A Loosely Coupled GNSS/PDR Integration Approach for Pedestrian Navigation

by

Connor Steele Jones

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of science

Auburn, Alabama
August 5, 2023

Keywords: Pedestrian Dead Reckoning, Step Length Estimation, Misalignment Estimation, Magnetometer Fault Detection

Approved by

David Bevly, Chair, Professor of Mechanical Engineering
Howard Chen, Research Professor of Mechanical Engineering
Chad Rose, Assistant Professor of Mechanical Engineering

Abstract

Pedestrian navigation systems embody small, lightweight hardware, techniques, and biomechanical information to provide positioning information in an architecture that substitutes, or supplements, traditional systems such as GNSS. Two unique challenges of pedestrian navigation systems are the hardware size and weight constraints to keep the user comfortable, while the other challenge is the ability to provide accuracy and stability of the navigation solution in certain environments. Existing systems rely on the availability of small, lightweight, GNSS and inertial hardware for position, velocity, and attitude information in challenging environments. This thesis presents methods of integrating GNSS with a torso-mounted IMU to estimate three physical parameters of the user and the system hardware as a means of providing longer stability of the position, velocity, and attitude estimates when GNSS is no longer available. The presented methods will include showcasing a method of estimating the user's step length with existing models, as well as a new model; a method of hardware misalignment compensation for heading estimation; and an approach to detecting erroneous magnetometer measurements to reduce errors in the user's heading. GNSS is utilized in conjunction with the IMU to provide discrete step length pseudo-measurements for the user's step length estimation; GNSS course measurements will be used to estimate heading misalignment between the user and the IMU; and, finally, a threshold metric of the magnetometer measurements is used to compensate for errors in heading that would occur from perturbed magnetic field measurements. Performance analyses of each method is shown using real data with simulated GNSS outages. The methods are implemented using IMU data from a Vectornav 9-DoF VN-100 and a Ublox EVK-7 GNSS receiver for some data sets, and a Ublox ZED-F9P GNSS receiver for other data sets. Conclusions drawn from results of each method implementation are discussed and summarized.

Acknowledgements

First and foremost, I would like to thank my parents Frank and Debbie for their love, support, and motivation. I cannot thank them enough for all they have done to help me accomplish my goals and surmount the challenges I've faced in my life. I will never be able to repay them for all the love they have shown me, though I will forever strive to do so. This thesis is dedicated to them. I would also like to thank my younger sister Claire for helping to shape me into the person I am today, and always making herself available to talk.

I would like to thank all the teachers, coaches, and role models in my academic, and personal, career for everything they showed me to help me succeed in life. I credit my high school Engineering teacher Scottie Wilson for introducing the world of Engineering and how to approach solving unique problems. Her passion for shaping young minds has added my name to the long list of students that owe her a debt of gratitude.

I must recognize the many coworkers I have had the pleasure of working with during my time in the GAVLAB. All of my achievements and successes in the lab are, directly, attributed to these men and women. The first person I must thank is Dr. Howard Chen, who took me under his wing to mentor me always make himself available for any assistance I needed in the lab. A debt of gratitude goes to Brendan Schretter for his help with using the Lincoln MKZ when I was put on my first project. Similarly, I would like to thank John David Sprunger for helping me expand my knowledge of computer science to reduce the amount of time I would have spent on Stack Exchange. I would also like to thank Andrew Weir for his assistance with collecting data and being very patient with me. I would like to thank those that have shared an office space with me for offering helpful insight and answering my many questions. I would also like to thank Sam Morgan and Tyler Long for humoring me with intense games of ping pong while writing this thesis.

I would like to thank my friends Kevin Ambrose and Joseph Naro for putting up with me as a roommate, and offering aid while writing this thesis. What began as roommates, blossomed into a true friendship that I will always cherish. Similarly, I would like to thank Jake Ward and Tahn Thawainin for opening up their homes to me as a temporary roommate while finishing this thesis.

I would like to thank my former music instructors Dan Hermany and Erath Old for sparking a love of piano and guitar that supplied a critical creative outlet to help reduce stress. I would also like to thank Matt Boler for his many discussions of guitars and introducing me to artists I would not have been exposed to.

I would also like to thank my committee member Dr. Chad Rose for reviewing this document and advising me during my time in graduate school.

Finally, I would like to thank Dr. Bevly for the opportunity to work in the GAVLAB, as it has changed the course of my life forever. Dr. Bevly and Dr. Martin have exposed me to new ways of thinking and solving complex problems, that I would have never been able to experience otherwise.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

The problem of navigation has existed since the dawn of mankind. The goal of navigation is to answer two important questions: *where am I* and *where am I going* [1]. The former question is referred to as the problem of *positioning*, where one must determine a known location on Earth [2]; while the latter question refers the problem of *guidance*, or the idea of planning a path, or trajectory, to move to a new location. All advancements in science and technology of this field are motivated by solving these problems.

Many modern advancements and applications require reliable information to solve the problems of positioning and guidance such as: marine travel for ships, orbit determination and characteristics of celestial bodies and hardware, and pedestrian navigation. The study of pedestrian navigation has become a major area of interest in the last couple decades for it's many pragmatic applications. These applications are able to provide various means of assistance to individuals like the blind and disabled community, geolocation of dismounted ground soldiers, or even tracking first responders in emergencies. However, these applications come with many difficulties and limitations. One such limitation is the degradation of reliable GNSS signals in many urban and rural environments. For users that are indoors or in dense foliage, this degraded signal can lead to errors in both positioning and guidance that could have detrimental, or fatal, results. Other limitations such as satellite outages in times of war, or signal jamming and spoofing can also greatly hinder the accuracy of these

systems. With these limitations in mind, many alternative navigation methods have been explored to help provide accurate solutions under various constraints.

As the technology of various handheld devices, such as smartphones, has advanced, the ability to implement these alternative navigation systems has become very attractive. Modern smartphones are equipped with sensors like a global positioning system (GPS) receiver that provides an individual with a reliable position solution with errors that remain bounded over time. However, smartphone users are often operating in environments that can hinder the accuracy and reliability of these GNSS signals and require external information, independent of GNSS signals, to improve the desired navigation solution. These external sources can be maps of the existing environment and often use signals or visual data to identify a user's location in reference to known landmarks within the environment. These signals can be from bluetooth low-energy (BLE), wireless local area network (WLAN), ultra-wideband radios (UWB), and even radio-frequency identification (RFID). Incorporation of these technologies into existing navigation systems can be valuable supplements or replacements for GNSS signals when degraded or unavailable. However, much like the problems facing GNSS signals, these systems are subject to vulnerabilities, blockages, and other factors that contribute to a poor signal environment. Inclusion of map data also proves to be very costly and can even be time-consuming when being installed. Therefore, the desire to incorporate alternative techniques without these physical infrastructure references is very attractive.

A technique that operates independently of externally-referenced information is the technique of dead reckoning. In pedestrian navigation, dead reckoning involves propagating a user's position and orientation through the use of accelerometers and gyroscopes within an Inertial Measurement Unit (IMU), and magnetometers worn by the pedestrian. With recent advancements in technology, using inertial and magnetic sensors for pedestrian navigation purposes have been made possible. Mainly, improvements in the processing power of mobile computers for real-time implementations [3], and the advancements in sensor technology to reduce the size and power consumption of the inertial and magnetic sensors to be worn by a pedestrian [2]. These advancements have paved a path for implementing the dead reckoning technique on a pedestrian without the need for external information about the pedestrian's environment.

However, this technique is faced with challenges that limit the accuracy of a pedestrian navigation solution. The first challenge stems from the need of high quality IMU and mag-

netometer measurements. While advancements in technology have allowed for sensors to be worn on the body, erroneous measurements can compound over time and lead to a degraded navigation solution. To aid this problem, research has been conducted to supplement these measurements with models and assumptions of the human walking motion [4, 5, 6, 3, 7, 8]. Application of these walking motion models and assumptions to supplement inertial sensors are based on the anatomical location of the sensors. Primarily, whether the inertial sensors are mounted to the pedestrian's foot, or body [4]. In the case of body mounted sensors, the navigation solution is constructed by using a model of the pedestrian's step length and propagating position during the occurrence of a new step [7, 9, 10, 11]. These step length models involve forming a mathematical relationship between the pedestrian's step length with various parameters and characteristics that are unique to an individual pedestrian, and can be difficult to construct in real-time implementations.

Another challenge faced with dead reckoning for pedestrian navigation is the need for position and orientation initialization information. Dead reckoning provides the necessary information of how a pedestrian's position and orientation changes from an original point in space, but is unaware of this needed initialization information. Existing literature has explored this problem by utilizing GNSS for position and velocity initialization, as well as, measurements of the Earth's gravitational force, magnetic field, transfer alignment from external reference, or even user defined inputs to initialize orientation [9, 8, 12, 13]. Providing the necessary initialization information allows for the exploitation of dead reckoning to construct a navigation solution without the need for any other external information.

The final challenge of dead reckoning for pedestrian navigation is the degradation of the pedestrian's estimated heading when hardware misalignment and disturbances in the local magnetic field are present. While magnetometers measurements can be used to determine heading with respect to magnetic north, these heading estimates are unique to the magnetometer, itself, and not the pedestrian. Simply put, the orientation of the magnetometer and the pedestrian are not always the same. However, even if the magnetometer is perfectly in line with the pedestrian's orientation, the existence of magnetic declination between true north and magnetic north will still lead to errors in heading over time [14]. Combining this issue of misalignment with ferrous materials that corrupt the measurements of the local magnetic field, introduce additional errors in heading that hinder the integrity of the navigation solution over time. Existing literature has explored methods of detecting disturbances

in local magnetic field measurements to avoid deviations in true heading estimates [14, 15, 16].

In short, pedestrian dead reckoning is plagued with the problem of smaller, less accurate inertial sensors being rigidly mounted to body parts that experience irregular motion, as well as, pedestrian's traveling in environments not conducive to traditional positioning methods reducing validity of the navigation solution. To combat the challenges faced with dead reckoning for pedestrian navigation, recent developments have aimed to integrate GNSS hardware with inertial and magnetic sensors to produce a fused navigation solution [2, 17, 18, 6, 19, 20]. With these developments, this thesis will explore the integration of GNSS with pedestrian dead reckoning, and the application of alternative methods in step length estimation, magnetometer misalignment estimation, and magnetic fault detection.

1.2 Research Objectives and Contributions

In general, this thesis aims to develop an integration framework of GNSS with PDR to improve the PDR solution when GNSS is no longer available. To develop this framework, this thesis will:

- Present a novel step length estimation model, and the use of GNSS to estimate these model parameters.
- Develop a method of using GNSS to estimate misalignment between the pedestrian and the magnetometer.
- Examine an alternative approach to detecting faulty magnetometer measurements.
- Evaluate the performance of the proposed system framework on real-world data.

1.3 Thesis Outline

This thesis will discuss the general principles of navigation and how those principles are applied to more specific and niche situations in the remaining chapters. Chapter 2 will discuss the general navigation background, and will focus on the various coordinate frames and the mathematical methods of transforming between these various frames. Chapter 3 will discuss the principles of inertial navigation and the Kalman Filter framework for state

estimation. Chapter 4 will discuss the current methods of pedestrian dead reckoning for foot mounted and torso mounted inertial sensors. Chapter 5 discusses the loosely coupled architecture of integrating GNSS measurements with the pedestrian dead reckoning techniques. Chapter 6 will present the proposed improvements on the integration of the torso mounted pedestrian dead reckoning techniques with GNSS. Chapter 7 will show the results of the system implementation. The final chapter will include a summary, conclusions, and future work.

Chapter 2

Coordinate Frames and Transformations

2.1 Introduction

In navigation systems, there is a requirement for a particular frame, or frames, of reference and the ability to transform between them in order to construct an optimal navigation solution. In particular, the ability to transform between the body frame of a pedestrian and the navigating frame to capture the dynamics of the user relative to the Earth. However, the dynamics of the user are rendered futile if they cannot be transformed into a navigation frame. This chapter will cover the various coordinate frames that are used and how to transform between them.

2.2 Coordinate Frames

This section will introduce the different coordinate frames used for navigation purposes. They are ordered by establishing the frame of a measurement device, then the frame of a body of interest, and then the global and local navigating frames. This order is meant to mimic the order of how information is passed when used to construct the navigation solution. Some additional global coordinate frames are provided to show the various reference frames that can be used when developing a navigation solution.

2.2.1 Sensor Frame

Sensors are often described as devices that detect, or measure, physical properties. Sensors that quantify these properties as a vector(s), indicate the measurement with a magnitude and direction in reference to a certain coordinate frame. For certain sensors, the measurements are with respect to the sensor itself. This reference frame is defined as the sensor frame, and describes the directional component of the vector quantities. This is often represented in a rectangular coordinate system (x, y, z). Consider a LiDAR sensor that measures a distant object's range relative to the sensor. The measurement is provided in the x, y, and z components of the range to describe the actual location of the object relative to the sensor, rather than just the magnitude of the distance away.

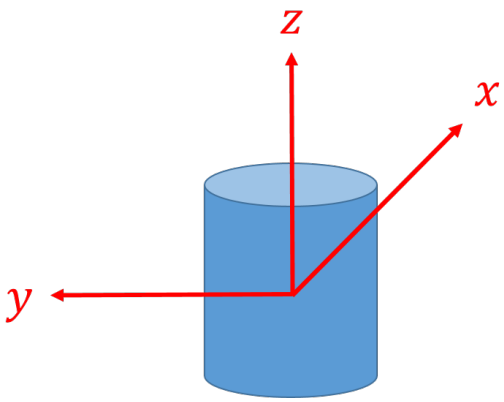


Figure 2.1: Sensor Frame

Figure 2.1 shows a generic cartesian sensor frame. Establishing the sensor frame allows for measurements to be quantified as a vector and becomes imperative when analyzing an object relative to the sensor or vice versa.

2.2.2 Body Frame

The body frame refers to the object or body of interest that system is providing a navigation solution for. In the case of pedestrian navigation, the body frame refers to the reference frame of the user and is free to translate and rotate with the user while in motion. This body frame of the user is typically defined where x is the forward facing direction of the user, y lies parallel to the user's shoulders and points in the direction of the user's right

shoulder, and z is given as the downward direction towards the user's feet when standing straight. Reference to the body frame will be denoted by the symbol b. In practice with inertial navigation it is common to align the axes of inertial device to correspond with the body frame, either physically or by means of an initial rotation during computation.

2.2.3 Local Tangent Plane Frame

As the body frame is free to move along in space, the navigation solution must resolve these dynamics into a frame relative to a global position on Earth. The common solution is to express a global solution in geodetic or geocentric latitude, longitude, and altitude. Figure 2.2 shows the relationship between geodetic and geocentric, where geocentric is fixed to the center of the earth and can be fully expressed by the latitude, longitude, and altitude of the object in question, whereas, geodetic is based on a tangent plane relative to the surface of the earth. The latter allows for the object in question to be represented by only latitude and longitude and does not change based on the object's altitude.

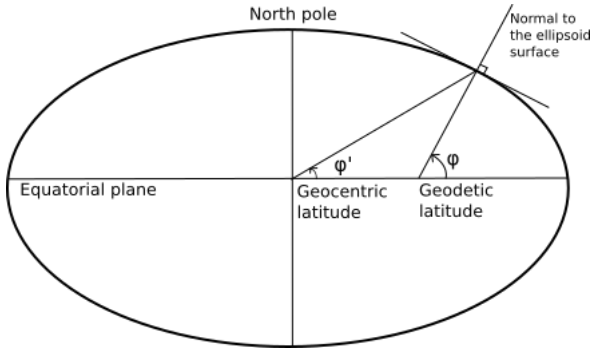


Figure 2.2: Geodetic and Geocentric Latitude

Another solution to resolving the body frame into a navigation frame is to use the local tangent plane and fix it to an initial global position along the geodetic reference ellipsoid. The local tangent plane allows for the dynamics of the user to be expressed relative to an initial point of reference. This navigation frame is denoted by the symbol n and is typically expressed in coordinates of North (x), East (y), and Down (z). This frame is referred to as the NED frame and allows the user's position and velocity to be transformed into motion along the tangent plane. However, resolving in this frame makes the assumption that dynamics are captured along a flat surface and not the spherical surface of the earth.

Therefore, resolving in this frame should only be used for the positions and velocities close to the origin of the local tangent plane to prevent growth of errors over large distances. The navigation frame is viable for any right-handed tri-axes such as: East-North-Up (ENU), North-West-Up, East-Down-South, etc. The two most common navigation frames are NED and ENU, however, for simplicity, the NED frame will be used as the navigation frame for the remainder of this thesis.

2.2.4 Global Frame

The global frame refers to the coordinate frame where positions and velocities are relative to the Earth. One such frame is the Earth-Centered Earth-Fixed (ECEF) frame denoted by the symbol e. This coordinate frame allows for a user to know his or her location with reference to any point on earth in the form of cartesian, or rectangular, coordinates as shown by Figure 2.3.

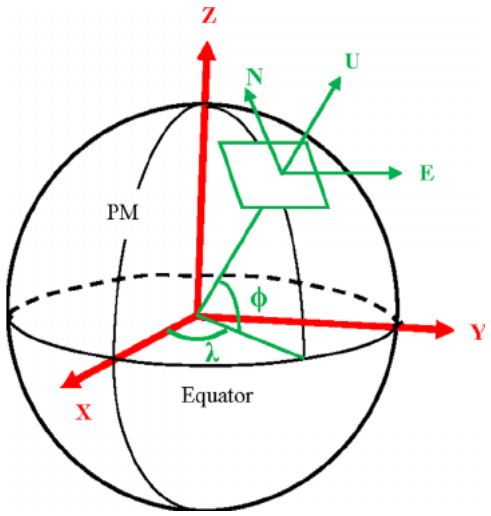


Figure 2.3: Earth-Centered Earth-Fixed

Where the z-axis is directed along the earth's axis of rotation from the center of the earth through the north pole and the x-axis is directed from the center of the earth through the intersection of the equator and the prime meridian. For objects of interest on the surface of the earth, this frame is stationary. However, for objects above the earth's surface such as aircraft and satellites, the reference frame is rotating at the earth's rate of rotation $\omega_{ie} =$

7.292115×10^{-5} rad/s. For pedestrian navigation, this reference frame is very attractive to resolve in, since most applications desire a navigation solution relative to the surface of the earth.

2.3 Coordinate Frame Transformations

As new data becomes available, it is important to be able to transform the data between the various frames to better understand the motion and orientation of the user. In the case of tracking a pedestrian, it can be very difficult to interpret the user's motion in a rectangular coordinate frame (i.e. ECEF frame); however, transforming the data into the local tangent plane can better capture the motion of the user and make the data easier to interpret. This conversion between frames is referred to as coordinate frame transformation and can be expressed in several ways.

2.3.1 Direction Cosine Matrix

One method of transforming the data between reference frames is to use the direction cosine matrix (DCM). A DCM allows a body's orientation, or attitude, to be represented with respect to another reference frame in the form of a matrix. This matrix representation allows for a vector given in a specified coordinate frame to be transformed to produce the same vector in a different reference frame by locating the unit vectors of the reference frame. For a 2 dimensional case, as shown in Figure 2.4, a vector given in one reference frame, denoted by the symbol $\boldsymbol{a}$, can be transformed into a new reference frame, denoted by $\boldsymbol{b}$, based on the rotation angle(s) between the two frames.

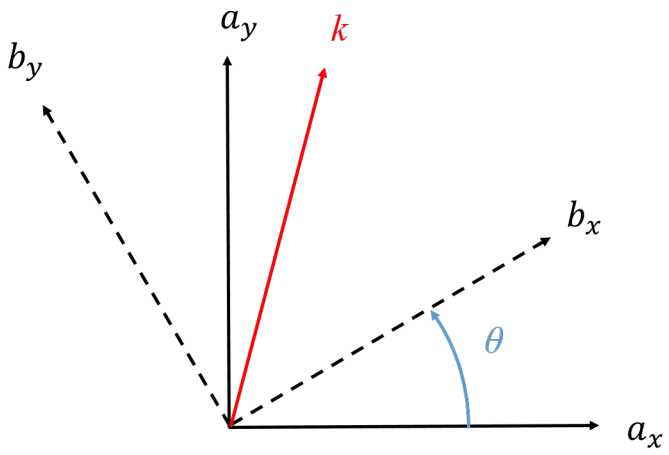


Figure 2.4: Vector resolved in two coordinate frames

The vector, $\boldsymbol{k}$, is given in frame $\boldsymbol{a}$ but can also be expressed in frame $\boldsymbol{b}$. In order to generate the DCM to perform this transformation, the angle of rotation between the two coordinate frames, θ, is used where,

$$k_x^b = k_x^a cos(\theta) + k_y^a sin(\theta) \tag{2.1}$$

$$k_y^b = k_y^a cos(\theta) - k_x^a sin(\theta) \tag{2.2}$$

This transformation of the vector, $\boldsymbol{k}$, into frame $\boldsymbol{b}$, can now be represented by a 2-dimensional square matrix that, when multiplied, gives the same vector within the rotated coordinate frame.

$$\begin{bmatrix} k_x^b \\ k_y^b \end{bmatrix} = C_a^b \begin{bmatrix} k_x^a \\ k_y^a \end{bmatrix} = \begin{bmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{bmatrix} \begin{bmatrix} k_x^a \\ k_y^a \end{bmatrix} \tag{2.3}$$

The generated DCM from frame $\boldsymbol{a}$ to frame $\boldsymbol{b}$, C_a^b, does not require any information about the magnitude of vector $\boldsymbol{k}$, and is able to produce the transformed vector using only the angle of rotation between the two coordinate frames. If the vector was to be rotated back into frame $\boldsymbol{a}$, the inverse of the DCM would need to be taken, which could be found by taking the transpose of the matrix:

$$C_b^a = (C_a^b)^{-1} = (C_a^b)^T \tag{2.4}$$

For the example given above, the DCM to rotate the vector back into frame $\boldsymbol{a}$ would be determined to be,

$$\begin{bmatrix} k_x^a \\ k_y^a \end{bmatrix} = (C_a^b)^T \begin{bmatrix} k_x^b \\ k_y^b \end{bmatrix} = \begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{bmatrix} \begin{bmatrix} k_x^b \\ k_y^b \end{bmatrix} \tag{2.5}$$

This example is a very simplified explanation and only looks at a case with a 2-dimensional frame. Most real-world implementations look to generate the DCM for a 3-dimensional case to track the orientation of a body relative to the earth. In the case of finding a body's orientation in the navigation frame (NED, ENU, ECEF, ECI, etc.), the DCM would need to be constructed,

$$\vec{x}^n = C_b^n \vec{x}^b \tag{2.6}$$

However, this formulation can prove to be very difficult for 3-dimensions since knowledge of the rotation angle of each coordinate frame to the reference frame may not be inherently obvious.

2.3.2 Euler Angles

A slightly more intuitive method of transforming vectors between reference frames is through the use of *Euler Angles*. Euler Angles define a rotation as a consecutive rotation around three axes [21]. A rotation about the z-axis is defined as the yaw angle, a rotation about the y-axis is defined as the pitch angle, and a rotation about the x-axis is defined as the roll angle. Figure 2.5 depicts a non-sequential 'right-hand' rotation about each axis:

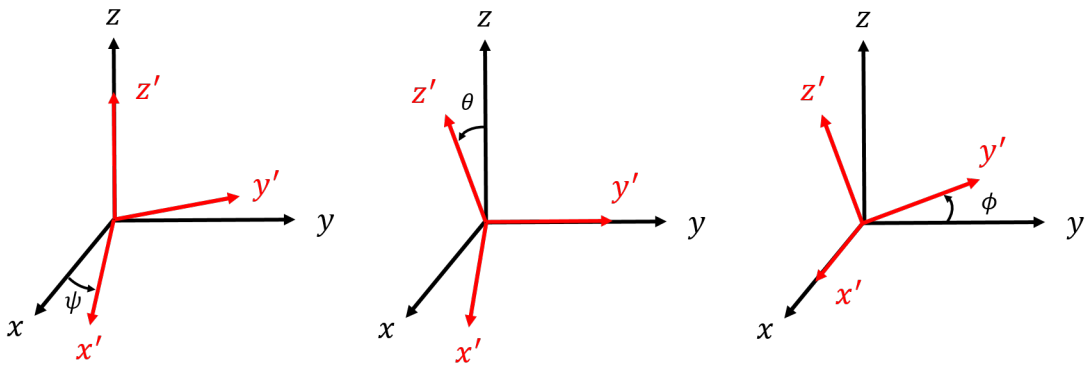
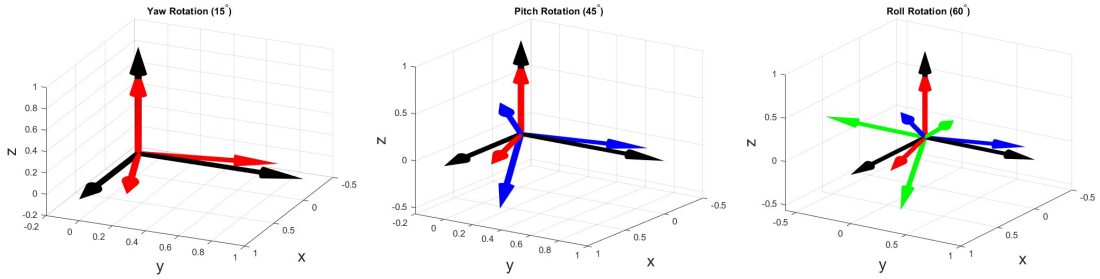


Figure 2.5: Euler angles in the convention (yaw, pitch, roll) or (ψ, θ, ϕ)

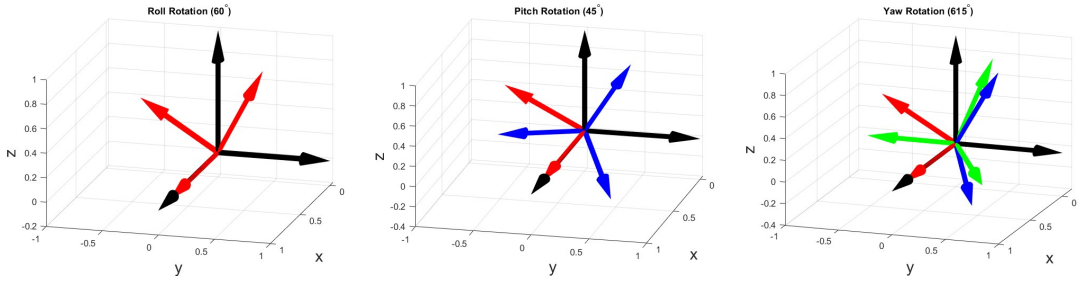
Euler angles are not a unique representation of three-dimensional orientation. As mentioned, Euler angles define orientation by a sequential rotation about each axis. However, the order in which the rotation sequence is performed will determine the final orientation. In other words, the final orientation of an Euler angle rotation for one sequence will be different than the final orientation of an Euler angle rotation for a different sequence. For example, Figure 2.6 shows the sequential rotation of three-dimensional axes by an initial yaw angle rotation of 15° (red), followed by a pitch angle rotation of 45° (blue), and a final roll angle rotation of 60° (green). This rotation sequence is referred to as a ZYX rotation where the initial frame (black) is rotated into the final frame (green).



(a) Yaw Rotation (b) Pitch Rotation (c) Roll Rotation

Figure 2.6: Euler Angle Orientation with ZYX Sequence

Now consider the same rotation angles performed in reverse order where the initial frame (black) is, first, rotated by a roll angle of 60° (red), followed by a pitch angle rotation of 45° (blue), and, finally, a yaw angle rotation of 15° (green). This rotation sequence is referred to as a XYZ rotation and is demonstrated in Figure 2.7.



(a) Roll Rotation (b) Pitch Rotation (c) Yaw Rotation

Figure 2.7: Euler Angle Orientation with XYZ Sequence

The final orientation for a ZYX rotation sequence is different than the final orientation for an XYZ rotation sequence. For the remainder of this thesis, the ZYX rotation sequence will be used.

While maintaining a constant rotation sequence is important, Euler angle representations are still subject to singularities. One singularity is the issue of angle wrapping. This problem occurs due to the fact that rotations of 0 radians and 2π radians about an axis are equivalent. For instance, a rotation of ($\psi = 0$, $\theta = 0$, $\phi = 0$) is equivalent to a rotation of ($\psi = 0$, $\theta = 0$, $\phi = 2\pi$). The other singularity of Euler angles is referred to as gimbal lock and occurs at pitch angles of $\theta = \pm\frac{\pi}{2}$ radians. This singularity poses a problem because the yaw and roll axes become aligned and a yaw, or roll, rotation becomes indistinguishable from the other.

2.3.3 Euler Angles and Direction Cosine Matrices

Euler angles can provide the necessary angles of rotation about each axis that are needed to produce the DCM as mentioned in section 2.3.1. Both methods of coordinate transformations can be used in conjunction to produce a DCM from one coordinate frame to another using the Euler angles. An individual DCM can be produced for the rotation about each axis using the angles (ψ, θ, and ϕ) from Figure 2.5. The DCM's produced are,

$$C_z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.7}$$

$$C_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{2.8}$$

$$C_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{2.9}$$

Recall in section 2.3.2 that the order for which Euler angle rotations occur is imperative for correctly performing the coordinate frame transformation. A common convention for performing this rotation is to do an ZYX rotation, or a rotation about the yaw axis, then the pitch axis, then the roll axis. To determine the full DCM to transform frame $\boldsymbol{b}$ into frame $\boldsymbol{n}$, the individual DCM's need to be multiplied in the same ZYX convention yielding a new DCM that is often referred to as a rotation matrix,

$$C_b^n = C_z(\psi)C_y(\theta)C_x(\phi) = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi & \sin\psi\sin\phi + \cos\psi\cos\phi\sin\theta \\ \cos\theta\sin\psi & \cos\psi\cos\phi + \sin\psi\sin\theta\sin\phi & \cos\phi\sin\psi\sin\theta - \cos\psi\sin\phi \\ -\cos\theta\sin\phi & \sin\theta & \cos\theta\cos\psi \end{bmatrix} \tag{2.10}$$

The rotation matrix shown above is denoted as C_b^n to indicate a transformation of vectors in the body frame into the navigation frame. To transform vectors from the navigation frame to the body frame the transpose, or inverse, of the matrix in equation 2.11 needs to be taken. Extracting the corresponding Euler angles from the ZYX rotation matrix can be accomplished by:

$$\psi = \textbf{atan2}\left(\frac{C_n^b(3,2)}{C_n^b(3,3)}\right) \tag{2.11}$$

$$\theta = -\textbf{arcsin}\left(C_n^b(3,1)\right) \tag{2.12}$$

$$\phi = \textbf{atan2}\left(\frac{C_n^b(2,1)}{C_n^b(1,1)}\right) \tag{2.13}$$

It is also possible to construct a rotation matrix that transforms vectors from one coordinate frame to another with an intermediate frame in between. One such case could be to transform vectors from the body frame to the global frame by defining a rotation matrix

from the body frame to the local tangent plane and then a transformation from the local tangent plane to the global reference frame. This would be performed by,

$$C_b^e = C_b^n C_n^e \tag{2.14}$$

This allows for transformations from the body frame $\boldsymbol{b}$ to the earth frame $\boldsymbol{e}$ to be performed without the need to construct a single rotation matrix and, instead, use existing rotation matrices to produce the equivalent transformation.

2.3.4 Quaternions

The final method for which a coordinate frame transformation can be expressed is through the use of *quaternions*. Quaternions were first introduced in 1843 by the mathematician William R. Hamilton and were used to describe mechanics in three-dimensional space by a generalization of complex numbers [22]. The common formulation of a quaternion is by:

$$q = q_0 + q_1 i + q_2 j + q_3 k \tag{2.15}$$

When using quaternions to describe spacial orientation and rotations, the unit quaternion defines an angle of rotation about a unit axis and can be written in the form:

$$\vec{q} = \begin{bmatrix} \cos\left(\frac{\phi}{2}\right) \\ \mathbf{u} \cdot \sin\left(\frac{\phi}{2}\right) \end{bmatrix} \tag{2.16}$$

where the terms ϕ represents the real part that expresses an angle of rotation in 3-D space, and the terms $\mathbf{u} = u_x i + u_y j + u_z k$ is the unit vector that generates an axis by which the rotation is performed [23]. Unlike Euler angles, quaternions have no singularities for which certain aspects of orientation become indeterministic making them very attractive for real-world implementations. However, quaternions are not very intuitive and come with their own set of properties such as: quaternions products, conjugates, and normalization.

A quaternion product, denoted by the symbol $\otimes$, is the product of two quaternions in the typical formulation (1), and is expressed in vector notation as:

$$\mathbf{q} \otimes \mathbf{p} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \otimes \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} q_0p_0 - q_1p_1 - q_2p_2 - q_3p_3 \\ q_1p_0 + q_0p_1 + q_3p_2 - q_2p_3 \\ q_2p_0 - q_3p_1 + q_0p_2 + q_1p_3 \\ q_3p_0 + q_2p_1 - q_1p_2 + q_0p_3 \end{bmatrix} \tag{2.17}$$

This expression for the quaternion product does reveal that this operation is not commutative:

$$\mathbf{q} \otimes \mathbf{p} \neq \mathbf{p} \otimes \mathbf{q} \tag{2.18}$$

However this non-commutative aspect is limited to events where the two quaternions are both real and have no imaginary parts, or if both quaternions have imaginary parts that are parallel [23].

The quaternion conjugate, denoted by the symbol $*$, is the same form as (1) but has the opposite sign on the imaginary parts and is defined as:

$$\mathbf{q}^* = q_0 - q_1i - q_2j - q_3k = \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix} \tag{2.19}$$

This operation reveals how the quaternion product of a quaternion and it's conjugate produce a quaternion with no imaginary parts and is referred to as a *real* quaternion:

$$\mathbf{q} \otimes \mathbf{q}^* = \mathbf{q}^* \otimes \mathbf{q} = \begin{bmatrix} q_0^2 + q_1^2 + q_2^2 + q_3^2 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.20}$$

The quaternion normalization is the process of scaling a quaternion vector into a unit vector and is defined as:

$$\| \mathbf{q} \| = \sqrt{\mathbf{q} \otimes \mathbf{q}^*} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} \tag{2.21}$$

While quaternion properties and operations come with a different set of procedures than that of rotation matrices, they have similar identities when being used to express

orientation. Show below are some properties of rotation matrices and the corresponding quaternion properties:

$$\mathbf{R}_A^B = \left(\mathbf{R}_B^A\right)^T \leftrightarrow \vec{\mathbf{q}}_A^B = \left(\vec{\mathbf{q}}_B^A\right)^* \tag{2.22}$$

$$\mathbf{R}_C^A = \mathbf{R}_B^A \mathbf{R}_C^B \leftrightarrow \vec{\mathbf{q}}_C^A = \vec{\mathbf{q}}_B^A \otimes \vec{\mathbf{q}}_C^B \tag{2.23}$$

$$\mathbf{R}_B^A = \mathbf{R}_C^A \left(\mathbf{R}_C^B\right)^T \leftrightarrow \vec{\mathbf{q}}_B^A = \vec{\mathbf{q}}_C^A \otimes \left(\vec{\mathbf{q}}_C^B\right)^* \tag{2.24}$$

$$\mathbf{R}_B^A = \left(\mathbf{R}_C^A\right)^T \mathbf{R}_B^C \leftrightarrow \vec{\mathbf{q}}_B^A = \left(\vec{\mathbf{q}}_A^B\right)^* \otimes \vec{\mathbf{q}}_C^B \tag{2.25}$$

$$\vec{x}^A = \mathbf{R}_B^A \vec{x}^B \leftrightarrow \begin{bmatrix} 0 \\ \vec{x}^A \end{bmatrix} = \vec{\mathbf{q}}_B^A \otimes \begin{bmatrix} 0 \\ \vec{x}^B \end{bmatrix} \otimes \left(\vec{\mathbf{q}}_B^A\right)^* \tag{2.26}$$

Further details and proofs of quaternion properties can be found in [23]. As mentioned above, quaternion representations are not always an intuitive means of interpreting orientation. Luckily, Euler angles can be extracted from quaternions by means of the conversion:

$$\psi = \mathbf{atan2}\left(2q_w q_z + q_x q_y,\, q_w^2 + q_x^2 - q_y^2 - q_z^2\right) \tag{2.27}$$

$$\theta = \mathbf{asin}\left(2q_w q_y - 2q_z q_x\right) \tag{2.28}$$

$$\phi = \mathbf{atan2}\left(2q_w q_x + 2q_y q_z,\, q_w^2 - q_x^2 - q_y^2 + q_z^2\right) \tag{2.29}$$

The conversion allows for quaternion operations and representations to be expressed in the form of Euler angle rotations of a specific sequence. Quaternions can also be converted into a rotation matrix, however, this is still dependent on the sequence of the construction of the rotation matrix. Since the remainder of this thesis is based on the ZYX sequence for constructing the rotation matrix, the conversion from a quaternion to a rotation matrix will be given as:

$$\mathbf{R} = \begin{bmatrix} 2q_w q_z + 2q_x^2 & q_w^2 - q_x^2 + q_y^2 - q_z^2 & 2q_z q_y - 2q_w q_x \\ q_w^2 + q_x^2 - q_y^2 - q_z^2 & 2q_y q_x - 2q_w q_z & 2q_w q_y + 2q_z q_x \\ 2q_w q_y - 2q_x q_z & -2q_w q_x - 2q_y q_z & q_x^2 - q_w^2 + q_y^2 - q_z^2 \end{bmatrix} \tag{2.30}$$

Chapter 3

Inertial Navigation

3.1 Introduction

Navigation techniques and systems utilize various methods, or devices, to determine an object of interest's position, velocity, and direction of travel. While navigation systems are developed with various sensors and devices, the methods used for navigation techniques are, typically, based around two fundamental categories: position fixing and dead reckoning [2]. Position fixing involves the determination of an object's position based on the knowledge of landmarks at known locations. This can be as simple as referencing a map of physical terrain or infrastructure of a specific region or environment. For wilderness explorers, comparing distances from physical features such as: rivers, mountains, and valleys with maps of the physical environment can provide a relative position with respect to these landmarks. A more precise and technical approach computes ranges and/or bearings to known objects based on the time of arrival (ToA), received signal strength (RSS), or angle of arrival (AoA) to determine an object's position via trilateration. One such example of this precise position fixing is through GNSS [2, 24].

Dead reckoning involves tracking an object's position and orientation based on linear and angular motion summation from a known reference point. This is commonly achieved by either measuring the change in position or integrating measured linear and angular velocity over time [2]. An intuitive example is comparing the change in a car's odometer when traveling from one place to another.

Both navigation techniques offer advantages and disadvantages for producing a navigation solution. Position fixing can allow for accurate position determination, but this is dependent on prior knowledge of the surrounding environment (e.g. mapping data), as well as, existing infrastructure. Dead reckoning is often self contained and requires no pre-existing knowledge of the surrounding environment. However, this is dependent on the quality of measurements that can result in errors over time. Also, since dead reckoning just determines a displacement in position and orientation, an initial absolute position and orientation must be provided in order to navigate in the global frame. That being said, dead reckoning is a very popular navigation technique based on it's advantages.

3.2 Continuous-Time Inertial Navigation

A common implementation of the dead reckoning methodology is through inertial navigation. An inertial navigation system (INS) produces a 3-dimensional dead reckoning solution from the sensor outputs of an inertial measurement unit (IMU). The IMU usually consists of three orthogonal accelerometers that measure specific forces ($\tilde{f}_{ib}^{b}$), and three orthogonal gyroscopes that measure angular velocity ($\tilde{\omega}_{ib}^{b}$). The fully realized dead reckoning solution from IMU measurements determines the body's position, velocity, and attitude states over time. In a continuous-time domain, the instantaneous change of these states is given by:

$$\dot{C}_{b}^{n} = C_{b}^{n}\tilde{\Omega}_{nb}^{b} \tag{3.1}$$

$$\dot{v}_{b}^{n} = \tilde{f}_{nb}^{n} \tag{3.2}$$

$$\dot{r}_{b}^{n} = v_{nb}^{n} \tag{3.3}$$

The term $\tilde{\Omega}_{nb}^{b}$ is the skew-symmetric matrix of the angular velocity measurements, as well as, the skew-symmetric matrix of the Earth's rotation with respect to an inertial frame and the transport rate of the local navigation frame with repect to the Earth. C_{b}^{n} is the rotation matrix from the body frame to the navigation frame, and $\tilde{f}_{nb}^{n}$ is the accelerometer measurements accounting for the force due to gravity.

When updating the position and velocity from accelerometer measurements, the IMU is not experiencing positional displacement. However, the accelerometer is measuring the reaction force exerted by the surface of the Earth due to the force of gravity. With this in mind, a gravity model is used to compensate for this effect. While the true measure of the

acceleration due to gravity is based on the IMUs location on Earth, a simplistic model in the local navigation frame can be defined by:

$$\gamma_{nb}^{n} = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} m/s^2 \tag{3.4}$$

The z-axis component is perpendicular to the surface of the Earth and pointed downwards. With the gravity model defined, the accelerometer measurements can be accounted for yielding:

$$\tilde{f}_{nb}^{n} = \tilde{a}_{nb}^{n} - \gamma_{nb}^{n} \tag{3.5}$$

The term $\tilde{a}_{nb}^{n}$ is the specific force measurements rotated into the local navigation frame. Integrating the time derivative of the navigation states to obtain the actual position, velocity, and attitude (PVA) is given by:

$$C_{b}^{n} = \int \dot{C}_{b}^{n} dt \tag{3.6}$$

$$v_{b}^{n} = \int \dot{v}_{b}^{n} dt = \int \tilde{f}_{nb}^{n} dt \tag{3.7}$$

$$r_{b}^{n} = \int v_{b}^{n} dt \tag{3.8}$$

This process stated above, provides the continuous-time domain for PVA determination from an IMU. These equations provide the foundation of an inertial navigation system.

3.2.1 INS Error Modeling

The accuracy of the INS's PVA determination is heavily dependent on the quality of the IMU measurements. A conceptual way of modeling the IMU measurements is with the inclusion of error terms:

$$\tilde{f}_{ib}^{b} = f_{ib}^{b} + b_a + w_a \qquad w_a \sim N(0, \sigma_a^2) \tag{3.9}$$

$$\tilde{\omega}_{ib}^{b} = \omega_{ib}^{b} + b_g + w_g \qquad w_g \sim N(0, \sigma_g^2) \tag{3.10}$$

The terms $\tilde{f}_{ib}^{b}$ and $\tilde{\omega}_{ib}^{b}$ are the measured specific force and angular velocity, f_{ib}^{b} and ω_{ib}^{b} are the true specific force and angular velocity, b_a and b_g are the accelerometer and gyroscope biases, and w_a and w_g represent zero-mean, Gaussian white noise of the accelerometer

and gyroscope measurements. It should be noted that these error models do not account for potential misalignment of the axes that may exist. However, these misalignments are, typically, removed by the manufacturer(s) during calibration and will be neglected in this thesis.

As the PVA estimates are computed over time, errors in the IMU measurements will cause the accuracy of these measurements to degrade over time too. The instantaneous change in time of the error in the PVA states is given by:

$$\delta\dot{\psi}^n_{nb} = C^n_b(b_g + w_g) \tag{3.11}$$

$$\delta\dot{v}^n_{nb} = -\left[f^n_{nb}\right]_\times \delta\psi^n_{nb} + C^n_b(b_a + w_a) \tag{3.12}$$

$$\delta\dot{r}^n_{nb} = \delta v^n_{nb} \tag{3.13}$$

The term $[f^n_{nb}]_\times$ is the skew-symmetric matrix of the specific force and the terms ($\delta\dot{\psi}$,$\delta\dot{v}$,$\delta\dot{r}$) represent the time derivative of the errors in attitude, velocity, and position. Integrating these terms with respect to time yields the PVA errors. The errors in the PVA states are related to the true PVA by:

$$C^n_b = \left(I_3 - [\delta\psi^n_{nb}]_\times\right)\hat{C}^n_b \tag{3.14}$$

$$v^n_{nb} = \hat{v}^n_{nb} + \delta v^n_{nb} \tag{3.15}$$

$$r^n_{nb} = \hat{r}^n_{nb} + \delta r^n_{nb} \tag{3.16}$$

The terms ($\hat{C}^n_b$,$\hat{v}^n_{nb}$,$\hat{r}^n_{nb}$) represent the estimated PVA from the INS in the navigation frame.

