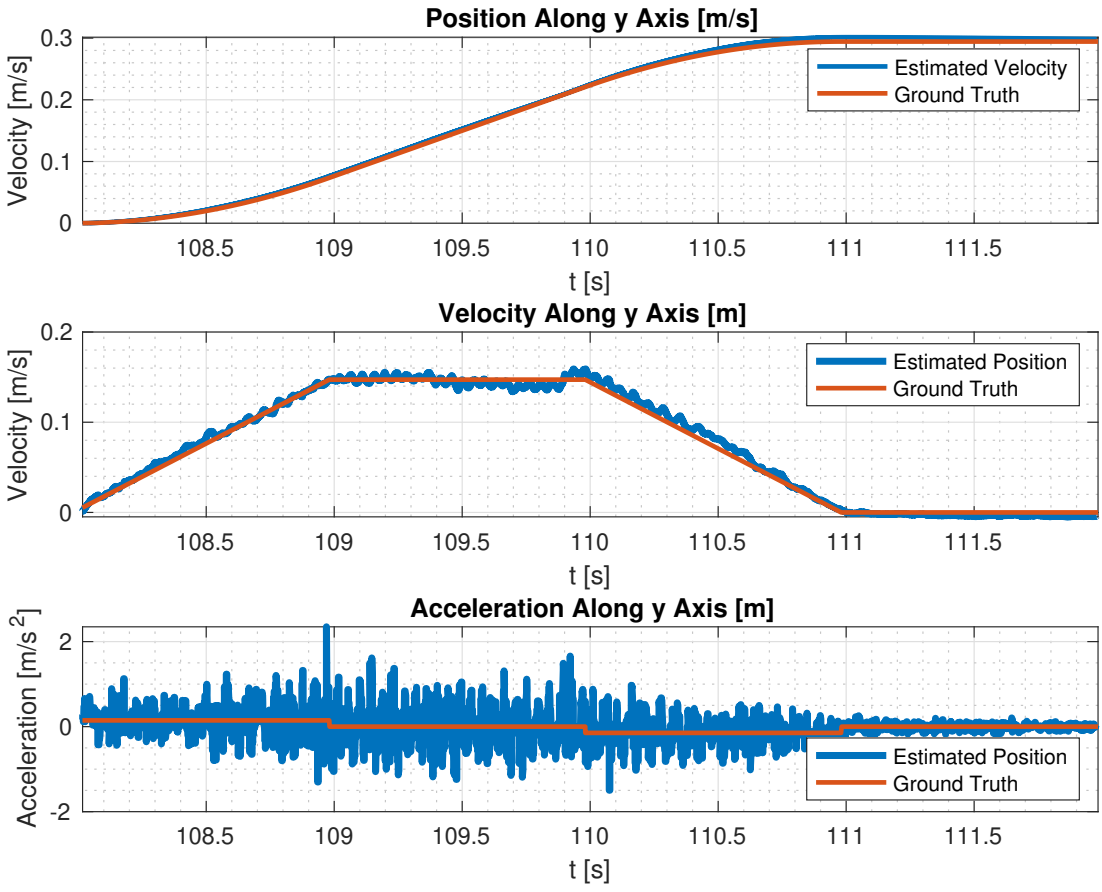


Figure G.8: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along y Axis Using Cellphone Embedded IMU Measurement Data Based on NLE in Asynchronous Model with Stack Length of 5 When Moving x- y- z- Axes All Together Simultaneously.

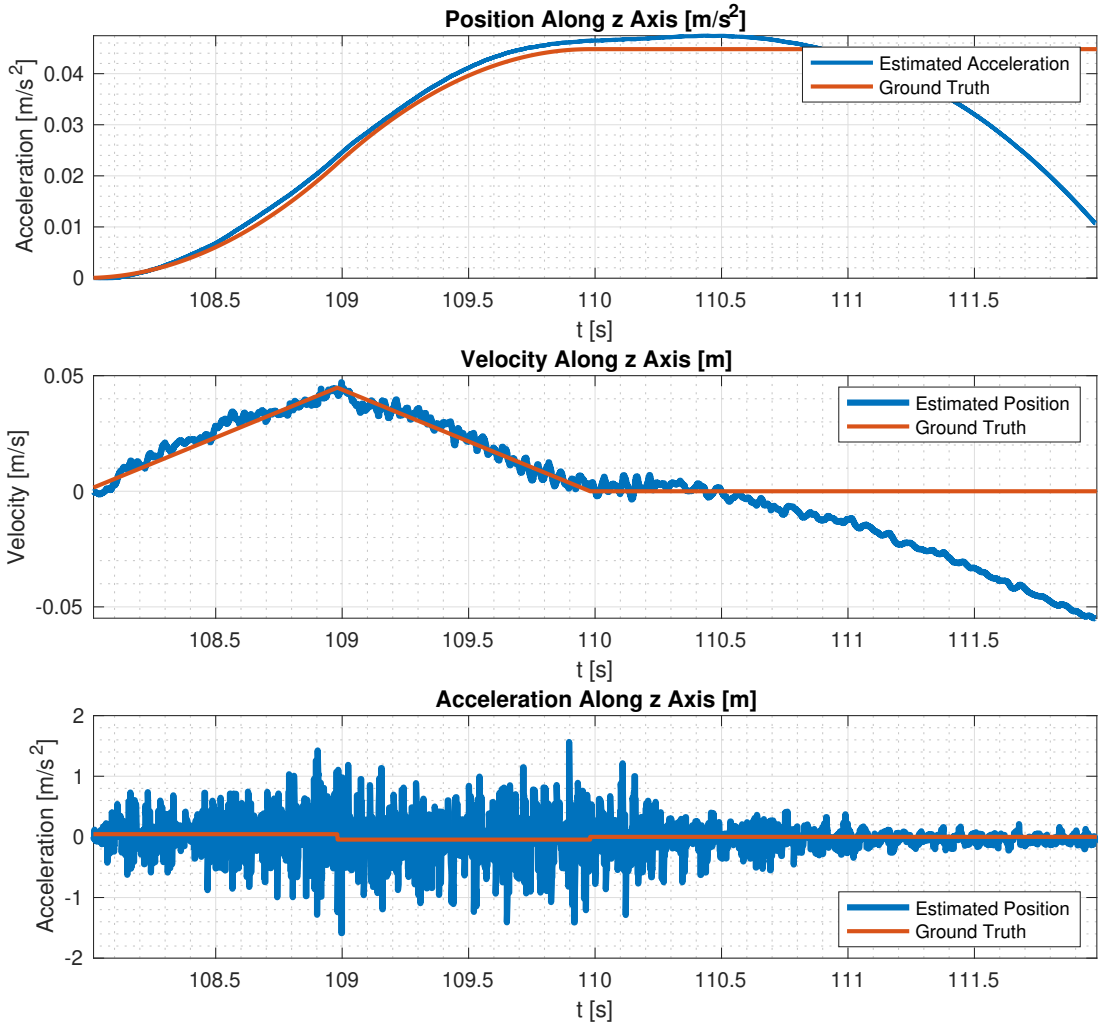


Figure G.9: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along z Axis Using Cellphone Embedded IMU Measurement Data Based on NLE in Asynchronous Model with Stack Length of 5 When Moving x- y- z- Axes All Together Simultaneously.