3.3 Discrete-Time Inertial Navigation

While the continuous-time INS equations (Equations (3.1) - (3.8)) and INS error equations (Equations (3.11) - (3.16)) provide the foundation for dead reckoning, these equations are not very suitable for sensors and computers that operate in discrete periods. To handle practical implementations of dead reckoning, the equations must be discretized based on the sampling periods of the IMU and the digital computer. The discretization of the PVA states is given by:

$$C^n_b(t + \Delta t) = C^n_b(t) + C^n_b(t)\tilde{\Omega}^b_{nb}\Delta t \tag{3.17}$$

$$v^n_{nb}(t + \Delta t) = v^n_{nb}(t) + \tilde{f}^n_{nb}(t)\Delta t \tag{3.18}$$

$$r^n_{nb}(t+\Delta t) = r^n_{nb}(t) + v^n_{nb}(t+\Delta t)\Delta t \tag{3.19}$$

The attitude update, $C^n_b(t+\Delta t)$, is computed with a first-order approximation. The same full state PVA discretization method can be applied to the INS's PVA errors by:

$$\delta\psi^n_{nb}(t+\Delta t) = C^n_b(b_g + w_g)\Delta t + \delta\psi^n_{nb}(t) \tag{3.20}$$

$$\delta v^n_{nb}(t+\Delta t) = \delta v^n_{nb}(t)\Delta t - [f^n_{nb}(t)]_\times \, \delta\psi^n_{nb}(t+\Delta t)\Delta t + C^n_b(b_a + w_a)\Delta t \tag{3.21}$$

$$\delta r^n_{nb}(t+\Delta t) = \delta r^n_{nb}(t) + \delta v^n_{nb}(t+\Delta t)\Delta t \tag{3.22}$$

3.4 Discrete-Time Linear Kalman Filter

The Kalman filter was first introduced in 1960 by a Hungarian-American engineer named Rudolf E. Kalman, who developed a recursive solution to the discrete-data filtering problem [25]. The Kalman filter has many applications surrounding navigation and controls of vehicles such as aircraft, marine ships, and ground vehicles. Due to innovations in digital computing, the implementation of the discrete-time Kalman filter has gained major popularity. The goal of this algorithm is to estimate various states of a discrete-time linear dynamic process from measurements observed in time. This discrete linear process is given by:

$$\mathbf{x}_{k+1} = \mathbf{\Phi}_k \mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_{d,k} \tag{3.23}$$

The terms $\mathbf{x}_{k+1}$ is the $(n \times 1)$ state vector at time t_{k+1}, $\mathbf{\Phi}_k$ is the $(n \times n)$ state transition matrix at time t_k that relates the states at time t_k to the states at time t_{k+1}, $\mathbf{B}$ is the $(n \times 1)$ vector relating the control inputs, $\mathbf{u}_k$, to the states at time t_k. For this thesis, the application of the Kalman filter is used to estimate INS errors and no inputs are included in the process model. The term $\mathbf{w}_{d,k}$ is the $(n \times 1)$ stochastic discrete process noise input at time t_k. The process noise input is a random variable with a zero-mean and Gaussian probability distribution

$$p(\mathbf{w}_{d,k}) \sim N(0,\, \mathbf{Q}_{d,k}) \tag{3.24}$$

$$\mathbf{Q}_{d,k} = E[\mathbf{w}_{d,k}\mathbf{w}^T_{d,k}] \tag{3.25}$$

The term $\mathbf{Q}_{d,k}$ is the $(n \times n)$ discrete process noise covariance matrix.

With the process model defined, an observation model is needed to describe the relationship between the measurements and the desired states in discrete time. This observation

model is given as:

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \tag{3.26}$$

The terms $\mathbf{y}_k$ is the $(m \times 1)$ measurement vector at time t_k and $\mathbf{H}_k$ is the $(m \times n)$ observation matrix that map the measurements to the state vector as a linear relationship at time t_k. The term $\mathbf{v}_k$ is the $(m \times 1)$ stochastic measurement noise vector at time t_k. Like the process noise input, the measurement noise vector is also a random variable with a zero-mean and Gaussian probability distribution with a $(m \times m)$ measurement covariance matrix $\mathbf{R}_k$.

$$p(\mathbf{v}_k) \sim N(0,\, \mathbf{R}_k) \tag{3.27}$$

$$\mathbf{R}_k = E[\mathbf{v}_k \mathbf{v}_k^T] \tag{3.28}$$

The Kalman filtering algorithm is comprised of two major steps: the *time update* and the *measurement update*. The time update uses the estimates of the current states to 'predict' the estimates of the states a single time-step in the future, while the measurement update uses the measurements of the current states to 'correct' for errors in the predicted states of the time update. The equations for the time update are given below with the state estimate prediction and the state estimate covariance prediction:

$$\hat{\mathbf{x}}_{k+1}^- = \mathbf{\Phi}_k \hat{\mathbf{x}}_k \tag{3.29}$$

$$\mathbf{P}_{k+1}^- = \mathbf{\Phi}_k \mathbf{P}_k \mathbf{\Phi}_k^T + \mathbf{Q}_{d,k} \tag{3.30}$$

the propagation of the state estimate and state estimate covariance are denoted by the superscript $(-)$ to represent them as an estimate before correction by the measurement update. The measurement update computes a gain matrix that is used to correct the state estimate and the state estimate covariance. The equations for the measurement update are give as:

$$\mathbf{K}_k = \frac{\mathbf{P}_k^- \mathbf{H}_k^T}{\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k} \tag{3.31}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \tag{3.32}$$

$$\mathbf{P}_k = (I_{n \times n} - \mathbf{K}_k \mathbf{H}_k)\mathbf{P}_k^- \tag{3.33}$$

The term $\mathbf{K}_k$ is the $(n \times m)$ gain matrix and $I_{n \times n}$ is a square identity matrix. The gain matrix consists of scalars between 1 and 0 that can be thought of as a confidence metric of whether to place higher "trust" in the measurements or the model prediction.

The implementation of the Kalman filtering algorithm requires that an initial estimate of the state and state covariance be given where:

$$\hat{\mathbf{x}}_0 = \hat{\mathbf{x}}_{t0} \tag{3.34}$$

$$\mathbf{P}_0 = E[\hat{\mathbf{x}}_0 \hat{\mathbf{x}}_0^T]. \tag{3.35}$$

Once initialized, the time update is recursively performed until measurements are made available and the measurement update can make corrections to the state estimates. Further explanations of the Kalman filter and its applications are given in [25].

3.5 The Extended Kalman Filter

A drawback to the linear Kalman filter from Section 3.3.5 is that the system model and the measurement model are required to be linear in nature. In the presence of systems that are defined by very nonlinear dynamics, the linear Kalman filter cannot properly estimate the desired states. Therefore, an alternative to the linear Kalman filter is needed to estimate states of a system with nonlinear dynamics. One such method is the extended Kalman filter (EKF). Consider a system of nonlinear dynamics and a nonlinear observation model:

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{w}_{d,k}) \tag{3.36}$$

$$\mathbf{y}_k = h(\mathbf{x}_k, \mathbf{v}_k) \tag{3.37}$$

The function $f(\bullet)$ describes a nonlinear relationship between the current state and process noise at time t_k to the future state at time t_{k+1}. A different nonlinear function, $h(\bullet)$, describes the relationship between the state and measurement noise to the measurement, y, at time t_k.

With this in mind, the state covariance ($\mathbf{P}$) propagation is performed by linearizing the nonlinear function, $f(\bullet)$, in order for the measurement update to be performed. The linearization of the describing function is a first-order Taylor Series approximation, often referred to as a Jacobian, given by:

$$\mathbf{A}_k = \frac{\partial f}{\partial \mathbf{x}_k}(\mathbf{x}_k) \tag{3.38}$$

$$\mathbf{H}_k = \frac{\partial h}{\partial \mathbf{x}_k}(\mathbf{x}_k) \tag{3.39}$$

The term $\mathbf{A}$ is the first-order Taylor Series approximation of the nonlinear process model, $f(\bullet)$, and $\mathbf{H}$ is the first-order approximation of the the nonlinear observation model, $h(\bullet)$.

The same steps from the linear Kalman filter are applied to the EKF with the inclusion of the linearized process model and measurement model where the *time update* is given by:

$$\hat{\mathbf{x}}_{k+1}^{-} = f(\hat{\mathbf{x}}_k) \tag{3.40}$$

$$\mathbf{P}_{k+1}^{-} = \mathbf{A}_k \mathbf{P}_k \mathbf{A}_k^T + \mathbf{Q}_{d,k} \tag{3.41}$$

The term $\mathbf{Q}_{d,k}$ is the discrete process noise covariance matrix mapped onto each state. It should be noted that the process noise cannot always be represented as a scalar matrix, in which case, the Jacobian of the nonlinear process function will have to be found with respect to the process noise $\mathbf{w}$ at time t_k. The *measurement update* equations are given by:

$$\mathbf{K}_k = \frac{\mathbf{P}_k^{-}\mathbf{H}_k^T}{\mathbf{H}_k\mathbf{P}_k^{-}\mathbf{H}_k^T + \mathbf{R}_k} \tag{3.42}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^{-} + \mathbf{K}_k(\mathbf{y}_k - h(\hat{\mathbf{x}}_k)) \tag{3.43}$$

$$\mathbf{P}_k = (I_{n\times n} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^{-} \tag{3.44}$$

where $\mathbf{R}_k$ is the discrete measurement noise covariance matrix. Again this noise term cannot always be represented by a scalar matrix and the Jacobian of the nonlinear observation function will have to be found with respect to the measurement noise $\mathbf{v}$ at time t_k.

3.6 Error-State Kalman Filter for INS Aiding

The ability to dead reckon to estimate a position over time proves to be very beneficial when position fixing systems and methods are not readily available. However, as mentioned in Section 3.2.1, the quality of measurements within an INS directly affect the accuracy of the dead reckoned solution. This inherent problem allows for a method of fusing a dead reckoning system with a position fixing system to be implemented. The Kalman filter algorithms discussed in Sections 3.4 and 3.5 allow for such an implementation to exist. The Kalman filter framework provides a way of determining a PVA that minimizes the difference between the INS PVA and an aiding system's PVA based on the uncertainty within each system.

An intuitive approach to applying the Kalman filter framework is to use the full PVA as the states. However, this approach comes with the difficulty of the PVA states being

nonlinear in the propagation equations. One solution is the use of the EKF framework as shown in Section 3.5, or by using the PVA errors as the states of the Kalman filter in an approach known as the error-state Kalman Filter (ESKF) [5, 3]. The ESKF assumes the notion that the PVA errors are linear combinations of the state propagation equations as shown in Section 3.3. The states of the ESKF are given by:

$$\delta\mathbf{x} = \begin{bmatrix} \delta r^n_{nb} & \delta v^n_{nb} & \delta\psi^n_{nb} & b_a & b_g \end{bmatrix} \tag{3.45}$$

The terms for the accelerometer and gyroscope biases (b_a, b_g) are included in the state vector. The estimation of the IMU biases help to reduce errors within the measurements for the full PVA estimation.

The first step of the ESKF is to propagate the full PVA estimates forward in time with the IMU measurements that had the estimated biases removed where:

$$C^n_b(t+\Delta t) = C^n_b(t) + C^n_b(t)\left(\tilde{\Omega}^b_{nb} - b_g\right)\Delta t \tag{3.46}$$

$$v^n_{nb}(t+\Delta t) = v^n_{nb}(t) + \left(\tilde{f}^n_{nb} - b_a\right)\Delta t \tag{3.47}$$

$$r^n_{nb}(t+\Delta t) = r^n_{nb}(t) + v^n_{nb}(t+\Delta t)\Delta t \tag{3.48}$$

The accelerometer measurements have compensated for the reaction force due to gravity. The next step is to propagate the error state vector forward in time as given by Equation (3.29) where the state transition matrix, $\mathbf{\Phi}$, is given by:

$$\mathbf{\Phi} = \begin{bmatrix} I_3 & I_3\Delta t & 0_3 & 0_3 & 0_3 \\ 0_3 & I_3 & -\left[\tilde{f}^b_{nb}\right]_\times \Delta t & C^n_b\Delta t & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 & C^n_b\Delta t \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{3.49}$$

The rotation matrix, C^n_b, is the propagated full state estimate from Equation (3.46). The continuous process noise matrix is given as the variance of the IMU measurements, as well as, the variance in the biases where:

$$\mathbf{Q} = \begin{bmatrix} \sigma^2_a & 0_3 & 0_3 & 0_3 \\ 0_3 & \sigma^2_g & 0_3 & 0_3 \\ 0_3 & 0_3 & \sigma^2_{b_a} & 0_3 \\ 0_3 & 0_3 & 0_3 & \sigma^2_{b_g} \end{bmatrix} \tag{3.50}$$

with the continuous process noise being discretized and mapped onto the current states by:

$$\mathbf{Q}_d = \mathbf{G}\mathbf{Q}\mathbf{G}^T\Delta t \tag{3.51}$$

and a mapping matrix:

$$\mathbf{G} = \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ C_b^n & 0_3 & 0_3 & 0_3 \\ 0_3 & C_b^n & 0_3 & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{3.52}$$

Now that the states are being estimated, the next phase is providing a measurement update from the INS aiding system. The goal of the aiding system is to provide measurements that allow for reference insight into the PVA states being estimated. A common approach is to aid the INS with GNSS due to the simplicity of GNSS position and velocity measurements. This aiding method is commonly referred to as loosely coupled GNSS-INS integration [2, 19]. The measurement update takes the error of the measured full state position and velocity with the estimated full state position and velocity and compares it with the estimated error state position and velocity from the ESKF given by:

$$\delta y = \begin{bmatrix} \delta y_r \\ \delta y_v \end{bmatrix} = \begin{bmatrix} \left(r_{nG}^n - r_{nb}^b\right) - \delta r_{nb}^n \\ \left(v_{nG}^n - v_{nb}^n\right) - \delta v_{nb}^n \end{bmatrix} \tag{3.53}$$

Since the measurement update is not directly using a measurement of the INS errors, the measurement vector is denoted as δy. The subscript, G, represents measurements from the GNSS receiver and result in the measurement model given by:

$$\mathbf{H} = \begin{bmatrix} -I_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & -I_3 & 0_3 & 0_3 & 0_3 \end{bmatrix} \tag{3.54}$$

and a measurement noise matrix of:

$$\mathbf{R} = \begin{bmatrix} \sigma_{r_G}^2 & 0_3 \\ 0_3 & \sigma_{v_G}^2 \end{bmatrix} \tag{3.55}$$

While the time update and measurement update follow the same procedure as the EKF and the Linear KF, the ESKF requires the state estimates be set back to zero after the measurement update. This application does not take into account the offset between the IMU's location and GNSS receiver's location. For more accurate implementations this offset,

also known as lever arm and misalignment, will need to be taken into consideration. The algorithm presented above makes the assumption that both sensors are mounted in such a way that there is no misalignment or lever arm. This algorithm will be explored further in the next chapter with an alternative aiding system approach.

Chapter 4

Pedestrian Dead Reckoning

4.1 Introduction

Dead reckoning (DR) is a term used in navigation to describe the process of estimating the current position of an object by utilizing a previously determined position, or fix, in conjunction with the distance traveled and the direction of travel over a period of time [26]. This method of position estimation is most common with ships attempting to travel across oceans or hikers searching for a destination in remote regions where certain technologies are unavailable. In the case of tracking an individual person, this technique of dead reckoning is applied to capture the motion of the individual in a method known as Pedestrian Dead Reckoning (PDR). For a pedestrian, the knowledge of the distance and direction traveled relative to the Earth is not always well known or available, therefore, many scenarios of pedestrian tracking require some information about the pedestrian's movement and orientation. In practice, PDR allows the position, velocity, and attitude (PVA) of a pedestrian to be estimated by fusing data from small micro-electric mechanical system (MEMS) grade Inertial Measurement Units (IMU) with models of the pedestrian walking motion. The PVA navigation state estimates use information from the accelerometer and angular velocity measurements that make certain assumptions about the motion of the pedestrian based on the location of the IMU on the body. Advancements in the technology of smaller grade IMUs have made implementations of PDR more feasible and heavily explored area of research. This has led to a variety of techniques and algorithms are being explored to better capture

the pedestrian's motion to reduce errors produced by the smaller and lower quality IMUs. The rest of this chapter will cover an INS approach to dead reckoning for foot-mounted IMUs, as well as a step detection approach to dead reckoning for chest and torso mounted IMUs.

4.2 PDR-INS Approach

A popular approach to tracking a pedestrian's motion is through an Inertial Navigation System (INS) technique of dead reckoning. This technique uses a "beacon-free" method of propagating a pedestrian's position and orientation where no reference points or pre-existing knowledge of infrastructure is needed [5]. This is performed by mathematical integration of tri-axial accelerometer and gyroscope outputs from an IMU mounted on the foot of the pedestrian, which allows for a navigation solution to be constructed by estimating the PVA states. This mathematical integration is known as 6-DoF mechanization. However, as mentioned previously, these IMU measurements are prone to errors that can cause the PVA estimates to diverge, unbounded. In order to account for these errors and reduce the drift of the PVA estimates, several techniques are used that estimate errors in the navigation states, as well as estimate estimate sensor errors and biases. These techniques are:

1. Zero Velocity Update (ZVU)
2. Zero Angular Rate Update (ZARU)
3. Heuristic Drift Reduction (HDR)

The ZVU, or ZUPT, technique is performed by the assumption that the IMU is stationary and the linear velocity in any direction is zero. The ZARU technique is similar to the ZVU technique, except that the stationary assumption also has no angular velocity. The HDR technique is based on the assumption that the direction of motion is held constant and changes in the heading are solely due to sensor errors within the IMU.

For foot-mounted IMUs, the INS techniques can be applied by fusing the inertial measurements with a model of the pedestrian's walking pattern. The remainder of this section will introduce the Gait Cycle model of the human walking motion for identifying periods of application of the INS techniques, the PDR-INS system model, and an implementation and summary of the system.

4.2.1 Pedestrian Gait Cycle Events

The goal of PDR is to track a pedestrian as he or she moves from one point to another. To achieve this, the motion of the pedestrian must be taken into account. This ambulatory walking motion uses two legs providing support and propulsion in a fluctuating pattern that is cyclical in nature. Human gait is referred to as the individual's manner, or style, of walking and is more unique to a single pedestrian instead of an all encompassing method of locomotion.

Due to the cyclical nature of the human walking motion, this pattern is referred to as a gait cycle. One full gait cycle consists of when one event in the cycle occurs and is then repeated. While the gait cycle can examine the events at any point in the walking motion, it is much more intuitive to think of one gait cycle as being the time of one foot coming in contact with the ground to the time when that same foot strikes the ground again. For the remainder of this thesis the gait cycle will be expressed by the latter explanation and the foot that the cycle begins with will be referred to as the *leader* foot. As shown in Figure 4.1, a full gait cycle can be broken down into two major phases: the *stance* phase and the *swing* phase. The stance phase of the gait cycle consists of the events when the leader foot is in constant contact with the ground, and the swing phase consists of the events when the leader foot is swinging in the air to take the next step. Within the stance phase of the gait cycle there are several events that occur to define this phase:

1. **Loading Response**: Time of initial contact of the leader foot with the ground to the toe of the opposite foot leaving the ground.

2. **Mid-stance**: Time from the toe of the opposite foot leaving the ground to the heel rise of the leader foot.

3. **Terminal Stance**: Time from the heel rise of the leader foot to the initial contact of the opposite foot with the ground.

4. **Pre-swing**: The time from initial contact of the opposite foot with the ground to the point where the leader foot leaves the ground.

With the leader foot leaving the ground, the swing phase of the gait cycle begins. Similar to the stance phase, the swing phase can also be broken down into separate events:

1. **Initial Swing**: The time from the leader foot leaving the ground to the point where the feet are adjacent with one another.

2. **Mid-swing**: The time from the feet being adjacent to the point where shin of the leader foot is perpendicular to the ground.

3. **Terminal Swing**: The time from the shin of the leader foot being perpendicular to the ground to the point where the leader foot makes contact with the ground.

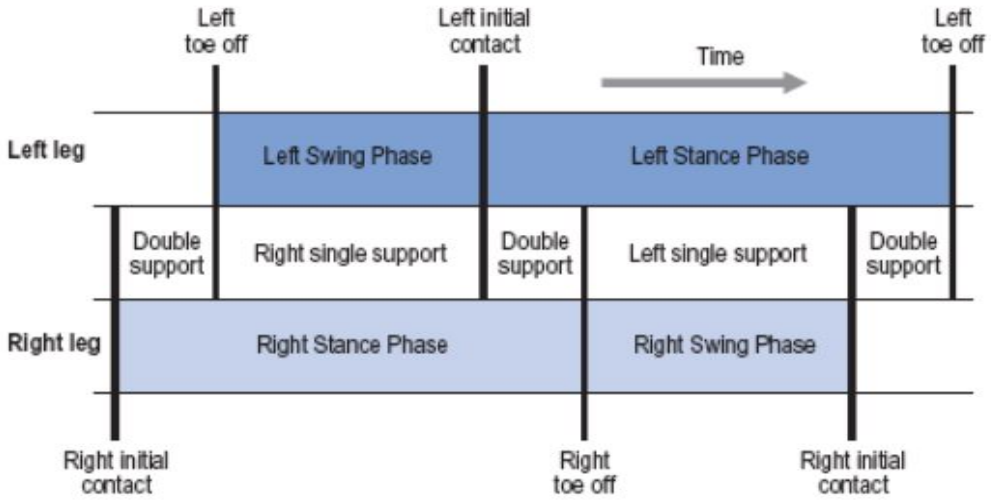


Figure 4.1: Stance and Swing phases of Gait Cycle [27]

There are periods during both phases where only one foot is in contact with the ground, and periods where both feet are in contact with the ground. During the periods where both feet are in contact with the ground, the body of the pedestrian is being supported by both feet in a period of *double support*. During this period of double support, one foot is positioned in front of the other having made initial contact with the ground and the other foot is positioned in the back about to be lifted off of the ground. When either foot is suspended in the air to swing forward for the next step, there is a period when only one foot is in contact with the ground. This period is known as *single support* where the pedestrian's body is being supported by only one foot.

When attempting to monitor and analyze the various phases of a pedestrian's gait cycle mentioned in the previous section, there are three cases that need to be considered. The

first case is the determination of the stance phase where the foot is in a stationary state during walking motion for the application of the ZVU technique, the second case is the determination of *still* periods where the pedestrian is stationary and not walking for the application of both the ZVU and ZARU techniques, and the final case is the determination of when the pedestrian's walking direction is held constant for the application of the HDR technique.

4.2.2 Stance Detection

As mentioned in the previous, the stance phase occurs when one or both of the feet are in contact with the ground or walking surface. During this phase, the linear velocity, as well as, the angular velocity of the foot is very nearly zero. However, due to errors within the IMU and the realization that the user's foot is never truly stationary, the ability to detect these periods of zero-velocity is an inherent problem. The detection of zero-velocity periods can be performed using either the accelerometer measurements or the gyroscope measurements from an IMU mounted on the pedestrian's foot [5, 28]. When using one, or both, of the measurements produced by the IMU, the magnitude of the measurements is needed by calculating the Euclidean norm given by:

$$\| \tilde{a}^b_{nb} \| = \sqrt{(\tilde{a}^b_{nb,x})^2 + (\tilde{a}^b_{nb,y})^2 + (\tilde{a}^b_{nb,z})^2} \tag{4.1}$$

$$\| \tilde{\omega}^b_{nb} \| = \sqrt{(\tilde{\omega}^b_{nb,x})^2 + (\tilde{\omega}^b_{nb,y})^2 + (\tilde{\omega}^b_{nb,z})^2} \tag{4.2}$$

The terms $\tilde{a}^b_{nb,x}$, $\tilde{a}^b_{nb,y}$, and $\tilde{a}^b_{nb,z}$ are the linear acceleration measurements along each axis, and $\tilde{\omega}^b_{nb,x}$, $\tilde{\omega}^b_{nb,y}$, and $\tilde{\omega}^b_{nb,z}$ are the angular velocity measurements along each axis. When the quantity of the angular velocity norm calculation is near zero, or when the norm of the acceleration calculation is near the acceleration due to gravity, the IMU can be assumed to be in a stationary state. The stance and swing phases of the pedestrian's gait cycle can be determined by applying this principle with the angular velocity norm. The stance phase can be seen from the point where the leader foot makes initial contact with the ground, then the foot's angular velocity flattens out near zero indicating that the foot is flat on the ground. From there, the swing phase begins as the leader foot is lifted off of the ground.

For implementation of a stance detection algorithm, certain conditions must be met by using a thresholding methodology proposed by Jiménez [5] and Pierce [29]. When consid-

ering both the accelerometer and gyroscope measurements, their are three conditions to satisfy a stance phase:

1. The magnitude of the normalized acceleration must be within a specified upper and lower bound for a window of measurements. The specified bounds are within small deviations of the acceleration due to gravity ($TH_{a,min} = 8.5 m/s^2$ and $TH_{a,max} = 11 m/s^2$). This can be written in the form of a logical statement:

$$C_1 = \begin{cases} 1 & \text{if} \quad TH_{a,max} > \| \tilde{a}^b_{nb} \| > TH_{a,min} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{4.3}$$

   If none of the acceleration norm measurements in the window are outside the upper or lower bounds, then the logical expression will return a 1 for *true*. If any of the measurements in the window are outside the upper and lower bounds, then the logical expression will return a 0 for *false*. A window size of 11 samples at a sampling rate of 100 Hz was used to produce a window roughly 0.1 seconds in length was chosen for this thesis and produced the best results for step detection.

2. The magnitude of the normalized angular velocity must be less than a specified upper bound for the window of measurements ($TH_{\omega,max} = 0.7\, rad/s$). This can be written in the form of a logical expression:

$$C_2 = \begin{cases} 1 & \text{if} \quad \| \tilde{\omega}^b_{nb} \| < TH_{\omega,max} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{4.4}$$

   Again, if none of the angular velocity norm measurements in the sample window are greater than the specified upper bound then the logical expression will return a 1 for *true*. The lower bound for the angular velocity norm measurements can be thought of as $0\, rad/s$ since angular velocity measurements magnitudes can never be less than zero. For the sake of simplicity, a lower bound is not specified. A window size of 11 samples at 100 Hz was also used for this condition.

3. The standard deviation of the acceleration norm must also be within a threshold. However, this threshold is determined as a function of the current mean and standard deviation of the acceleration norm measurements. This can be written in the form of

a logical expression:

$$C_3 = \begin{cases} 1 & \text{if} \quad (\parallel \tilde{a}^b_{nb} \parallel - \mu_{window}) \; < \; 1.5 \cdot \sigma_{window} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{4.5}$$

The terms μ_{window} and σ_{window} are the current mean and standard deviation of the acceleration norm measurements. A window size of 11 samples sampled at 100 Hz was also used for this condition.

If all three of the conditions (C_1, C_2, and C_3) are satisfied then the stance phase of the gait cycle is occurring and the ZVU technique can be applied.

4.2.3 Still Detection

Still periods where the pedestrian is stationary is not part of the gait cycle as mentioned in Section 4.2.1, since the gait cycle is based on the dynamic motion when walking. While it is intuitive to think that the pedestrian is motionless during stationary periods, the reality is that these periods are never truly motionless and there are errors within the sensor measurements. These errors in the sensor measurements will be discussed in later sections. Similar to the methodology described for *stance* detection, a thresholding technique specified by [5, 28] is used to satisfy two conditions to register a *still* period:

1. The magnitude of the angular velocity norm measurements must be less than a specified upper bound for the window of measurements ($TH_{\omega,max} = 0.2 rad/s$). This upper bound is much less than the upper bound set for the *stance* detection as mentioned in section 3.2.2. This can be written in the form of a logical expression:

$$C_4 = \begin{cases} 1 & \text{if} \quad \parallel \tilde{\omega}^b_{nb} \parallel < \; TH_{\omega,max} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{4.6}$$

   If none of the angular velocity norm measurements in the sample window are greater than the specified upper bound then the logical expression will return a 1 for *true*. A window size of 11 samples sampled at 100 Hz was used for this condition.

2. The standard deviation of the angular velocity norm measurements must be within a threshold that is a function of the current mean and standard deviation of the angular

velocity measurements. This can be written in the form of a logical expression:

$$C_5 = \begin{cases} 1 & \text{if} \quad (\parallel \tilde{\omega}_{nb}^{b} \parallel - \mu_{window} \; < \; 2 \cdot \sigma_{window} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{4.7}$$

where the terms μ_{window} and σ_{window} are the current mean and standard deviation of the angular velocity norm measurements. A window size of 11 samples sampled at 100 Hz was also used for this condition.

If both conditions (C_4 and C_5) are satisfied then a *still* period is occurring, allowing for the application of both the ZVU and ZARU techniques.

Both the *stance* phase detection and the *still* period detection methods were applied to a real-world data set where a pedestrian was fitted with a Vectornav VN-100 [30] mounted to the right foot. Data was then recorded of the pedestrian beginning in a static position and then walking around a parking lot. The results are shown below in Figure 4.2. For almost a full minute the pedestrian is standing still with both feet planted on the ground. Detection of this *still* period is represented by the black markers. Once the pedestrian begins walking, the *stance* periods were detected and shown with the green markers. This test was used to validate the performance of the *stance* and *still* phase detection algorithms and conditions presented above.

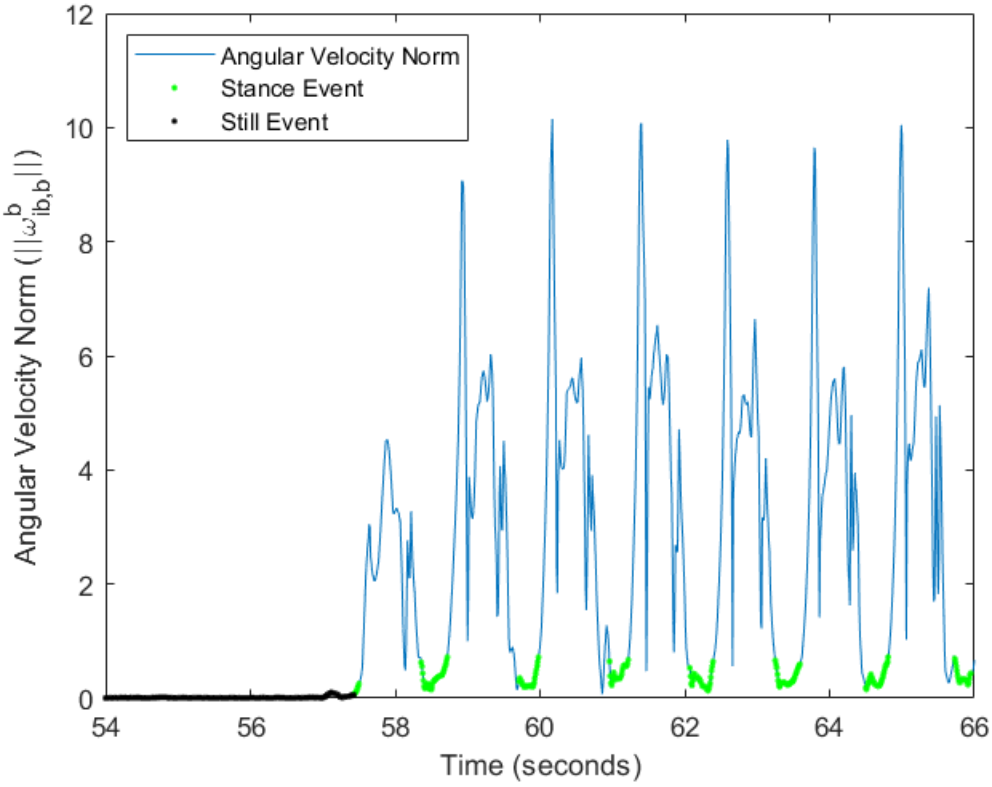


Figure 4.2: Still Period and Stance Phase Detection

4.2.4 Straight Walking Detection

As mentioned in Sections 4.2.2 and 4.2.3, the ability to detect periods of stance, swing, and stillness allow for errors within the IMU measurements to be reduced when determining the pedestrian's PVA during propagation. However, a critical element of this error reduction is determining the z-axis gyroscope bias for heading estimation. During the *still* period detection, the heading drift can be reduced by estimating the bias along the z-axis, however, if no *still* periods occur, then estimation of this bias becomes unachievable. In the case of a pedestrian walking along a corridor in a building or along the side of the road on a sidewalk, the assumption can be made that the direction of travel remains fairly constant. In order to reduce the drift in the heading estimate from the z-axis gyroscope bias, the change in the pedestrian's heading between steps ($\delta\psi_s$) can be assumed to be zero if the change is

relatively small [5]. The change in the pedestrian's heading between steps can be evaluated as:

$$\Delta\psi_k = \psi_k - \frac{1}{2}(\psi_{k,s} + \psi_{k,s-1}) \tag{4.8}$$

The term ψ_k is the pedestrian's current heading, $\psi_{k,s}$ is the pedestrian's heading at the previous instance of a detected stance, and $\psi_{k,s-1}$ is the pedestrian's heading at the instance of the detected stance prior to $t_{k,s}$. Similar to the *stance* and *still* detection methodologies, a threshold can be applied to the change in the pedestrian's heading for Straight Walking Detection (SWD). This can be written in the form of a logical expression:

$$C_6 = \begin{cases} 1 & \text{if} \quad |\,\Delta\psi_k\,| \leq TH_{\delta\psi} \\ 0 & \text{if} \quad \text{otherwise} \end{cases} \tag{4.9}$$

The threshold for the change in the pedestrian's heading is specified as $TH_{\delta\psi} = 4$ degrees [5] and the pedestrian's heading is computed from the estimated attitude by $\psi_k = \arctan(\tilde{C}_b^n(1,2), \tilde{C}_b^n(1,1))$. If the condition above is satisfied and the change in the pedestrian's heading is below the threshold, then the logical expression returns a 1 for *true* and the change in heading is reduced to zero to allow for the application of the HDR technique.

4.2.5 INS-EKF-ZVU

The methodology for the PDR-INS presented in this thesis was developed by Jiménez et al. [5] and the work from Foxlin [3]. This methodology utilizes a discrete-time error-state Kalman filter (ESKF). Contrary to the full-state methods given by the linear Kalman filter and the extended Kalman filter, the ESKF estimates the error between estimated states and the true states instead of estimating the the states directly. The advantage of this indirect filter formulation is that the error dynamics are low frequency and can suitably represented as linear. The low frequency dynamics also allow for a lower sampling rate to be used, whereas, the full-state EKF needs a much higher sampling rate to better capture high frequency dynamics that are not linear. The rest of this section will discuss the state selection, system model, and measurement models for the ESKF to estimate the PVA states mentioned in Section 4.2.2.

1. **State Selection**

The state vector for the ESKF is a 15 element column vector comprised of 9 navigation error states and 6 IMU bias states given by:

$$\delta\mathbf{x} = [\delta\mathbf{r}\ \ \delta\mathbf{v}\ \ \delta\psi\ \ \mathbf{b}_a\ \ \mathbf{b}_g]^T \tag{4.10}$$

The navigation states are defined as the (3×1) position error ($\delta\mathbf{r}$), the (3×1) velocity error ($\delta\mathbf{v}$), and the (3×1) attitude error ($\delta\psi$) in the North, East, Down navigation frame. The IMU bias states are the (3×1) accelerometer biases ($\mathbf{b}_a$) and the (3×1) gyroscope biases ($\mathbf{b}_g$) in the x, y, z sensor frame. With the states defined, their error dynamic equations can also be defined.

2. **Discrete-time System Model**

The discretized inertial navigation error models in the local navigation frame are given below as:

$$\delta\mathbf{r}_{k+1}^{-} = \delta\mathbf{r}_k + \delta\mathbf{v}_k\Delta t \tag{4.11}$$

$$\delta\mathbf{v}_{k+1}^{-} = \delta\mathbf{v}_k - \delta\psi_k[\hat{a}_k^n\wedge]\Delta t + C_{b,k}^n\mathbf{b}_{a,k}\Delta t \tag{4.12}$$

$$\delta\psi_{k+1}^{-} = \delta\psi_k + C_{b,k}^n\mathbf{b}_{g,k}\Delta t \tag{4.13}$$

$$\mathbf{b}_{a,k+1}^{-} = \mathbf{b}_{a,k} \tag{4.14}$$

$$\mathbf{b}_{g,k+1}^{-} = \mathbf{b}_{g,k} \tag{4.15}$$

The term $[\hat{a}_k^n\wedge]$ is the skew-symmetric matrix of the estimated acceleration in the navigation frame after bias and gravity vector removal. It should be noted that the IMU biases are modeled as random walk processes with a zero-mean and Gaussian distribution. In matrix notation, the linear error equations are given as:

$$\delta\mathbf{x}_{k+1}^{-} = \mathbf{\Phi}_k\delta\mathbf{x}_k + \mathbf{w}_{d,k} \tag{4.16}$$

The term $\mathbf{w}_{d,k}$ is the discrete system noise and $\mathbf{\Phi}_k$ is the discrete state transition matrix for time t_k given by:

$$\mathbf{\Phi}_k = \begin{bmatrix} I_3 & \Delta tI_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & I_3 & -\Delta t[f_k^b\wedge] & C_{b,k}^n\Delta t & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 & C_{b,k}^n\Delta t \\ 0_3 & 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{4.17}$$

The formulation of the discrete process noise covariance matrix ($Q_{d,k}$) maps the continuous process noise covariance matrix (Q) to the corresponding states can be approximated by the noise input matrix, G, and is given by:

$$\mathbf{Q}_{d,k} = \mathbf{G}_k \mathbf{Q} \mathbf{G}_k^T \Delta t \tag{4.18}$$

The process noise covariance matrix and the noise input matrix are given as:

$$\mathbf{Q} = \begin{bmatrix} \sigma_a^2 I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & \sigma_g^2 I_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & \sigma_{\mathbf{b}_a}^2 I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & \sigma_{\mathbf{b}_g}^2 I_3 \end{bmatrix} \tag{4.19}$$

$$\mathbf{G}_k = \begin{bmatrix} 0_3 & 0_3 & 0_3 & 0_3 \\ C_{b,k}^n & 0_3 & 0_3 & 0_3 \\ 0_3 & C_{b,k}^n & 0_3 & 0_3 \\ 0_3 & 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{4.20}$$

The terms ($\sigma_a^2,\ \sigma_g^2$) represent the variance of the noise of the acceleration and angular velocity measurements, and ($\sigma_{\mathbf{b}_a}^2,\ \sigma_{\mathbf{b}_g}^2$) represent the variance of the accelerometer and gyroscope biases, respectively.

3. **ZVU Measurement Model**

For the discrete-time linear system model of the ESKF, the measurement model is given by:

$$\delta \mathbf{y}_k = \mathbf{H}_k \delta \mathbf{x}_k + \mathbf{v}_k, \quad \mathbf{v}_k \sim N(0, \mathbf{R}_k) \tag{4.21}$$

The term $\delta \mathbf{y}_k$ is the error measurement, $\mathbf{H}_k$ is the observation matrix, and $\mathbf{v}_k$ is the measurement noise at time t_k. For a ZVU, the foot is stationary and the linear velocity is assumed to be zero. Therefore, the measurement vector for the ZVU is given by:

$$\delta \mathbf{y}_{k_{ZVU}} = [\hat{\mathbf{v}}_{k+1}^- - 0_{3\times 1}] \tag{4.22}$$

The error measurement is given as the difference between estimated velocity and the true velocity. The corresponding measurement matrix for the ZVU is given by:

$$\mathbf{H}_{k_{ZVU}} = \begin{bmatrix} 0_3 & I_3 & 0_3 & 0_3 & 0_3 \end{bmatrix} \tag{4.23}$$

This allows the measurement update to correct the states based on the difference between the true velocity error and the estimated velocity error.

4. **ZARU/ZVU Measurement Model**

   During a still period where the pedestrian is not walking, both the linear velocity and angular velocity of the foot can be assumed to be zero. This allows for both a ZARU and ZVU measurement update to occur and is given by the measurement error update:

$$\delta \mathbf{y}_{k_{ZARU}} = \begin{bmatrix} \hat{\mathbf{v}}_{k+1}^{-} - 0_{3\times 1} \\ \omega_{ib}^{b} - 0_{3\times 1} \end{bmatrix} \tag{4.24}$$

   The angular velocity measurement error is defined by the difference between the gyroscope measurement and the true angular velocity. However, the angular velocity of the pedestrian's foot is not a state of the system model, so the angular velocity error is said to be the gyroscope bias error. This yields the measurement matrix:

$$\mathbf{H}_{k_{ZARU}} = \begin{bmatrix} 0_3 & I_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & I_3 \end{bmatrix} \tag{4.25}$$

5. **SWD/Magnetic heading Measurement Model**

   During the periods where the pedestrian is walking, the assumption is made that the pedestrian is walking forward. The SWD update and the magnetic heading update constrain the forward motion to be either walking in a straight line, or turning while walking. During periods where the pedestrian is walking in a straight line, the measurement error update is given as:

$$\delta \mathbf{y}_{k_{SWD}} = \begin{bmatrix} \psi_s - \psi_{s-1} \end{bmatrix} \tag{4.26}$$

   The term ψ_s is the heading of the pedestrian's foot at the most recent step occurrence, and ψ_{s-1} is the heading of the pedestrian's foot at the previous step. This yields the measurement matrix to be:

$$\mathbf{H}_{k_{SWD}} = \begin{bmatrix} 0_{1\times 3} & 0_{1\times 3} & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} & 0_{1\times 3} & 0_{1\times 3} \end{bmatrix} \tag{4.27}$$

   The SWD measurement update only applies an update to the heading in the navigation frame rather than the full attitude of the pedestrian in the navigation frame to help reduce the yaw drift. During periods where the pedestrian is turning, a magnetic

heading measurement update is needed to constrain the yaw drift. The measurement error is given by:

$$\delta \mathbf{y}_{k_{MAG}} = \begin{bmatrix} \psi_{nb}^{n} - \psi_{n} \end{bmatrix} \tag{4.28}$$

The term ψ_n is the magnetic heading. The measurement matrix for the magnetic heading update is the same as the measurement matrix for the SWD update since only the heading is being corrected:

$$\mathbf{H}_{k_{MAG}} = \begin{bmatrix} 0_{1\times3} & 0_{1\times3} & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} & 0_{1\times3} & 0_{1\times3} \end{bmatrix} \tag{4.29}$$

Measurement Covariance

The measurement noise is assumed to be zero-mean and Gaussian with a covariance matrix $\mathbf{R}_k$. Since the measurement associated with a ZVU during the stance phase is not a true measurement from an external source but is, rather, an assumed measurement, the measurement noise covariance matrix must be selected. The method for setting the measurement noise covariance matrix was developed by Foxlin in [3], to give:

$$\mathbf{R}_{k_{ZVU}} = I_3 \cdot max(trace(\mathbf{H}_{k_{ZVU}} \mathbf{P}_{k+1}^{-} \mathbf{H}_{k_{ZVU}}^{T})) \tag{4.30}$$

The term I_3 is a 3×3 identity matrix. Intuitively, this formulation can be seen to be equal to the current velocity covariance before the measurement update. This allows for the velocity estimate to be corrected gradually during the stance phase to avoid any instabilities in large covariance reduction that might occur if the foot is not truly stationary. For the ZARU measurement correction during still periods, the measurement noise covariance matrix is constructed similarly:

$$\mathbf{R}_{k_{ZARU}} = I_6 \cdot max(trace(\mathbf{H}_{k_{ZARU}} \mathbf{P}_{k+1}^{-} \mathbf{H}_{k_{ZARU}}^{T})) + \begin{bmatrix} 0_3 & 0_3 \\ 0_3 & \mathbf{Q}_{d,k(4:6,4:6)} \end{bmatrix} \tag{4.31}$$

The measurement noise covariance also includes the discrete process noise associated with the gyroscope. This results in a 6×6 matrix as both the ZVU and ZARU are being applied simultaneously. For the SWD measurement update, the measurement noise covariance is formed by:

$$\mathbf{R}_{k_{SWD}} = max(trace(\mathbf{H}_{k_{SWD}} \mathbf{P}_{k+1}^{-} \mathbf{H}_{k_{SWD}}^{T})) \tag{4.32}$$

where the result is a scalar value since the SWD measurement update is only being applied to the error in the pedestrian's heading.

4.2.6 INS-EKF-ZVU Implementation and Summary

Implementation of the ESKF methodology presented in Section 4.2.5 was performed for a real-world data set of a pedestrian walking a closed loop around a parking lot in the clockwise direction. The pedestrian was outfitted with a Vectornav VN-100 9-DoF IMU [30] and a Ublox GPS receiver with a single antenna [31]. The IMU was mounted to the shoelaces of the pedestrian's right foot, and the antenna for the GPS receiver was mounted to the pedestrian's right shoulder. Measurements from the IMU were post-processed in the ESKF implementation, while the GPS measurements were used as a truth reference of the walking path. The results of the implementation will be shown in succession of the different measurement models as they were introduced with only the ZVU measurement update first, then the ZVU+ZARU measurement updates, and finally the ZVU+ZARU+HDR measurement updates.

The architecture for the PVA estimation of the pedestrian is comprised of eight steps that occur in the order they are presented in. This architecture is based on the work by Foxlin in [3] and Jiménez [5].

1. **PVA Initialization**

   To initialize the position, velocity, and attitude of the pedestrian, an external reference is required. The position was initialized from the GPS and the initial velocity was assumed to be zero indicating a *still* period for initialization. The initial attitude was determined by obtaining the pitch and roll angles from the expected gravity vector in the navigation frame in a process known as *leveling*. During a stationary period, the accelerometers of the IMU measure the reaction to the force of gravity and can obtain the pitch (θ) and roll (ϕ) angles using simple trigonometric equations:

   $$\theta = \text{atan}(-f^b_{ib,y}/-f^b_{ib,z}) \tag{4.33}$$

   $$\phi = \text{atan2}\left(f_{ib,x}, \sqrt{(f^b_{ib,x})^2 + (f^b_{ib,y})^2}\right) \tag{4.34}$$

   The function atan() is the inverse tangent function and atan2() is the four-quadrant inverse tangent function. The terms $f^b_{ib,x}$,$f^b_{ib,y}$, and $f^b_{ib,z}$ are the specific force measurements from the IMU along the axes of the IMU sensor frame. For the initial heading, the yaw angle is determined from the magnetometer.

2. **IMU Bias Compensation**

The next step is to remove the estimated accelerometer and gyroscope biases from the specific force and angular velocity measurements as described in Section 3.2 where the estimated specific force and angular velocity are given by:

$$\hat{f}_{ib}^{b} = \tilde{f}_{ib}^{b} - \mathbf{b}_{a} \tag{4.35}$$

$$\hat{\omega}_{ib}^{b} = \tilde{\omega}_{ib}^{b} - \mathbf{b}_{g} \tag{4.36}$$

The terms $\hat{f}_{ib}^{b}$ and $\hat{\omega}_{ib}^{b}$ are the estimated specific force and angular velocity, and the terms $\tilde{f}_{ib}^{b}$ and $\tilde{\omega}_{ib}^{b}$ are the measured specific force and angular velocity.

3. **Attitude Propagation**

After removal of the estimated gyroscope biases, the attitude must be propagated from the body frame to the local navigation frame. From Section 3.2, the continuous-time attitude update of the rotation matrix from the body frame to the local navigation frame is given by:

$$\dot{C}_{b}^{n} = C_{b}^{n}\Omega_{ib}^{b} \tag{4.37}$$

The term Ω_{ib}^{b} is the skew-symmetric matrix of the bias compensated angular velocity measurements. The skew-symmetric matrix is defined by:

$$\Omega_{ib}^{b} = \begin{bmatrix} 0 & -\hat{\omega}_{ib,z}^{b} & \hat{\omega}_{ib,y}^{b} \\ \hat{\omega}_{ib,z}^{b} & 0 & -\hat{\omega}_{ib,x}^{b} \\ -\hat{\omega}_{ib,y}^{b} & \hat{\omega}_{ib,x}^{b} & 0 \end{bmatrix} \tag{4.38}$$

However, since the system hardware samples in discrete-time, the attitude propagation is discretized by the use of a *Padé* approximation [5]:

$$\hat{C}_{b_{k+1}}^{n^{-}} = \hat{C}_{b_{k}}^{n}\left(\frac{2\cdot I_{3} + \Omega_{ib_{k}}^{b}\Delta t}{2\cdot I_{3} - \Omega_{ib_{k}}^{b}\Delta t}\right) \tag{4.39}$$

The term $\hat{C}_{b_{k+1}}^{n^{-}}$ is the estimated rotation matrix from the body frame to the navigation frame before a measurement update correction at time t_{k+1}.

4. **Gravity Compensation**

With the updated rotation matrix defined, the accelerometer measurements can be rotated into the local navigation frame. However, since accelerometers measure changes in specific force and not gravity, the gravity vector in the local navigation frame must

be removed in order to properly update the velocity and position estimates. This compensation is given by:

$$\hat{a}^n_{ib_{k+1}} = \hat{C}^{n^-}_{b_{k+1}} f^b_{ib_k} + g^n \tag{4.40}$$

The term g^n is the gravity vector in the local navigation frame. While the acceleration due to gravity is not constant around the Earth, it can be approximated as $9.81 m/s^2$. Since the PVA propagation is performed in a North, East, and Down local navigation frame, the gravity vector is given as:

$$g^n = \begin{bmatrix} 0 \\ 0 \\ 9.81 \end{bmatrix} m/s^2 \tag{4.41}$$

In an ideal static case, after rotating the accelerometer readings into the local navigation frame and removing the gravity vector, the measurements would be $f^b_{ib,x} = 0$ m/s^2, $f^b_{ib,y} = 0$ m/s^2, and $f^b_{ib,z} = 0$ m/s^2.

5. **Velocity and Position Propagation**

   With the specific force measurements rotated into the navigation frame and compensated for acceleration due to gravity, the estimated velocity is calculated by numerically integrating the acceleration in the navigation frame over the sampling period:

$$\hat{v}^{n^-}_{ib_{k+1}} = \hat{v}^n_{ib_k} + \hat{a}^n_{ib_{k+1}} \Delta t \tag{4.42}$$

   The term $\hat{v}^{n^-}_{ib_{k+1}}$ is the estimated velocity in the navigation frame before a measurement update correction at time t_{k+1}. The propagation of the estimated position in the navigation frame is given by:

$$\hat{r}^{n^-}_{ib_{k+1}} = \hat{r}^n_{ib_k} + \hat{v}^{n^-}_{ib_{k+1}} \Delta t \tag{4.43}$$

   The same rectangular integration technique is used to determine the estimated position before any measurement update correction is performed. It should be noted that the superscript $(-)$ is not present with the estimated acceleration because the ESKF does not estimate the error in acceleration, and, therefore, no correction is made to the acceleration estimate.

6. **ESKF Time Update**

The order that the ESKF Time Update and the estimated position and velocity propagation occur is not required to provide a unique solution. The time update is purely a function of the estimated rotation matrix and acceleration in the navigation frame, therefore, the order that steps 5 and 6 occur does not matter. The order given is how they were implemented for this experiment.

The states of the ESKF are updated based on the discrete-time update equation in Section 3.3. The estimated states and state covariances are given by:

$$\delta \mathbf{x}_{k+1}^{-} = \mathbf{\Phi}_k \delta \mathbf{x}_k \tag{4.44}$$

$$\mathbf{P}_{k+1}^{-} = \mathbf{\Phi}_k \mathbf{P}_k \mathbf{\Phi}_k^T + \mathbf{Q}_{d,k} \tag{4.45}$$

The $\mathbf{P}_{k+1}^{-}$ term is the state covariance before a measurement update correction at time t_{k+1} and $\mathbf{Q}_{d,k}$ is the discrete process noise covariance matrix at time t_k.

7. **Measurement Update and PVA Correction**

As measurements become available with the different measurement updates from the ESKF, the estimated position, velocity, and attitude are corrected by removing the error between the estimated PVA and the true PVA. The correction of the position and velocity is given by:

$$\hat{r}_{b_{k+1}}^{n} = \hat{r}_{b_{k+1}}^{n^{-}} - \delta \mathbf{r}_{b_{k+1}}^{n} \tag{4.46}$$

$$\hat{v}_{b_{k+1}}^{n} = \hat{v}_{b_{k+1}}^{n^{-}} - \delta \mathbf{v}_{b_{k+1}}^{n} \tag{4.47}$$

and the attitude correction is given by:

$$\hat{C}_{b_{k+1}}^{n} = \left(\frac{2 \cdot I_3 + [\delta \mathbf{\Psi}_{b_{k+1}}^{n} \wedge] \Delta t}{2 \cdot I_3 - [\delta \mathbf{\Psi}_{b_{k+1}}^{n} \wedge] \Delta t} \right) \hat{C}_{b_{k+1}}^{n^{-}} \tag{4.48}$$

The term $[\delta \mathbf{\Psi}_{b_{k+1}}^{n} \wedge]$ is the skew-symmetric matrix of the attitude errors:

$$[\delta \mathbf{\Psi}_{b_{k+1}}^{n} \wedge] = \begin{bmatrix} 0 & -\delta \hat{\psi}_{b_{k+1}}^{n} & \delta \hat{\theta}_{b_{k+1}}^{n} \\ \delta \hat{\psi}_{b_{k+1}}^{n} & 0 & -\delta \hat{\phi}_{b_{k+1}}^{n} \\ -\delta \hat{\theta}_{b_{k+1}}^{n} & \delta \hat{\phi}_{b_{k+1}}^{n} & 0 \end{bmatrix} \tag{4.49}$$

The terms $(\delta \hat{\phi}, \delta \hat{\theta}, \delta \hat{\psi})$ are the estimated errors in roll, pitch, and yaw.

8. **Resetting Error State estimates**

After the corrections have been applied to the full states, the error states are set to zero to avoid accumulating previous error after the full states have been corrected. Since the PVA states are corrected from the error states, then it's reasonable to claim the error between the estimated and true states are zero.

$$\delta\hat{\psi}^n_{b_{k+1}} = 0_{3\times1}$$

$$\delta\hat{v}^n_{b_{k+1}} = 0_{3\times1}$$

$$\delta\hat{r}^n_{b_{k+1}} = 0_{3\times1}$$

From here the steps are repeated beginning at step two.

With only the ZVU measurement update provided during implementation, the z-axis gyroscope bias is unobservable and results in heading drift as shown in Figure 4.3. The pedestrian begins at the bottom left corner of the plot and walks in a clockwise path around the parking lot. Initially, the pedestrian's heading is tracked properly until reaching the first corner. From there, the pedestrian's heading begins to drift clockwise to drive the PDR solution more southward.

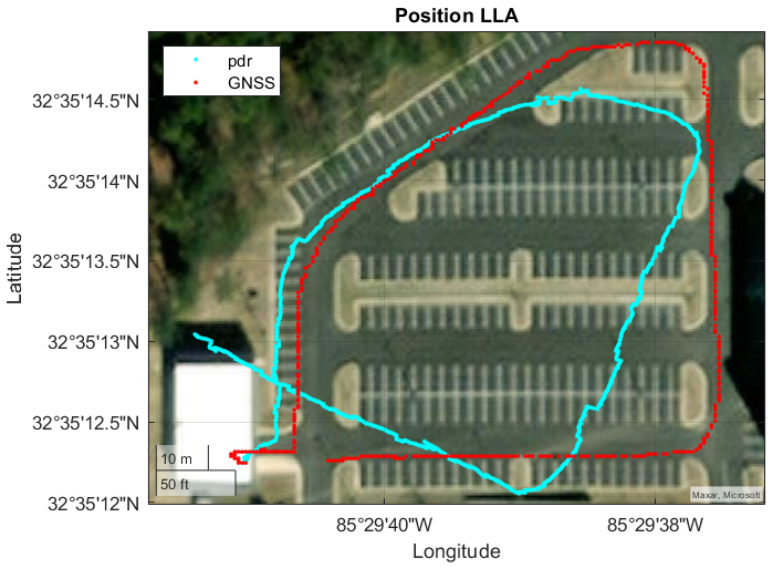


Figure 4.3: PDR-INS trajectory with only zero-velocity updates

Now the ZARU measurement update is added in addition to the ZVU to estimate the z-axis gyroscope bias in order to help reduce the drift in heading. Incorporating measurements from both a ZVU and a ZARU help to estimate the gyroscope bias and constrain the drift in heading as shown in Figure 4.4. However, the *still* periods are only detected at the beginning and end of the route. While more observability of the z-axis gyroscope bias is provided during the first *still* period, the heading drift still remains present and heading estimates continue to be noisy.

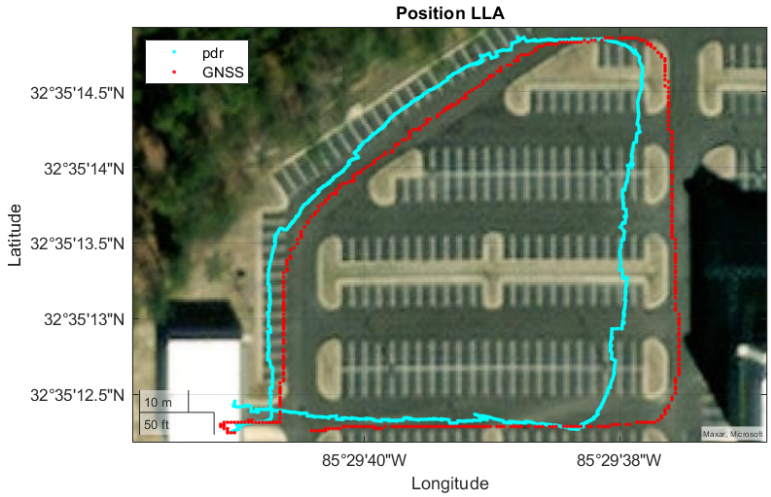


Figure 4.4: PDR-INS trajectory with zero-velocity and zero-angular rate updates

Incorporating measurements from the ZVU, ZARU, and SWD allow for the drift in heading to be constrained based on detection of constant heading as shown in Figure 4.5. The incorporation of SWD updates help to smooth the noise in the heading estimates. However, a significant amount of drift in heading can be observed.

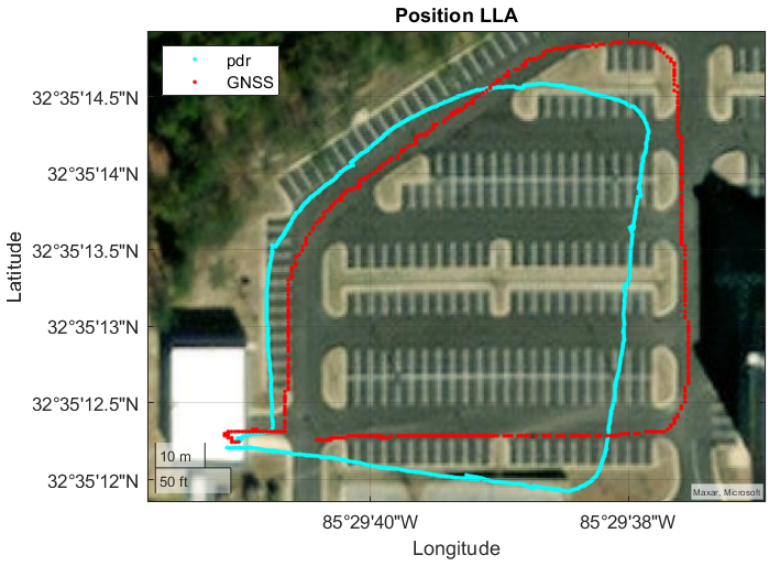


Figure 4.5: PDR-INS trajectory with ZVU, ZARU, and SWD

Figure 4.6 shows the position error and root mean squared error for the PDR-INS implementation with ZVU, ZARU, and SWD. The inclusion of SWD updates from straight walking detection help to smooth the noisy heading estimates when the pedestrian is walking in a constant direction. However, SWD updates do not eliminate the drift in heading when the pedestrian is walking around a bend. Filter tuning might help to reduce this effect.

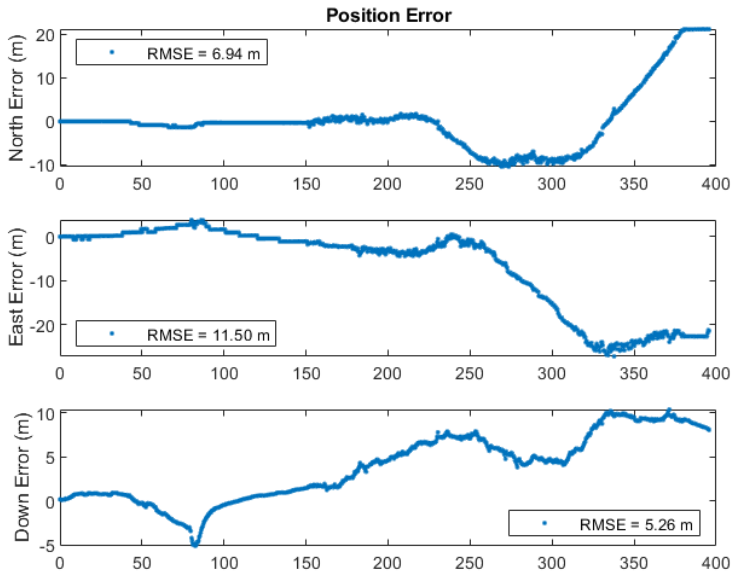


Figure 4.6: PDR-INS Position Error in North and East directions

The PDR-INS approach offers insight into errors that can corrupt the navigation solution using the various techniques. However, this implementation can prove to be computationally expensive, and is limited by the presence of events in the gait cycle that are needed to perform these techniques. The next section introduces the step detection approach to PDR.

4.3 PDR Step Detection Approach

An alternative approach to PDR performs step detection, step length estimation, and heading estimation of the pedestrian. Unlike the INS approach for a foot-mounted IMU, a full 6-DoF mechanization is not performed. Rather, the step detection approach utilizes the natural bouncing motion of the pedestrian's torso while walking to construct a navigation solution. This is done by fusing the data from an IMU mounted on the body with a model of how the pedestrian's torso moves between. Implementations of this approach have been proven by [10, 4]. This approach provides a method of PDR that is more often simplistic for implementation than that of the INS approach mentioned in the previous section, as full IMU mechanization is not required. Rather than examining the phases of the pedestrian's

gait cycle, the PDR with step detection (PDR-SD) approach is broken into four different methods: step detection, step length estimation, heading determination, and the navigation position update. The remainder of this section will cover the different methods of the step detection approach to PDR, the system model, and an implementation and summary of the system.

4.3.1 Step Detection

The goal of the step detection aspect of the PDR-SD implementation is to determine the occurrence of steps when the pedestrian is walking regularly. This is very critical to the performance of the PDR-SD implementation since the step length estimation and navigation position update are dependent on the ability to detect when steps occur. Steps can be detected using only an accelerometer [17, 4, 10]. However, identifying step occurrences by analyzing each axis of the accelerometer is often difficult and can lead to erroneous step detections. Therefore, examining the Euclidean norm of the accelerometer can provide a more simplistic means of detecting a step given by:

$$\| \tilde{\mathbf{f}}_{ib}^{b} \| = \sqrt{\left(\tilde{f}_{x,ib}^{b}\right)^2 + \left(\tilde{f}_{y,ib}^{b}\right)^2 + \left(\tilde{f}_{z,ib}^{b}\right)^2} \tag{4.50}$$

The effectiveness of the step detection algorithm analyzing only one axis of the accelerometer requires accurate measurements of the Earth's force due to gravity. This requires the analyzed axis of the accelerometer to be mounted perpendicular to the surface of the Earth to capture this motion.

Figure 4.7 shows experimental data of an IMU mounted onto a pedestrian's chest while walking with the x-axis facing the upward direction of the pedestrian's frame. Additionally, Figure 4.8 shows the same plot but with the inclusion of the resulting Euclidean norm. Analysis of the Euclidean norm of the tri-axial accelerometer allows for measurements of the Earth's force due to gravity to constantly be obtained. For the remainder of this thesis, the Euclidean norm of the accelerometer mounted on the pedestrian's chest will be examined.

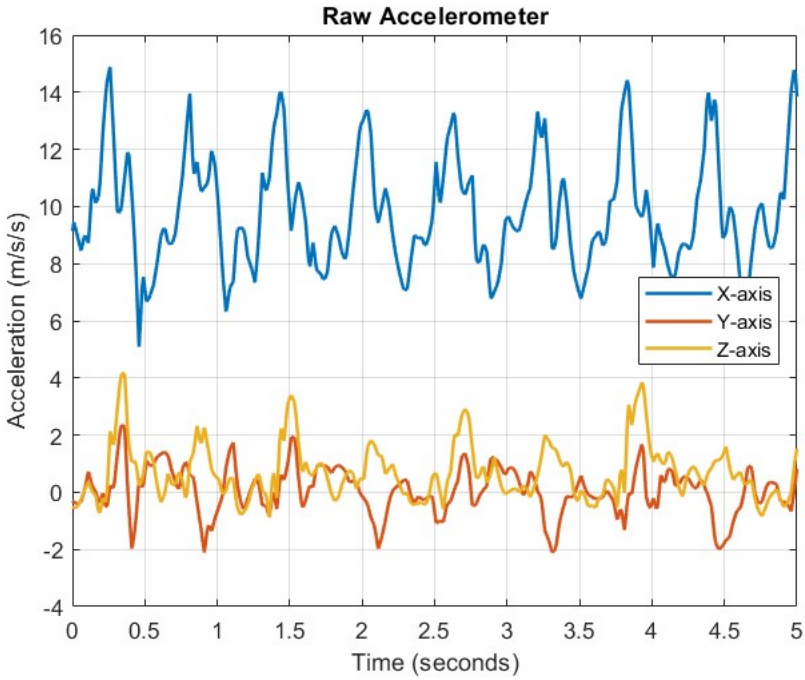


Figure 4.7: Tri-axial acclerometer measurements from chest-mounted IMU

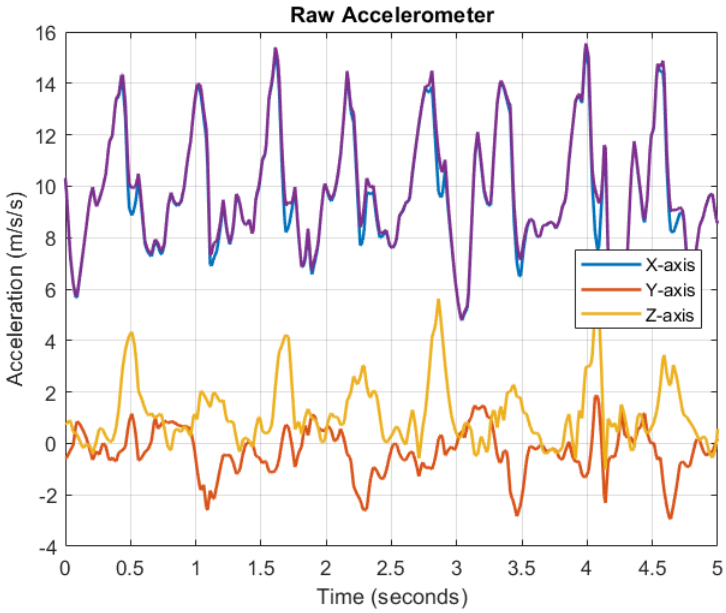


Figure 4.8: Tri-axial acclerometer measurements with Euclidean Norm

Methods for detecting steps range from peak detection [32, 33, 11, 12], autocorrelation and spectral analysis [34], and thresholding [17, 35]. Due to the robustness and relative simplicity of implementation, the thresholding method of step detection will be used for this thesis. A simplistic approach to the thresholding method examines the zero-crossing of the Euclidean norm of the accelerometer measurements with the acceleration due to gravity ($g = 9.81\ m/s^2$). This assumes that the step begins during the footfall of the leading foot where the acceleration experienced by the body is equal to the acceleration due to gravity ($\| \mathbf{f}_{ib}^b \| = g$).

A more advanced method of the thresholding technique uses a window of samples to define a maximum, minimum, and average threshold [33]. For effectiveness, the sample window is dependent on the sampling rate of the accelerometer. For example, an IMU being sampled at $100\,Hz$ would have the thresholds updated every 100 samples. Within the analyzed window of samples, four different criteria need to be met in order for a step to be detected:

1. The current magnitude of the accelerometer signal must be less than the previous magnitude of the accelerometer signal. In other words, the accelerometer signal must have a negative slope.

2. The previous magnitude of the accelerometer must be greater than the average threshold and the current accelerometer magnitude must be less than the average threshold.

3. The time difference between the current sample time and the most recent detected step time must be greater than the specified time between steps. This ensures that steps are detected at a reasonable walking rate.

4. The minimum and maximum thresholds must be below and above specified values that are indicative of human walking motion. This is to ensure steps aren't detected during static periods where the torso is still in motion.

An implementation of the more advanced thresholding method, stated above, was performed on the same segment of IMU data shown in Figure 4.7. The results of this implementation are shown below in Figure 4.9. At each sample, the maximum, minimum, and average threshold are determined over the 11 sample window. The average acceleration norm is then computed and compared to the previous window. If the current acceleration norm

average is less than the preivous; less than the previous average threshold; the time between steps is less than 0.1 seconds; the minimum threshold is below 9.75 m/s^2; and the maximum threshold is above 10.25 m/s^2 then a step has been detected. The detected steps are marked in red.

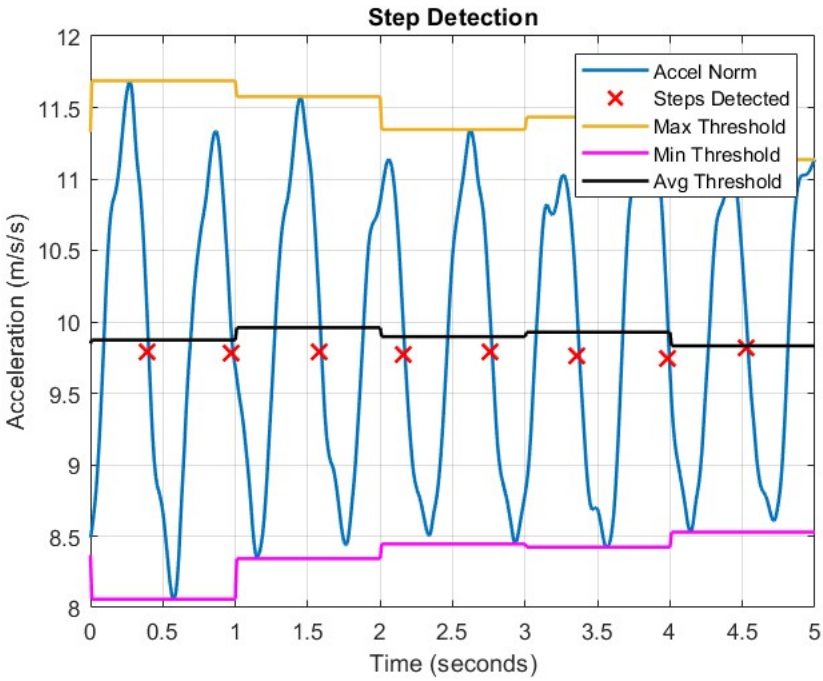


Figure 4.9: Thresholding method of step detection for torso-mounted dataset

4.3.2 Step Length Estimation

With the step detection process of the PDR-SD defined, the next component is the estimation of the pedestrian's step length. This estimation process is important for propagating the pedestrian's position forward in time for the navigation solution. Many step length estimation methods attempt to relate the step length to different physical parameters. These parameters range from the total distance traveled, the pedestrian's height, accelerometer measurements, and signal characteristics. In the case of relating step length to the total distance traveled, the step length is defined as the total distance traveled divided by the total number of steps taken:

$$SL_1 = \frac{d}{n} \tag{4.51}$$

where n is the total number of steps taken over a distance, d. This method of step length estimation is often referred to as constant step length. This approach is highly intuitive, but requires good knowledge or estimation of the total distance traveled.

Another approach to step length estimation is based on defining a relationship between the pedestrian's step length and height. This method assumes a constant step length proportional to the pedestrian's height and some pre-defined constant:

$$SL_2 = K \cdot h \tag{4.52}$$

where h is the pedestrian's height and K is an empirically determined constant that typically ranges from 0.413 to 0.415 [36]. While this method can prove to be useful for cases where the user is walking at a constant speed and step length, the reality is that the pedestrian's step length can vary dependent on the pace of the walking pattern. Another method using the pedestrian's height was proposed by Zhao [33], where the step length was augmented based on the speed of the pedestrian. The results were determined experimentally and are shown below in Table 4.1:

Table 4.1: SL_3 based on pedestrian pace

Steps per 2 seconds	SL_3
0-2	$h/5$
2-3	$h/4$
3-4	$h/3$
4-5	$h/2$
5-6	$h/1.2$
6-8	h
8+	$1.2 \cdot h$

The fixed height methodology requires that the pedestrian's height be used as an input into the estimator and must be described in the same units as the estimator.

Another method relates the pedestrian's step length to the accelerometer measurements between detected steps. This method was proposed by Weinberg [37] where a dynamic step length estimator based on the vertical displacement of the pedestrian's hip, referred to as the "hip-bounce". To reduce complexity, the formula was modified for estimation of step

length rather than stride length. This relationship of the pedestrian's step length to the "hip-bounce" is given by:

$$SL_4 = K \cdot \sqrt[4]{a_{max} - a_{min}} \tag{4.53}$$

where a_{max} and a_{min} are the maximum and minimum measured vertical acceleration during a step. One key assumption of this algorithm is the assumption of a constant angle of the hip between the minimum and maximum positions of a step, which is often untrue. However, the use of this technique without accounting for the angle of the hip positions still produces estimates that are within 8% of the true step length [37].

Another dynamic method of estimating step length based on the user's "hip-bounce" was determined by Bylemans [36]. This method is derived from Kim [8] who empirically determined step length from a foot mounted experiment of several walking tests. The resulting model was determined as:

$$SL_5 = 0.1 \cdot \sqrt[2.7]{\frac{\sum_{i=1}^{N}|a_i|}{N} \times \sqrt{\frac{K}{\Delta t \cdot (a_{max} - a_{min})}}} \tag{4.54}$$

where i is the current sample, a_i is the vertical acceleration made during the step, and N is the number of samples in the step, Δt is the duration of the step, and k is a calibration constant. Similar to Equation 4.53 the difference between the maximum and minimum acceleration is used.

A biomechanical model that utilizes knowledge of the pedestrian's leg length is based on work by [38], and presented by [10] and [39]. This model is given by:

$$SL_6 = K \cdot \sqrt{2LY - Y^2} \tag{4.55}$$

where L is a fixed leg length, Y is the vertical displacement of the pedestrian's hip, and K is a calibration constant.

The final method of step length estimation methods presented in this thesis relate the pedestrian's step length to certain signal characteristics, such as step frequency and specific force variance. One such algorithm was developed by Gusenbauer [7] and focused on the step frequency:

$$SL_7 = K_1 f_s + K_2 \tag{4.56}$$

where f_s is the step frequency (time between steps), and coefficients K_1 and K_2 are empirically determined for each user. An experiment was conducted by Li in [40] where a linear

trend was found between the pedestrian's step length and the time between steps. Further analysis of this method included the variance of the magnitude of specific force [11]:

$$SL_8 = K_1 f_s + K_2 \sigma_f^2 + K_3 \tag{4.57}$$

where f_s is the step frequency, σ_f^2 is the variance of magnitude of acceleration, and K_1, K_2, and K_3 are the coefficients determined for each user. Groves et al. continued on this modification by taking the slope of the walking path into account [4]:

$$SL_9 = K_1 f_s + K_2 \sigma_f^2 + K_3 S + K_4 \tag{4.58}$$

where the equation is identical to Equation 4.57 but includes the additional term, S, which is the slope of the elevation change of the walking path. However, analysis of the results showed that the step frequency and slope components of this formula were poorly identifiable and the model was reduced to:

$$SL_{10} = K_1 \sigma_f^2 + K_2 \tag{4.59}$$

where σ_f^2 is the variance of the specific force magnitude, and K_1 and K_2 are the model coefficients.

For Equations (4.56 - 4.59), the step length estimates are governed by signal characteristics of the walking pattern. The step frequency term f_s is defined by the time between successive steps and is given by:

$$f_{s,k} = \frac{1}{t_{s,k} - t_{s,k-1}} \tag{4.60}$$

the variance of the specific force is a function of the euclidean norm of the accelerometer measurements between steps:

$$\sigma_f^2 = \frac{\left(\sum_{k_{start}}^{k_{stop}} \left(\parallel f_{ib,b}^b \parallel - \mu_{k_{start}:k_{stop}} \right)\right)^2}{k_{stop} - k_{start}} \tag{4.61}$$

and for Equation 4.58 the slope of the change in elevation, S, is given by:

$$S = \frac{\Delta h}{\sqrt{(\Delta r_E)^2 + (\Delta r_N)^2}} \tag{4.62}$$

where Δh is the change in altitude in meters, and Δr_E and Δr_N are the changes in the North and East position in meters.

A full comprehensive comparison of all the mentioned methods of step length estimation is beyond the scope of this thesis. However, implementations of formulas 4.53, 4.56, and

4.51 are analyzed for the methodology proposed in Section 6.3. Equation 4.53 has been augmented as:

$$SL_{11} = K_1 \sqrt[4]{|f^b_{ib,b,max}| - |f^b_{ib,b,min}|} + K_2 \tag{4.63}$$

where $|f^b_{ib,b,max}|$ and $|f^b_{ib,b,min}|$ are the maximum and minimum of the euclidean norm of the specific force measurements, and K_1 and K_2 are the model coefficients.

4.3.3 Attitude Estimation

For the PDR-SD implementation, the pedestrian's step length is used to propagate the position in the navigation frame. In order to accurately perform this propagation, the pedestrian's attitude needs to be determined in order for the new position to be directed properly. This can be conducted by dead-reckoning the angular velocity measurements from the gyroscope, using the heading from magnetometer measurements with tilt angles, or combining dead-reckoned angular velocity, magnetometer heading, and tilt angles together in an attitude heading reference system (AHRS).

Angular Velocity Dead-Reckoning

Since IMUs are typically comprised of tri-axial accelerometers and tri-axial gyroscopes, the accelerometer is used for step detection and the gyroscope can be used for tracking changes in orientation. Parameterizing the pedestrian's attitude as a rotation matrix allows for the changes in yaw, pitch, and roll to be propagated in discrete time by:

$$C^n_{b_{k+1}} = (I_3 + \Omega^b_{nb_k} \Delta t_k) C^n_{b_k} \tag{4.64}$$

The inaccuracies of this method are due to errors within the gyroscope measurements. As mentioned previously, gyroscope measurements are subject to multiple sources of error which can cause the attitude solution to drift over time. These errors can be mitigated and accounted for by the aid of ZVUs, ZARUs, and SWDs, however, implementations of these techniques are not often available for PDR-SD approaches. With this in mind, the dead-reckoning of angular velocity measurements alone does not provide the accuracy needed for attitude determination.

Magnetometer and Tilt Angle

Another method of attitude determination with the PDR-SD approach, is utilizing models of the Earth's gravitational and magnetic field as a source of reference. The accelerometer provides measurements of the Earth's acceleration due to gravity to give the pitch and roll angles of the IMU relative to the local navigation frame [2]:

$$\theta = atan2\left(\frac{f^b_{nb,x}}{\sqrt{(f^b_{nb,y})^2 + (f^2_{nb,z})^2}}\right) \qquad \phi = atan2\left(\frac{f^b_{nb,y}}{f^b_{nb,z}}\right) \tag{4.65}$$

where f^b_{nb} is the accelerometer measurements for each axis. This method only works when the IMU is not accelerating. This assumes that the only acceleration is due to gravity. With pitch and roll defined, the yaw angle can be determined from calibrated measurements of the Earth's magnetic field from a calibrated magnetometer co-located with the IMU. A method of magnetometer calibration is presented in the next chapter. Determining the yaw angle from a magnetometer requires that the magnetometer be "leveled" where the z-axis of the magnetometer is parallel to the z-axis of the local navigation frame [2, 5]. The heading can then be determined by:

$$\psi_{mag} = atan\left(\frac{B^n_{nb,y}}{B^n_{nb,x}}\right) + \alpha_m \tag{4.66}$$

where B^n_{nb} is the magnetic field strength measurement along each axis, and α_m is a declination angle between magnetic north and true north. While this provides a consistent reference for heading determination, the magnetometer is still subject to disturbances from local magnetic fields such as: power lines, mobile phones, cars, generators, etc. [2]. With this in mind, the use of magnetometers with tilt angle measurements can still produce inaccuracies in the navigation solution.

Attitude Heading Reference System

Using either dead reckoned angular velocity measurements or measurements of the Earth's gravitational and magnetic fields for attitude determination come with their own benefits and drawbacks. Angular velocity dead reckoning allows for short-term stability, but is prone to drift over time due to measurement noise, biases, misalignment, and other sources of error. Using magnetic and gravitational field measurements provide a consistent reference for full attitude determination over longer periods of time, but are susceptible to

local perturbations and disturbances, as well as, acceleration that can produce erroneous measurements causing the estimates to deviate. With this in mind, a methodology that fuses the two techniques together proves to be much more beneficial. This methodology is described as an attitude heading reference system (AHRS) [2].

The attitude for an AHRS is determined by, first integrating the angular velocity measurements to determine the yaw, pitch, and roll angles. Next, the accelerometer measures the Earth's gravitational pull to correct the errors in the pitch and roll by the "leveling" process mentioned in the previous section. With the pitch and roll corrected, the magnetometer measures the strength of the Earth's magnetic field to provide a heading correction for errors in the dead reckoned yaw angle. Low gain smoothing filters are used to reduce the effect of short term errors in the gravitational and magnetic field measurements during the correction of the dead reckoned gyroscope. Many AHRS utilize maneuver detection for high dynamic cases to filter the accelerometer measurements, as well as, magnetic fault detection for errors in the magnetic field measurements [41]. The integration of these measurements within a Kalman filter allow for smoothing gains to be dynamically optimized and estimation of the gyroscope biases.

The Kalman filter approach to attitude estimation allows for the Euler angles to be estimated, while simultaneously reducing errors by removing gyroscope biases. The AHRS Kalman filter is presented based on the work by [23] and [41], that use a quaternion-based Kalman filter for attitude estimation. This is accomplished by tracking the orientation of the IMU as a full-state quaternion (X), as well as tracking the attitude errors as Euler angle error-states (x)and the gyroscope biases as full states where:

$$X = \begin{bmatrix} q \\ b_g \end{bmatrix} \qquad\qquad x = \begin{bmatrix} \delta\psi \\ b_g \end{bmatrix} \tag{4.67}$$

Both state vectors are propagated in discrete time at the IMUs sampling rate based on the continuous dynamic models given in [23] and [41] where:

$$\dot{X} = FX \qquad\qquad \dot{x} = Ax + w \tag{4.68}$$

This yields the state transition matrices of:

$$A_d = \mathbf{expm}\left(\begin{bmatrix} -\left[\omega_{ib}^n\right]_\times \Delta t & -I_3\Delta t \\ 0_3 & 0_3 \end{bmatrix}\right) \tag{4.69}$$

$$F_d = \begin{bmatrix} \mathbf{expm}\left(\frac{1}{2} \begin{bmatrix} 0 & -\omega_{ib}^{n^T} \Delta t \\ \omega_{ib}^n \Delta t & -[\omega_{ib}^n \Delta t]_\times \end{bmatrix} \right) & 0_{4\times 3} \\ 0_{3\times 4} & I_3 \end{bmatrix} \tag{4.70}$$

The function **expm** () is matrix exponential function, A_d and F_d represent the discretized dynamic matrices based on the IMU sampling time Δt, and ω_{ib}^n is the angular velocity measurements with respect to the navigation frame. It should be noted that no process noise is considered for the full state model. This is because the measurement updates are applied for correcting the errors in the Euler angle error estimates, which are then used to correct the full state estimates. The discrete process noise matrix of the error state model is given by:

$$Q_d = \begin{bmatrix} \mathbf{diag}\left(\sigma_g^2\right) & 0_3 \\ 0_3 & \mathbf{diag}\left(\sigma_{bg}^2\right) \end{bmatrix} \cdot \Delta t \tag{4.71}$$

The measurement updates are applied as they become available. In many cases, the sampling rate of the magnetometer and the accelerometer are not always the same and are not available at the same time. For simplicity the measurement updates are presented under the assumption that both sensors are at the same sampling rate, and available at the same time. The measurement update is applied in two steps: the accelerometer update, and the magnetometer update. The accelerometer update rotates the gravity vector into the IMU frame from the full state quaternion estimates for the pitch and roll angle corrections [41]. The rotation matrix derived from a quaternion is given by:

$$C_b^n = \begin{bmatrix} 2q_1^2 + 2q_2^2 + 2q_3^2 - 2q_0q_3 & q_0^2 + q_1^2 + 2q_2^2 + 2q_3^2 & -2q_0q_2 - 2q_1^2 - 2q_2^2 - 2q_3^2 \\ q_0^2 + 2q_1^2 + q_2^2 + 2q_3^2 & 2q_0q_3 + 2q_1^2 + 2q_2^2 + 2q_3^2 & 2q_0q_1 - 2q_1^2 - 2q_2^2 - 2q_3^2 \\ 2q_0q_1 + 2q_1^2 + 2q_2^2 + 2q_3^2 & 2q_1^2 + 2q_2^2 + 2q_3^2 - 2q_0q_2 & -q_0^2 - 2q_1^2 - 2q_2^2 - q_3^3 \end{bmatrix} \tag{4.72}$$

This yields the measurement vector:

$$y_a = \left[f_{ib}^{b^T} - C_n^b(\hat{q}) g \right] \tag{4.73}$$

The term f_{ib}^b is the accelerometer measurements with respect to the body frame, and $C_n^b(\hat{q})$ is the rotation matrix from the navigation frame to the body frame based on the estimated quaternion. The magnetometer update rotates the magnetic field measurements into the body frame for the yaw angle correction:

$$y_m = \left[m_{nb}^{b^T} - C_n^b \tilde{m} \right] \tag{4.74}$$

The term $\tilde{m}$ is given by:

$$\tilde{m} = \begin{bmatrix} cos(\alpha) \\ 0 \\ -sin(\alpha) \end{bmatrix} \tag{4.75}$$

The term α is the dip angle. The measurement errors are mapped onto the error states from the measurement matrix:

$$H = \begin{bmatrix} \left[C_n^b(q)g\right]_\times & 0_3 \\ \left[C_n^b(q)\tilde{m}\right]_\times & 0_3 \end{bmatrix} \tag{4.76}$$

with a measurement covariance matrix of:

$$R = \begin{bmatrix} \mathbf{diag}(\sigma_a^2) & 0_3 \\ 0_3 & \mathbf{diag}(\sigma_m^2) \end{bmatrix} \tag{4.77}$$

The final step of the AHRS Kalman filter is utilizing the corrected error states to update the full state estimates. The quaternion states are updated from the Euler angle error estimates [23] where:

$$q_{k+1}^+ = q_{k+1}^- \otimes q\{\hat{\delta\psi}\}_{k+1} \tag{4.78}$$

The term $q\{\hat{\delta\theta}\}$ is the quaternion error from the Euler angle errors which can be approximated from a first-order expansion yielding:

$$\hat{q}_{k+1}^+ = \hat{q}_{k+1}^- + \frac{1}{2}\begin{bmatrix} 0 & -\delta\psi_{k+1}^T \\ \delta\psi_{k+1} & -\left[\delta\psi_{k+1}\right]_\times \end{bmatrix}\hat{q}_{k+1}^- \tag{4.79}$$

The corrected quaternion estimate must then be normalized to describe a rotation:

$$\hat{q}_{k+1}^+ = \frac{\hat{q}_{k+1}^+}{\parallel \hat{q}_{k+1}^+ \parallel} \tag{4.80}$$

The gyroscope biases are then set from the error state vector bias estimates, and the error state vector is reset to zero:

$$X\{5:7\}_{k+1} = x\{4:6\}_{k+1} \rightarrow b_{g_{k+1}} = b_{g_{k+1}} \tag{4.81}$$

$$x_{k+1} = 0_{6\times 1} \tag{4.82}$$

A further analysis and implementation of this AHRS algorithm is explored in Chapter 6.

4.3.4 Navigation Position Update

Combining the knowledge of step detection, step length estimation, and attitude determination, the pedestrian's position can be propagated in the navigation frame. The PDR-SD approach relies on two key assumptions:

1. The pedestrian is walking along a planar surface (North and East) and the Downward component is not considered.

2. The pedestrian is facing the walking direction. The PDR-SD system would fail if the pedestrian were walk in a different direction from the direction the user is facing (i.e. walking backwards, side stepping, spinning while walking, etc.).

With these assumptions in mind, the pedestrian's position only needs to be updated by mapping the step length from the heading in the navigation frame. From [19, 4] the position propagation is given by:

$$\begin{bmatrix} E_{k+1} \\ N_{k+1} \end{bmatrix} = \begin{bmatrix} E_k \\ N_k \end{bmatrix} + SL_k \cdot \begin{bmatrix} sin\left(\psi_k\right) \\ cos\left(\psi_k\right) \end{bmatrix} \tag{4.83}$$

The new East and North positions at time t_{k+1} are propagated from the positions at time t_k by distance SL_k in the direction of ψ_k. A major difference between the PDR-SD and PDR-INS systems is the time update occurrences. The PDR-SD system maintains a constant position estimate until a new step is detected. With all the sections of PDR-SD defined, the PDR-SD methodology can be implemented for a chest-mounted IMU scenario.