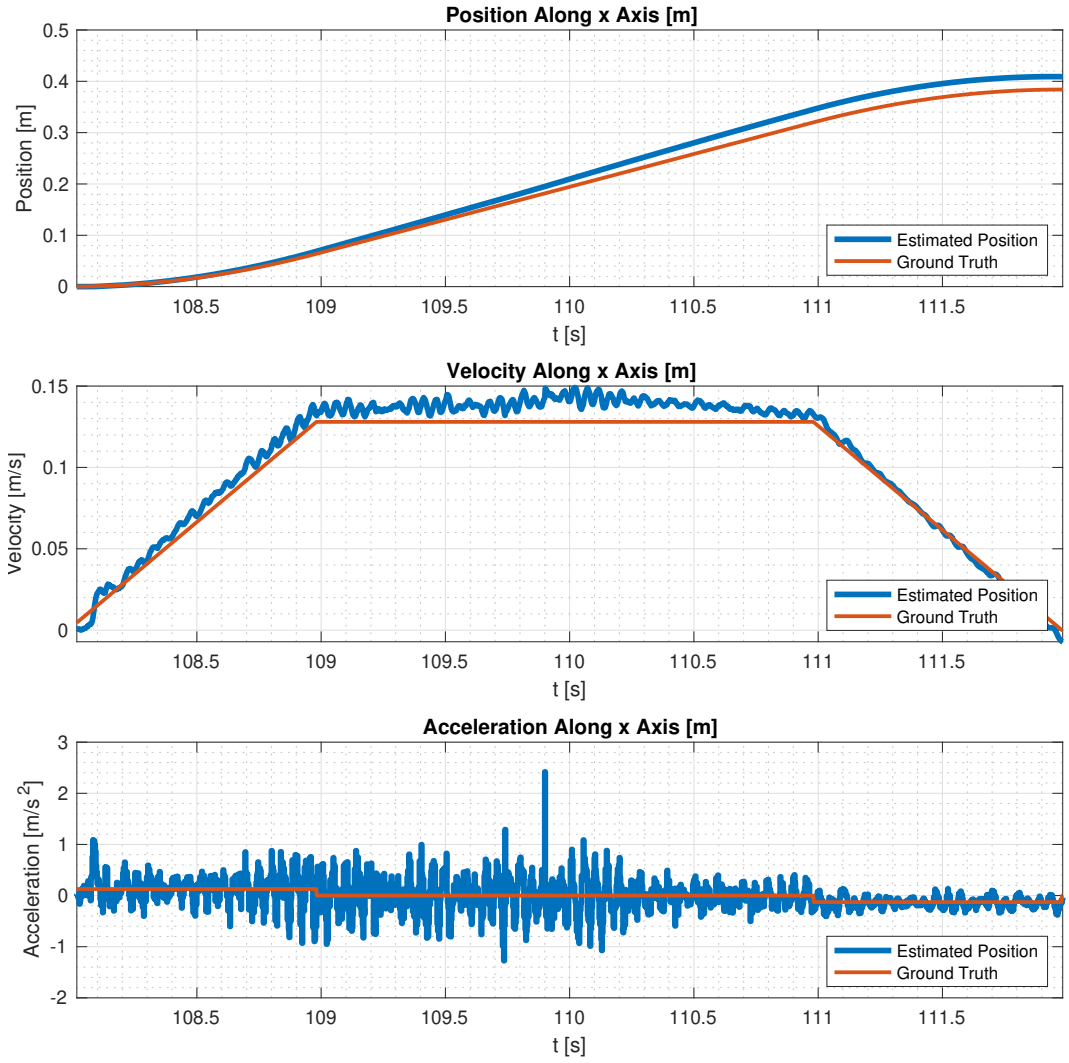


Figure G.10: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along x Axis Using Cellphone Embedded IMU Measurement Data Based on EKF in Asynchronous Model When Moving x- y- z- Axes All Together Simultaneously.

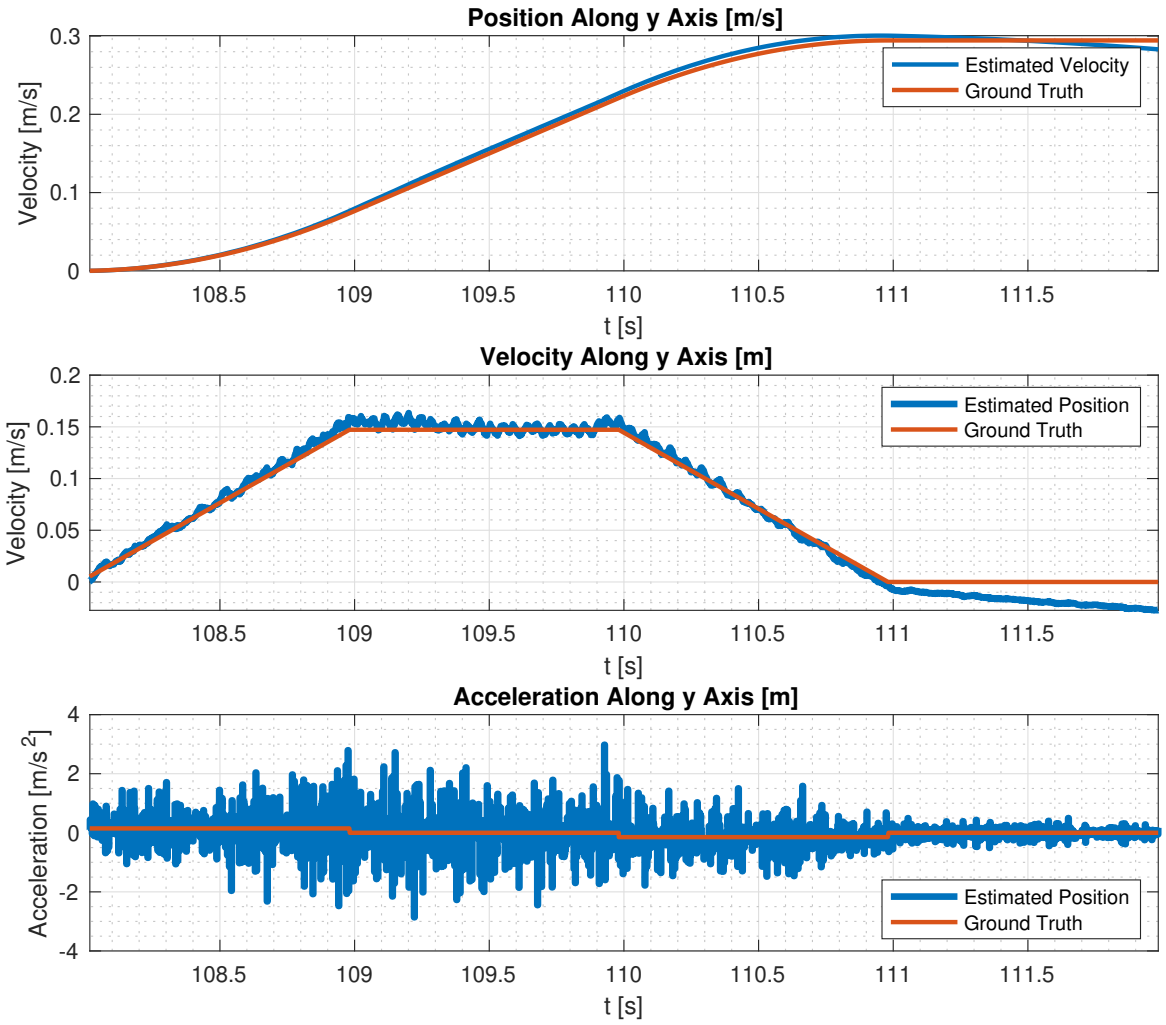


Figure G.11: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along y Axis Using Cellphone Embedded IMU Measurement Data Based on EKF in Asynchronous Model When Moving x- y- z- Axes All Together Simultaneously.