4.3.5 PDR-SD Implementation and Summary

Performance examination of the PDR-SD implementation was conducted on two different real-world data sets of a pedestrian walking a closed loop path along a sidewalk on one side of a street, then going around a round about to walk back to the starting point. For both data sets the pedestrian was equipped with a Vectornav VN-100 IMU [30] and a Ublox ZED-F9P GPS receiver [31] with a single antenna. The IMU was mounted to a vest in line with the pedestrian's sternum and the antenna for the GPS receiver mounted on the pedestrian's right shoulder. A reference path is provided by the GPS receiver where the reported positions were used for truth, and the IMU was used for step detection, as well as, tracking the pedestrian's attitude. For both of the results shown, the pedestrian's step

length is assumed constant at 0.8 meters and the heading is provided by the output of the AHRS algorithm.

The first data set was conducted with the pedestrian traveling at a consistent pace and step length with a recorded step count of $1,289$ steps taken. However, the second data set was conducted with the pedestrian walking at a constant pace and step length from the starting position at the beginning of the street to the position before the roundabout on the other side of the street, then the pedestrian's pace increased until reaching the crosswalk where the pace was decrease to less than the initial pace with a recorded step count of $1,089$ steps taken. Figure 4.10 shows the path results of the first data set with the PDR-SD implementation (cyan) with the GPS receiver positions (magenta) provided as a truth reference.

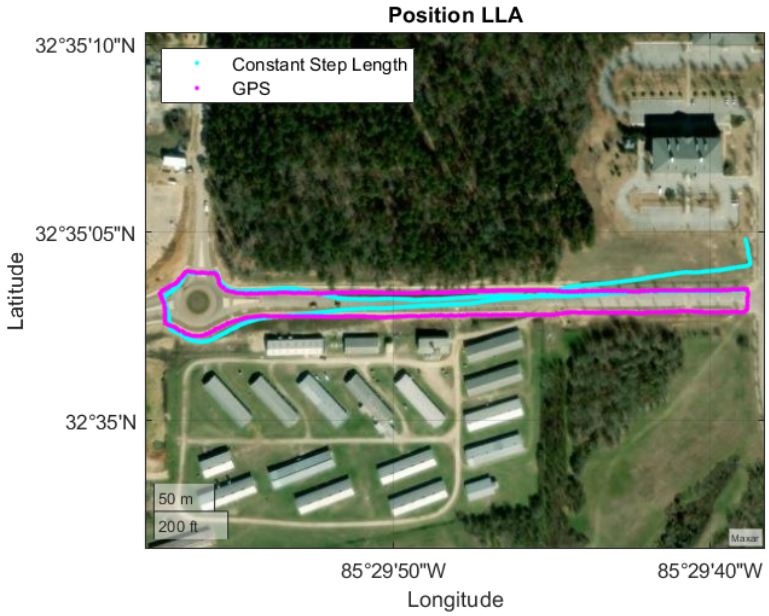


Figure 4.10: PDR-SD Path Results

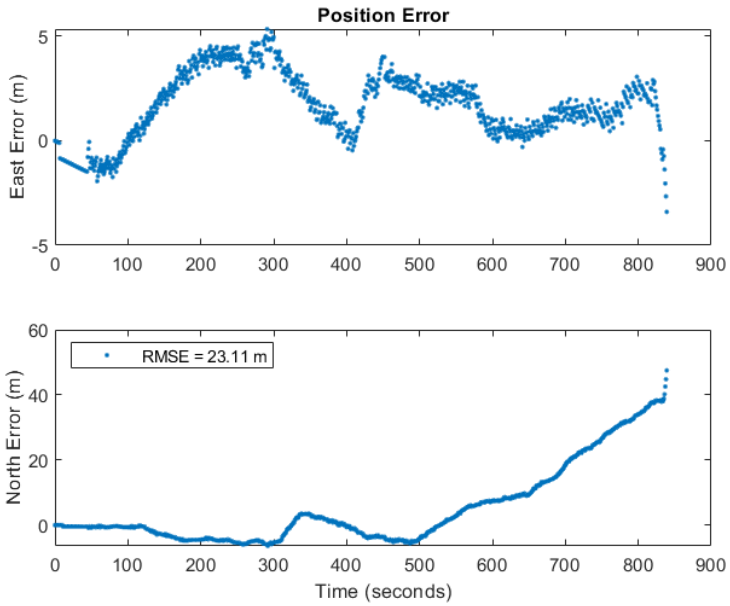


Figure 4.11: PDR-SD Path Error in East and North directions

Figure 4.11 shows the path errors for the first data set implementation in the East and North directions, as well as, the root mean squared error for each direction. The path errors were calculated by interpolating the PDR positions at the recorded receiver header times. The total distance traveled from the receiver positions was calculated to be $1,040.63$ meters, while the estimated distance traveled from the PDR-SD implementation was calculated to be $1,074.41$ meters. One thing to notice is the change in heading that results in most of the path errors shown. This is due to the presence of disturbances in the local magnetic field while walking.

Figure 4.12 shows GNSS positions and the portions of the route where the pedestrian changes pace from normal, to fast, and then slow. Figure 4.13 shows the path results of the second data set with the PDR-SD implementation (cyan) and the GNSS receiver positions (magenta) provided as a truth reference.

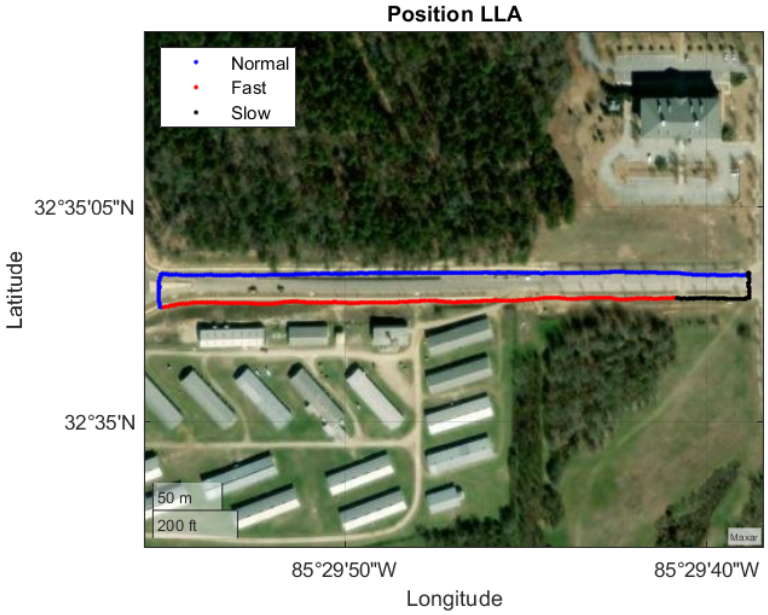


Figure 4.12: PDR-SD Path with Changing Pace

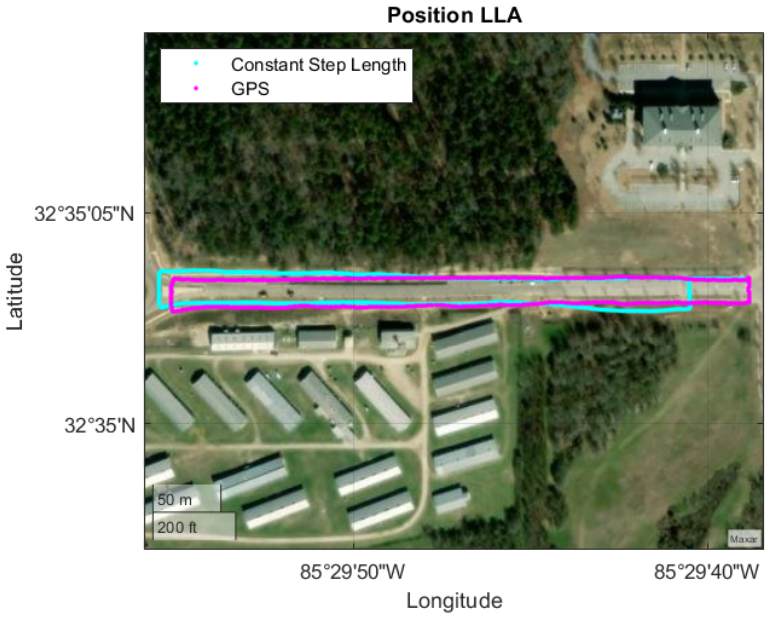


Figure 4.13: PDR-SD Path Results for Varying Walking Pace

Figure 4.14 shows the path errors for the second data set implementation in the East and North directions, and the corresponding root mean squared error for each direction. The path errors were calculated the same way as the first data set. The total distance traveled from the receiver positions was determined to be 901.13 meters, while the estimated distance traveled from the PDR-SD implementation was calculated to be 914.17 meters.

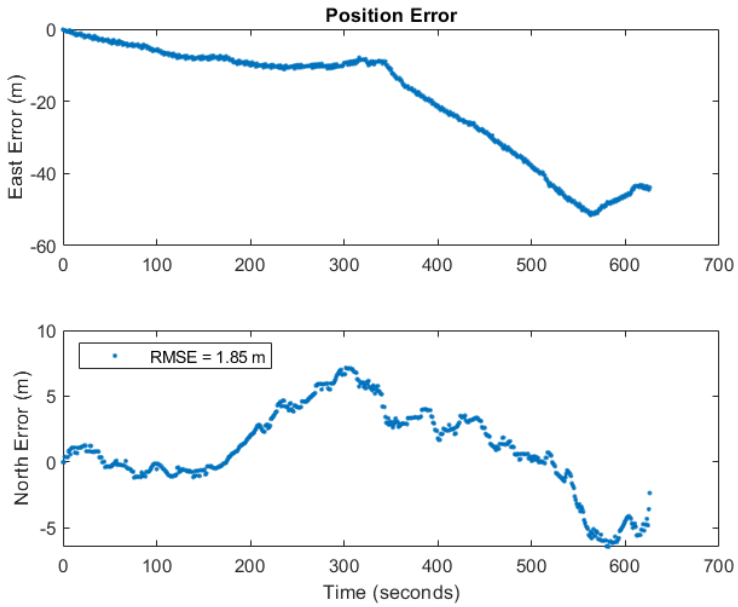


Figure 4.14: PDR-SD Path Error in East and North directions

The PDR-SD approach utilizes the pedestrian's torso bounce between steps, as well as, the lower dynamics experienced by an IMU mounted on the torso, rather than the foot. However, the accuracy of the PDR-SD approach is dependent on the accuracy of the step length estimation and the assumption that magnetic disturbances are not present to corrupt the heading estimates.

4.4 Drawbacks to PDR

As mentioned above, both approaches to PDR are prone to errors that will accumulate over time, another drawback to both approaches is the problem of initialization. In the case of the PDR-SD approach, initial pitch and roll estimates from gravitational measurements,

and initial heading estimates from magnetic field measurements provide an initial attitude in the navigation frame. The other requirement for initialization is an initial position estimate. With this in mind, a standalone PDR-INS or PDR-SD system is not sufficient for a global navigation solution. GNSS provides the missing link for both position initialization, and the mitigation of dead reckoning error growth over time. The next chapter will discuss the architecture of fusing GNSS and PDR for foot-mounted and torso-mounted implementations.

Chapter 5

Loosely Coupled GNSS/PDR Integration

5.1 Introduction

As mentioned in the previous chapter, GNSS measurements offer a solution to the initialization problem of pedestrian navigation. Due to its wide availability, GNSS provides a solution of geolocation for challenging environments that other localization techniques could not provide. For example, a maritime vehicle traveling across the ocean has little to no landmarks that can be used as a reference point when estimating its location. Whether reference landmarks are obtained by direct visibility, radios, or DR, their availability is not always ensured. However, GNSS allows for such a maritime vehicle to obtain readily available positioning and guidance information almost anywhere. However, these solutions are only readily available in environments that support visibility to the satellite system. In the case of positioning objects of interest whose visibility of these satellites is heavily obstructed (i.e. underground, indoors, urban cities, dense foliage, etc.), GNSS does not offer such a solution. Therefore, the process of integrating GNSS with dead reckoning techniques can offer a fused solution that provides such information for a much larger diversity of environments. This chapter presents the existing methods of integrating GNSS measurements with DR techniques to produce a fused solution for pedestrian navigation.

One of the most simplistic and intuitive methods of integrating GNSS measurements with

DR techniques is through a loosely coupled architecture. The term *loosely coupled* refers to using position, velocity, and timing (PVT) measurements for the system integration [2], rather than a *tightly coupled* integration that uses range and range-rate measurements between the receiver and the satellite. Such an integration allows for the GNSS measurements to correct uncertainties in the navigation state estimates resulting from errors in the DR estimates, and, in some cases, the DR estimates correcting navigation states due to errors in GNSS measurements. However, correcting estimates from GNSS measurements with DR estimates is not as common due to the higher fidelity in GNSS measurements and accumulation of errors from the DR system. The next two sections will present the integration of GNSS position and velocity measurements with the INS approach to PDR, and integration with the step detection approach to PDR.

5.2 GNSS/PDR-INS Integration

The INS approach to pedestrian navigation with foot-mounted IMUs use mechanization equations to produce estimates of the navigation states. Primarily, the position, velocity, and attitude of the pedestrian within the navigation frame. Aiding these navigation state estimates with GNSS position and velocity measurements correct errors in the state estimates from the INS. This aiding process, typically, falls into one of two architectures:

- Open-loop integration
- Closed-loop integration

Open loop integration estimates errors in the INS position and velocity states with GNSS measurements of position and velocity, which are then used to correct the INS to produce an integrated navigation solution [18, 2]. The *open-loop* term is based on the fact that the GNSS measurements do not interfere with the INS, and a standalone INS solution is still produced alongside the integrated solution. Closed-loop integration estimates the errors in the INS states, and feeds the error estimates back to the INS equations to produce the integrated solution. The closed-loop scheme offers better long term stability by, essentially, resetting the INS to the corrected state estimates and prevent the errors in the INS from growing unbounded. Generally, the closed loop integration scheme often includes the estimation of the sensor biases in the IMU, though this is not a requirement. Figure 5.1 shows the

closed-loop architecture for loosely coupled GNSS/INS integration, where the navigation error state estimates from the INS Kalman Filter are:

$$\hat{e} = \left[\delta\hat{r}^n_{INS} \;\; \delta\hat{v}^n_{INS} \;\; \delta\hat{\psi}^n_{INS} \;\; \hat{b}_g \;\; \hat{b}_a\right]^T \tag{5.1}$$

The error states are used to correct the INS navigation states, as well as, estimate sensor biases within the IMU error model to reduce errors in the IMU measurements.

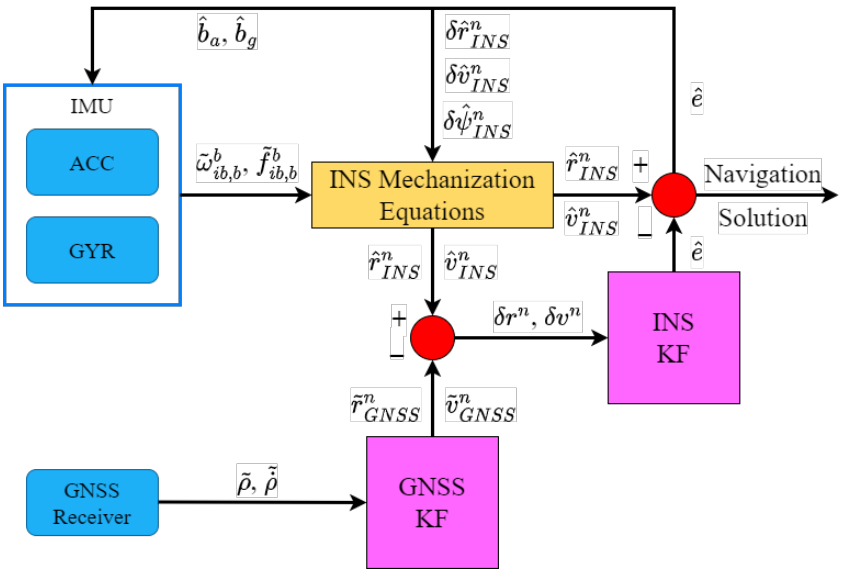


Figure 5.1: Loosely Coupled GNSS/INS Integration with closed-loop scheme

In many frameworks for GNSS/INS integration, incorporating the lever arm distance between the INS and the GNSS antenna may be required when constructing the measurement updates. For vehicles where a GNSS antenna and INS have been rigidly mounted at different positions, the measurement model would include the lever arm, to reduce the position and velocity error and is given by:

$$\hat{r}^i_{ia,G} = \hat{r}^i_{ib} + \hat{C}^i_{ib} l^b_{ba} \tag{5.2}$$

The term $\hat{r}^i_{ib}$ is the estimated position of the INS, $\hat{C}^i_{ib}$ is the orientation of the INS with respect to the vehicle frame, and l^b_{ba} is the position of the GNSS antenna with respect to the INS. The inclusion of the lever arm when coupling is often taken into consideration for systems where the INS and GNSS antenna relative positions are rather large and held constant. With GNSS/INS integration for pedestrian navigation with foot-mounted inertial

sensors, these criteria are, typically, invalid. When the pedestrian is walking, the lever arm distance, as well as, the orientation between the IMU and the GNSS antenna are constantly changing. When navigating along the horizontal plane, the horizontal lever arm is also relatively small. Therefore, the lever arm is often not considered when integrating the systems for foot-mounted pedestrian navigation.

For the INS approach to pedestrian navigation, coupling GNSS measurements with INS state estimates in a closed-loop scheme offer long-term stability, as long as GNSS measurements remain available with high fidelity. In the event GNSS is no longer available, the INS errors will continue to grow unbounded. Therefore, INS techniques such as: ZUPTs, ZARUs, and SWDs from Section 4.2 are often included within the loosely coupled framework to provide longer stability of the DR solution, as well as, observability of sensor biases in the IMU. Figure 5.2 uses a similar framework to that of the generic loosely coupled framework shown in Figure 5.1, with their measurement updates from the INS techniques supplied to the INS Kalman Filter.

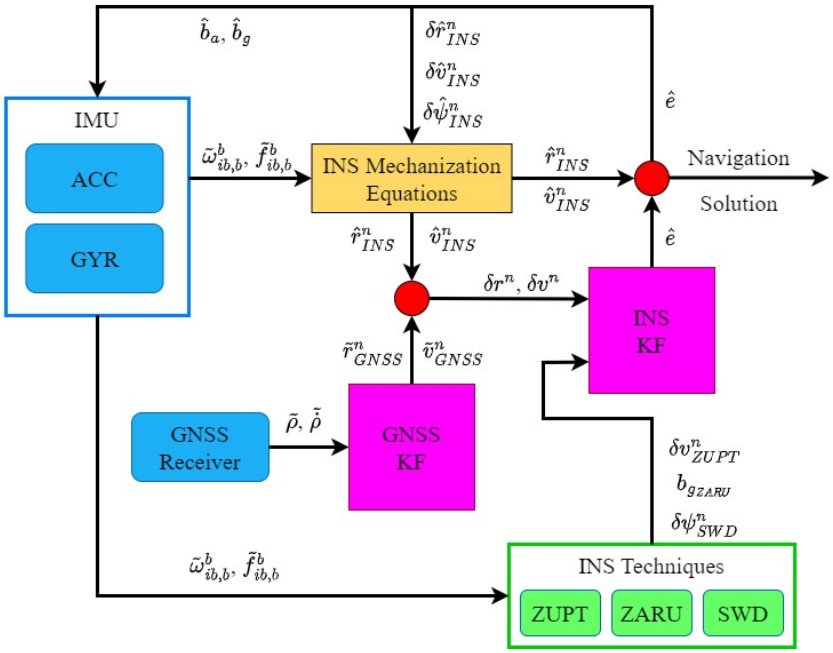


Figure 5.2: Loosely Coupled GNSS/INS Integration for PDR

5.2.1 GNSS Measurement Update

The GNSS position and velocity measurements are derived from the receiver's pseudorange and pseudorange rate measurements that have been passed through the receiver's Kalman Filter. Since this thesis uses the receiver's position and velocity as the only measurements from GNSS, the design of the GNSS KF and the derivation of these measurements will not be included. Information on these derivations can be found in [2]. The GNSS measurement update is used to provide observability of the position and velocity error states for the discrete-time system model presented in Section 4.2.5, where the measurement model is given by:

$$\delta \mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \nu_k \qquad \nu_k \sim N\left(0, \sigma^2\right) \tag{5.3}$$

The measurement vector ($\delta \mathbf{y}_k$) is given by:

$$\delta \mathbf{y}_k = \begin{bmatrix} \tilde{r}^n_{GNSS_k} - \hat{r}^n_{INS_k} \\ \tilde{v}^n_{GNSS_k} - \hat{v}^n_{INS_k} \end{bmatrix} \tag{5.4}$$

and the measurement matrix ($\mathbf{H}$) is given as:

$$\mathbf{H} = \begin{bmatrix} I_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & I_3 & 0_3 & 0_3 & 0_3 \end{bmatrix} \tag{5.5}$$

The measurement covariance matrix is the variance in the position and velocity measurements that are assumed to be uncorrelated in time given by:

$$\mathbf{R} = \begin{bmatrix} \sigma_r^2 & 0_3 \\ 0_3 & \sigma_v^2 \end{bmatrix} \tag{5.6}$$

These values are determined by the accuracy of the GNSS measurements and can be evaluated by models relating various parameters such as: satellite geometry, signal-to-noise ratio, clock errors, etc. Details of tuning the measurement noise covariance matrix can be found in [2].

5.2.2 GNSS/PDR-INS Implementation and Summary

Implementation of integrating GNSS position and velocity measurements with a foot-mounted INS in a loosely coupled framework was performed on a real-world data set with the pedestrian walking a clock-wise path around a parking lot. The pedestrian was fitted with

a Vectornav VN-100 IMU mounted to the right foot, and a Ublox ZED-F9P GNSS receiver with a single antenna mounted on the left shoulder. Both devices reported measurements that were post-processed in MATLAB and integrated in a loosely coupled framework. The results will be shown for three different scenarios in the order they are presented where:

1. GNSS is constantly available for the duration of the implementation.
2. GNSS is initially available, then an outage is simulated for a period of time and then becomes available again.
3. GNSS is initially available, then an outage is simulated for the rest of the implementation.

GNSS Constantly Available

The test for the first scenario examines the fused solution where GNSS measurements are constantly available for the duration of the implementation. Figure 5.3 shows the path results for this case.

Figure 5.3: GNSS/PDR-INS Path with constant GNSS availability

With the constant availability of GNSS, the PDR-INS estimates are corrected when a new position and velocity measurement is recorded. The path errors for this scenario are shown in Figure 5.4.

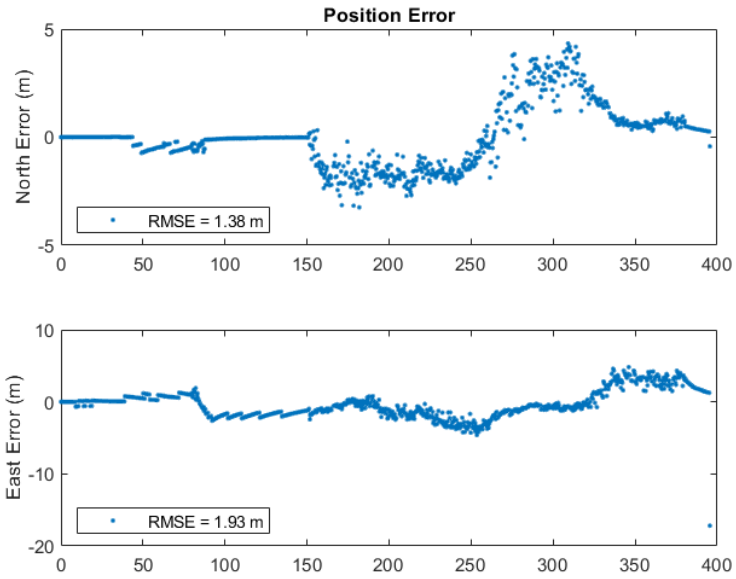


Figure 5.4: North and East Path Errors

GNSS Partial Outage

The second scenario for the implementation examines the fused solution when GNSS is not available for a small window of time during the experiment. Figure 5.5 shows the path results for the implementation.

Figure 5.5: GNSS/PDR-INS Path with GNSS partial outage

The outage is simulated 224 seconds into the integration, where no position or velocity measurements from GNSS were used. The outage lasts for about 60 seconds before it becomes available again to correct errors in the PDR-INS. The path errors for this scenario are shown below in Figure 5.6.

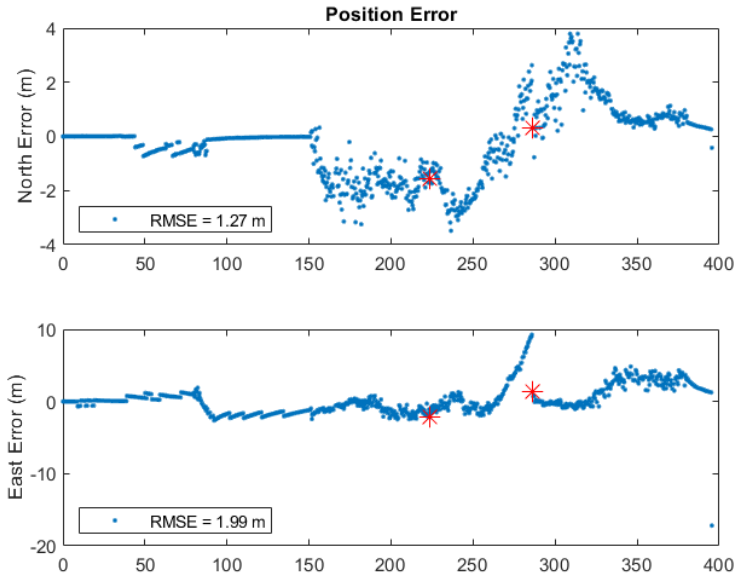


Figure 5.6: North and East Path Errors

GNSS Full Outage

The final scenario for implementation examines the fused solution when GNSS signal is lost at a certain instance in time and does not return. This scenario shows how the fused solution accuracy only relies on the PDR-INS solution for the remainder of the experiment after GNSS signal is lost. Figure 5.7 shows the path results for the implementation.

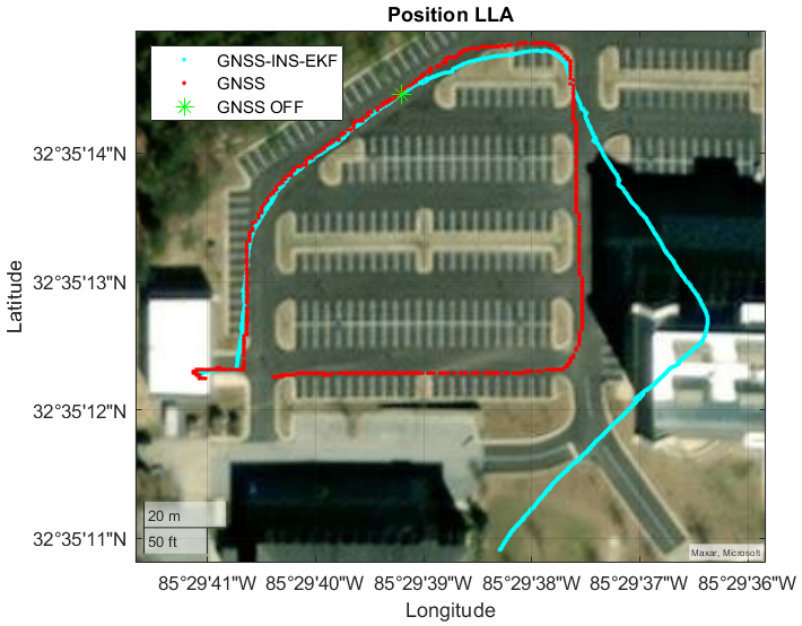


Figure 5.7: GNSS/PDR-INS Path with Loss of GNSS

Initially, the INS position and velocity estimates are corrected by the GNSS measurements. However, once GNSS is lost entirely, the navigation solution, solely, relies on the PDR-INS techniques to DR. The path errors for this scenario are shown below in Figure 5.8.

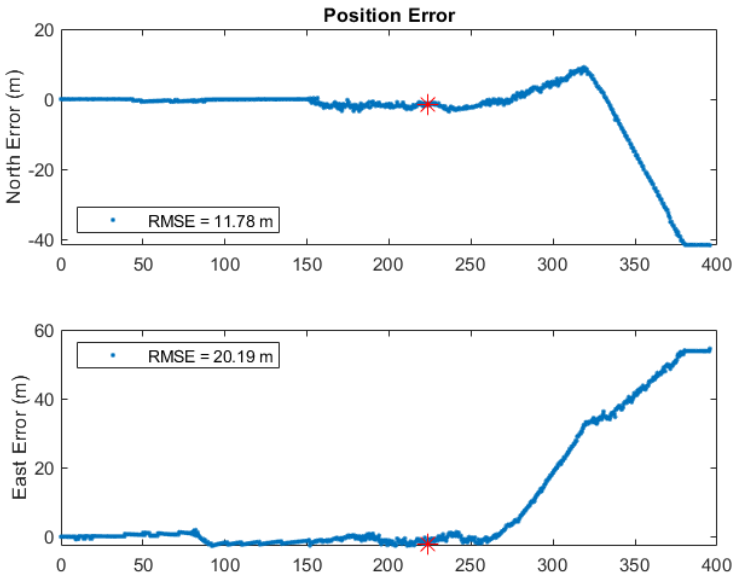


Figure 5.8: North and East Path Errors

As expected, without GNSS measurements to correct errors in the INS solution, the resulting path errors grow unbounded. This scenario of total loss in GNSS would be most likely to occur after the pedestrian moves into an environment where no satellite visibility exists (i.e. indoors, underground, dense foliage, etc.). Simulating this scenario shows how quickly the pedestrian's error in position grows without GNSS to aid it. The next section will discuss the integration of GNSS and PDR using a body-mounted approach.

5.3 GNSS/PDR-SD Integration

As discussed in Section 4.3, the step detection approach to pedestrian navigation utilizes a chest-mounted IMU to propagate the pedestrian's position when steps are detected. Unlike the INS approach, the IMU is not fully mechanized. Rather, the accelerometer acts as a pedometer to propagate the pedestrian's position based on the detection of steps and estimated heading. Coupling the step detection approach with GNSS position and velocity measurements correct errors in these navigation state estimates that arise from errors in step length estimation, as well as, heading estimation. The coupling architectures for GNSS with

the step detection approach to PDR vary based on the step length and heading estimation algorithms and have been proven in existing literature. A comprehensive analysis of the loosely coupled architectures is beyond the scope of this thesis. For simplicity, a zero-order hold system model with a constant step length estimator and dead reckoned angular velocity measurements is used in this thesis. Consider the discrete time navigation states from Section 4.3.4:

$$E_{k+1} = E_k + SL_k \cdot \sin(\psi_k) \tag{5.7}$$

$$N_{k+1} = N_k + SL_k \cdot \cos(\psi_k) \tag{5.8}$$

The current heading, ψ_k, is determined from dead reckoning angular velocity measurements as shown in Equation (4.64) and the step length estimate, SL_k, is determined by the constant step length model in Equation (4.51).

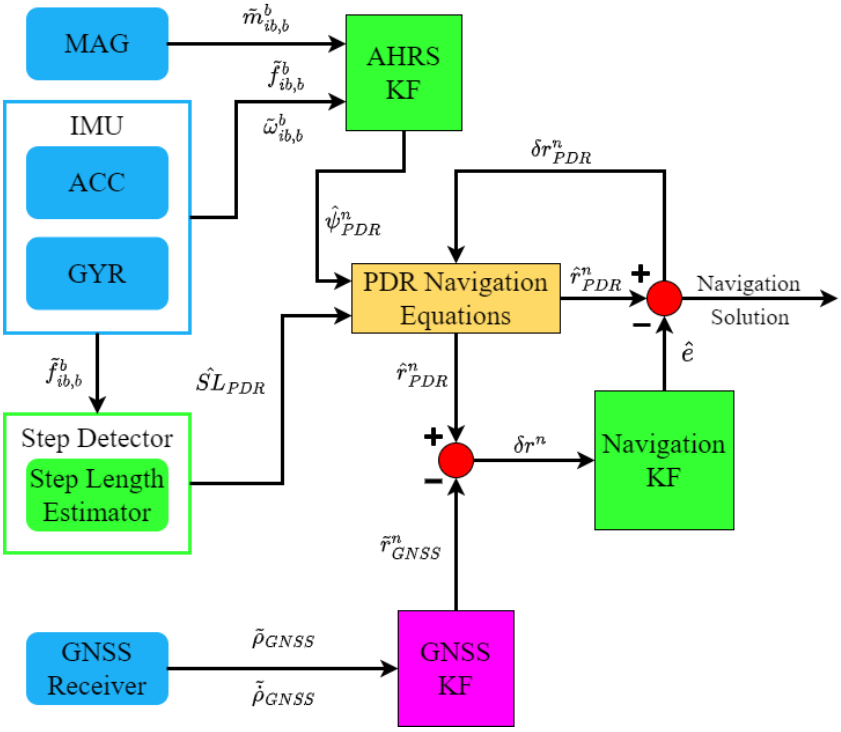


Figure 5.9: Loosely Coupled GNSS/PDR Integration Framework

Figure 5.9 shows the architecture for the loosely coupled integration of GNSS measurements with PDR. Since the position state estimates are held constant until a new step is

detected, a velocity state is not considered. Thus, GNSS velocity measurements are not necessary. Contrary to the GNSS/INS framework that estimates 15 states, only two states are estimated for this framework. The computational efficiency is improved, however, the accuracy of the navigation solution is determined by the fidelity in the step length and heading estimation.

5.3.1 GNSS Measurement Update

For the zero-order hold model, only the position measurements from the GNSS receiver's Kalman Filter are needed to update the current position in the North and East components, as well as, update the step length estimates. For the discrete-time system model in Section 4.3.4, the GNSS measurement model is given as the position measurement in the East and North directions:

$$\mathbf{y}_k = \begin{bmatrix} \tilde{E}^n_{GNSS_k} \\ \tilde{N}^n_{GNSS_k} \end{bmatrix} + \nu_k \qquad \nu_k \sim N\left(0, \sigma^2\right) \tag{5.9}$$

The measurement matrix ($\mathbf{H}$) is given as:

$$\mathbf{H} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.10}$$

Constructing the measurement noise covariance matrix ($\mathbf{R}$) is done with the variance in each component of the position measurement that are assumed to be uncorrelated and given by:

$$\mathbf{R} = \begin{bmatrix} \sigma^2_E & 0 \\ 0 & \sigma^2_N \end{bmatrix} \tag{5.11}$$

However, the variance in each component is not always available or simple to determine. Therefore, the measurement noise covariance matrix is given as:

$$\mathbf{R} = \begin{bmatrix} \sigma^2_r & 0 \\ 0 & \sigma^2_r \end{bmatrix} \tag{5.12}$$

where the variance in the East and North directions are just the variance in the position measurement.

5.3.2 GNSS/PDR Implementation and Summary

Implementation of integrating GNSS position measurements with a body-mounted IMU in a loosely coupled framework is performed on a real-world data set with the pedestrian walking the same clock-wise path along a sidewalk as shown in the PDR-SD implementation in Section 4.3. This data set was selected due to a lack in observed magnetic disturbances, and for the changing step length at different intervals of the route. The pedestrian was fitted with a Vectornav VN-100 IMU mounted to the right of the pedestrian's sternum, and a Ublox ZED-F9P GNSS receiver [31] with a single antenna mounted on the left shoulder. Both devices reported measurements that were post-processed in MATLAB and integrated in the loosely coupled architecture. The results will be shown for three similar scenarios as the foot-mounted approach on a different route. However, the results will include three different step length estimation approaches discussed in Section 4.3.2:

- The constant step length model
- The waist-swing model
- The step frequency model

The three step length estimation models will be shown for each of the scenarios. To reiterate, the scenarios are given as:

1. GNSS is constantly available for the duration of the implementation.
2. GNSS is initially available, then an outage is simulated for a period of time and then becomes available again.
3. GNSS is initially available, then an outage is simulated for the rest of the implementation.

The three scenarios will be presented in succession and the results for each step length model will be discussed.

GNSS Constantly Available

The first scenario for implementation examines the fused solution where GNSS position measurements are constantly available for the duration of the implementation. Figure 5.10 shows the results of the implementation with all three step length estimation approaches.

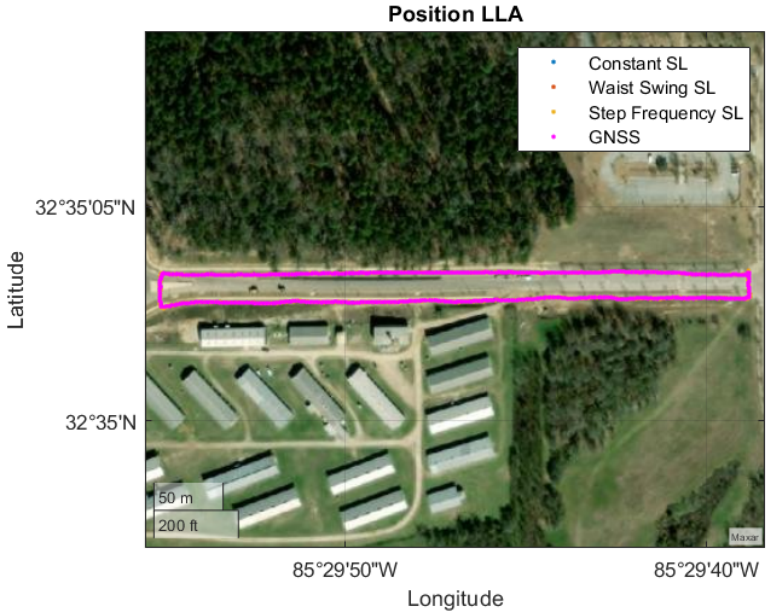


Figure 5.10: GNSS/PDR-SD Path with constant GNSS availability

As expected, the constant availability of GNSS allows for the position estimates from the PDR-SD to be corrected to reduce path errors. Figure 5.11 shows the individual path errors for each step length estimation method.

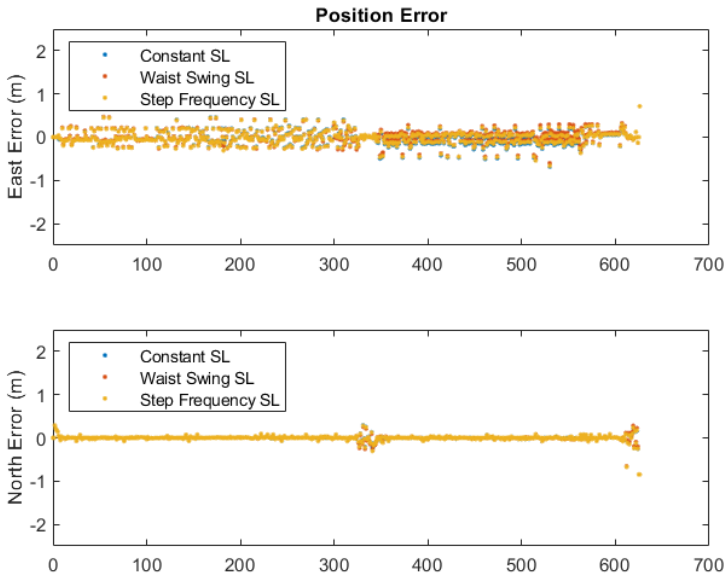


Figure 5.11: North and East Path Errors

Table 5.1: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Constant	0.159	0.070
Waist Swing	0.154	0.071
Step Frequency	0.157	0.071

The step length estimation path errors in the North and East directions are shown above in Figure 5.11. Additionally, the root mean squared error in each direction is provided in Table 5.1. With the constant availability of GNSS, all three methods perform to produce path errors of less than half a meter.

GNSS Partial Outage

The second scenario for implementation examines the fused solution when a GNSS outage occurs for a brief period of time. Figure 5.12 shows the path results for the implementation.

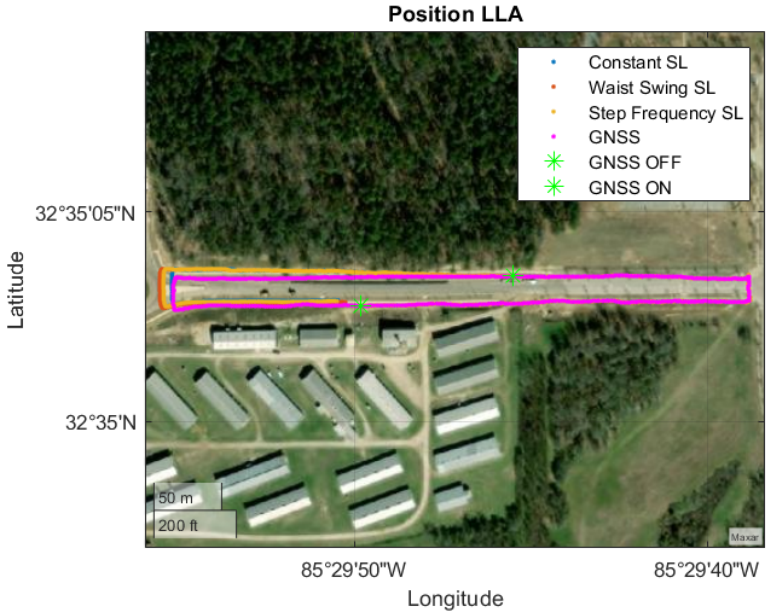


Figure 5.12: GNSS/PDR-SD Path with GNSS partial outage

It can easily be seen that the constant step length estimation method deviates from the true path, significantly, while the waist swing method and the step frequency method stay much closer to the true path during the outage. The corresponding path errors in the North and East directions are shown below in Figure 5.13.

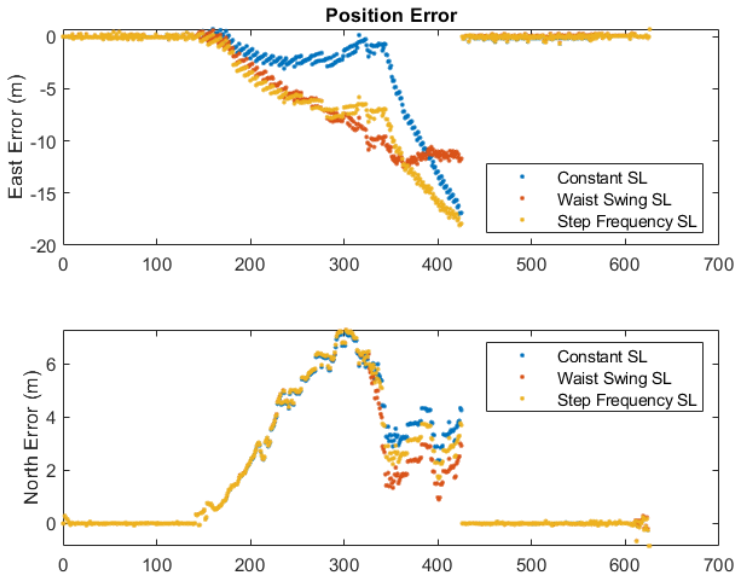


Figure 5.13: North and East Path Errors

Table 5.2: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Constant	4.117	2.919
Waist Swing	5.348	2.723
Step Frequency	5.985	2.848

During the partial outage, the constant step length estimation method shows less error in the Eastward direction once the outage begins. However, once the pedestrian's step length increases, the waist swing method shows a consistent error. Due to the comparatively shorter path changes in the North and South directions, as well as consistent heading estimates from the AHRS heading, path errors are constrained in the North direction for all three step length estimation methods.

GNSS Full Outage

The final scenario examines the fused solution when a GNSS outage occurs indefinitely. The GNSS outage begins at the same instance as the partial outage, except GNSS does not become available again. Figure 5.14 shows the path results for the full GNSS outage.

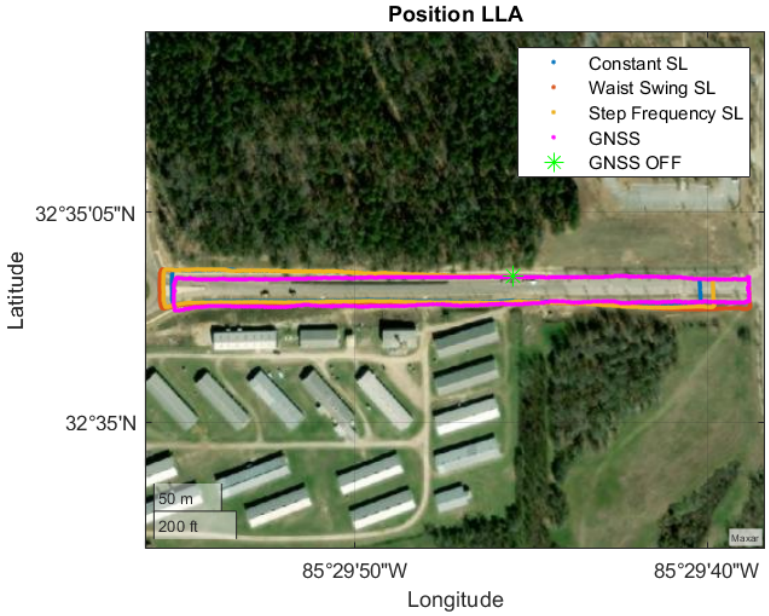


Figure 5.14: GNSS/PDR-SD Path with full outage of GNSS

After the full GNSS outage, the navigation is produced solely by the PDR-SD with each step length estimation method for the remainder of the experiment. The path error results are shown below in Figure 5.15 for the North and East directions and summarized in Table 5.3.