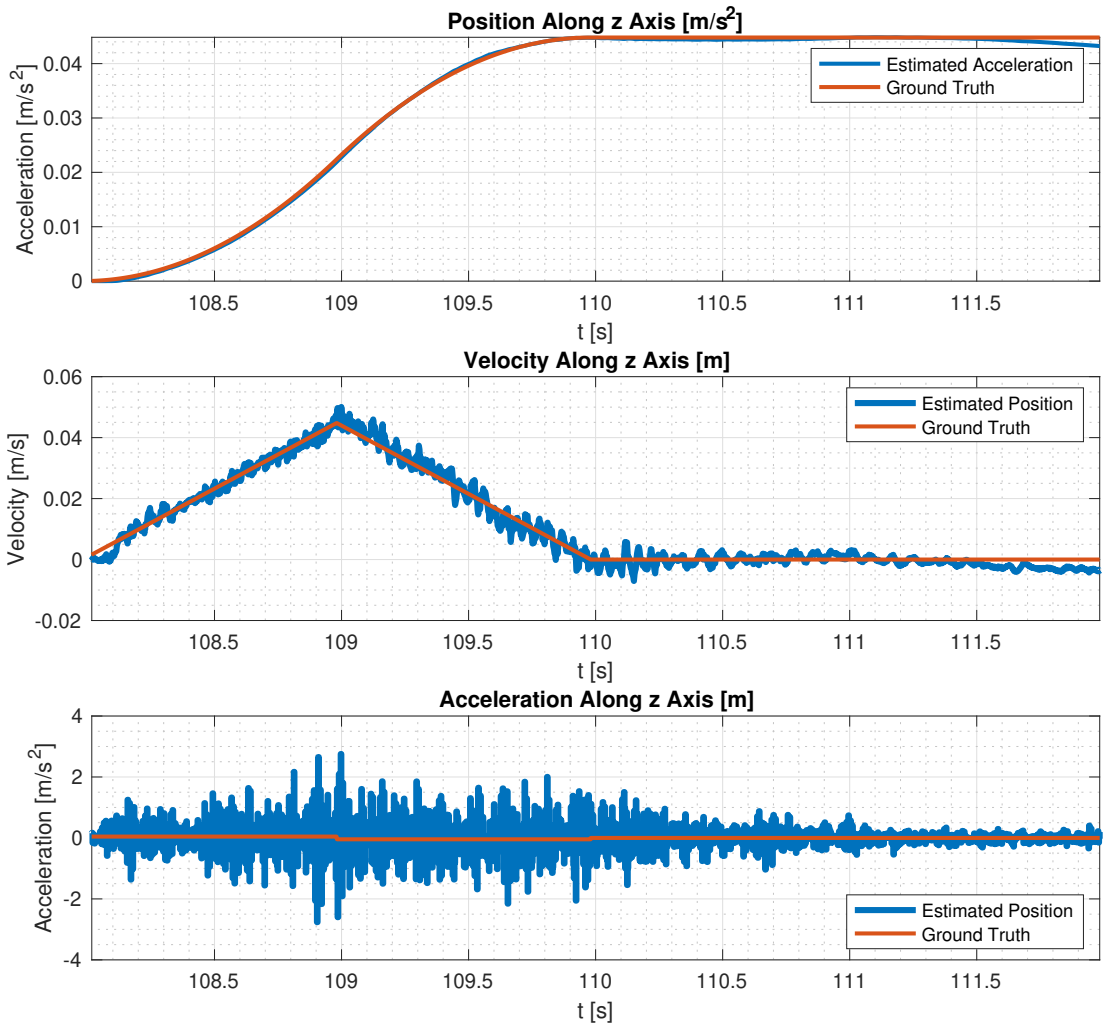


Figure G.12: Estimation Results of the 3D Motion of Position Velocities and Acceleration Along z Axis Using Cellphone Embedded IMU Measurement Data Based on EKF in Asynchronous Model When Moving x- y- z- Axes All Together Simultaneously.

REFERENCES

[1] *Nfl*, https://newatlas.com/rfid-tags-to-enable-real-time-tracking-of-nfl-players/33199/.

[2] *Rfrain*, https://rfrain.com/blog-posts/rfrain-maximizes-efficiency-with-rfid-tracking-technology-for-supply-chain-management/.

[3] *Wireless communication and tracking*, https://www.cdc.gov/niosh/mining/content/emergencymanagementandresponse/\commtracking/advcommtrackingtutorial1.html.

[4] *Motion capture*, https://en.wikipedia.org/wiki/Motion_capture.

[5] E. van Der Kruk and M. M. Reijne, "Accuracy of human motion capture systems for sport applications; state-of-the-art review," *European Journal of Sport Science*, vol. 18, no. 6, pp. 806–819, 2018.

[6] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system," *ACM Trans. Inf. Syst.*, vol. 10, no. 1, pp. 91–102, Jan. 1992.

[7] E. van der Kruk, "Modelling and measuring 3d movements of a speed skater," 2013.

[8] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '00, Boston, Massachusetts, USA: ACM, 2000, pp. 32–43.

[9] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, "The anatomy of a context-aware application," *Wireless Networks*, vol. 8, no. 2, pp. 187–197, Mar. 2002.

[10] M. A. Brodie, A. G. Walmsley, and W. Page, "Fusion motion capture: A prototype system using inertial measurement units and gps for the biomechanical analysis of ski racing," 2008.

[11] ScienceDaily, *Robots track moving objects with unprecedented precision: System uses rfid tags to home in on targets; could benefit robotic manufacturing, collaborative drones, and other applications.*

[12] J. Hightower, G. Borriello, and R. Want, "Spoton: An indoor 3d location sensing technology based on rf signal strength," Mar. 2000.

[13] F. Tlili, N. Hamdi, and A. Belghith, "Accurate 3d localization scheme based on active rfid tags for indoor environment," in *2012 IEEE International Conference on RFID-Technologies and Applications (RFID-TA)*, 2012, pp. 378–382.

[14] J. Maneesilp, C. Wang, H. Wu, and N.-F. Tzeng, "Rfid support for accurate 3d localization," *IEEE Transactions on Computers*, vol. 62, no. 7, pp. 1447–1459, 2013.

[15] L. Qiu, Z. Huang, N. Wirström, and T. Voigt, "3dinsar: Object 3d localization for indoor rfid applications," in *2016 IEEE International Conference on RFID (RFID)*, 2016, pp. 1–8.

[16] L. Yang, Y. Chen, X.-Y. Li, C. Xiao, M. Li, and Y. Liu, "Tagoram: Real-time tracking of mobile rfid tags to high precision using cots devices," in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '14, Maui, Hawaii, USA: Association for Computing Machinery, 2014, pp. 237–248, ISBN: 9781450327831.

[17] I. Pefkianakis and K.-H. Kim, "Accurate 3d localization for 60 ghz networks," in *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '18, Shenzhen, China: Association for Computing Machinery, 2018, pp. 120–131, ISBN: 9781450359528.

[18] F. Adib, Z. Kabelac, D. Katabi, and R. C. Miller, "3d tracking via body radio reflections," in *Proceedings of the 11th USENIX Conference on Networked Systems Design and Implementation*, ser. NSDI'14, Seattle, WA: USENIX Association, 2014, pp. 317–329, ISBN: 9781931971096.

[19] Y. Ma, X. Hui, and E. C. Kan, "3d real-time indoor localization via broadband nonlinear backscatter in passive devices with centimeter precision," in *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '16, New York City, New York: Association for Computing Machinery, 2016, pp. 216–229, ISBN: 9781450342261.

[20] P. Singh, A. Khosla, A. Kumar, and M. Khosla, "3d localization of moving target nodes using single anchor node in anisotropic wireless sensor networks," *AEU - International Journal of Electronics and Communications*, vol. 82, pp. 543–552, 2017.

[21] J. Maneesilp, C. Wang, H. Wu, and N.-F. Tzeng, "Rfid support for accurate 3d localization," vol. 62, no. 7, 2013.

[22] J. Wang, F. Adib, R. Knepper, D. Katabi, and D. Rus, "Rf-compass: Robot object manipulation using rfids," in *Proceedings of the 19th Annual International Confer-*

ence on Mobile Computing Networking, ser. MobiCom '13, Miami, Florida, USA: Association for Computing Machinery, 2013, pp. 3–14, ISBN: 9781450319997.

[23] A. Eleryan, M. Elsabagh, and M. Youssef, "Synthetic generation of radio maps for device-free passive localization," *2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, pp. 1–5, 2011.

[24] P. Davidson and R. Piche, "A survey of selected indoor positioning methods for smartphones," *IEEE Communications surveys and tutorials*, vol. 19, no. 2, pp. 1347–1370, 2017.

[25] K. A. Gotsis, I. Kyriakides, and J. N. Sahalos, "3d localization and frequency band estimation of multiple unknown rf sources using particle filters and a wireless sensor network," *Wireless personal communications*, vol. 90, no. 4, pp. 1889–1902, 2016.

[26] R. Zhang, A. Bannoura, F. Hoflinger, L. M. Reindl, and C. Schindelhauer, "Indoor localization using a smart phone," in *2013 IEEE Sensors Applications Symposium Proceedings*, IEEE, 2013, pp. 38–42, ISBN: 9781467346368.

[27] H.-H. Hsu, J.-K. Chang, W.-J. Peng, T. K. Shih, T.-W. Pai, and K. L. Man, "Indoor localization and navigation using smartphone sensory data," *Annals of operations research*, vol. 265, no. 2, pp. 187–204, 2018.

[28] E. Martin, O. Vinyals, G. Friedland, and R. Bajcsy, "Precise indoor localization using smart phones," in *Proceedings of the 18th ACM international conference on multimedia*, ser. MM '10, ACM, 2010, pp. 787–790, ISBN: 1605589330.