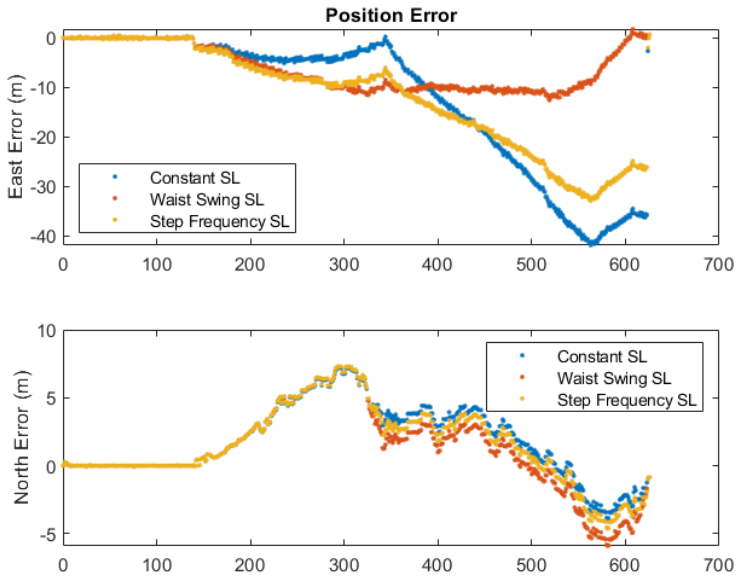


Figure 5.15: North and East Path Errors

Table 5.3: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Constant	18.443	3.236
Waist Swing	7.629	3.221
Step Frequency	15.767	3.198

Once the full outage begins, the Eastward path error for the constant step length estimation method outperforms the other step length estimation methods. However, once the pedestrian's step length increases and decreases, the waist swing method outperforms the other two methods in the East direction, while, simultaneously, being outperformed in the North direction.

Chapter 6

Extended Loosely Coupled GNSS-SD Integration for PDR Improvement

6.1 Introduction

The INS and Step Detection methods to PDR, as described in Chapter 4, offer techniques to mitigate the effect of sensor errors on the navigation solution. However, these techniques only offer knowledge of the pedestrian's change in position and orientation, and not an initial absolute position and orientation. Thus, the PDR system is faced with the additional problem of requiring a known initial global position and attitude in order to navigate within a global frame.

For initializing the pedestrian's attitude (heading, pitch, and roll), measurements of the Earth's gravitational pull and unperturbed magnetic field measurements can provide this necessary information. Using the Earth's gravity model allows for an initial pitch and roll angle to be determined, assuming that the pedestrian is standing perpendicular to the Earth's surface, while unperturbed magnetometer measurements provide a reference for the pedestrian's global heading with respect to magnetic north [3, 42, 21, 41]. However, this heading reference requires that the magnetometer's frame be in line with the pedestrian's

frame. For the problem of initializing the global position, GNSS measurements again provide such information when satellite visibility is high (i.e. outdoors with full view of the sky) [42, 43].

However, when GNSS is not readily available, the PDR system will act alone to construct the navigation solution whose accuracy is reflected in the system's ability to properly determine change in position and orientation. When comparing a foot-mounted and torso-mounted PDR system, the higher dynamics experienced by the pedestrian's foot combined with the changes in relative position and orientation of the IMU and GNSS receiver require high grade hardware and faster software to construct a robust PDR solution. Whereas, a torso-mounted system experiences lower dynamics and fewer changes in relative position and orientation. With this in mind, the remainder of this chapter presents the proposed novel methods of integrating GNSS measurements with PDR techniques for a torso-mounted IMU to provide an improved PDR solution when GNSS is no longer available.

6.2 System Architecture

The proposed methods of GNSS and PDR integration are based on the loosely coupled approach where GNSS position and velocity measurements are used in conjunction with the PDR-SD techniques to produce a fused navigation solution [2]. The overall system is divided into three subsystems:

1. Step length estimation
2. Modified AHRS for heading estimation
3. Navigation state estimation

While these subsystems embody the same navigation techniques as shown for the loosely coupled GNSS/PDR-SD integration from Section 5.3, each subsystem has been extended and use GNSS for parameter estimation.

The block diagram of the system architecture is shown below in Figure 6.1. The three subsystems, indicated by the green blocks, extend on the loosely coupled architecture described in Section 5.3. The standard loosely coupled architecture only utilizes GNSS position measurements to mitigate the errors in the PDR navigation states from Equation (5.7).

The architecture for the proposed integration system extends and modifies the standard architecture in four ways:

1. The step length estimator subsystem utilizes GNSS measurements to estimate parameters of a novel step length model in a KF framework.

2. The AHRS subsystem utilizes GNSS measurements to correct for misalignment between the pedestrian and hardware.

3. The AHRS subsystem utilizes a novel algorithm for detecting and rejecting faulty magnetometer measurements.

4. The navigation subsystem produces a navigation solution using alternative PDR navigation equations and utilizes GNSS measurements to correct errors in the PDR solution.

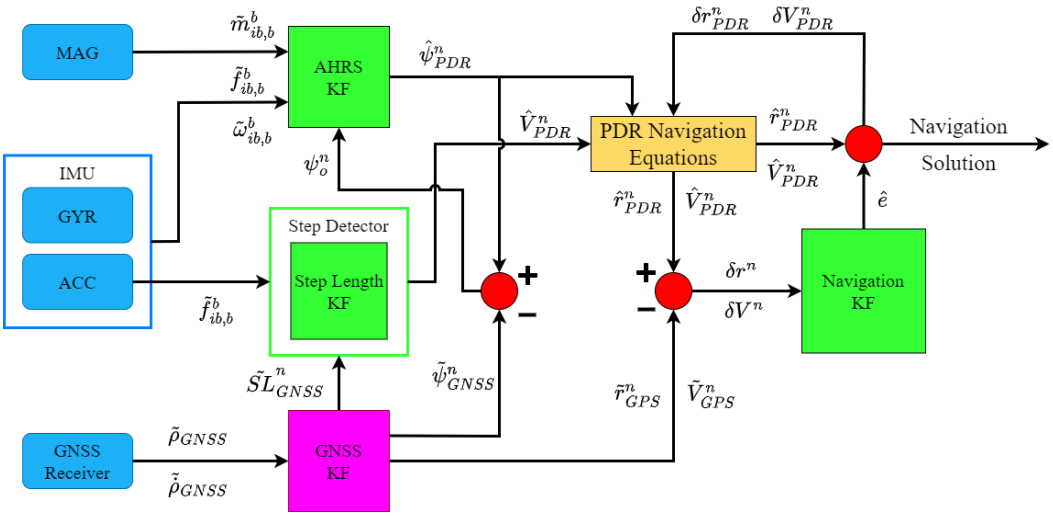


Figure 6.1: Loosely Coupled Architecture for GNSS/PDR-SD Integration

The following sections will present the modifications and extensions to each subsystem in succession of how they are listed above.

6.3 Step Length Estimation

In order to estimate the pedestrian's step length, a reference for the measured step length needs to be compared with the estimated step length to ensure accuracy of the step length estimates. One approach to providing a reference step length measurement is to interpolate GNSS position measurements at the times that the steps are detected. In the case of a constant step length estimator, averaging the interpolated GNSS positions converges to a step length that can be assumed to be constant when a new step is detected. However, a constant step length estimate can lead to erroneous position propagation when considering a change in the pedestrian's gait cycle. As discussed previously in Section 4.3.1, a waist-swing approach to step length estimation can provide a more robust estimate for erroneous steps. Consider the proposed waist-swing method given by Equation 6.1:

$$SL = K_1 W + K_2 \tag{6.1}$$

where the step length is a function of a scale factor constant (K_1) multiplied by the fourth root of the difference between the maximum and minimum acceleration norm (W), and a constant (K_2).

6.3.1 Interpolated GNSS Position RLS Approach

In order to determine the appropriate values of the constants in Equation 6.1, a recursive least squares (RLS) estimator is used for the model:

$$\mathbf{SL_k} = \Phi \mathbf{x_k} \tag{6.2}$$

The state vector ($\mathbf{x}$) and the state transition matrix (Φ) are given by:

$$\mathbf{x} = \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} \qquad \Phi = \begin{bmatrix} W_k & 1 \end{bmatrix} \tag{6.3}$$

Using the interpolated GNSS positions as true step length measurements, the measurement model is given as:

$$\mathbf{y}_k = \Phi_k \mathbf{x}_k \tag{6.4}$$

The RLS method was performed on a data set where a total of 424 steps were taken and the reported GNSS positions were interpolated to provide measurements of the pedestrian's

step length. Figure 6.2 shows the RLS results of the step length model estimates and the difference in interpolated GNSS position measurements as step length measurements of the RLS algorithm. After the final GNSS step length measurement updates the model, the parameters reach a final value of:

$$K_1 = 0.2939 \qquad K_2 = 0.2270 \tag{6.5}$$

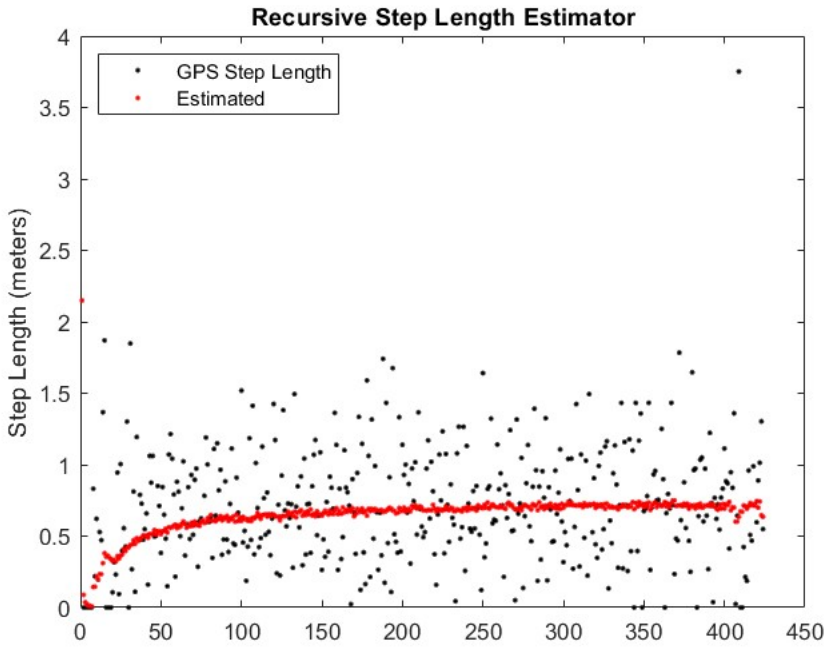


Figure 6.2: Recursive Least Squares Step Length Estimation

While using interpolated GNSS positions is an intuitive approach to measuring the pedestrian's step length, GNSS does not provide the accuracy needed to directly measure step length. As shown in Figure 6.2, the interpolated GNSS positions often reported step lengths greater than one meter, and occasional step lengths greater than a meter and a half. Interpolated GNSS positions also prove to be difficult to perform in real time, especially when GNSS is unavailable. An alternative method is to provide a pseudo-measurement of the pedestrian's step length using GNSS position and velocity.

6.3.2 GNSS Step Length Pseudo-Measurements

If a GNSS position and velocity measurement is available in between steps, then a step length pseudo-measurement of the pedestrian's step length can be determined and utilized as a measurement update for the step length estimator.

Position-Velocity Pseudo-Measurement

The first method determines the pedestrian's step length by calculating the distance between the position of the previous step and the measured GNSS position, and adding the integrated GNSS velocity measurement over the time between the measurement and the next step. This method makes the assumption that the pedestrian's velocity is constant between the time the measurement is received and the time the next step occurs. For example, in Figure 6.3 if a new step is detected at time $t = t_{s+1}$, a pseudo-measurement of step length can be provided by the GNSS:

$$\tilde{SL}_{GNSS} = d_r + d_v + \eta \tag{6.6}$$

where η is represented as zero-mean and Gaussian measurement noise, d_r is the change in position from time t_s to t_n, and d_v is the integrated GNSS velocity measurement from time t_n to t_{s+1} given by:

$$d_r = \sqrt{(E_n - E_s)^2 + (N_n - N_s)^2} \qquad d_v = V_n \cdot \Delta t = \sqrt{\dot{E}_n^2 + \dot{N}_n^2} \cdot (t_{s+1} - t_n) \tag{6.7}$$

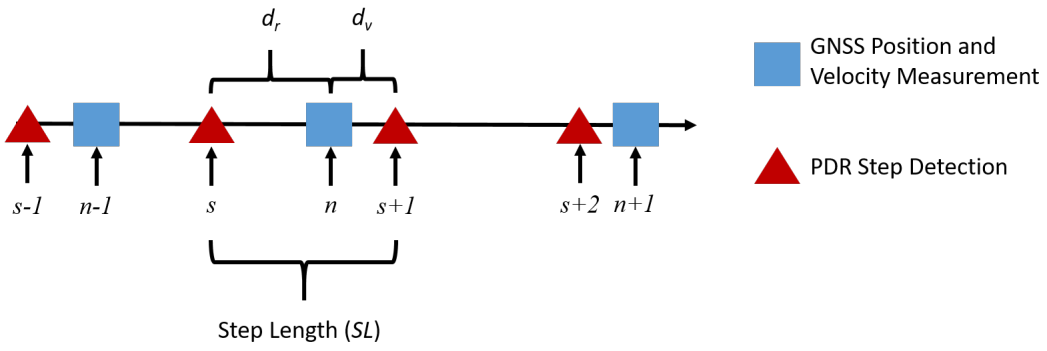


Figure 6.3: GNSS Step Length Position-Velocity Pseudo-measurement

This method is referred to as the position-velocity pseudo-measurement due to the fact that the GNSS step length is a function of the previous estimated step position and not

a true measurement of the pedestrian's step length. It should also be noted that GNSS step length pseudo-measurements are not always available even under ideal conditions for GNSS measurements. For example, from time t_{s+1} to t_{s+2}, as shown in Figure 6.3, a GNSS position and velocity measurement is not obtained during the time between steps. In this case, a step length measurement update is not performed and the step length is estimated by the model in Equation 4.63.

Velocity Pseudo-Measurement

The second method determines the pedestrian's step length by integrating the GNSS velocity measurement over the total time between steps. This method makes the assumption that the pedestrian's velocity is constant over the difference in time between steps, rather than the difference in time between receiving the GNSS measurement and the next step occurrence. For example, in Figure 6.4 if a new step is detected at time $t = t_{s+1}$, a pseudo-measurement of step length can be provided by the GNSS:

$$\tilde{SL}_{GNSS} = d_v + \eta \tag{6.8}$$

where η is represented as zero-mean and Gaussian measurement noise and d_v is the integrated GNSS velocity measurement from time t_s to t_{s+1} given by:

$$d_v = V_n \cdot \Delta t = \sqrt{\dot{E}_n^2 + \dot{N}_n^2} \cdot (t_{s+1} - t_s) \tag{6.9}$$

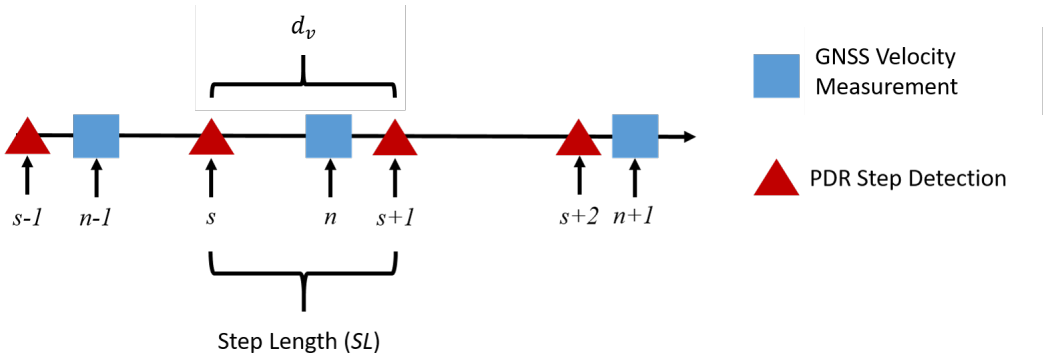


Figure 6.4: GNSS Step Length Velocity Pseudo-measurement

This method is referred to as the velocity pseudo-measurement due to the fact that the GNSS step length is only a function of the integrated velocity measurement between

steps and not a true measurement of the pedestrian's step length. Similar to the position-velocity pseudo-measurement, a measurement update is only applied when a GNSS velocity measurement is obtained between successive steps.

A drawback to the step length pseudo-measurements is the accuracy of the position and velocity measurements from the GNSS receiver. Inaccuracies in these measurements can cause large variations in the GNSS step length that are not indicative of realistic step lengths. As seen in Figure 6.5, there are still step length position-velocity pseudo-measurements that exceed what would be in the realm of a realistic step length (i.e. step lengths greater than 1.5 meters).

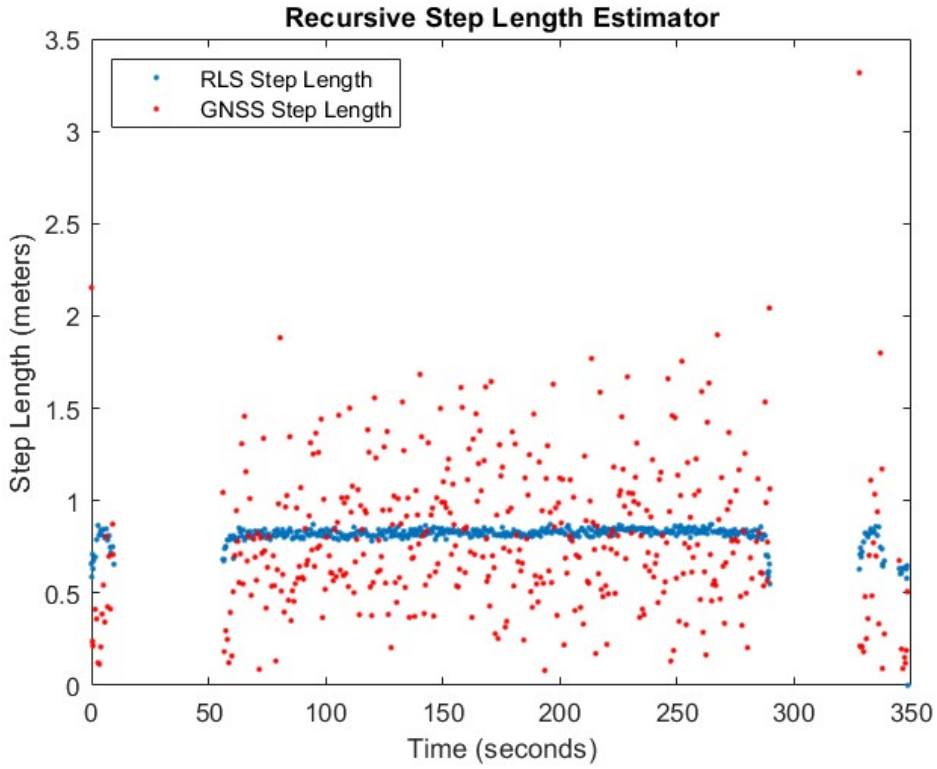


Figure 6.5: RLS Step Length Estimator with GNSS pos-vel pseudo-measurements

In Figure 6.6 below, an RLS approach was examined using the step length velocity pseudo-measurements. It can be seen that the variation in measurements is much lower than the position-velocity pseudo-measurements. However, there appear to be outliers that report step length measurements much larger than the largest position-velocity pseudo-

measurements.

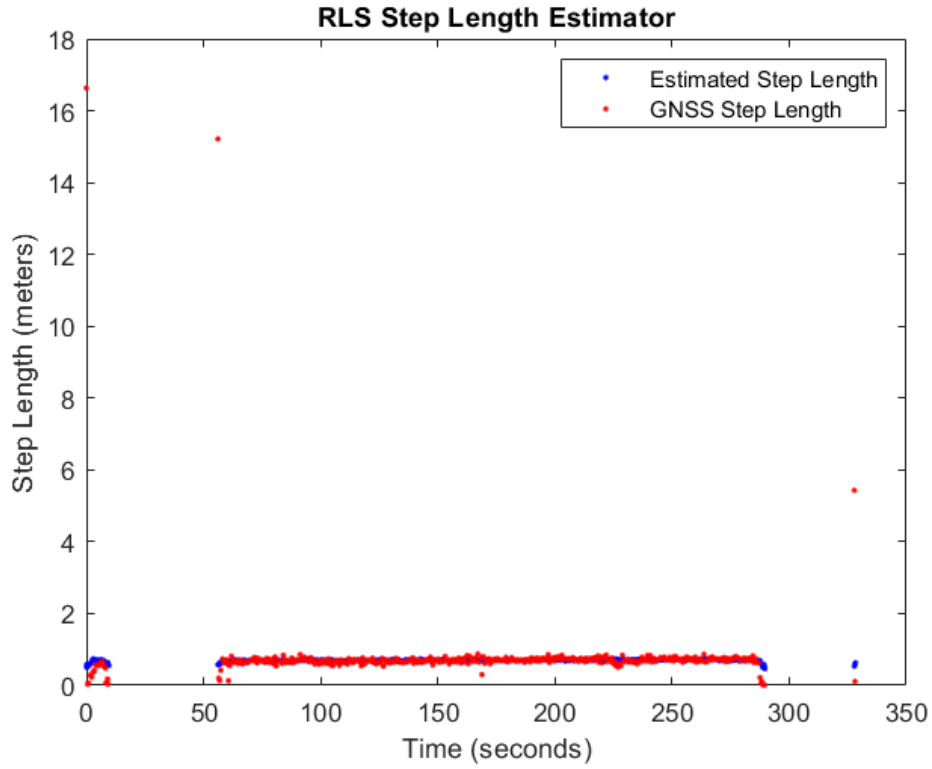


Figure 6.6: RLS Step Length Estimator with GNSS velocity pseudo-measurements

The accuracy of the RLS approach is dependent on the quality of the GNSS step length pseudo-measurements, as well as the initialization of the model parameters and their uncertainties. The RLS measurement update is given by:

$$x_k = x_k + P_k \Phi_k (y_k - \Phi_k x_k) \tag{6.10}$$

$$P_k = P_k - P_k \Phi_k^T (I + \Phi_k P_k \Phi_k^T)^{-1} \Phi_k P_k \tag{6.11}$$

The term $P_k \Phi_k \in (0, 1)$ can be seen as the gain term that adjusts the state estimate (x_k) by an amount proportional to the measurements innovation ($y_k - \Phi_k x_k$). If P_k is very small, $P_k \Phi_k$ will approach zero and the adjustments made to the model parameters by the measurement innovation will also be very small. This phenomenon is referred to as the filter "going to sleep". If P_k is very large, $P_k \Phi_k$ will approach 1 and the adjustments made to the model parameters will be the result of the measurement innovation. As new measurements

are received, P_k gets diminished over time resulting in smaller adjustments to the model parameters. This will introduce a problem in the RLS estimator that the model parameters will no longer be adjusted as new measurements are received. One method to prevent this problem is to introduce a forgetting factor term to "de-weight" older error samples. Another alternative is the use of a Kalman filter framework. The next section will introduce a Kalman filter framework for the step length model parameter estimation.

6.3.3 Step Length Kalman Filter (SLKF)

The linear Kalman filter framework accomplishes the same goal as the RLS algorithm, by estimating model parameters as states that evolve over time. As discussed in the RLS algorithm, no process noise is considered on the model parameters. As the term P_k is minimized with new measurements, the filter will "go to sleep" and new measurements will have little to no effect on the estimation of parameters. Using the KF approach, uncertainty in the model parameters and the measurements can be taken into account to allow new measurements to make adjustments to the model parameter estimates. The SLKF models the step length parameters as constants in a linear system with additive noise:

$$\mathbf{x}_{k+1} = \Phi \mathbf{x}_k + w_k \tag{6.12}$$

$$\mathbf{y}_k = H_k x_k \tag{6.13}$$

where the state transition matrix, Φ, and observation matrix, H_k, are given by:

$$\Phi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad\qquad H_k = \begin{bmatrix} W_k & 1 \end{bmatrix}$$

and the process noise, w, is zero-mean and Gaussian white noise:

$$w_{K_1} \sim N\left(0, \sigma^2_{K_1}\right) \tag{6.14}$$

$$w_{K_2} \sim N\left(0, \sigma^2_{K_2}\right) \tag{6.15}$$

Determining the process noise of the model parameters is not intuitive. The advantage of the step length model is the adaptive nature of determining the pedestrian's step length. This is due to the influence of the leading term $K_1 \cdot W$, where the estimated step length changes based on the acceleration experienced between steps. If the K_1 term is estimated to zero, then the resulting step length model becomes a constant step length estimator where:

$$SL = K_2 \tag{6.16}$$

This is not ideal for scenarios where the pedestrian's step length fluctuates. The evolution of the model parameters in the KF can be studied based on the ratio of the model parameters' noise terms:

$$\gamma = \frac{\sigma_{K_1}^2}{\sigma_{K_2}^2} \tag{6.17}$$

In the measurement update of the KF, the adjustment of the step model parameters is given by:

$$\mathbf{x}_k = \mathbf{x}_k + K_k(y_k - H_k\mathbf{x}_k) \tag{6.18}$$

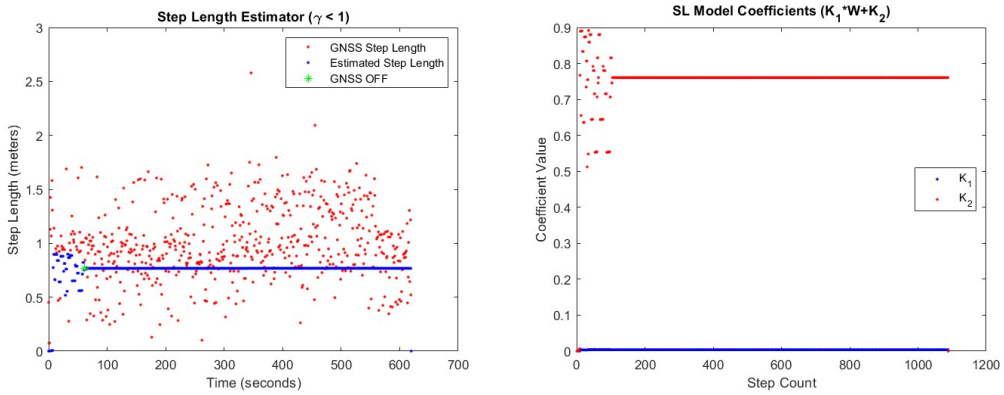
$$K_k = \frac{P_k H_k^T}{H_k P_k H_k^T + R} \tag{6.19}$$

The term P_k is the state covariance that is propagated forward in time by:

$$P_k = \Phi P_k \Phi^T + Q \tag{6.20}$$

The term Q represents the process noise matrix. Increasing the process noise of one parameter will result in a larger Kalman gain that adjusts that model parameters estimate more than the other.

A real-world data set was used to examine the effect of the process noise ratio on the resulting step length model. Figure 6.7 shows the results of the experiment when $\gamma << 1$ and how the model parameters converge in such a way that the step length estimates become constant with little to no variation. This would lead to large errors in the event the pedestrian changes pace.

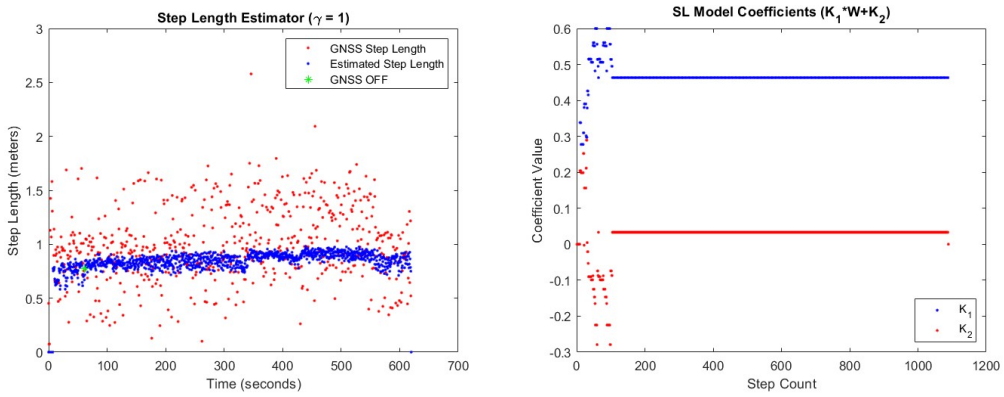


(a) Step Length Estimates

(b) Model Parameters

Figure 6.7: SLKF with $\gamma << 1$

Considering the effect of $\gamma = 1$, as shown in Figure 6.8, where the process noise terms of the model parameters are equivalent. In this case, both model parameters are able to converge in such a way that the step length estimates vary as the pedestrian changes pace. Around 340 seconds the pedestrian accelerates and increases the step length, and decelerates around the 550 second mark to a slower pace with a smaller step length. However, it can also be seen that the K_2 coefficient is estimated to be less than zero at several points during the estimation.

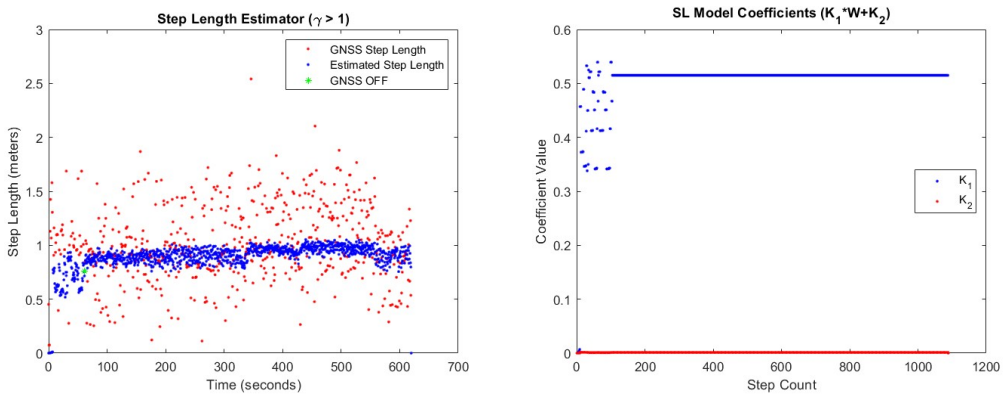


(a) Step Length Estimates

(b) Model Parameters

Figure 6.8: SLKF with $\gamma = 1$

In the case where $\gamma >> 1$, as shown in Figure 6.9, a similar effect is found with the model parameters behavior when $\gamma << 1$. But, instead of the leading term approaching zero, the constant term of the step length model converges to zero. This results in the step length model being represented very close to the waist-swing model in Equation (4.53) with a large variance in the step length estimates.



(a) Step Length Estimates

(b) Model Parameters

Figure 6.9: SLKF with $\gamma >> 1$

Ideally, both coefficients should be estimated to be greater than zero, while still yielding variable step length estimates. Consider the input term of the step length model:

$$W = \sqrt[4]{\max\left(\| f_{ib}^{b} \|\right) - \min\left(\| f_{ib}^{b} \|\right)} \tag{6.21}$$

Taking the squared inverse of the input term with each new step results in a nonzero value that is less than one, as shown in Figure 6.10. Using this information as the process noise of the leading coefficient, K_1, the process noise of coefficient K_2 can be set to always be less than that of K_1 by taking the inverse cubed of the input term. This allows for both coefficients to be estimated without forcing either coefficient to converge to zero.

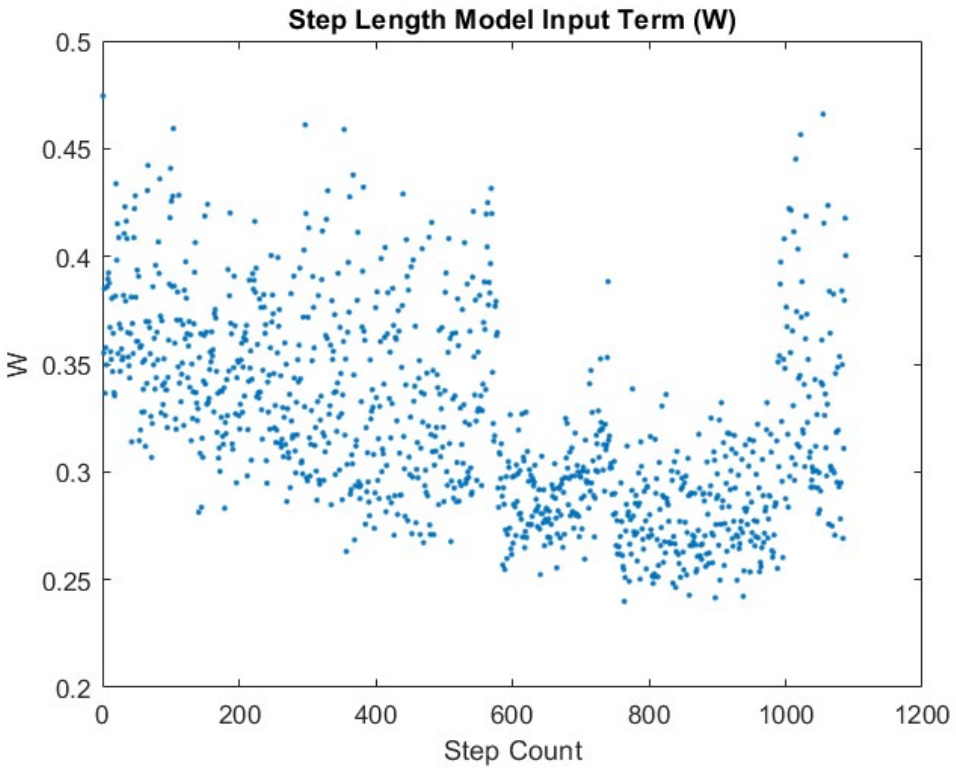
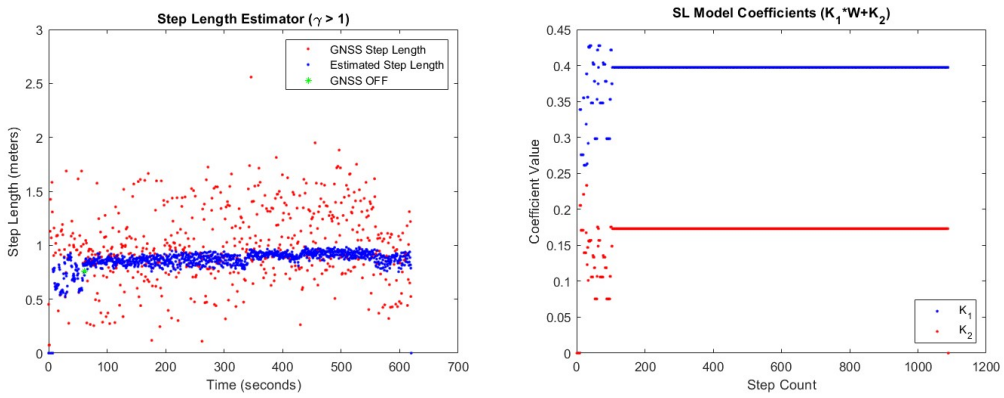


Figure 6.10: Step Length Model Inputs by Step Count

Figure 6.11 shows the implementation of selecting the process noise and how both model parameters converge to be nonzero, and neither coefficient is estimated to be less than zero. While the results of this implementation are very similar to that of the scenario where $\gamma >> 1$, the variation in estimated step lengths is much lower. During periods where the pedestrian maintains a relatively constant pace, the step length variance should be relatively low.



(a) Step Length Estimates

(b) Model Parameters

Figure 6.11: SLKF with $\gamma > 1$

The measurement model is similar to the RLS approach, but with the inclusion of measurement noise where:

$$\mathbf{y}_k = H_k \mathbf{x}_k + \nu_k \tag{6.22}$$

where the measurement matrix and measurement noise are given by:

$$H_k = \begin{bmatrix} W_k & 1 \end{bmatrix} \qquad \nu \sim N\left(0, \sigma^2_{GNSS}\right) \tag{6.23}$$

The advantage of the SLKF approach is that the inclusion of process noise on the dynamic model prevents the filter from "going to sleep" and new measurements will continue to adjust the model parameters as they are provided. Additionally, the inclusion of measurement noise provides the opportunity of weighting the step length pseudo-measurements to prevent bad measurements from being considered. In the case of pseudo-measurements that measure the step length to be greater than 1 meter, the measurement noise can be set as large to indicate a bad measurement and not have a great effect on the model parameters. Similarly, pseudo-measurements than measure a step length to be less than 0.5 meters can be weighted to also have a large measurement noise. This ensures that only measurements of good quality will effect the estimation of the model parameters.

This thresholding technique to weight the GNSS step length pseudo-measurements can

be given in the form of a logical expression where:

$$R_{GNSS} = \begin{cases} 0.1 & \text{if } 0.5\text{m} \leq SL_{GNSS} \leq 1\text{m} \\ 1000000 & \text{if otherwise} \end{cases} \tag{6.24}$$

The large value of R_{GNSS} based on whether the pseudo-measured step length was too large or too small, is arbitrarily set to indicate a low level of trust and mimic the idea of rejecting the measurement. The small value of R_{GNSS} was set to indicate a higher level of trust in the pseudo-measurement.

6.4 AHRS Heading Correction

As previously discussed in Section 4.3.3, the utilization of a magnetometer aids an AHRS by providing measurements of the Earth's magnetic field strength to give a global heading reference. While the incorporation of these measurements are useful to constrain the drift in heading from dead reckoning angular velocity measurements, the accuracy of this implementation relies on two key assumptions:

1. The magnetometer measurements are aligned with the body frame of the pedestrian.
2. Measurements of the Earth's magnetic field strength are static and are not perturbed by local magnetic disturbances.

These assumptions are critical for effective performance of an AHRS integrated with PDR.

6.4.1 Misalignment Estimation

The first assumption requires that the measurements of the IMU and the magnetometer be aligned with the body frame of the pedestrian at all times. As shown in Figure 6.12, if the forward direction of the magnetometer (ψ_m) differs from the forward direction of the pedestrian (ψ_p) by some offset angle (ψ_o), the PDR system will not propagate the pedestrian's position in the direction of the true heading.

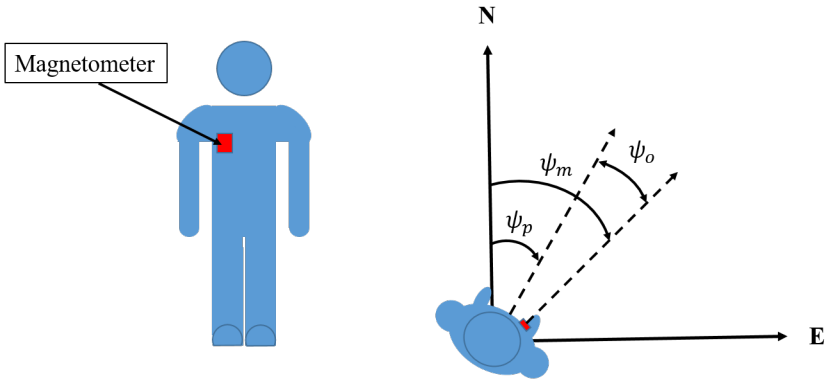


Figure 6.12: Heading offset from Magnetometer mounting

One method of reducing the offset angle between the magnetometer and pedestrian heading is to mount the magnetometer to be perfectly in line with the body frame. While this is an intuitive solution, it can often be difficult to achieve as knowledge of the pedestrian's true heading is not always known. However, even minute offsets will introduce errors that grow over time.

With this problem in mind, a novel method is proposed to reduce the heading offset with GNSS course measurements. While course and heading angles can often be the same, they have very different definitions. Course refers to the angle of the path traveled from one location to another, while heading is the angle that the body of interest is facing. This key difference is best realized when considering an aircraft in motion as depicted in Figure 6.13.

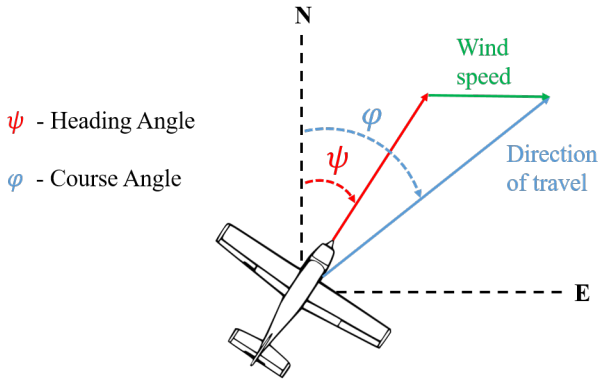


Figure 6.13: Aircraft heading and course angle

While the aircraft's nose is directed in one angle, the effect of the direction of the oncoming wind causes the aircraft to travel in a different direction. In the case of ground vehicles and pedestrian's, the heading angle and course angle are often equal due to the fact that vehicles and pedestrian are, typically, facing in the direction of travel. Therefore, course measurements from GNSS can be directly calculated in two ways:

1. The change in at least two position measurements in the North and East direction.

2. Velocity measurements in the North and East direction.

Both methods provide insight to the direction the pedestrian is traveling with different degrees of accuracy.

Determining course from the change in position measurements requires two position measurements at separate epochs in time given by:

$$\phi = \tan^{-1}\left(\frac{r_{E_k} - r_{E_{k-n}}}{r_{N_k} - r_{N_{k-n}}}\right) \tag{6.25}$$

The terms $r_{E_{k-n}}$ and $r_{N_{k-n}}$ are the east and north position measurements from the receiver at epoch t_{k-n}. It should be noted that the change in position measurements do not require sequential epochs. For example, if a GNSS receiver reported position measurements using a single antenna traveling straight towards north with no deviation in eastern position, then the angle of travel would be determined to be 0 degrees N from epoch t_k to t_{k+1}, as well as, from epoch t_k to t_{k+2}. However, the effectiveness of this method for course determination is dependent on the accuracy of the position measurements, as well as the position differences between epochs. To examine the effectiveness of this implementation, the same data set from Section 5.3.2 was examined using a Vectornav VN-100 IMU [30] with an onboard magnetometer mounted to the right of the pedestrian's sternum, as well as, a Ublox ZED-F9P GNSS receiver [31] with a single antenna mounted to the pedestrian's right shoulder. The pedestrian travels due west for about 400 meters, and then crosses a street and walks due east back to the initial position.

Figure 6.14 and 6.15 shows the results of the experiment with the heading from the AHRS compared to the course measurements from position differences. Initially, the course measurements from position are erroneous due to the small differences with respect to the initial position measurement. However, over larger position differences, the course measurements approach the true heading (90 degrees W). After the pedestrian has traveled about

32 meters west, 32 position measurements have been received and the course error is 0.89 degrees, or 1% of error. However, the course measurements from position differences are not able to consistently stay within 1% of error until the pedestrian has traveled about 117 meters and 97 position measurements have been received.