[29] J. Hightower and G. Borriello, "A survey and taxonomy of location systems for ubiquitous computing."

[30] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications*, vol. 2, Mar. 2000, pp. 775–784.

[31] L. M. Ni, Yunhao Liu, Yiu Cho Lau, and A. P. Patil, "Landmarc: Indoor location sensing using active rfid," in *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, (PerCom 2003)*, Mar. 2003, pp. 407–415.

[32] W. Tidd, J. W. Raymond, Y. Huang, and Y. Zhao, "A rf source localization and tracking system," in *2010 - MILCOM 2010 MILITARY COMMUNICATIONS CONFERENCE*, 2010, pp. 858–863.

[33] D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and localization with rfid technology," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 1, Apr. 2004, 1015–1020 Vol.1.

[34] T. Deyle, H. Nguyen, M. Reynolds, and C. C. Kemp, "Rf vision: Rfid receive signal strength indicator (rssi) images for sensor fusion and mobile manipulation," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2009, pp. 5553–5560.

[35] B. Choi, J. Lee, and K. Park, "A hierarchical algorithm for indoor mobile robot localization using rfid sensor fusion," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 6, pp. 2226–2235, Jun. 2011.

[36] H. Samuel, S. Connell, I. Milligan, D. Austin, T. Hayes, and P. Chiang, "Indoor localization using pedestrian dead reckoning updated with rfid-based fiducials," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011, pp. 7598–7601.

[37] M. Stikic, T. Huynh, K. Van Laerhoven, and B. Schiele, "Adl recognition based on the combination of rfid and accelerometer sensing," in *2008 Second International Conference on Pervasive Computing Technologies for Healthcare*, Jan. 2008, pp. 258–263.

[38] Y. Wang, Z. Jiang, Y. Li, J.-N. Hwang, G. Xing, and H. Liu, "Rodnet: A real-time radar object detection network cross-supervised by camera-radar fused object 3d localization," *IEEE Journal of Selected Topics in Signal Processing*, vol. PP, pp. 1–1, Feb. 2021.

[39] M. B. Akbar, D. G. Taylor, and G. D. Durgin, "Hybrid inertial microwave reflectometry for mm-scale tracking in rfid systems," *IEEE transactions on wireless communications*, vol. 14, no. 12, pp. 6805–6814, 2015.

[40] Y. H. Lee and I. Marsic, "Object motion detection system using passive uhf rfid technology for a trauma bay," 2017.

[41] C. Hekimian-Williams, B. Grant, X. Liu, Z. Zhang, and P. Kumar, "Accurate localization of rfid tags using phase difference," *2010 IEEE International Conference on RFID*, pp. 89–96, 2010.

[42] G. Li, D. Arnitz, R. Ebelt, U. Muehlmann, K. Witrisal, and M. Vossiek, "Bandwidth dependence of cw ranging to uhf rfid tags in severe multipath environments," in *2011 IEEE International Conference on RFID*, 2011, pp. 19–25.

[43] Y. Ma and E. C. Kan, "Accurate indoor ranging by broadband harmonic generation in passive nltl backscatter tags," *IEEE Transactions on Microwave Theory and Techniques*, vol. 62, no. 5, pp. 1249–1261, 2014.

[44] J. S. Day, G. A. Dumas, and D. J. Murdoch, "Evaluation of a long-range transmitter for use with a magnetic tracking device in motion analysis," *Journal of Biomechanics*, vol. 31, no. 10, pp. 957–961, 1998.

[45] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proceedings of 1995 American Control Conference - ACC'95*, vol. 3, Jun. 1995, 1628–1632 vol.3.

[46] M. Nørgaard, N. K. Poulsen, and O. Ravn, "New developments in state estimation for nonlinear systems," *Automatica*, vol. 36, no. 11, pp. 1627–1638, 2000.

[47] P. S. Maybeck, *Stochastic models, estimation and control. Volume 2*, ser. Mathematics in science and engineering ; v. 141. New York: Academic Press, 1982, ISBN: 1-282-29029-0.

[48] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*. New York, NY, USA: Wiley-Interscience, 2006.

[49] D. G. Taylor, "Design of nonlinear least-squares velocity estimation algorithms for automotive vehicles," in *49th IEEE Conference on Decision and Control (CDC)*, IEEE, 2010, pp. 6409–6414.

[50] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Society for Industrial and Applied Mathematics, 1996.

[51] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[52] J. D. Griffin and G. D. Durgin, "Complete link budgets for backscatter-radio and rfid systems," *IEEE antennas and propagation magazine*, vol. 51, no. 2, pp. 11–25, 2009.

[53] F. Amato, H. M. Torun, and G. D. Durgin, "Rfid backscattering in long-range scenarios," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2718–2725, 2018.

[54] G. D. Durgin, *Space-Time Wireless Channels*. Prentice Hall Press, 2002.

[55] D. Katz, J. Baptista, S. P. Azen, and M. C. Pike, "Obtaining confidence intervals for the risk ratio in cohort studies," *Biometrics*, vol. 34, no. 3, pp. 469–474, 1978.

[56] G. L. Stüber, *Principles of mobile communication*. 1996, ISBN: 0792397320.

[57] C. Tepedelenlioglu, A. Abdi, and G. Giannakis, "The ricean k factor: Estimation and performance analysis," *IEEE transactions on wireless communications*, vol. 2, no. 4, pp. 799–810, 2003.

[58] L. Greenstein, D. Michelson, and V. Erceg, "Moment-method estimation of the ricean k-factor," *IEEE communications letters*, vol. 3, no. 6, pp. 175–176, 1999.

[59] W. Hocking, "Reduction of the effects of non-stationarity in studies of amplitude statistics of radio wave backscatter," *Journal of atmospheric and terrestrial physics*, vol. 49, no. 11, pp. 1119–1131, 1987.

[60] P. K. Rastogi and O. Holt, "On detecting reflections in presence of scattering from amplitude statistics with application to d region partial reflections," *Radio Science*, vol. 16, no. 06, pp. 1431–1443, 1981.

[61] D. Arnitz, K. Witrisal, and U. Muehlmann, "Multifrequency continuous-wave radar approach to ranging in passive uhf rfid," *IEEE transactions on microwave theory and techniques*, vol. 57, no. 5, pp. 1398–1405, 2009.

[62] Q. Yang, D. G. Taylor, M. B. Akbar, and G. D. Durgin, "Analysis of kalman filter-based localization for himr rfid systems," *IEEE Journal of Radio Frequency Identification*, vol. 3, no. 3, pp. 164–172, 2019.

[63] K. LEVENBERG, "A method for the solution of certain non-linear problems in least squares," *Quarterly of applied mathematics*, vol. 2, no. 2, pp. 164–168, 1944.

[64] C. Qi, F. Amato, M. Alhassoun, and G. D. Durgin, "Breaking the range limit of rfid localization: Phase-based positioning with tunneling tags," in *2019 IEEE International Conference on RFID (RFID)*, 2019, pp. 1–8.

[65] *Fcc*, https://transition.fcc.gov/Bureaus/Engineering_Technology/Orders/2000/fcc00312.pdf.

[66] I. Frosio, F. Pedersini, and N. A. Borghese, "Autocalibration of mems accelerometers," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 6, pp. 2034–2041, 2009.

[67] S. Poddar, V. Kumar, and A. Kumar, "A comprehensive overview of inertial sensor calibration techniques," *Journal of Dynamic Systems, Measurement, and Control*, vol. 139, no. 1, p. 011 006, 2017.

[68] A. Grammenos, C. Mascolo, and J. Crowcroft, "You are sensing, but are you biased? a user unaided sensor calibration approach for mobile sensing," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 2, no. 1, p. 11, Mar. 2018.

[69] P. Aggarwal, Z. Syed, X. Niu, and N. El-Sheimy, "A standard testing and calibration procedure for low cost mems inertial sensors and units," *Journal of Navigation*, vol. 61, no. 2, pp. 323–336, 2008.

[70] F. Caron, E. Duflos, D. Pomorski, and P. Vanheeghe, "Gps/imu data fusion using multisensor kalman filtering: Introduction of contextual aspects," *Information fusion*, vol. 7, no. 2, pp. 221–230, 2006.

[71] F. Mirzaei and S. Roumeliotis, "A kalman filter-based algorithm for imu-camera calibration: Observability analysis and performance evaluation," *IEEE transactions on robotics*, vol. 24, no. 5, pp. 1143–1156, 2008.

[72] S. Sukkarieh, E. Nebot, and H. Durrant Whyte, "A high integrity imu/gps navigation loop for autonomous land vehicle applications," *IEEE transactions on robotics and automation*, vol. 15, no. 3, pp. 572–578, 1999.

[73] D. Tedaldi, A. Pretto, and E. Menegatti, "A robust and easy to implement method for imu calibration without external equipments," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 3042–3049.

[74] O. Sarkka, T. Nieminen, S. Suuriniemi, and L. Kettunen, "A multi-position calibration method for consumer-grade accelerometers, gyroscopes, and magnetometers to field conditions," *IEEE sensors journal*, vol. 17, no. 11, pp. 3470–3481, 2017.

[75] M. Kok, J. D. Hol, and T. B. Schön, "Using inertial sensors for position and orientation estimation," *Foundations and trends in signal processing*, vol. 11, no. 1-2, pp. 1–153, 2017.

[76] D. Gebre-Egziabher, G. H. Elkaim, J. David Powell, and B. W. Parkinson, "Calibration of strapdown magnetometers in magnetic field domain," *Journal of aerospace engineering*, vol. 19, no. 2, pp. 87–102, 2006.

[77] S. Griffin, *Quaternions : theory and applications*. Nova Science Publishers, Incorporated, 2017.

[78] D. A. Aligia, B. A. Roccia, C. H. De Angelo, G. A. Magallán, and G. N. González, "An orientation estimation strategy for low cost imu using a nonlinear luenberger observer," *Measurement*, vol. 173, p. 108 664, 2021.