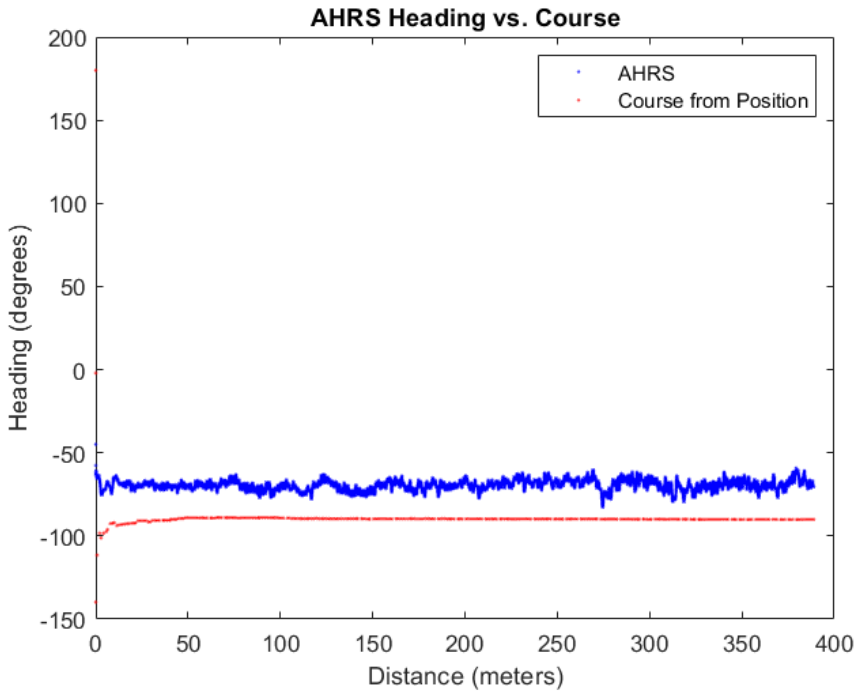


Figure 6.14: Heading and Course from Position Measurements Comparison

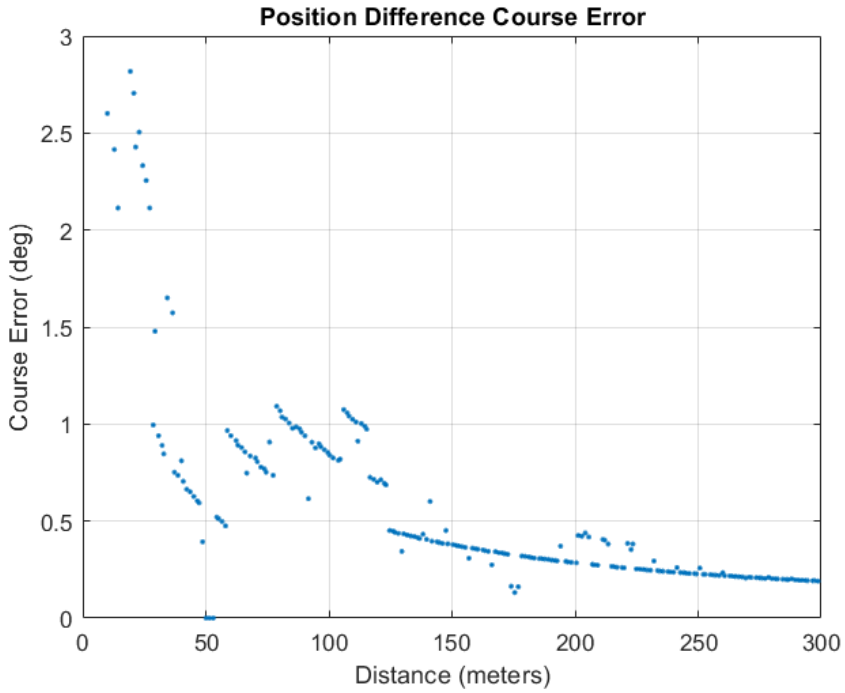


Figure 6.15: Position Difference Course Error

Examining the error in course measurements from position differences reveals brief periods where the error suddenly increases and begins to approach zero again. Over the distance of 25 meters to 125 meters, there are four periods showing these sudden increases in course error. The root of this trend can be found when examining the changes in the north direction during this period. Figure 6.16 shows the north position measurements over the walking distance. Changes in the pedestrian's north position are very small while walking directly west. However, examining the north position from 50 to 125 meters reveals a sudden change in the north position by about 0.4 meters at each interval. From Equation (6.25) a course measurement of 90 degrees west requires that the change in north position relative to the origin be zero degrees. As the difference in north positions increase over these intervals, the measured course drifts away from 90 degrees west and begin to approach zero as the difference in east positions grow larger.

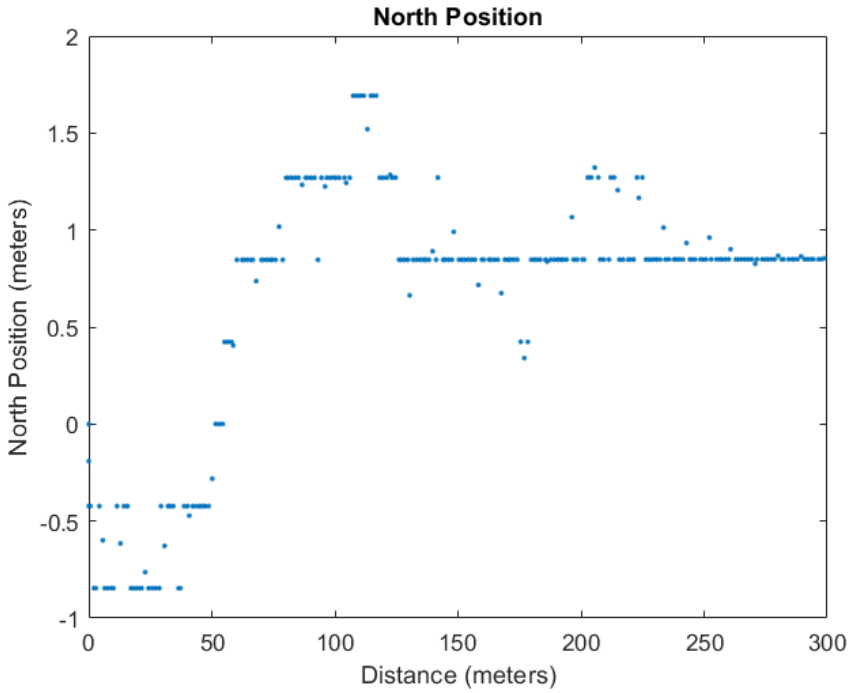


Figure 6.16: North Position Over Walking Distance

Using larger position differences proves to be more suitable for determining course. However, if the pedestrian is not traveling in a constant direction, this method will diverge from the true course and the misalignment will no longer be apparent. Figure 6.17 shows the results of course determination for the same data set when the pedestrian changes the direction of travel. The course measurements show a constant direction of travel for the first 300 seconds when the pedestrian is traveling due west. After about 300 meters, the pedestrian crosses a street and changes direction walking due east and returns to the initial starting point.

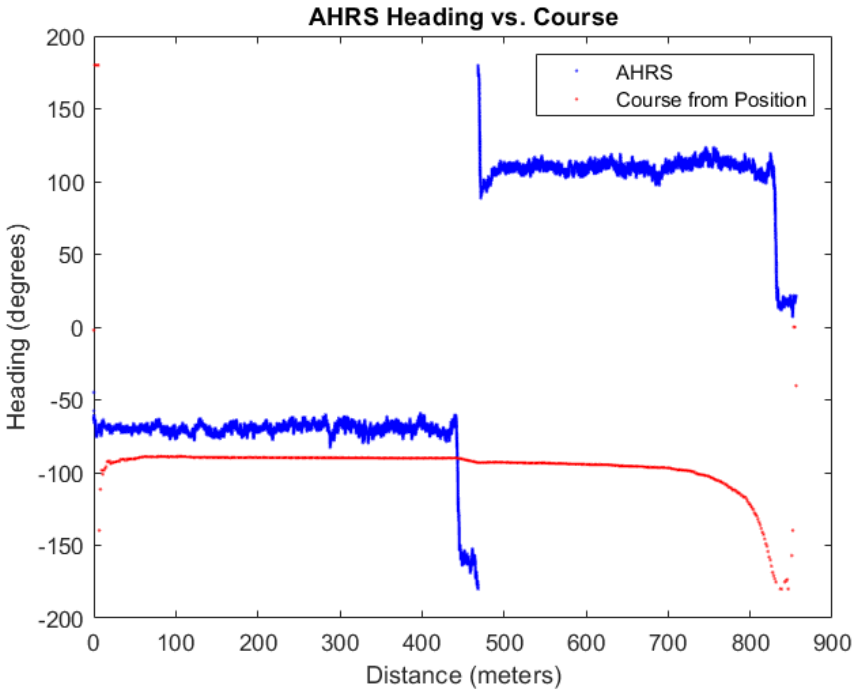


Figure 6.17: Course from Large Position Differences with Changing Direction

The use of position differences for course measurements prove to be accurate and stable over large distances when the pedestrian is traveling in a constant direction. However, the need for long distances in a constant direction may not always be available.

Determining course from the measured velocity in the north and east direction can be calculated with single velocity measurements at a single epoch with a single antenna. The velocity derived course measurement is given by:

$$\phi = \tan^{-1}\left(\frac{V_E}{V_N}\right) \tag{6.26}$$

where V_E is the measured velocity in the eastward direction, and V_N is the measured velocity in the northward direction from GNSS.

The ability to account for the misalignment between the magnetometer's measured heading and the pedestrian's heading requires an assumption that the magnetometer is rigidly mounted to the pedestrian and not free to sway. If this criteria is met, the offset between the magnetometer heading and the pedestrian's heading is constant. To visualize this effect of constant offset, the same data set from Section 6.3 was used where a Vectornav VN-

100 IMU [30] with an onboard magnetometer was mounted to the right of the pedestrian's sternum. As seen in Figure 6.18, the misalignment results in a constant offset between the magnetometer heading and the velocity derived course measurements from a walking experiment.

Figure 6.18 shows the experimental results of using the course from velocity measurements. The first 7 velocity measurements produce course measurement errors that are over 15% of the true heading and remain within 8% of the true heading for the remainder of the route.

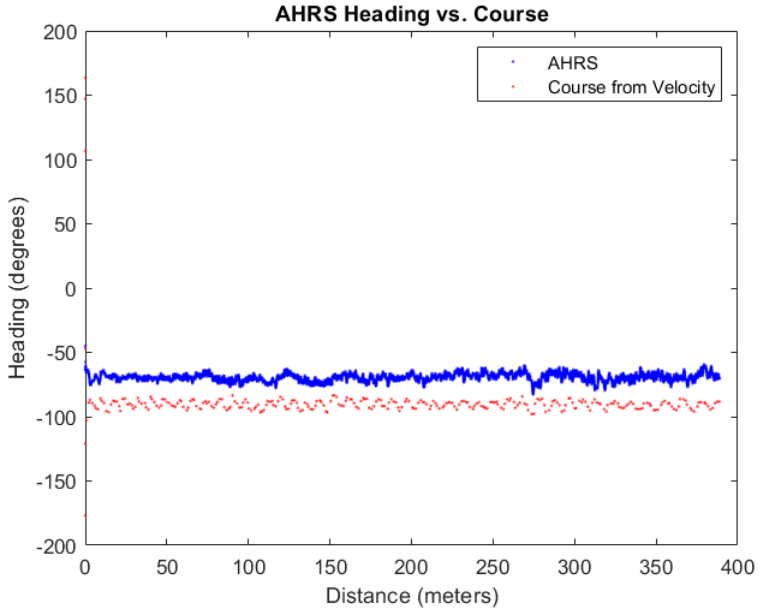


Figure 6.18: Heading and Velocity Course comparison

A drawback to using course measurements to estimate the misalignment is the variance of the course measurements. Figure 6.19 shows the error in course measurements over time. The first 7 course measurements are not due to large errors ranging from 12 to 163 degrees during the initial static period. 235 velocity measurements are received over the 300 meter walking distance. Excluding the first 7 velocity measurements during the static period, the average error over 300 meters is 2.7 degrees, or 3% of error. Only 35 of the velocity measurements were able to achieve a course measurement error below 1%. The maximum error after the static period is almost 8 degrees, or almost 9% of error.

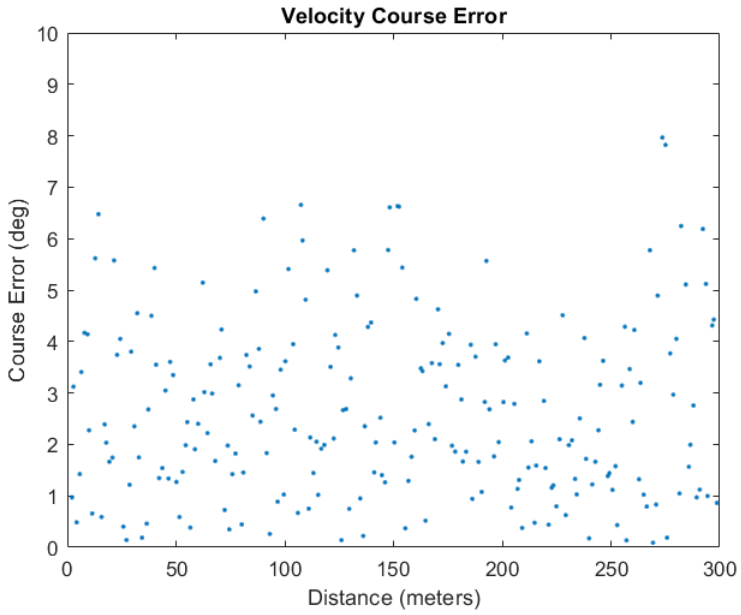


Figure 6.19: Velocity Course Error

With this in mind, course measurements from single GNSS velocity measurements show no improvement over the position difference method for capturing the misalignment angle. However, an average window of measurements can improve to capture the misalignment angle. A large window to be averaged reduces the noise of the resulting course measurements. However, it requires that the pedestrian walk in a constant direction for a longer duration of time. Figure 6.20 shows the results of the window averaging for a 10, 20, and 30 sample window. The pedestrian's average speed over the experiment was determined to be 1.3 m/s. With the GNSS sample rate set to 1 Hz this corresponds to the pedestrian traveling 13 meters for the 10 sample window, 26 meters for the 20 sample window, and 39 meters for the 30 sample window.

The 10 sample window has a total average error of 3.2 degrees, or 3.6%. However, the erroneous course measurements during the initial static period of the experiment produce errors between 11 and 160 degrees for the first 15 samples of the window average. These large errors are a result of the window average being buffered for the first 10 velocity measurements. After the first 15 samples of the window average, the average error for the rest of the walking route reduces to 0.5 degrees, or 0.57% error. For the 20 sample window,

the total average error is 4.3 degrees, and does not begin to drop below 1% of error until after the first 26 samples. The average error after 26 samples of the route is 0.48 degrees, or 0.54% error. A similar trend is visible for the 30 sample window where the total average error is about 5 degrees and does not begin to drop below 1% error until after the first 36 samples of the route. After this period, the average error for the remainder of the route is 0.48 degrees, or 0.52%.

Figure 6.20: Course Error with Window Averages

Averaging the course measurements from GNSS velocity over a window of samples reduces the noise in the measurements and improves the average error for a larger sample window. However, a large sample window averages older samples that are erroneous and reduce the accuracy of the window average. The large sample window results in the improvement of the window average accuracy being delayed in time based on the size of the sample window. Additionally, the average error after the delay only improves by about 0.02% as the window size increases by an additional 10 samples.

Compared to the position difference method, averaging a 10 GNSS velocity measurements window improves the accuracy of course measurements over a smaller period of time and distance. Additionally, the examination of the spread of course measurements in the

sample window allows for analysis on changes in the pedestrian's heading. For these reasons, the average moving window of GNSS course measurements will be used for the remainder of this thesis.

6.4.2 Magnetometer Fault Detection

The other key factor on the effectiveness of an AHRS implementation relies on unperturbed measurements of the Earth's magnetic field. These perturbations are often caused by structures and devices that generate magnetic fields using ferrous materials that perturb the local magnetic field [14]. Consider the data set analyzed above, where a constant misalignment angle was determined between the magnetometer and the pedestrian. Figure 6.21 shows how the presence of a local magnetic disturbance causes the AHRS heading to almost match the GPS velocity course measurements.

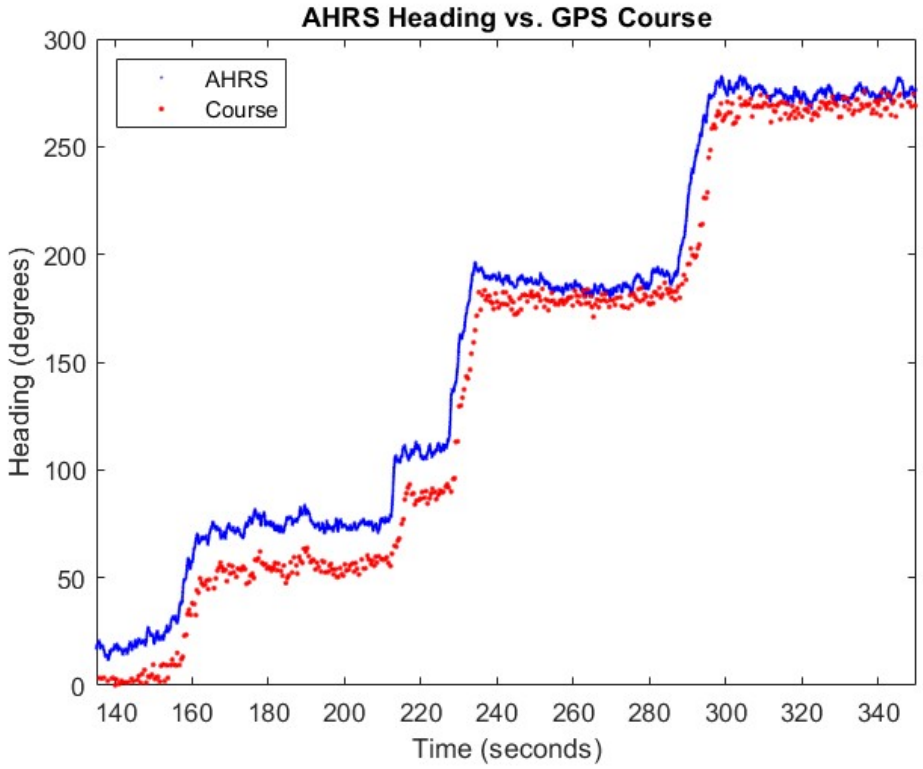


Figure 6.21: AHRS Heading Corrupted by Magnetic Disturbance

Around 240 seconds the AHRS heading drifts closer to the course measurements for about 40 seconds and then again around 300 seconds for another minute. This change in offset is a result of a disturbance in the local magnetic field from some unknown source. The strength and proximity of the source of the magnetic disturbance is often unknown and difficult to estimate. Therefore, it's more practical to disregard the magnetometer measurement entirely if it's being affected by a local disturbance. However, this requires the ability to recognize instances where the magnetic field has been disturbed.

A common approach to detecting faulty magnetometer measurements is examining the magnitude of the normalized innovation squared (NIS) [44]. The NIS is given by:

$$\epsilon_z = (y_k - Hx_k)^T S_k^{-1} (y_k - Hx_k) \tag{6.27}$$

where S^{-1} is the innovation covariance given by:

$$S_k = H_k P_k^- H_k^T + R_k \tag{6.28}$$

The NIS approach provides a method for checking if the KF is consistent with the measurement innovation and the innovation covariance. If the magnetometer measurements are corrupted by a disturbance, the normalized innovation squared metric will compute a large value due to large error between the estimated rotation of the magnetic field vector and the measured magnetic field vector.

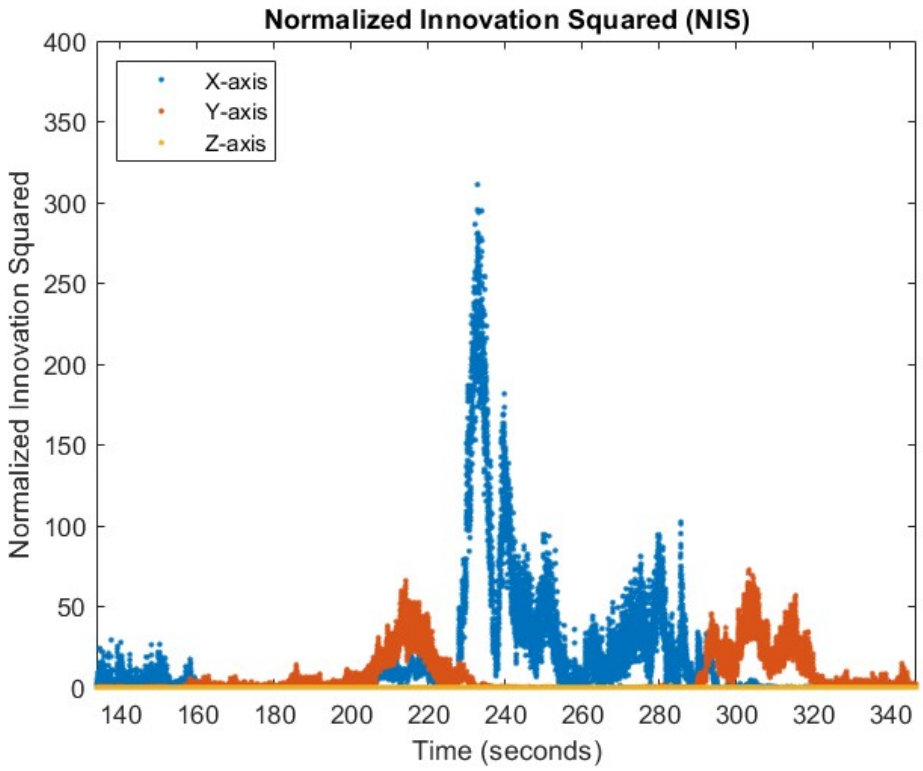


Figure 6.22: Normalized Innovation Squared of Magnetometer Innovation

The normalized innovation squared for the data shown in Figure 6.21 is shown in Figure 6.22. Notice the large NIS value in the x and y axes during the time intervals of the magnetic disturbance. For these intervals, the magnetometer measurements in the X and Y components need to be discarded as they have become corrupted. A simple thresholding technique can be applied to discard magnetometer measurements that exceed a specified NIS value. In this case, a NIS threshold of 50 will be used for this thesis.

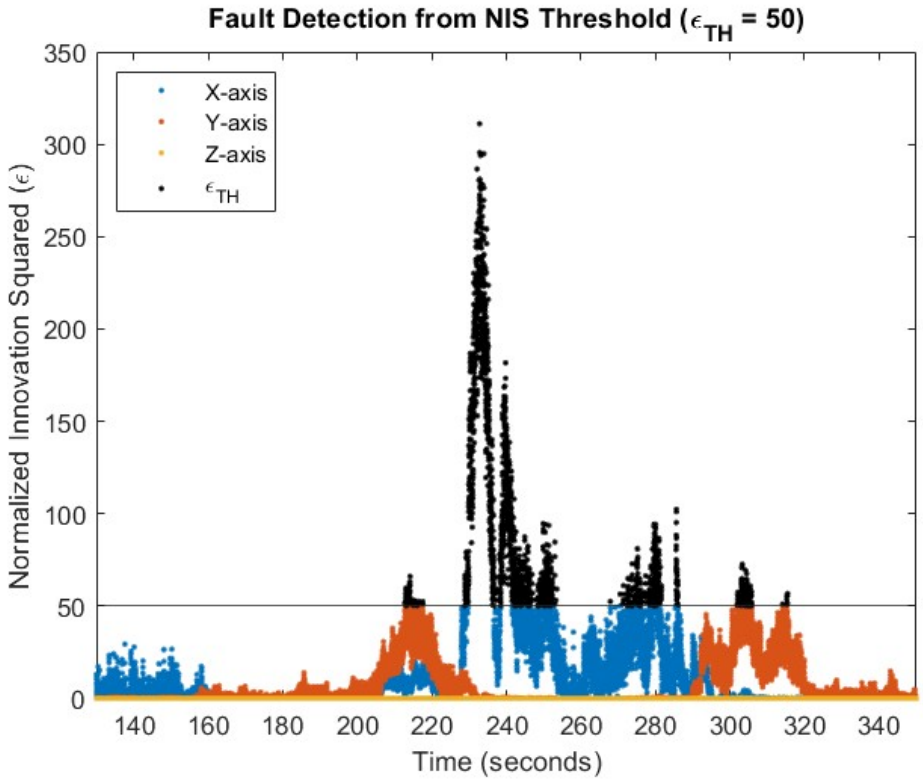


Figure 6.23: NIS Thresholding for Magnetometer Fault Detection ($TH = 50$)

Figure 6.23 shows how the magnetometer measurements that exceed the specified NIS threshold are discarded from the measurement update. With these measurements discarded, the accelerometer is the only measurement update applied to correct the pitch and roll angles, and the heading is determined from the dead reckoned angular velocity measurements. Figure 6.24 shows the results of applying the magnetic fault detection method.

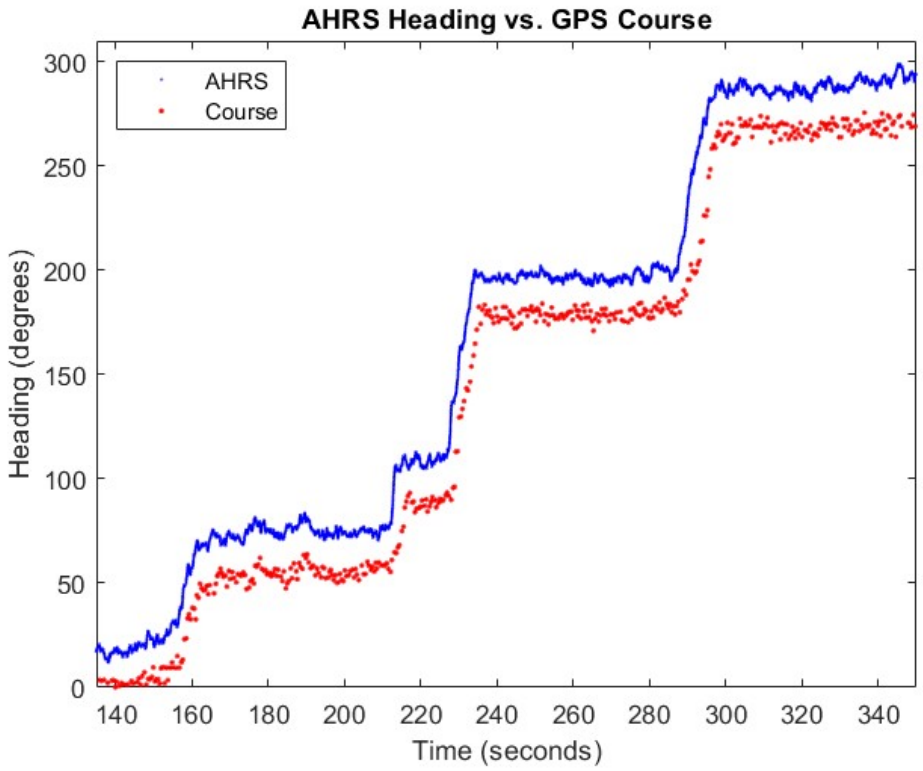
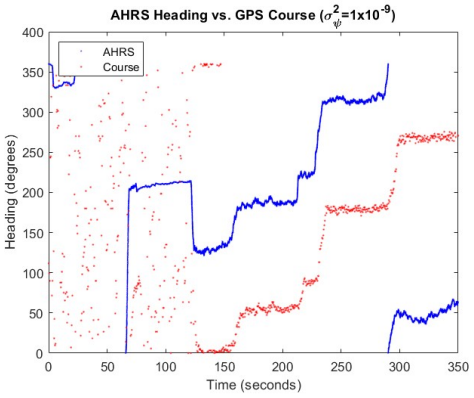
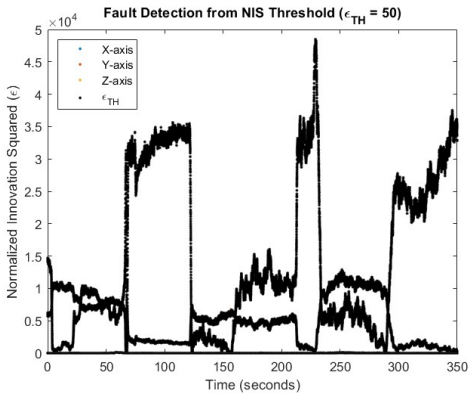


Figure 6.24: AHRS Heading with Magnetic Fault Detection

The rejection of the disturbed magnetometer measurements allow the trends for the AHRS heading to better match the GPS course, and the misalignment between the magnetometer frame and the pedestrian frame is held constant for the duration of the walking route. However, only using the NIS metric threshold to reject faulty magnetometer measurements relies on the initial estimate of the heading covariance, as well as, the initial estimate of the heading itself. If the covariance is initialized to be too small and the heading is erroneously initialized, then the NIS metric will be very large for all, or most, of the walking route. Figure 6.25 shows how a bad initial heading estimate with a low initial covariance will cause all magnetometer measurements to be rejected and the heading will be determined by only dead reckoning angular velocity measurements.



(a) AHRS Heading vs. GPS Course



(b) Normalized Innovation Squared (NIS)

Figure 6.25: AHRS Heading with Low σ_ψ^2 Initialization

With this in mind, the initial covariance can be set to be very large. However, this can cause the heading to converge much slower and may allow faulty magnetometer measurements to be used if the covariance has not converged low enough. Therefore, an additional constraint is added to ensure faulty magnetometer measurements are not considered: The misalignment angle is well estimated with a low covariance. Providing this additional constraint ensures that magnetometer measurements are not discarded during sudden changes in heading and allows for flexibility when initializing the heading covariance.

The proposed AHRS with misalignment correction and magnetic fault detection is similar to the AHRS methodology presented in Section 4.3.3, but with the inclusion of an additional heading misalignment state on the error state vector:

$$X = \begin{bmatrix} q \\ b_g \end{bmatrix} \qquad\qquad x = \begin{bmatrix} \delta\psi \\ b_g \\ \psi_o \end{bmatrix} \tag{6.29}$$

where (ψ_o) represents the misalignment between the magnetometer heading and the pedestrian heading. The discrete time propagation of both states from the continuous dynamic models remain the same, with the augmentation of the state transition matrix for the error state propagation given by:

$$A_d = \begin{bmatrix} \mathbf{expm}\left(\begin{bmatrix} -[\omega_{ib}^n]_\times \Delta t & -I_3 \Delta t \\ 0_3 & 0_3 \end{bmatrix}\right) & 0_{6\times 1} \\ 0_{1\times 6} & 1 \end{bmatrix} \tag{6.30}$$

The misalignment state is assumed to be constant with additive zero-mean Gaussian noise as governed by the continuous time model:

$$\dot{\psi}_o = w \tag{6.31}$$

The magnetometer and accelerometer measurement updates remain the same as described in Section 4.3.3. After the measurement update and full state corrections are applied, the error state vector is reset to zeros for all states except for the misalignment state which is held constant where:

$$x = \begin{bmatrix} 0_{6\times 1} \\ \psi_o \end{bmatrix} \tag{6.32}$$

The misalignment angle is then added to the magnetometer heading angle calculated from the conversion of quaternion to Euler angles as derived by Equation 2.27 where:

$$\psi = \psi_{AHRS} + \psi_o \tag{6.33}$$

Course Measurement Update

As mentioned previously, a moving average window filter is implemented to reduce the errors that result from noisy course measurements. For implementation in this thesis, a window size of 10 samples is used when two conditions are satisfied:

- The window of samples does not exceed a specified variance threshold. This can be expressed as a logical expression where:

$$C_{\sigma_\phi^2} = \begin{cases} 1 & \text{if } \sigma_\phi^2 < TH_{\sigma^2} \\ 0 & \text{if otherwise} \end{cases} \tag{6.34}$$

  This condition is included for sudden changes in heading that are not due to measurement errors.

- The speed of the pedestrian exceeds a specified threshold. This can be expressed as a logical expression where:

$$C_{\|\mathrm{v}\|} = \begin{cases} 1 & \text{if} \quad \| \mathrm{v} \| < TH_{\|\mathrm{v}\|} \\ 0 & \text{if otherwise} \end{cases} \tag{6.35}$$

  where $\| \mathrm{v} \|$ is the Euclidean norm of the north and east components of the GNSS velocity vector. This condition accounts for erroneous course measurements during periods where the pedestrian is stationary.

The above conditions are best satisfied when the pedestrian is walking in a constant direction to capture the misalignment angle and are determined by the quality of GNSS measurements available. For this thesis, the variance and speed thresholds were set to be:

$$TH_{\sigma^2} = 0.005\,\mathrm{rad}^2 \qquad\qquad TH_{\|\mathrm{v}\|} = 0.5\,\mathrm{m/s} \tag{6.36}$$

A window of 50 samples is averaged for the AHRS heading to account for the variance in the estimates, however, a constraint of the AHRS heading variance is neglected due to the high sampling rate. The measurement model for the misalignment angle is given by:

$$y = Hx + v \tag{6.37}$$

The term v represents zero-mean Gaussian noise and the observation matrix is given as:

$$H = \begin{bmatrix} 0_{1\times 6} & 1 \end{bmatrix} \tag{6.38}$$

and the measurement vector is given as:

$$y = \left[\frac{1}{10} \sum_{n-10}^{10} \phi_{GNSS_n} - \frac{1}{20} \sum_{n-20}^{20} \psi_{m_n} \right] \tag{6.39}$$

The measurement noise covariance matrix is given as a single scalar equating to the calculated variance of the course measurement window where:

$$R_\phi = \sigma_\phi^2\,\mathrm{rad}^2 \tag{6.40}$$

This method of calculating the measurement noise covariance allows a higher weight to be applied to measurement windows that provide more consistent course measurements

during the misalignment correction. With the measurement update parameters defined, the misalignment angle estimate is corrected and then applied to the global heading estimate:

$$\psi_n^+ = \psi_{AHRS,k} + \psi_o^+ \tag{6.41}$$

where ψ_n^+ represents the corrected global heading using the corrected misalignment angle ψ_o^+. After the correction is applied, the error state vector is reset to all zeros, except for the misalignment angle state that is set to the corrected angle.

Magnetometer Fault Detection Update

Before a magnetometer measurement update is applied at each epoch, the magnetometer measurements are examined to determine if they are unperturbed by a local disturbance. This requires calculating a NIS value of the measurement innovation:

$$\epsilon_{mag} = (\delta y_k - H_{mag}x)^T \cdot S_k^{-1} \cdot (\delta y_k - H_{mag}x) \tag{6.42}$$

where δy_k is given as the measurement error between the magnetometer measurement in the body frame and the magnitude of the local magnetic field strength rotated into the body frame from the estimated quaternion:

$$\delta y_k = \left[m_{ib}^b - \hat{C}_n^b \tilde{m}^n \right] \tag{6.43}$$

The term H_{mag} is the magnetometer observation matrix given below:

$$H_{mag} = \left[\left[\hat{C}_n^b \tilde{m}^n \right]_\times \; 0_{3\times 4} \right] \tag{6.44}$$

After the NIS value is calculated, it must satisfy the conditions stated previously to verify if the magnetometer measurements are viable:

$$C_1 = \begin{cases} 1 & \text{if } \epsilon_{mag} < 50 \text{ AND } \sigma_{\psi_o}^2 > 0.0001 \\ 0 & \text{if otherwise} \end{cases} \tag{6.45}$$

If the logical statement, returns a 1, then the magnetometer measurement is utilized. If the logical statement returns a 0, then the magnetometer measurement is considered faulty and the measurement update is not applied.

6.5 Navigation System Update

The pedestrian navigation system requires that the navigation states be estimated in order to provide a stable navigation solution. The PDR-SD approach in Section 4.3.4 uses the step length and heading to propagate the pedestrian's position in the north and east components. The proposed navigation system is based on the work from Rehman et al. [19] that uses an EKF approach and extends the estimated states to include the pedestrian's velocity magnitude and the rate of change in the heading angle as given by the state vector:

$$\mathbf{x} = \begin{bmatrix} E \\ N \\ V \\ \psi \\ \dot{\psi} \end{bmatrix} \tag{6.46}$$

with the states listed as east position (E), north position (N), magnitude of north and east velocity (V), heading angle (ψ), and the rate of change in the heading angle ($\dot{\psi}$). These states are propagated forward in discrete time by the dynamic model:

$$\mathbf{x}_{k+1} = \mathbf{\Phi}_k \mathbf{x}_k + \mathbf{w}_k \tag{6.47}$$

where the state transition matrix from time t_k to t_{k+1} is derived as:

$$\mathbf{\Phi}_k = \begin{bmatrix} 1 & 0 & \sin(\psi_k)\,\Delta t & 0 & 0 \\ 0 & 1 & \cos(\psi_k)\,\Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.48}$$

As can be seen from the state transition matrix, the dynamic model for the system is nonlinear, and must be linearized about the current state estimate using a first order Taylor Series approximation in order to propagate the state covariance matrix forward in time.

The linearized state transition matrix is given as:

$$\mathbf{A}_d = \frac{\delta \mathbf{\Phi}_k}{\delta \mathbf{x}_k} = \begin{bmatrix} 1 & 0 & \sin(\psi_k)\Delta t & V_k \cdot \cos(\psi_k)\Delta t & 0 \\ 0 & 1 & \cos(\psi_k)\Delta t & -V_k \cdot \sin(\psi_k)\Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.49}$$

and the state covariance propagation is performed as shown in Equation (3.41) with the discrete process noise matrix:

$$\sigma_{d,k}^2 = \begin{bmatrix} \sigma_E & 0 & 0 & 0 & 0 \\ 0 & \sigma_N & 0 & 0 & 0 \\ 0 & 0 & \sigma_V & 0 & 0 \\ 0 & 0 & 0 & \mathbf{w}_\psi & 0 \\ 0 & 0 & 0 & 0 & \mathbf{w}_{\dot{\psi}} \end{bmatrix} \cdot \Delta t \tag{6.50}$$

The system state estimates are propagates by the time step (Δt), which is set by the sampling period of the IMU. The state estimate corrections are performed in two separate measurement updates:

1. GNSS measurement update
2. PDR measurement update

These two measurement updates utilize measurements output from the GNSS receiver, as well as, pseudo-measurements from the step detector, step length estimator, and AHRS filter discussed in the previous sub-sections.

GNSS Measurement Model

The GNSS measurement update corrects the estimates of the east and north position states, and the estimates of the magnitude of horizontal velocity in the north and east frame using position and velocity measurements. The GNSS measurement model is given by:

$$\mathbf{y}_{GNSS} = \mathbf{H}_{GNSS}\mathbf{x}_k + \mathbf{v}_{GNSS,k} \tag{6.51}$$

where the measurement vector ($\mathbf{y}_{GNSS}$) and observation matrix ($\mathbf{H}_{GNSS}$) are given as:

$$\mathbf{y}_{GNSS} = \begin{bmatrix} E_{GNSS} \\ N_{GNSS} \\ V_{GNSS} \end{bmatrix} \qquad \mathbf{H}_{GNSS} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{6.52}$$

The east and north position measurements are derived from the LLA measurements converted into the local navigation frame. The process of converting LLA measurements to the local NED frame is often conducted by calculating the equivalent Cartesian position in the ECEF frame and its distance from the initial ECEF position, followed by a rotation of this cartesian distance into the local navigation frame. For this thesis, the conversion is performed using MATLAB's **lla2ned**() function.

The measurement update for the magnitude of horizontal velocity requires taking the norm of the GNSS velocity measurements in the north and east directions where:

$$V_{GNSS} = \sqrt{\left(V_{GNSS,E}\right)^2 + \left(V_{GNSS,N}\right)^2} \text{ m/s} \tag{6.53}$$

This calculation produces a positive scalar quantity that is more akin to receiver's measured speed since only the velocity magnitude is considered, rather than the direction. However, for the sake of simplicity and notation, the speed measurement update will be referred to as the velocity measurement update. The GNSS measurement noise covariance matrix is given by:

$$R_{GNSS} = \begin{bmatrix} \sigma_E^2 & 0 & 0 \\ 0 & \sigma_N^2 & 0 \\ 0 & 0 & \sigma_V^2 \end{bmatrix} \tag{6.54}$$

The GNSS measurement update allows for position and velocity state estimates to be bounded by errors that can arise from a pure PDR-SD solution.

PDR Measurement Update

The second measurement update uses the step detection, GNSS aided step length estimation, and GNSS aided AHRS methods discussed in Sections 4.3.1, 6.3, and 6.4 to correct the estimates in the velocity, heading, and rate of change in heading states. The PDR measurement model is given by:

$$\mathbf{y}_k = \mathbf{H}_{PDR}\mathbf{x}_k + \mathbf{v}_{PDR,k} \tag{6.55}$$

where the measurement vector ($\mathbf{y}_{PDR}$) and the observation matrix ($\mathbf{H}_{PDR}$) are found to be:

$$\mathbf{y}_{PDR} = \begin{bmatrix} V_{PDR} \\ \psi_{PDR} \\ \dot{\psi}_{PDR} \end{bmatrix} \qquad \mathbf{H}_{PDR} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.56}$$

The PDR velocity measurement update is defined as the change in distance traveled from the current estimated step length over the time between steps taken where:

$$V_{PDR,k} = \frac{SL_k}{t_{step_k} - t_{step_{k-1}}} \text{ m/s} \tag{6.57}$$

This formulation utilizes the step detection algorithm from Section 4.3.1 to determine the time difference between steps taken, and the adaptive step length estimation algorithm from Section 6.3 to determine the magnitude of the distance traveled between detected steps. This measurement update only occurs when a new step has been detected.

The PDR rate of change in the heading angle measurement update calculates the change in heading angle over the change in time between detected steps where:

$$\psi_{PDR,k} = \frac{\psi_{step_k} - \psi_{step_{k-1}}}{t_{step_k} - t_{step_{k-1}}} \text{ rad/s} \tag{6.58}$$

The change in heading uses the step detection and GNSS aided AHRS algorithms to calculate this pseudo-measurement. This measurement update is also only applied during a time of a detected step.

The PDR heading angle measurement update uses the estimated GNSS aided AHRS heading algorithm from Section 6.4 to update the navigation system heading state:

$$\psi_{PDR,k} = \psi_{AHRS} + \psi_o \text{ rad} \tag{6.59}$$

One key difference between the heading update and the velocity and rate of change of heading updates is the constant availability of the GNSS aided AHRS heading estimates. Since the AHRS filter is continuously updated at the rate of the IMU sampling time, the navigation system heading state is constantly corrected immediately following the time update of the navigation system states. It seems intuitive to apply the rate of change in heading update at the same rate as the heading update, but the high noise characteristics of the GNSS aided AHRS heading estimates introduce high variance in rate of change in heading when differentiating over smaller intervals of time. Constraining the occurrence of the heading

rate update to occur during periods of detected steps help to constrain these errors due to differentiation.

The measurement noise covariance matrix is given by a diagonal matrix of the variance in the measurements given by:

$$R_{PDR} = \begin{bmatrix} \sigma_V^2 & 0 & 0 \\ 0 & \sigma_{\dot{\psi}}^2 & 0 \\ 0 & 0 & \sigma_{\dot{\psi}}^2 \end{bmatrix} \tag{6.60}$$

Still Detection

One caveat to the velocity update is to account for periods where the pedestrian is no longer walking. This is necessary to prevent the position being propagated continuously from the PDR velocity update during stationary periods. One method to account for this is to use the same still detection methodology from Section 4.2.3, where thresholds are set for signal characteristics of the Euclidean norm of the angular velocity measurements. If *still* periods are detected at any point, then a zero-velocity update (ZVU) is applied for the PDR velocity update where:

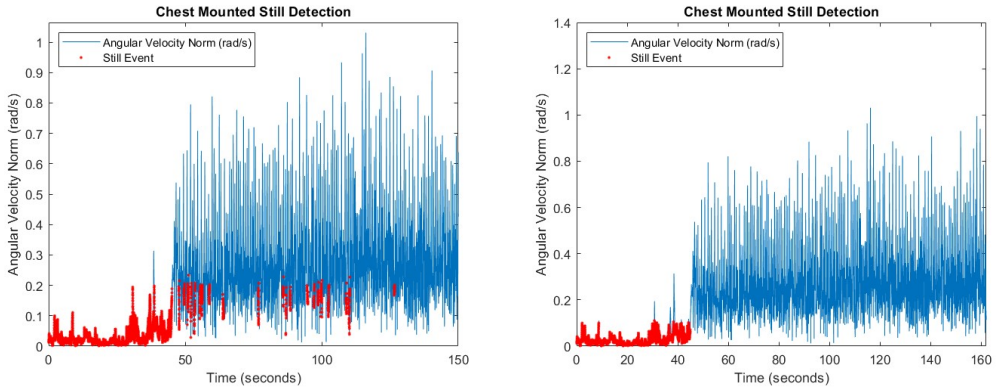
$$V_{PDR,k} = \begin{cases} \frac{SL_k}{t_{step_k} - t_{step_{k-1}}}, & \text{if } \text{still} = 0 \\ 0, & \text{if } \text{still} = 1 \end{cases} \tag{6.61}$$

However, due to the lower dynamics experienced by an IMU mounted to the pedestrian's chest, rather than the foot, identifying these stationary periods requires a lower threshold of the maximum angular velocity norm. Figure 6.26 shows how the still detection threshold for the foot mounted scenario (left) continues to detect still periods during the walking phases, while the modified threshold for the body mounted scenario (right) is able to avoid detecting still periods during walking phases.

Applying a threshold of $TH_{\omega,max} = 0.1\, rad/s$ allows for *still* periods to be determined more effectively for body mounted scenarios in order to apply ZUPTs when the pedestrian is not walking.

6.6 System Summary

The presented loosely coupled architecture for GNSS/PDR-SD integration analyzes the extensions of each subsystem to improve upon the PDR-SD solution in the event GNSS mea-



(a) Still Detection $TH_{\omega,max} = 0.2$rad/s

(b) Still Detection $TH_{\omega,max} = 0.1$rad/s

Figure 6.26: Still Detection for Chest Mounted IMU

surements are no longer available. Rather than using GNSS to only correct errors in position estimates from the PDR-SD methodology in Section 5.3, velocity and course measurements are also utilized to provide parameter, and state, estimation of each subsystem.

For the step length estimation subsystem, a novel step length model is presented, as well as the use of GNSS step length pseudo-measurements to estimate parameters of the new step length model. Utilization of step length pseudo-measurements allows for the step length model parameters to be adjusted to fit the discrete step lengths of a pedestrian with unknown physical characteristics (i.e. height, leg length, and average walking pace).

For the AHRS heading estimation subsystem, methods of misalignment correction and magnetometer fault detection were presented. The use of GNSS course measurements provide estimates of the the misalignment angle between the pedestrian frame and the leveled magnetometer. The misalignment estimation and correction allows for placement of the co-located IMU and magnetometer to be rigidly mounted in any position and orientation on the pedestrian's torso. The magnetic fault detection algorithm proposes a method of rejecting magnetometer measurements that have been corrupted by the presence of a disturbance in the local magnetic field when GNSS is, either available or not available.

Finally, the Navigation state estimation subsystem uses velocity state estimates from the step length estimation subsystem, heading state estimates from the AHRS subsystem, and position and velocity measurements from GNSS to produce a fused navigation solution. When GNSS is no longer available, the step length estimation subsystem continues to provide

velocity state estimates to the navigation system, in order to account for when the pedestrian is no longer in motion. A modified *still* period detection algorithm prevents the position state estimates from propagating when the pedestrian is stationary. The next chapter will present the experimental results for the implementation of this system for various real-world scenarios.

Chapter 7

Extended Loosely Coupled GNSS-SD Implementation and Results

7.1 Introduction

The implementation of the GNSS/PDR-SD system is conducted on multiple real-world data sets to examine the performance of each subsystem in different environmental conditions. The results will be compared to the implementation of the loosely coupled GNSS/PDR-SD system described in Section 5.3.2 as a test case in the same environments. The examined environmental scenarios will include:

1. Unperturbed magnetic field measurements with no magnetometer misalignment.
2. Unperturbed magnetic field measurements with magnetometer misalignment.
3. Perturbed magnetic field measurements with magnetometer misalignment.

For each environmental condition, the implementation will include different scenarios of GNSS availability.

The first environmental condition will examine the proposed system performance against the step length estimation methods from the test case when the magnetometer is fully aligned

with the pedestrian frame, and no magnetic disturbances affect the heading estimates from the AHRS. The goal of this experiment is to validate the performance the proposed step length model based on the accumulation of step length pseudo-measurements for different periods of GNSS availability.

The second environmental condition will examine the performance of the magnetometer misalignment correction with GNSS course measurements. The goal of this experiment is to validate the method of misalignment estimation, as well as determine periods of GNSS availability that yield the best misalignment estimates.

The final environmental condition will examine the performance of the magnetometer fault detection algorithm in conjunction with the magnetometer misalignment estimation and correction. The goal of this experiment is to evaluate if corrupted magnetic field measurements can be detected and rejected to reduce heading errors with varying periods of GNSS availability.

All the presented results will include the test case implementation for performance evaluation, and experimental setup and procedure for the data sets in each environmental condition. For all the experimental results that will be shown, certain assumptions are made about the pedestrian:

- The pedestrian is walking in the same direction the pedestrian is facing.
- The pedestrian makes no irregular steps or motions (i.e. tripping, side stepping, sitting down, shrugging shoulders, etc.).
- The pedestrian exhibits only a walking motion (no running, skipping, jumping, etc.).

7.2 Step Length Model and Pseudo-Measurement Test

The data set for the first environmental condition was collected by a pedestrian walking a closed loop along a sidewalk up and down a section of a road as shown in Figure 7.1.

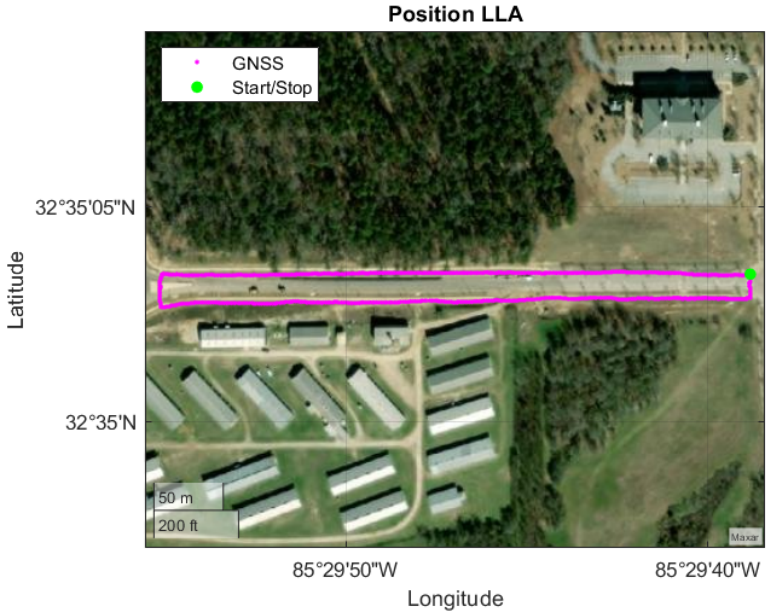


Figure 7.1: GNSS reference path

The pedestrian starts at the top right position (green), then walks West along the sidewalk to the top right left position and cross the street to the bottom left position. From there, the pedestrian walks East to the bottom right position and cross the street to return back to the top right position (green) to conclude the route. This is the same data set used in Section 5.3.2 where no magnetic field disturbances were observed in the post-processed data, and the misalignment and magnetic declination angle were initially corrected to be in line with the pedestrian frame. This data set also includes periods of varying step length and walking pace. Figure 7.2 shows the reference path with the change in the pedestrian's pace.

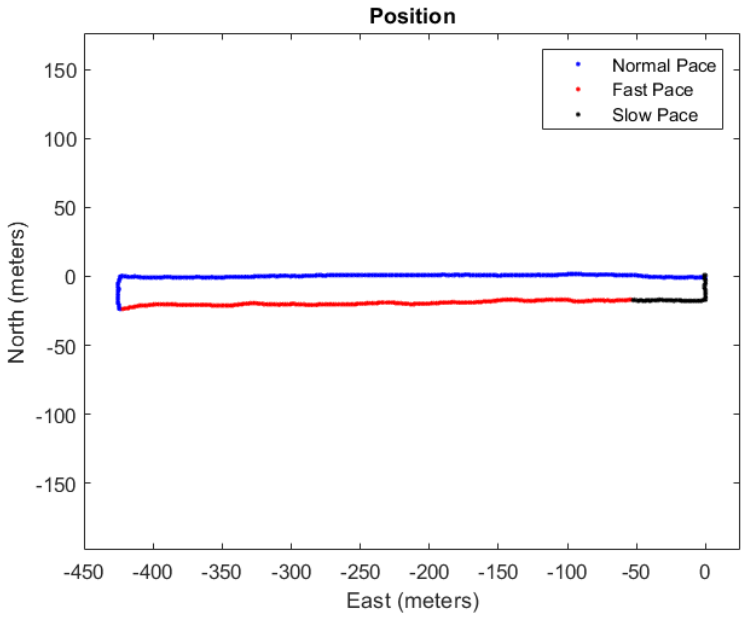


Figure 7.2: GNSS reference path with different walking paces

For the first leg of the walking route (blue), the pedestrian maintains a normal walking pace for about 345 seconds. In the second leg of the walking route (red), the pedestrian accelerates and increases the step length for 217 seconds. For the final leg of the walking route (black), the pedestrian's walking pace and step length reduces to be slower and smaller than the first leg of the walking route for the remaining 64 seconds. These periods will help to examine the subsystem's performance in the presence of varying step length.

The pedestrian was fitted with a tightened military vest to ensure rigidity between the hardware and the pedestrian. A Vectornav VN-100 9-DoF IMU [30] was mounted to the right of the pedestrian's sternum, and a Ublox EVK-7P GNSS receiver [31] was included with a single antenna mounted to the pedestrian's left shoulder. The IMU and magnetometer are sampled at 100 Hz, and the GNSS receiver is sampled at 1 Hz. As mentioned, the implementation will be evaluated for different periods of GNSS availability. The step length estimation methods from the test case assume static model parameters that are estimated assuming three different step length models:

- Constant step length model.
- Waist swing step length model.

- Step frequency model.

The constant step length model parameter is calculated by Equation (7.1)

$$SL_c = \frac{d}{s} \tag{7.1}$$

where d is the total distance traveled from the origin at the current epoch, and s is the total number of step detected at the current epoch. The waist swing model parameter and step frequency model parameters are determined experimentally with a RLS approach using the GNSS step length pseudo-measurements. The experimental results for the SLKF implementation will be shown using the position-velocity pseudo-measurements, and then the velocity pseudo-measurements.

7.2.1 GNSS Constant Availability

Position-Velocity Pseudo-Measurements

For the first scenario of the environmental condition, GNSS is assumed to be constantly available for the duration of the experiment. Figure 7.3 shows the path error results for constant GNSS availability. Additionally, Table 7.1 shows the RMSE in the east and north directions for all 4 step length models.

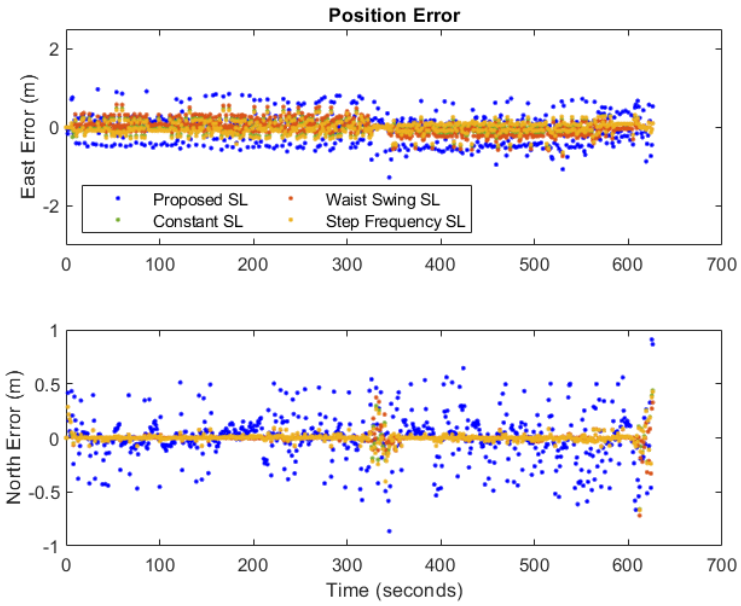


Figure 7.3: North and East Path Errors

Table 7.1: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Proposed	0.378	0.225
Constant	0.156	0.056
Waist Swing	0.193	0.056
Step Frequency	0.155	0.058

With constant GNSS availability, the proposed step length model produces position errors within 1 meter of the true positions from GNSS. However, the model parameters are initialized at zero and estimated over time using viable GNSS step length pseudo-measurements. Therefore, the accuracy of the proposed step length model is dependent on the number of viable step length position-velocity pseudo-measurements. Figure 7.4 shows the results of the proposed step length model estimates with the GNSS step length position-velocity pseudo-measurements.

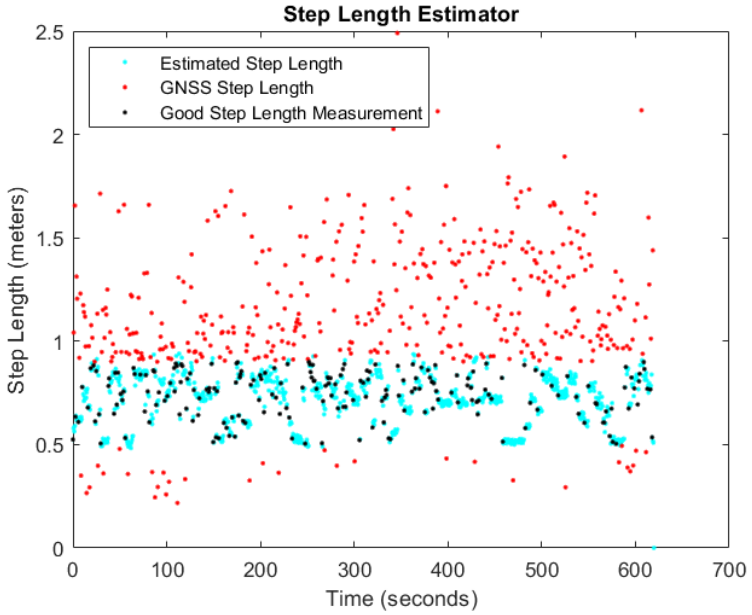


Figure 7.4: Step Length Estimation for constant GNSS availability

The total number of steps taken for the walking route was $1,089$, however, when estimating the step length model parameters, GNSS step length position-velocity pseudo-measurements are only available when a position and velocity measurement is received between successive steps. Therefore, the use of step length position-velocity pseudo-measurements is dependent on the update rate of the GNSS receiver and the walking pace of the pedestrian. With GNSS available for the entire duration of the walking path, 620 step length position-velocity pseudo-measurements were available, and of those only 198 step length position-velocity pseudo-measurements were viable to be used for model parameter estimation.

Table 7.2: Walking Pace with Position-Velocity Pseudo-Measurements

Pace	SLKF			GNSS Position-Velocity Pseudo-Measurements					
	steps	μ (m)	σ (m)	total	μ (m)	σ (m)	good	μ (m)	σ (m)
Normal	577	0.73	0.12	340	0.94	0.31	126	0.74	0.11
Fast	413	0.72	0.1	219	1.17	0.36	51	0.89	0.27
Slow	98	0.7	0.12	61	0.9	0.32	21	0.88	0.28

Table 7.2 shows the comparison between of the SLKF with the GNSS step length position-velocity pseudo-measurements during the different walking paces of the route. During the first leg of the route, the pedestrian takes a total of 577 steps and the proposed step length model estimates an average step length of 0.73 meters with a standard deviation of 0.12 meters using 126 viable pseudo-measurements from GNSS. However, during the second leg, the pedestrian's average estimated step length reduces to 0.72 meters even with the 51 viable pseudo-measurements that indicate an increase in step length. For the final leg of the route, the average estimated step length reduces to 0.7 meters when the pedestrian is walking slower when the viable pseudo-measurements indicate an average step length that still exceeds the average step length during the first leg of the route.

The high variance in the step length position-velocity pseudo-measurements introduce a large variance in the estimation of the step length model parameters. Figure 7.5 shows the change in step length model coefficients with new pseudo-measurements.

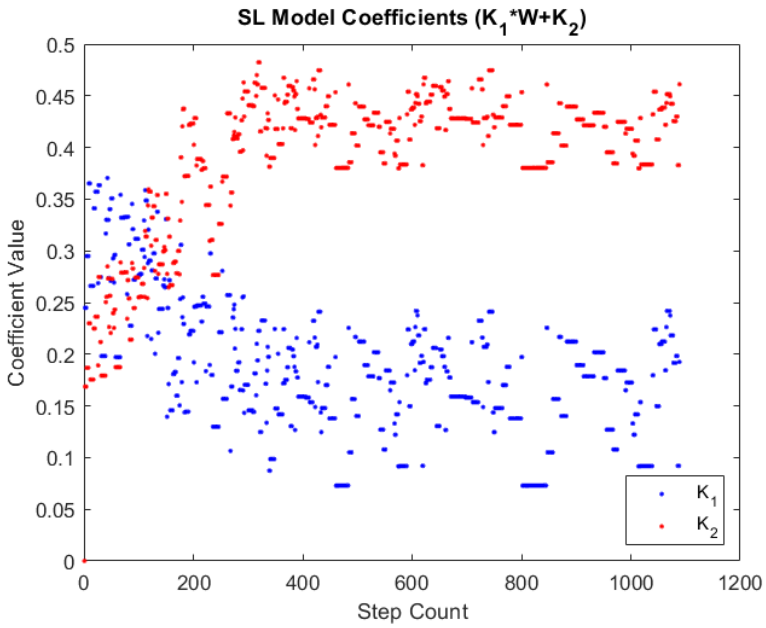


Figure 7.5: Step Length Model Coefficients

Initially, the scaling coefficient, K_1, is larger than the constant coefficient, K_2, then as more pseudo-measurements are used, the constant coefficient is increased to be greater and has less variation. The larger process noise of the scaling coefficient introduces the higher

variance when applying the pseudo-measurement. The final coefficient terms yield a step length model of:

$$SL = 0.193 \cdot W + 0.461 \tag{7.2}$$

The resulting step length model will be examined in the final results of the unperturbed magnetic field with no misalignment condition.

Velocity Pseudo-Measurements

Using the GNSS step length velocity pseudo-measurements, the experimental path errors are shown below in Figure 7.6. Table 7.3 also shows the corresponding RMSE in the north and east directions for the four step length models. It can be seen in that performance is slightly reduced for the proposed step length method. However, the step frequency approach shows dramatic errors in both the east and north directions.

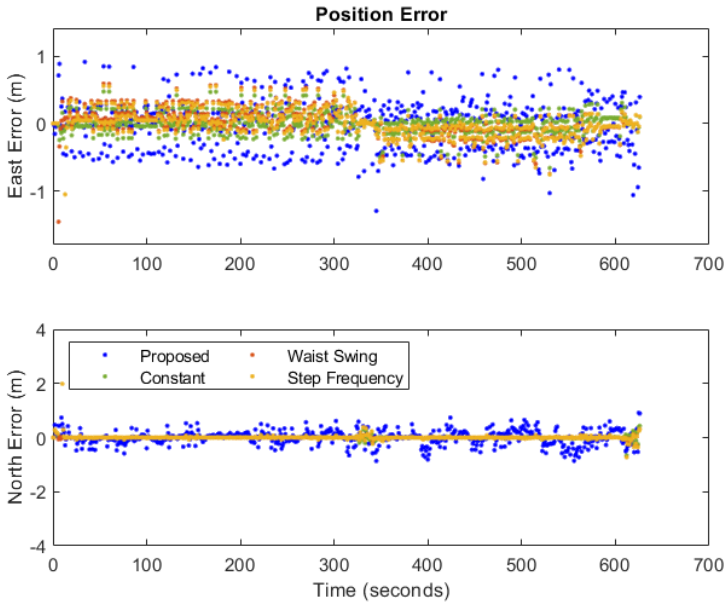


Figure 7.6: North and East Path Errors

Table 7.3: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Proposed	0.370	0.258
Constant	0.156	0.056
Waist Swing	0.211	0.057
Step Frequency	250.193	18.367

The large errors in the step frequency method are primarily attributable to the initial static period where the large delay in time from when the pedestrian is stationary to walking improperly calibrating the step length model coefficients.

Figure 7.7 shows the results of the proposed step length model estimates with the GNSS step length velocity pseudo-measurements. The trends of the pedestrian's change in step length is now much more apparent as the step length velocity pseudo-measurements increase and decrease at the periods where the pedestrian's pace changes.

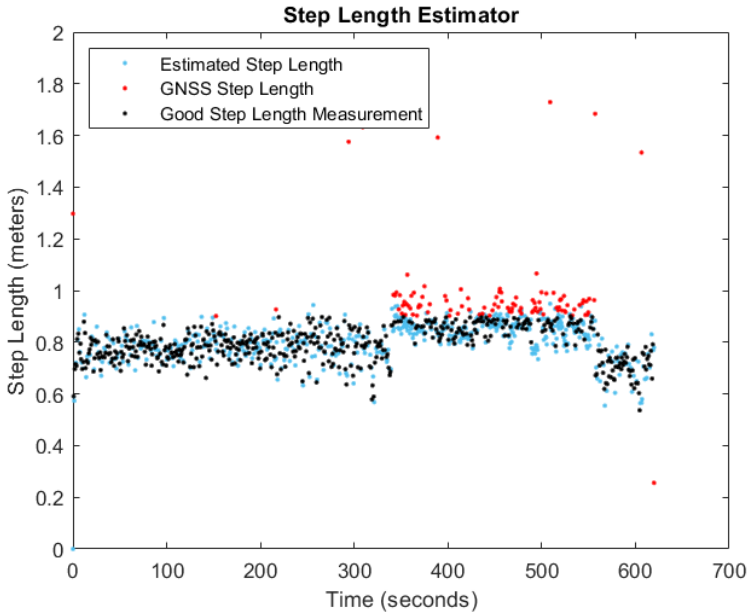


Figure 7.7: Step Length Estimation for constant GNSS availability

While the performance of the proposed step length model appears to have reduced in east and north accuracy, the number of viable velocity pseudo-measurements is much greater than the position-velocity pseudo-measurements. Table 7.4 shows the comparison of viable velocity pseudo-measurements relative to the total pseudo-measurements available. It can be seen that the majority of velocity pseudo-measurements are viable and lower in variance for each pace of the route. However, during the fast paced leg of the route fewer velocity pseudo-measurements are considered viable as they are clustered close to the maximum threshold of 0.9 meters.

Table 7.4: Walking Pace with Velocity Pseudo-Measurements

Pace	SLKF			GNSS Velocity Pseudo-Measurements					
	steps	μ (m)	σ (m)	total	μ (m)	σ (m)	good	μ (m)	σ (m)
Normal	577	0.78	0.06	340	0.78	0.09	335	0.78	0.05
Fast	413	0.85	0.04	219	0.91	0.13	126	0.77	0.07
Slow	98	0.70	0.06	61	0.71	0.13	59	0.77	0.09

Figure 7.8 shows the evolution of the proposed step length model coefficients over time when being estimated using the velocity pseudo-measurements. The variance in the model coefficients is much lower compared to the model coefficients when using the position-velocity pseudo-measurements.

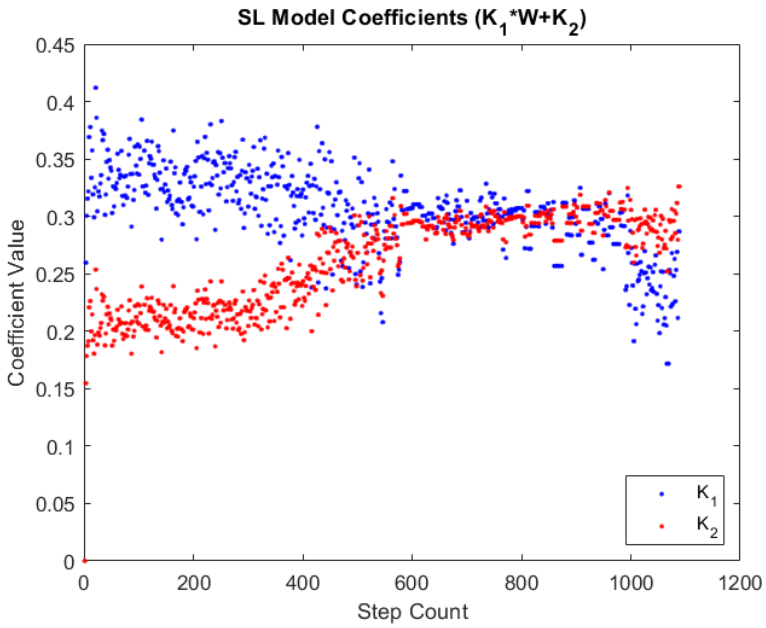


Figure 7.8: Step Length Model Coefficients with Velocity Pseudo-Measurements

After the last measurement update is applied at the end of the slow pace of the route the step length model coefficients reach a final value that produces a step length model of:

$$SL = 0.2868 \cdot W + 0.3260 \tag{7.3}$$

The resulting step length model will be examined in the final results of the unperturbed magnetic field with no misalignment condition. The next scenario will examine the same environmental conditions when GNSS signal is fully unavailable towards the end of the walking route.

7.2.2 GNSS Mostly Available

Position-Velocity Pseudo-Measurements

The second scenario simulates a loss of GNSS signal just before the pedestrian reduces the pace in the final leg using the same walking route. The goal of this scenario is to examine the accuracy of the PDR solution with the converged step length model parameters once GNSS pseudo-measurements are no longer available during the final leg of the route. For this scenario GNSS was simulated to be unavailable about 64 seconds before the end of the

walking path. Shown below in Figure 7.9 are the path error results for the second scenario. Additionally, the RMSE in the east and north directions are given in Table 7.5.

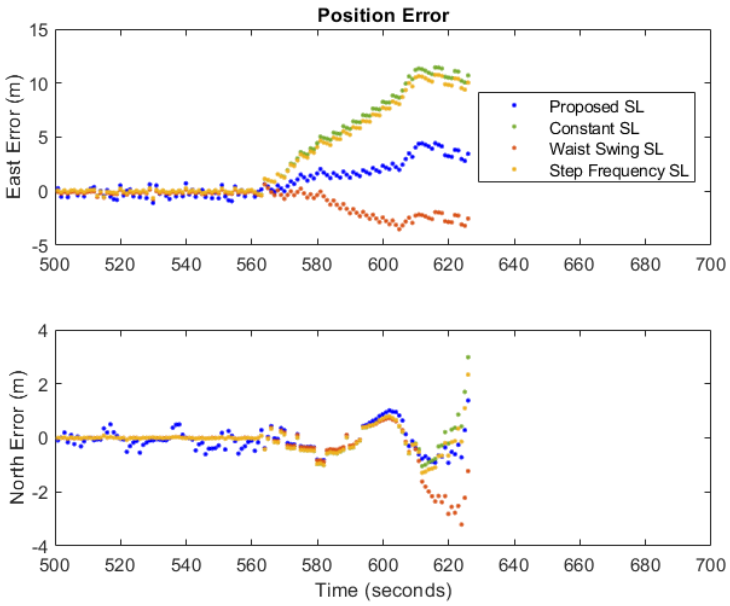


Figure 7.9: North and East Path Errors

Table 7.5: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Proposed	0.876	0.273
Constant	2.442	0.219
Waist Swing	0.664	0.381
Step Frequency	2.280	0.212

With GNSS being available for the majority of the walking route, the proposed step length model produces errors of up to 5 meters in the East direction after GNSS is no longer available, while the constant and step frequency step length models maintain East errors less than 15 meters. However, the step length model shows a lower RMSE in the North direction, which can be attributed to the modified AHRS removing any errors in

misalignment or faulty magnetometer measurements that were not accounted for. Figure 7.10 shows the results of the proposed step length model estimates with the GNSS step length pseudo-measurements.

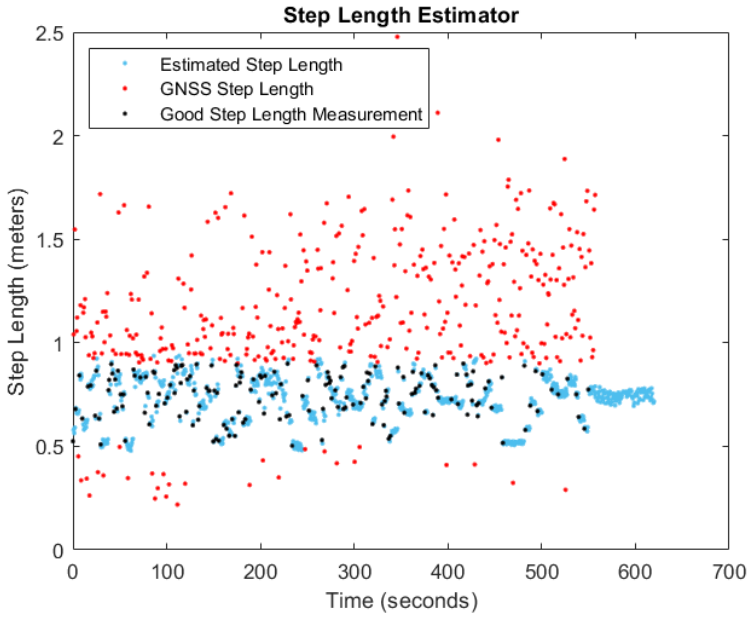


Figure 7.10: Step Length Estimation for GNSS mostly available

Lowering the amount of time that GNSS is available during the walking route, also lowers the number of step length pseudo-measurements that can be obtained. The total number of step length pseudo-measurements that were recorded was 558, and of those, only 176 were viable to be used for parameter estimation. Figure 7.11 shows the change in the step length model parameters for each step. Table 7.6 provides the SLKF results for the different walking paces of the route.

Table 7.6: Step Length Estimation RMSE

Pace	SLKF		
	steps	mean	sigma
Normal	577	0.73	0.12
Fast	413	0.72	0.10
Slow	98	0.74	0.02

In the final leg of the walking route, the pedestrian's average step length is estimated to be 0.74 meters with a standard deviation of 0.02 meters. Contrary to expectation, the pedestrian's average step length is estimated to be larger than the average step length during the first two legs of the walking route.

After 970 steps have been taken in the walking route, the model coefficients converge to a final estimate to determine the pedestrian's step length for the remainder of the route. The final coefficient terms yield a step length model of:

$$SL = 0.184 \cdot W + 0.425 \tag{7.4}$$

Again, the resulting step length model for when GNSS is mostly available will be examined at the end of this environmental condition.

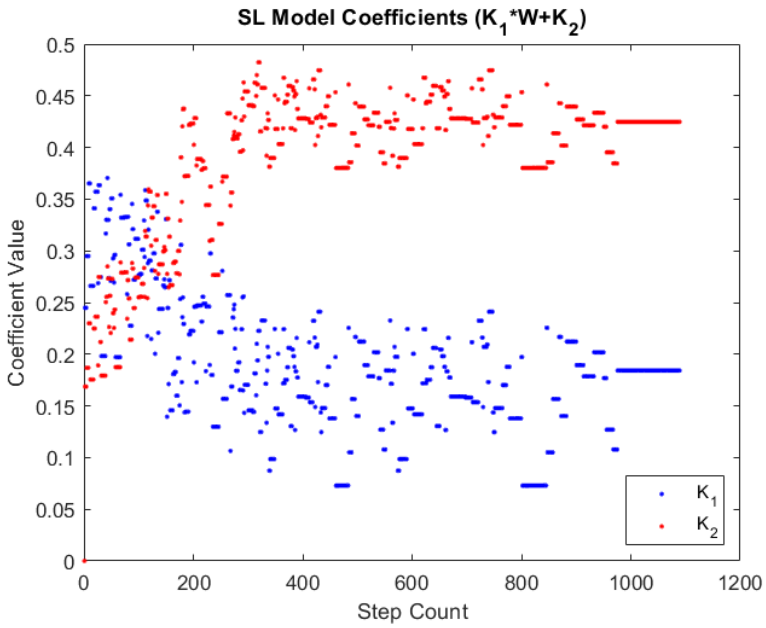


Figure 7.11: Step Length Model Coefficients

Velocity Pseudo-Measurements

Using the GNSS step length velocity pseudo-measurements, the experimental path errors are shown below in Figure 7.12 to show the evolution in the PDR errors when GNSS is no longer available. Table 7.7 also shows the corresponding RMSE in the north and east directions for the four step length models. With GNSS being available for the majority of the walking route, the proposed step length model produces greater errors in the East direction with a maximum of 6.2415 meters. With GNSS unavailability being simulated just before the start of the slow pace of the route, the path errors indicate that the proposed step length model is not able to adapt to the pedestrian's step length when the pace abruptly changes from fast to slow when compared to the results using the position-velocity pseudo-measurements.

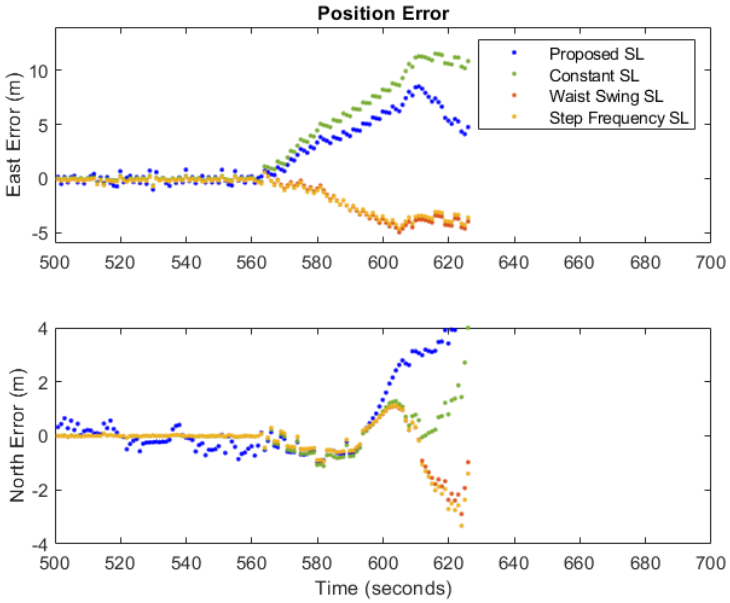


Figure 7.12: North and East Path Errors

Table 7.7: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Proposed	1.648	0.752
Constant	2.446	0.313
Waist Swing	1.009	0.333
Step Frequency	250.195	18.370

Figure 7.13 shows the results of the proposed step length model estimates with the GNSS step length velocity pseudo-measurements. While the number of viable velocity pseudo-measurements is reduced when simulating GNSS unavailability, the proposed step length model shows only a slight downward trend once the slow pace section begins. This trend is desirable as the pedestrian's step length was reduced during this portion.

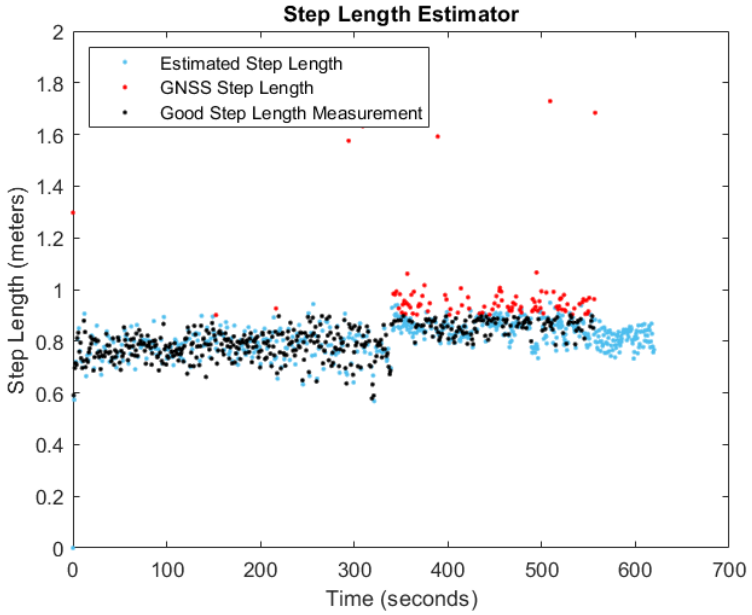


Figure 7.13: Step Length Estimation for GNSS mostly available

Table 7.8 below shows the results of the SLKF during the different paces of the route. Initially, the average estimated step length during the different paces of the route show more desirable trends than when using position-velocity pseudo-measurements. The average step length estimate goes up from 0.78 meters during the normal pace to 0.85 during the fast pace; and goes down from 0.85 meters during the fast pace to 0.81 meters during the slow pace. However, the pedestrian's average estimated step length is still 0.03 meters greater during the slow pace than the normal pace.

Table 7.8: Step Length Estimation RMSE

Pace	SLKF		
	steps	mean	sigma
Normal	577	0.78	0.06
Fast	413	0.85	0.04
Slow	98	0.81	0.03

Figure 7.14 below shows the evolution the proposed step length model coefficients during

the walking route. Additionally, Figure 7.15 shows the results of the step length estimation and the change in average estimated length over time. The model coefficients show a somewhat constant average in estimated values for the first 300 steps of the route and then a slight linear increase for the remainder of the normal pace. A more logarithmic trend can be seen in the step length estimation results during the normal pace where the average estimated step length quickly grows from 0 meters to about 0.75 meters. After the first 300 steps, the average step length continues to grow at a slower rate. This growth trend indicates that the total average step length during the normal pace of the walking route is most likely greater than 0.78 meters.

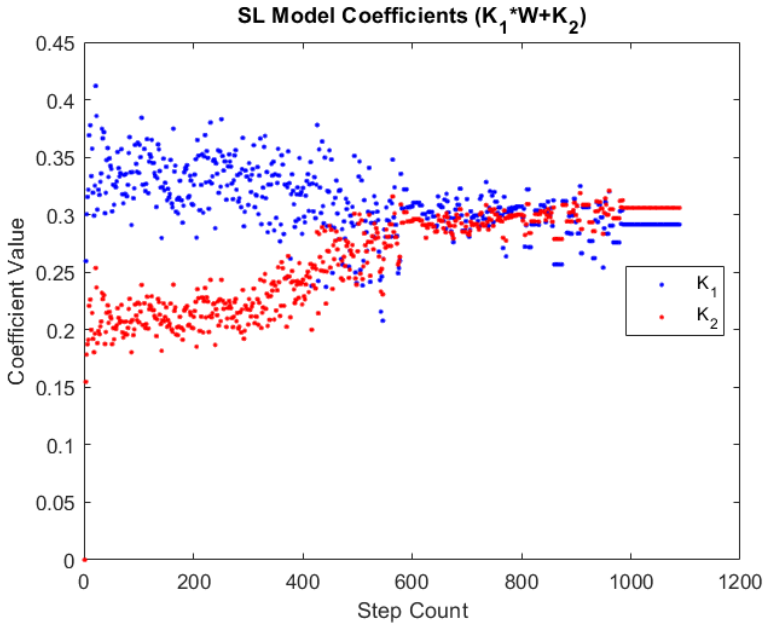


Figure 7.14: Step Length Model Coefficients

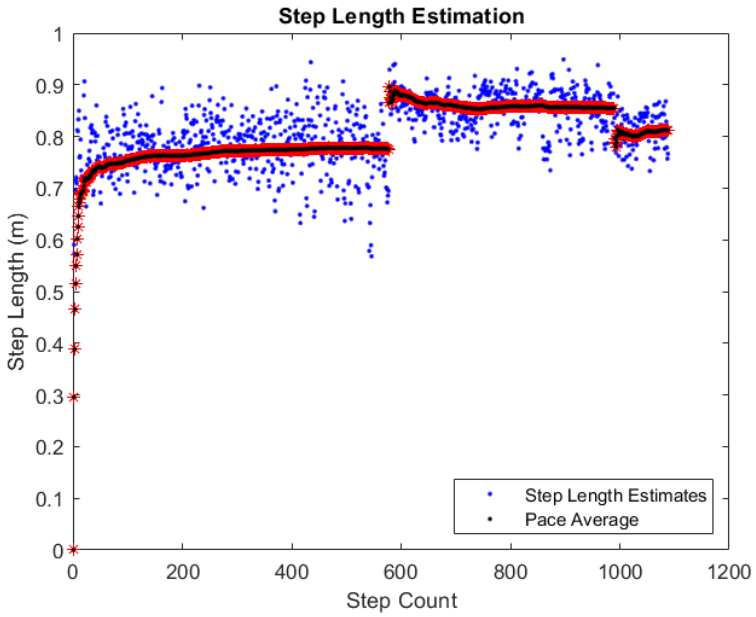


Figure 7.15: Step Length Estimation for GNSS mostly available

After 970 steps have been taken the step length model coefficients reach a final estimate to determine the pedestrian's step length for the remainder of the walking route. The resulting step length model is given by:

$$SL = 0.2915 \cdot W + 0.3059 \tag{7.5}$$

The resulting step length model for when GNSS step length velocity pseudo-measurements are mostly available will be examined at the end of this section. The next scenario will examine the same environmental condition when GNSS is unavailable for the majority of the walking route.

7.2.3 GNSS Rarely Available

Position-Velocity Pseudo-Measurements

The final scenario for the specified environmental condition simulates a loss of GNSS signal before the start of the second leg of the route when the pedestrian's pace increases. The goal of this scenario is similar to the goal of the previous scenario, but with only step length pseudo-measurements available during the normal walking pace of the route. For this

scenario, GNSS was simulated to be available for only the first 345 seconds of the walking route. Shown below in Figure 7.16 are the path error results for GNSS being available for a shorter amount of time. Additionally, the RMSE for the east and north directions are given in Table 7.9.

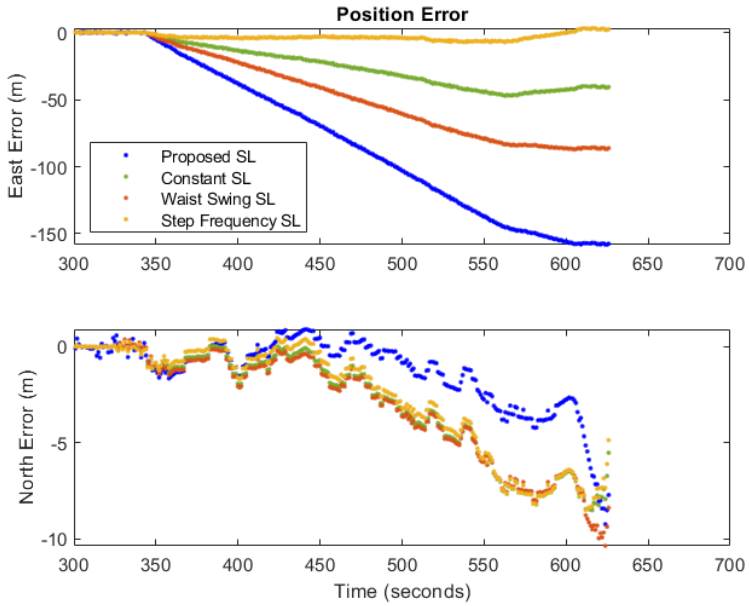


Figure 7.16: North and East Path Errors

Table 7.9: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Proposed	69.506	1.673
Constant	21.018	2.924
Waist Swing	39.745	3.221
Step Frequency	2.779	2.827

With GNSS being unavailable for almost half of the walking route, the proposed step length model produces a maximum of over 150 meters of error in the East direction. The proposed step length model underestimates the path traveled greatly underestimates the

distance traveled for the final two legs of the walking route. However, from Table 7.9, it can be seen that step frequency step length model outperforms all of the other step length models for RMSE for the East direction and estimates the remaining distance traveled with a higher level of accuracy.

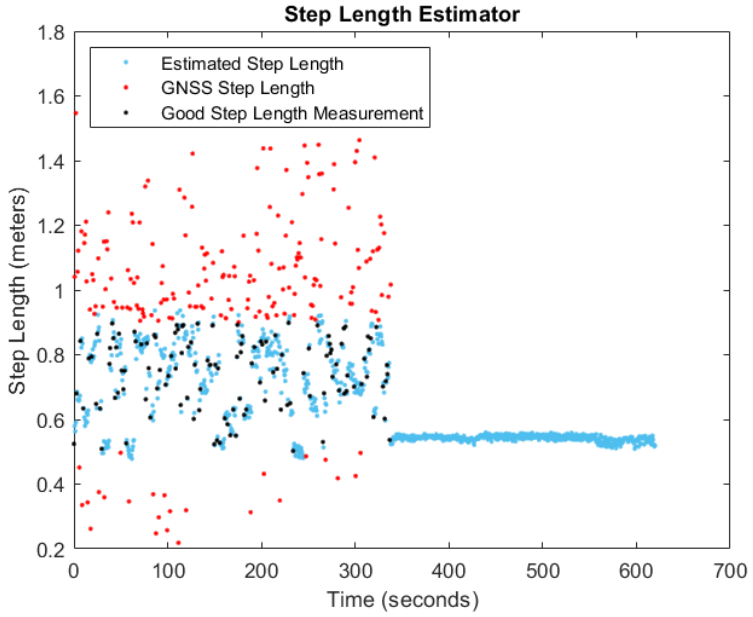


Figure 7.17: Step Length Estimation for GNSS rarely available

Figure 7.17 shows the results of the proposed step length model estimates with the GNSS step length pseudo-measurements. With GNSS only being available for the first 344 seconds of the walking route, only 339 pseudo-measurements were recorded. Of the 339 total pseudo-measurements, only 125 were viable to be used for parameter estimation. However, the final viable step length pseudo-measurement used for parameter estimation was recorded as 0.537 meters which is within 2 standard deviations of the mean. Table 7.10 shows the SLKF results for the different legs of the route.

Table 7.10: Step Length Estimation RMSE

Pace	SLKF		
	steps	μ (m)	σ (m)
Normal	577	0.73	0.12
Fast	413	0.54	0.01
Slow	98	0.53	0.01

During the second leg of the walking route, the pedestrian's average step length is estimated to be 0.54 meters, which is almost 0.2 meters less than the average step length during the normal walking pace in the first leg.