[79] P. Moraal and J. Grizzle, "Observer design for nonlinear systems with discrete-time measurements," *IEEE transactions on automatic control*, vol. 40, no. 3, pp. 395–404, 1995.

[80] "Ieee xplore," New York, N.Y.: IEEE, 2000.

[81] N. Ben Halima and H. Boujemâa, "3d wls hybrid and non hybrid localization using toa, tdoa, azimuth and elevation," *Telecommunication systems*, vol. 70, no. 1, pp. 97–104, 2019.

[82] Y. Geng and K. Pahlavan, "Design, implementation, and fundamental limits of image and rf based wireless capsule endoscopy hybrid localization," *IEEE Transactions on Mobile Computing*, vol. 15, no. 8, pp. 1951–1964, 2016.

[83] H.-S. Gang and J.-Y. Pyun, "A smartphone indoor positioning system using hybrid localization technology," *Energies*, vol. 12, no. 19, 2019.

[84] E. Ataer-Cansizoglu, Y. Taguchi, and S. Ramalingam, "Pinpoint slam: A hybrid of 2d and 3d simultaneous localization and mapping for rgb-d sensors," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1300–1307.

[85] A. Fink and H. Beikirch, "Hybrid indoor tracking with bayesian sensor fusion of rf localization and inertial navigation," in *Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems*, vol. 2, 2011, pp. 823–827.

[86] L. M. Ni, D. Zhang, and M. R. Souryal, "Rfid-based localization and tracking technologies," *IEEE Wireless Communications*, vol. 18, no. 2, pp. 45–51, 2011.

[87] S. Traini, L. Sciullo, A. Trotta, and M. Di Felice, "Practical indoor localization via smartphone sensor data fusion techniques: A performance study," in *2019 16th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2019, pp. 1–7.

[88] W. T. Fong, S. K. Ong, and A. Y. C. Nee, "Methods for in-field user calibration of an inertial measurement unit without external equipment," *Measurement Science and Technology*, vol. 19, no. 8, p. 085 202, 2008.

[89] A. Maria and A. Sabatini, "A wavelet-based bootstrap method applied to inertial sensor stochastic error modelling using the allan variance," *Measurement Science and Technology*, vol. 17, no. 11, pp. 2980–2988, 2006.

[90] B. Wagstaff, V. Peretroukhin, and J. Kelly, "Robust data-driven zero-velocity detection for foot-mounted inertial navigation," *IEEE Sensors J.*, vol. 20, no. 2, pp. 957–967, Jan. 2020.

[91] M. S. Amin, M. B. I. Reaz, S. S. Nasir, M. A. S. Bhuiyan, and M. A. M. Ali, "A novel vehicle stationary detection utilizing map matching and imu sensors," *The Scientific World Journal*, vol. 2014, pp. 1–13, 2014.

[92] X. Meng, H. Wang, and B. Liu, "A robust vehicle localization approach based on gnss/imu/dmi/lidar sensor fusion for autonomous vehicles," *Sensors*, vol. 17, no. 9, 2017.

[93] Z. F. Syed, P. Aggarwal, C. Goodall, X. Niu, and N. El-Sheimy, "A new multi-position calibration method for mems inertial navigation systems," *Measurement Science and Technology*, vol. 18, no. 7, pp. 1897–1907, 2007.

[94] F. Ferraris, U. Grimaldi, and M. Parvis, in *Procedure for effortless in-field calibration of three-axial rate gyro and accelerometers*, 1995.

[95] I. Frosio, F. Pedersini, and N. Borghese, "Autocalibration of mems accelerometers," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 6, pp. 2034–2041, 2009.

[96] S. P. Won and F. Golnaraghi, "A triaxial accelerometer calibration method using a mathematical model," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 8, pp. 2144–2153, 2010.

[97] J. Metge, R. Mégret, A. Giremus, Y. Berthoumieu, and T. Décamps, "Calibration of an inertial-magnetic measurement unit without external equipment, in the presence of dynamic magnetic disturbances," *Measurement Science and Technology*, vol. 25, no. 12, p. 125 106, 2014.

[98] P. Zhang, J. Gu, E. Milios, and P. Huynh, "Navigation with imu/gps/digital compass with unscented kalman filter," *IEEE International Conference Mechatronics and Automation*, vol. 3, pp. 1497–1502, 2005.

[99] I. Skog and P. Händel, "Calibration of a mems inertial measurement unit," in *in Proc. XVII IMEKO WORLD CONGRESS, (Rio de Janeiro)*, 2006.

[100] *Mems inertial measurement unit calibration*, http://www.fivebyfivedevelopment.com/Downloads/IMU%20Calibration.pdf.

[101] *Geoplaner*, https://www.geoplaner.com/.

[102] *Surface gravity prediction*, https://geodesy.noaa.gov/cgi-bin/grav_pdx.prl.

[103] O. Sarkka, T. Nieminen, S. Suuriniemi, and L. Kettunen, "A multi-position calibration method for consumer-grade accelerometers, gyroscopes, and magnetometers to field conditions," *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3470–3481, 2017.