Figure 7.18 shows the change in the step length model parameters with only 125 viable step length pseudo-measurements used for estimation.

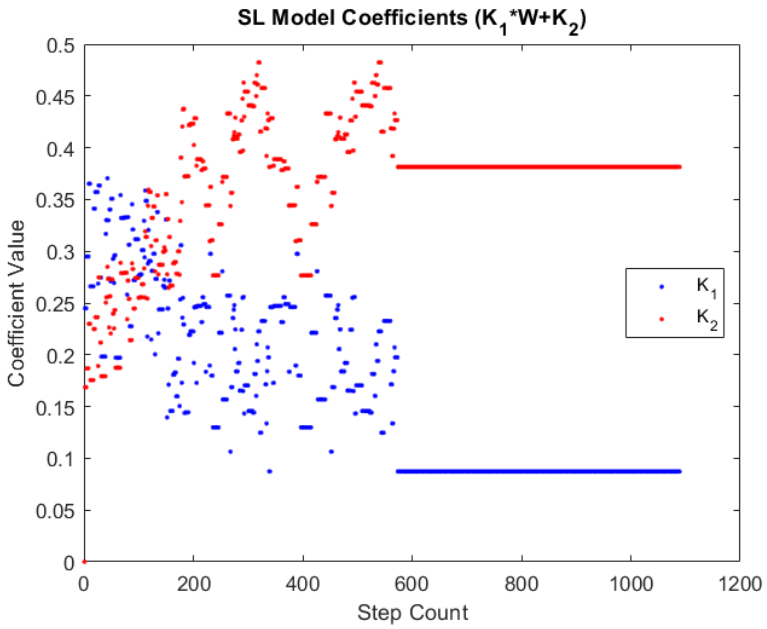


Figure 7.18: Step Length Model Coefficients

The total number of steps taken before the model coefficients converge to constant values is 577. From there, the final coefficient terms yield a resulting step length model of:

$$SL = 0.087 \cdot W + 0.381 \tag{7.6}$$

The resulting model coefficients are much closer in values than the previous two scenarios. The next section will examine the PDR solution using the three resulting step length models shown above.

Velocity Pseudo-Measurements

Using the GNSS step length velocity pseudo-measurements, the experimental path errors are shown below in Figure 7.19 to show the evolution in the PDR errors when GNSS is no longer available. Additionally, Table 7.11 shows the corresponding RMSE in the north and east directions for the four step length models. With GNSS only being available for about the first 344 seconds of the route, the proposed step length model produces a maximum error in the east direction of almost 100 meters. While this is marginally better than the waist-swing step length model, the constant step length model outperforms all the other models overall, and especially during the PDR period. However, this is still an improvement over the errors when using step length position-velocity pseudo-measurements. Not shown in the plot is the large errors produced by the step frequency model much earlier in the route. Although, this model appears to show improvements during the PDR period.

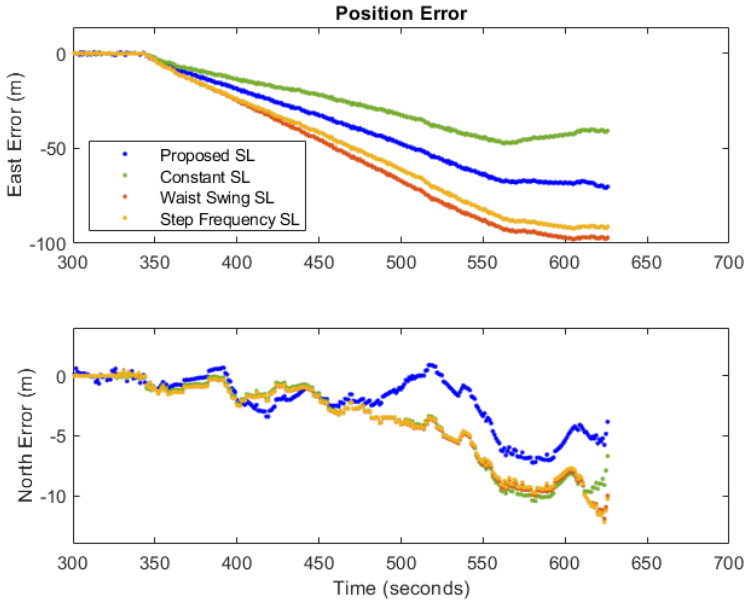


Figure 7.19: North and East Path Errors

Table 7.11: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
Proposed	31.664	2.223
Constant	21.092	3.624
Waist Swing	44.300	3.672
Step Frequency	253.584	18.721

Figure 7.20 below shows the results of the proposed step length model estimates with the GNSS step length velocity pseudo-measurements. When simulating the GNSS loss for this scenario, only 5 velocity pseudo-measurements were not considered to be viable. However, the step length estimation results show that the estimated step length is underestimated during the fast pace of the route and no visible change can be seen when transitioning to the slow pace of the route. This is very undesirable as the model should estimate a higher step length during the fast pace, and a lower estimate during the slow pace.

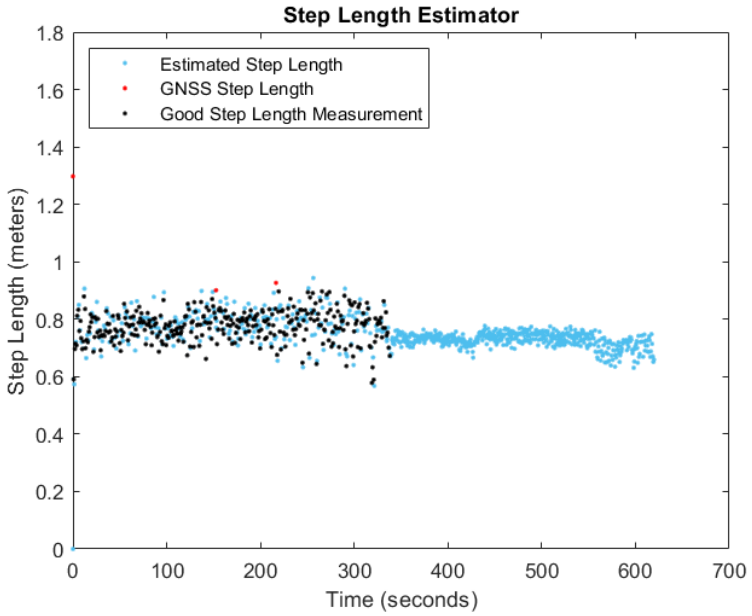


Figure 7.20: Step Length Estimation for GNSS rarely available

Table 7.12 below shows the results of the SLKF during the different paces of the route. It can be seen that the step length model is not able to adapt to the change in step length during the fast pace of the walking route. However, the step length model is able to estimate the reduction in step length during the slow pace. While it is desireable for the model to estimate smaller step lengths when the pace decreases, the step lengths are underestimated during the fast pace of the route.

Table 7.12: Step Length Estimation RMSE

Pace	SLKF		
	steps	μ (m)	σ (m)
Normal	577	0.78	0.06
Fast	413	0.73	0.02
Slow	98	0.70	0.03

Figure 7.21 below shows the evolution of the proposed step length model coefficients during the walking route. Additionally, Figure 7.22 shows the results of the step length estimation and the change in average estimated length over time. Initially, both models appear to evolve in a linear trend while velocity pseudo-measurements are available. However, the final velocity pseudo-measurement before the loss occurs drives the model coefficients to final values that are smaller than their average trend would suggest. The final step length velocity pseudo-measurement is calculated to be 0.6723 meters. Recall in Table 7.4 from Section 7.2.1 that the average step length velocity pseudo-measurements during the normal pace was 0.78 meters with a standard deviation of 0.05 meters. This means that the final step length velocity pseudo-measurement used for estimating the model coefficients is greater than 2σ less than the average velocity pseudo-measurements during the normal pace. This indicates a flaw in the SLKF measurements update since all viable velocity pseudo-measurements have equivalent measurement covariances.

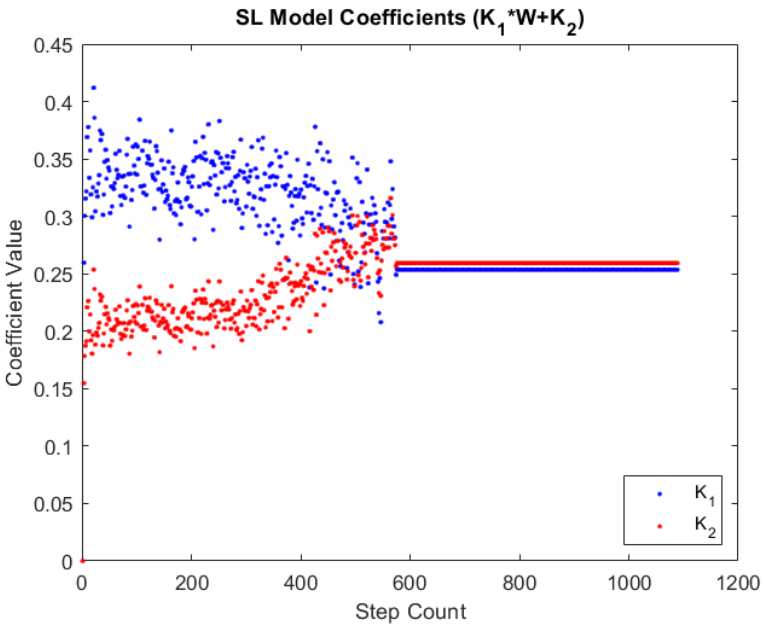


Figure 7.21: Step Length Model Coefficients

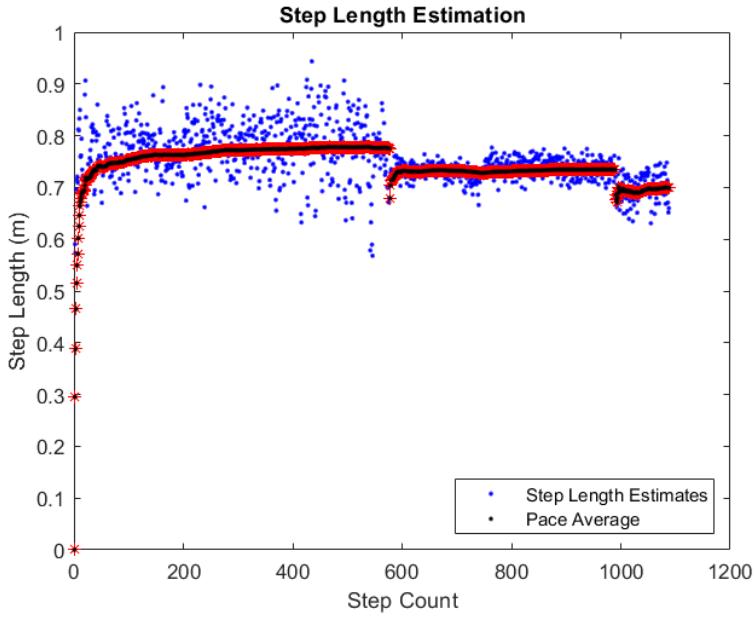


Figure 7.22: Step Length Estimation for GNSS rarely available

After the last velocity pseudo-measurement is applied to the SLKF, the model coefficients reach a final value for the remainder of the walking route. The final step length model is given by:

$$SL = 0.2535 \cdot W + 0.2594 \tag{7.7}$$

The resulting model will be evaluated for performance in the next section.

7.2.4 PDR Solution

For the final scenario of the specified environmental condition, the three resulting step length model for different periods of GNSS availability will be compared to the test case step length models for PDR alone. The goal of this scenario is to examine the performance of the proposed step length model, and to examine the performance of the use of the GNSS step length pseudo-measurements.

Position-Velocity Pseudo-Measurements

Shown below in Figure 7.23 are the path and path errors for the PDR-SD solution for the three proposed step length model results using step length position-velocity pseudo-measurements, and the test case step length models. The satellite basemap is not included in the path plot to show all the resulting paths.

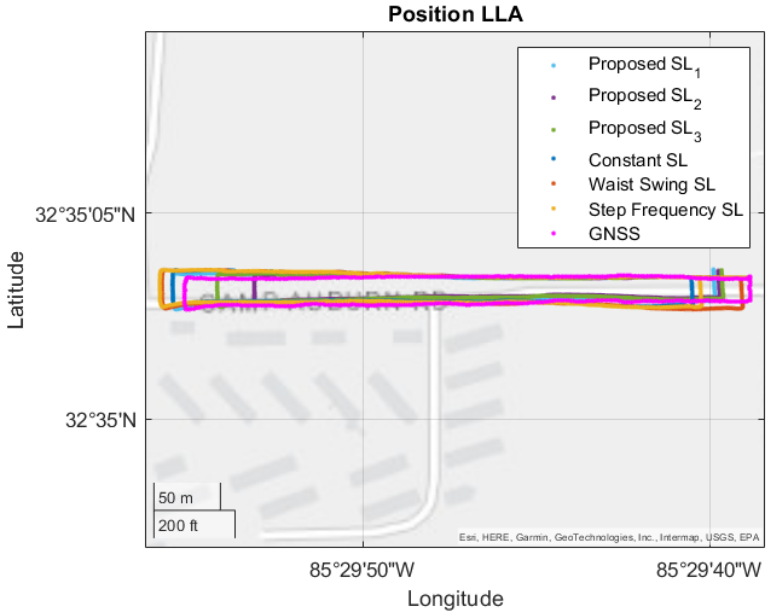


Figure 7.23: PDR-SD Path with Step Length Models

Figure 7.23 shows the path results where proposed step length models are listed in succession of how they were presented:

- SL_1 is the resulting model with constant GNSS availability.
- SL_2 is the resulting model when GNSS is mostly available.
- SL_3 is the resulting model when GNSS is rarely available.

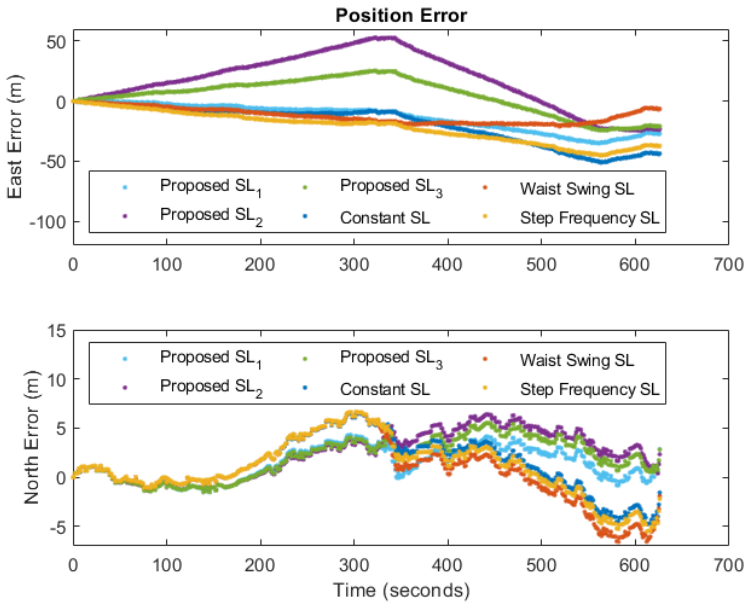


Figure 7.24: North and East Path Errors

Table 7.13: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
SL_1	17.415	2.128
SL_2	28.562	3.273
SL_3	15.333	2.777
Constant	24.854	3.009
Waist Swing	13.779	3.157
Step Frequency	25.359	3.044

Velocity Pseudo-Measurements

Figure 7.25 below shows the PDR solution for the proposed step length model of varying GNSS availability, and test case step length models. Additionally, Figure 7.26 shows the resulting path errors of each model's PDR solution. The results are shown in succession of how they were presented:

- SL_1 is the resulting model with constant GNSS availability.
- SL_2 is the resulting model when GNSS is mostly available.
- SL_3 is the resulting model when GNSS is rarely available.

Interestingly, the proposed step length model when GNSS is rarely available shows large positive errors in the east direction. This indicates that the proposed step length model overestimated the step length more frequently than the other step length models. However, this model appears to perform better in the north direction than most of the other models.

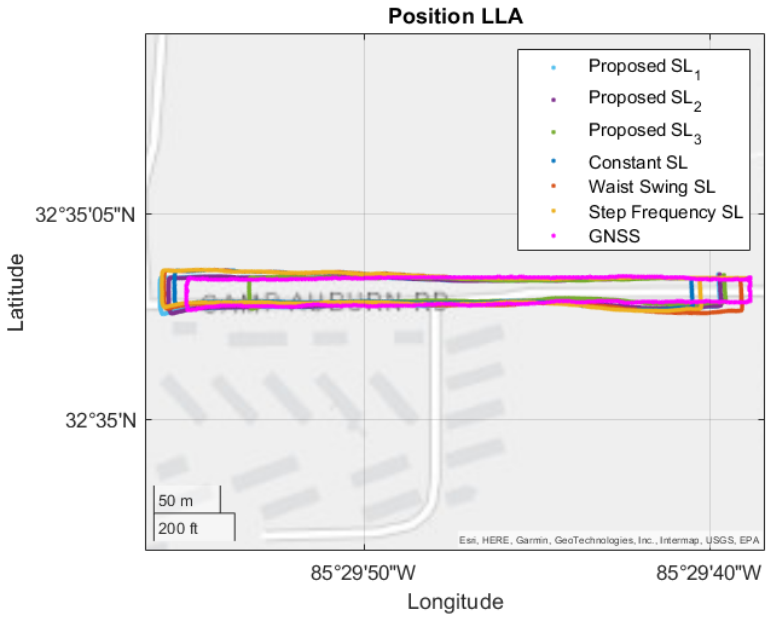


Figure 7.25: PDR-SD Path with Step Length Models

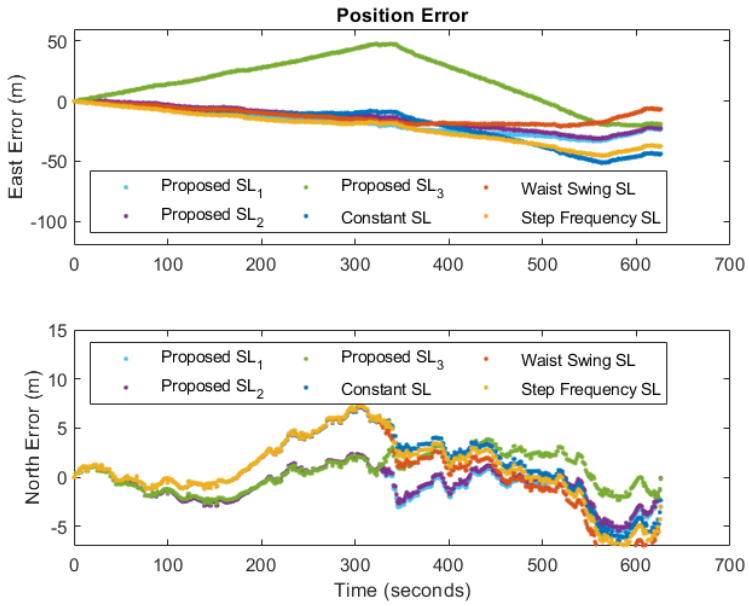


Figure 7.26: North and East Path Errors

Table 7.14 below shows the RMSE in the east and north directions for all six step length models. The waist-swing model continues to show the best performance in the PDR solution when compared to the other models. The proposed step length model when GNSS is mostly available has the best RMSE of the three scenarios using velocity pseudo measurements with a maximum error of 31.45 meters in the east direction. The resulting model from GNSS being rarely available when using position-velocity pseudo-measurements has the lowest RMSE of all the proposed models. However, the resulting models from using step length velocity pseudo-measurements have a much lower variation than the models using position-velocity pseudo-measurements. The lower variation in step length velocity pseudo-measurements show an RMSE difference of 8.483 m^2 between the best performing step length model (SL_2) and the worst step length model (SL_3); whereas, the RMSE difference between the best and worst performing step length models using position-velocity pseudo-measurements is 13.23 m^2.

Table 7.14: Step Length Estimation RMSE

Step Length Method	RMSE (m^2)	
	East	North
SL_1	20.461	2.013
SL_2	17.259	1.891
SL_3	25.742	1.855
Constant	25.014	3.372
Waist Swing	13.940	3.701
Step Frequency	25.515	3.472

The lowest RMSE in the PDR solution from the proposed step length models using position-velocity pseudo-measurements was when the leading coefficient (K_1) was more than four times less than the second coefficient (K_2), and the highest RMSE was when the leading coefficient was only 2.3 times smaller than the second coefficient. However, the resulting step length model from constant GNSS availability has an RMSE much closer to the better performing model with the leading coefficient being only 2.4 times less than the second coefficient. Overall, this would indicate a higher performance when the leading coefficient is less than the second coefficient.

The lowest RMSE in the PDR solution from the proposed step length models using velocity pseudo-measurements was when the leading coefficient was 1.05 times less than the second coefficient. Unlike the analysis of the models using position-velocity pseudo-measurements, the RMSE increased as the ratio increased to 1.14 for SL_1 and increased even larger as the ratio decreased to 1.02. This differs from the expected trend of the position-velocity step length models, as the RMSE did not improve as K_2 grew larger than K_1. These results indicate that the performance of the step length model may not be defined by the difference in the coefficients. Additionally, all of the resulting step length models from both pseudo-measurement approaches show a smaller leading coefficient than second coefficient.

7.3 Misalignment Estimation Test

The second environmental condition examines the performance of the system when heading misalignment between the magnetometer and the pedestrian is present. The experimental setup for this test is the same data set from the first environmental condition with a Vectornav VN-100 9-DoF IMU [30] rigidly mounted to the right of the pedestrian's sternum, and a Ublox EVK-7P GNSS receiver with a single antenna mounted to the pedestrian's left shoulder. The IMU and magnetometer are sampled at 100 Hz, and the GNSS receiver is sampled at 1 Hz. Two of the same scenarios of GNSS availability from the previous environmental condition will be examined for this scenario where:

- GNSS is constantly available.
- GNSS is rarely available.

The results of this experiment will focus on the system's ability to estimate the misalignment angle for various periods of GNSS availability. With this in mind, the alternative step length estimation methods from the previous section will not be examined.

With the misalignment angle between the magnetometer and pedestrian unaccounted for, the pedestrian's position is propagated in the heading direction determined by the magnetometer. Figure 7.27 shows the PDR path results without removing the misalignment.

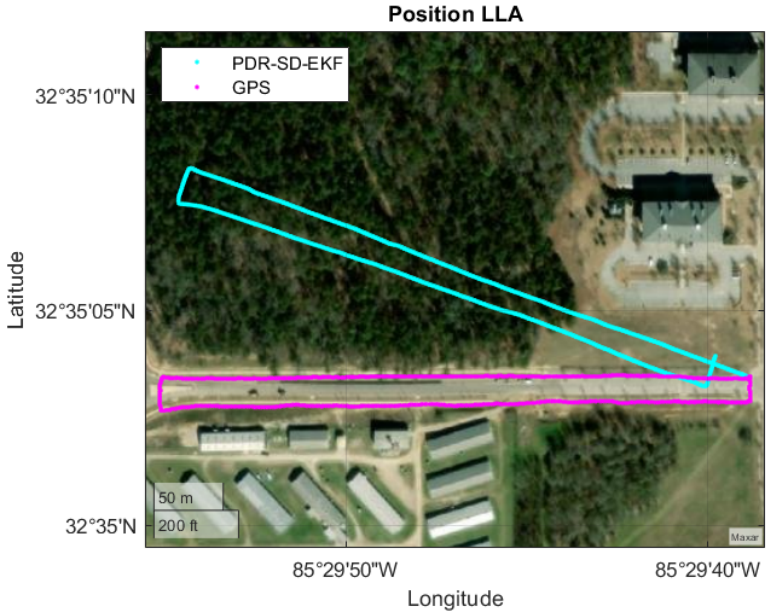


Figure 7.27: PDR Path with Magnetometer Misalignment

The path was generated using the proposed step length model with constant GNSS availability. The heading is determined by the AHRS subsystem and does not account for the misalignment between the magnetometer and the pedestrian. Figure 7.28 shows the results of the heading estimated by the AHRS compared to the reference GNSS course measurements. As shown, there exists a, relatively, constant misalignment angle of about 20 degrees for the duration of the walking route. The following scenarios of GNSS availability will examine the performance of the AHRS subsystem's ability to estimate the misalignment angle in order to account for it.

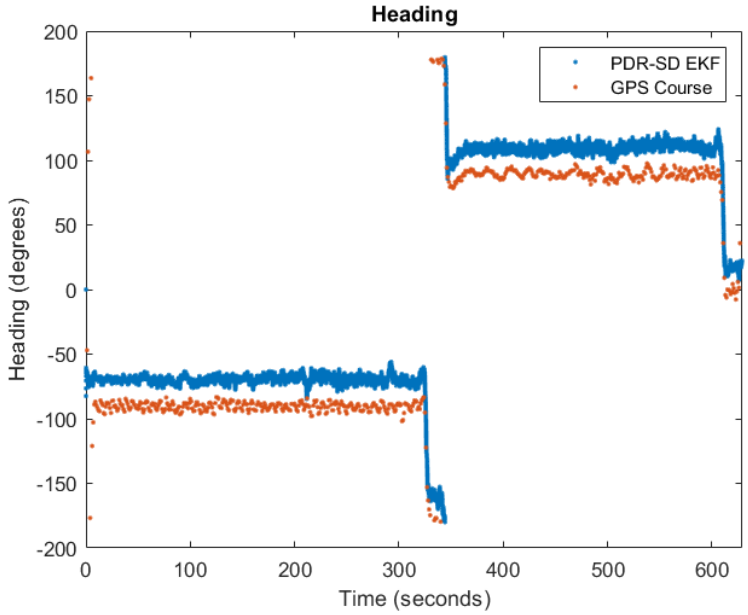


Figure 7.28: Heading and GNSS Course comparison with Misalignment

7.3.1 GNSS Constant Availability

For the first scenario of the environmental condition, GNSS is assumed to be constantly available for the duration of the experiment. Figure 7.29 shows the results of the pedestrian's heading estimation with constant GNSS availability. With the misalignment estimation, the heading from the AHRS subsystem is constantly corrected by estimates of the misalignment angle. This results in the estimated heading and GNSS Course to be very similar. Results of the misalignment estimation are shown below in Figure 7.30. The misalignment angle has a final convergence value of -20.6 degrees. The total average of the estimated misalignment angle is determined to be -20.3 degrees with a standard deviation of 2.4 degrees.

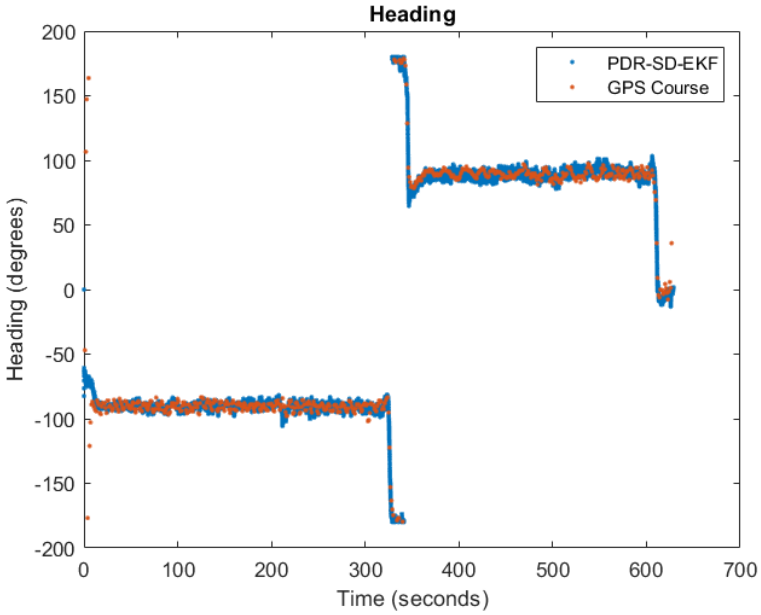


Figure 7.29: Heading and GNSS Course comparison

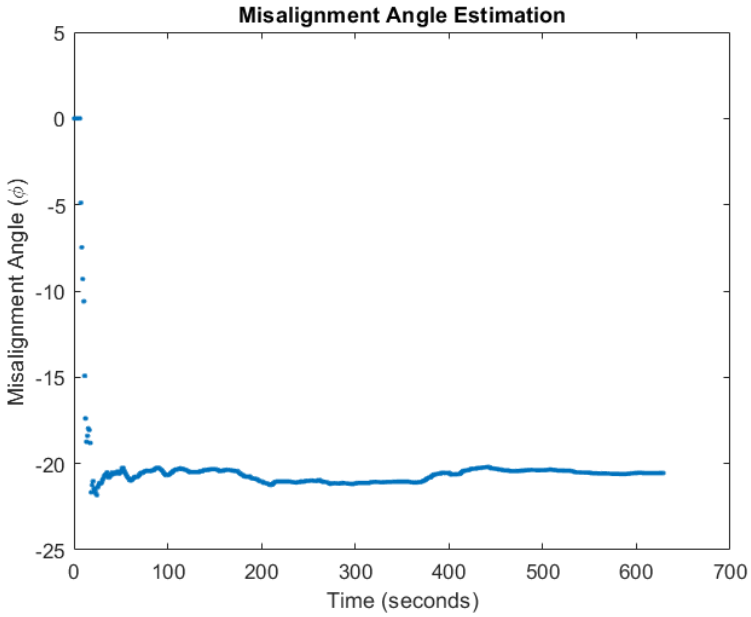


Figure 7.30: Misalignment Estimation

The misalignment angle estimate converges to within 1σ of the average after about 13 seconds, or 13 GNSS Course measurement updates and remains within 1σ for the remainder of the route. However, the time to convergence is based on the availability of viable GNSS measurements when the pedestrian is traveling over 0.5 m/s. With the pedestrian beginning in a static position for the first 6 seconds of the route, the first 6 GNSS Course measurements are not viable for misalignment estimation. With this in mind, the misalignment estimation converged with only 7 Course measurement updates.

In Figure 7.31, the dead reckoned path errors in the North direction are shown for zero, one, two, and three standard deviations from the average misalignment angle when GNSS is constantly available. For the majority of the route, the pedestrian is traveling in the Eastern direction where path errors are primarily attributed to errors in the selected step length estimation model. Examining the errors in the North direction allow greater insight into how the misalignment angle affects the performance of the system. Misalignment angle estimates 3σ from -20.3 degrees introduce heading errors that result in path errors greater than 50 meters in the North direction.

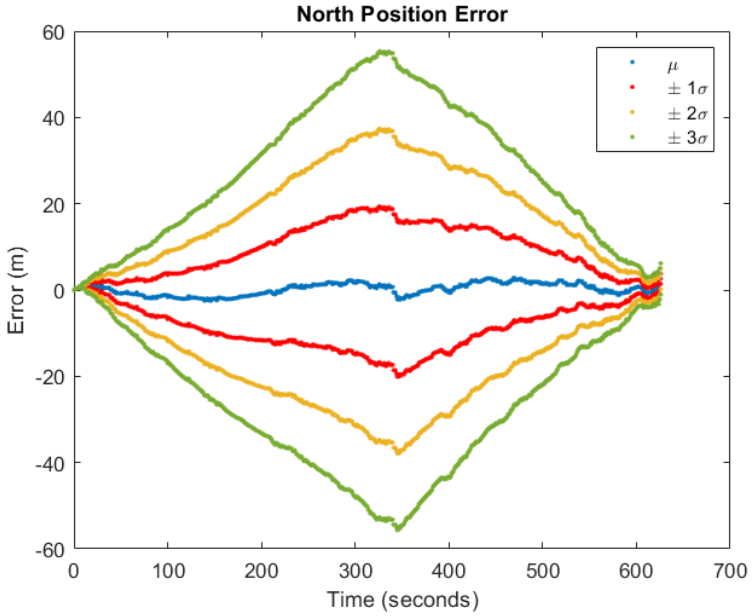


Figure 7.31: North Error with Misalignment Error bounds

7.4 Magnetic Disturbance Test

The final environmental condition examines misalignment between the magnetometer and the pedestrian, and the system's ability to detect and reject magnetometer measurement affected by disturbances in the local magnetic field. The experimental setup for this test uses a clockwise walking route in a parking lot where the pedestrian walks a single loop beginning and ending at the southwest corner of the path. This route also includes a period where the magnetometer measurements are perturbed by a disturbance in the local magnetic field. The pedestrian is outfitted with a Vectornav VN-100 9-DoF IMU [30] rigidly mounted to the right of the pedestrian's sternum and a Ublox EVK-7P GNSS receiver with a single antenna mounted to the pedestrian's left shoulder. The IMU and magnetometer are sampled at 100 Hz, and the GNSS receiver is sampled at 2 Hz. This test examines three levels of GNSS availability:

- GNSS is constantly available
- GNSS is lost during the period of magnetic disturbance.
- GNSS is lost before the period of magnetic disturbance.

The results of this test will focus on the system's ability to account for the misalignment angle between the magnetometer and the pedestrian, as well as, detect periods of magnetic disturbances and reject those measurements.

For the preliminary results, the pedestrian's dead reckoned position is shown in Figure 7.32, where the misalignment angle is unaccounted for and the disturbed magnetometer measurements are not removed. The path is generated using the converged step length model parameters when GNSS is constantly available given by Equation (7.8) below.

$$SL = 0.267 \cdot W + 0.325 \tag{7.8}$$

Again, this test focuses on the system's ability to account for misalignment and reject magnetometer measurements influenced by local disturbances. Therefore, this step length model will be used for the remainder of this test and will not be evaluated for performance.

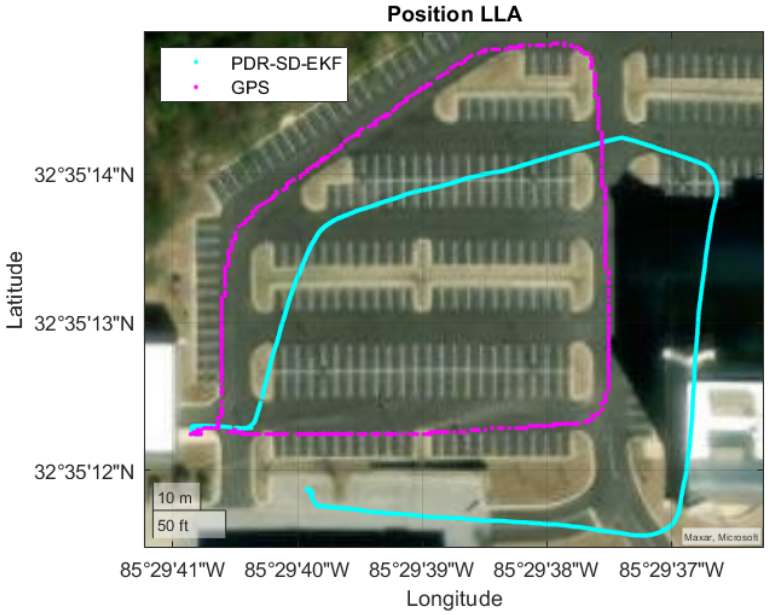


Figure 7.32: PDR Path with Misalignment and Magnetic Disturbance

The results for the heading without accounting for misalignment or magnetic disturbances is shown below in Figure 7.33. Initially, a relatively constant misalignment angle of about 17 degrees is present while the pedestrian is walking. However, around 230 seconds into the route, the misalignment angle reduces closer to 10 degrees of misalignment. While this appears as though the heading is correcting itself, it's actually due to the presence of a disturbance in the local magnetic field for the remainder of the route.

Figure 7.34 below shows the measured magnetic field strength from the calibrated magnetometer for the route. During the periods where the pedestrian's heading is held relatively constant on straightaways, the magnetic field strength is also relatively constant. However, around the 240 seconds mark, the magnetic field strength becomes less consistent with the expected trends due to a local disturbance. The remainder of this section will examine the periods of GNSS availability and the subsystem's performance of misalignment estimation and rejection of faulty magnetometer measurements.

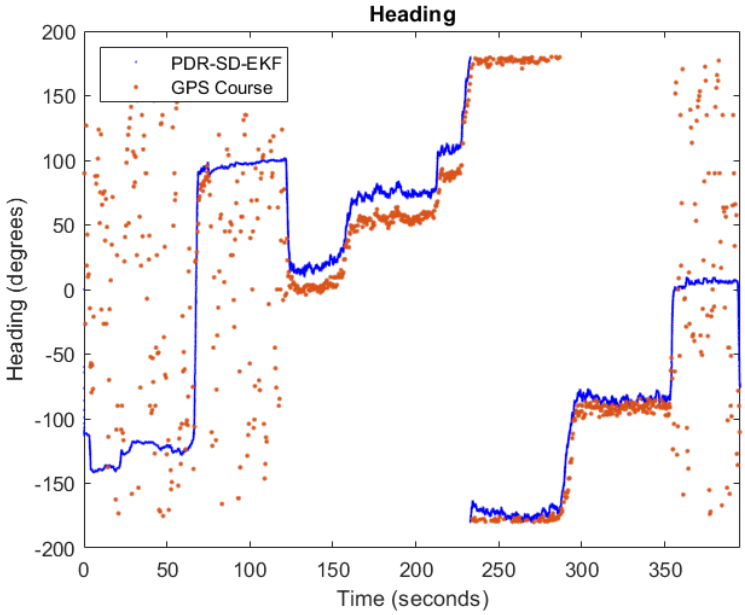


Figure 7.33: Heading and GNSS Course comparison

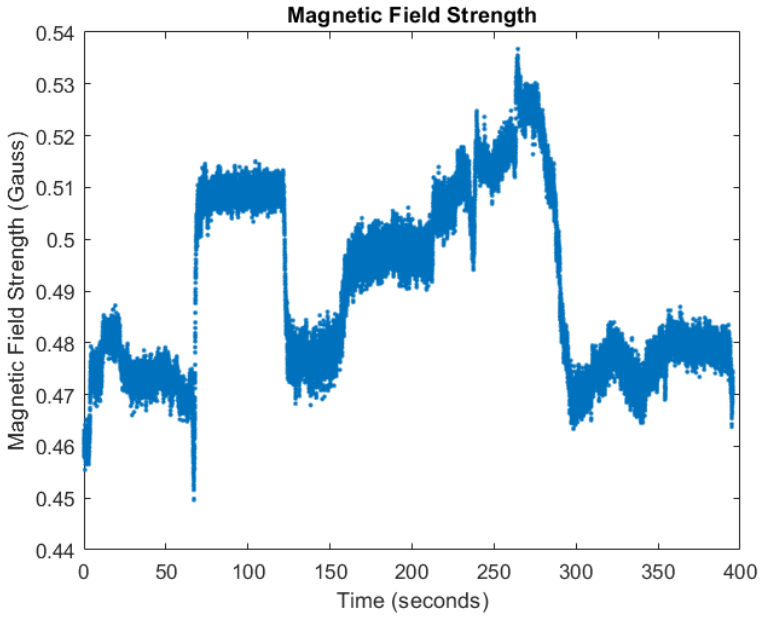


Figure 7.34: Measured Magnetic Field Strength

7.4.1 GNSS Constantly Available

The first scenario will examine the subsystem performance when GNSS is constantly available. Without rejecting the disturbed magnetometer measurements, the misalignment angle estimate would deviate from the converged value due to the effect of the magnetic disturbance on the estimated heading. Figure 7.35 below shows the misalignment estimation when disturbed magnetometer measurements are not rejected. After about 230 seconds, the estimated misalignment angle has converged to about -17 degrees. From there, the disturbed magnetometer measurements drive the misalignment estimate to about -13.7 degrees.

Figures 7.36 and 7.37 show the PDR position and heading solution when the misalignment angle is held constant at -17 degrees. Using the first converged misalignment angle estimate removes the errors in the heading estimates for the first three straightaways of the route. However, once the pedestrian turns and begins traveling South on the fourth straightaway, the misalignment estimate does not remove the errors in the heading estimates from the disturbed magnetometer measurements for the remainder of the route.

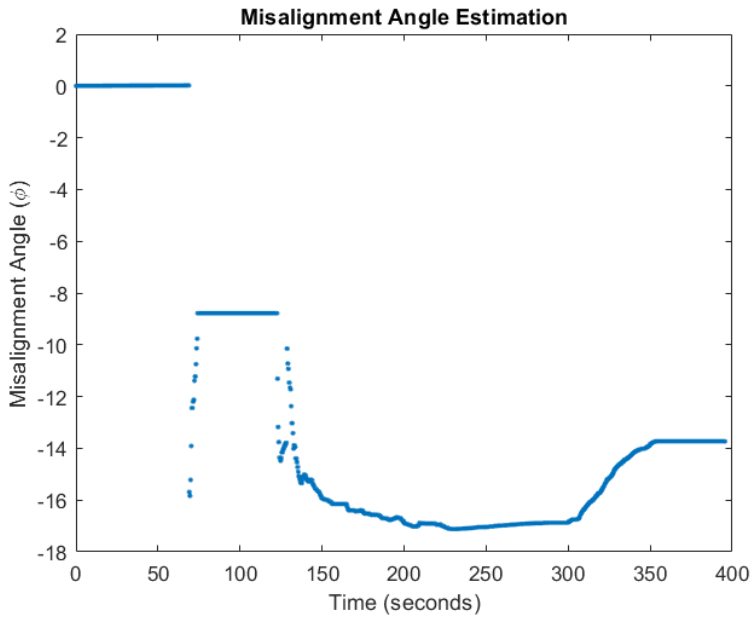


Figure 7.35: Misalignment Estimation with Constant GNSS Availability

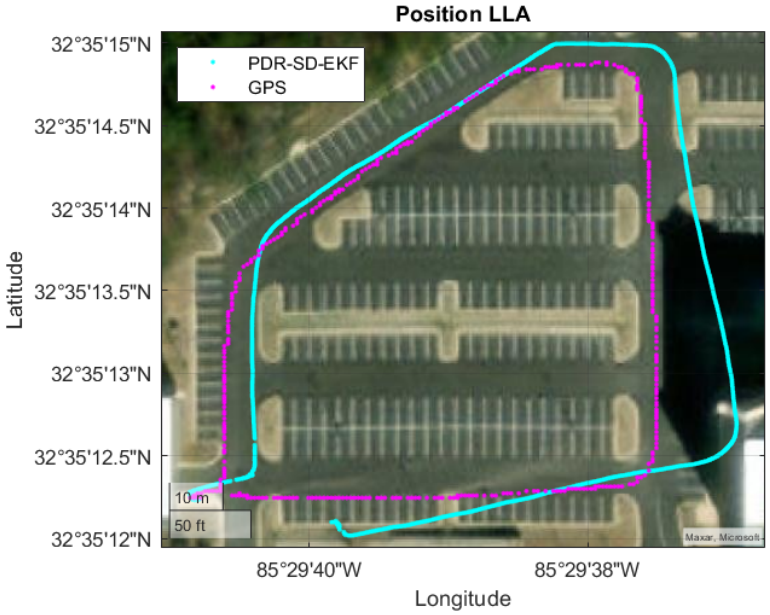


Figure 7.36: PDR Path With Misalignment Angle ($\psi_o = -17$ degrees)

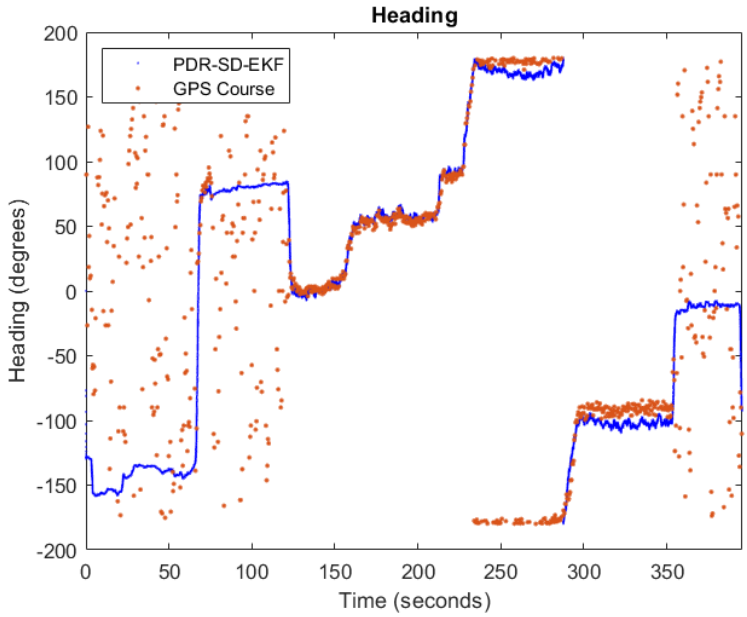


Figure 7.37: AHRS Heading and GNSS Course With Misalignment Angle ($\psi_o = -17$ degrees)

With the inclusion of the magnetic fault detection, the misalignment angle should remain constant, even after the magnetometer measurements have become corrupted. Recall from Section 6.4.2, the conditions for detecting a disturbed magnetometer measurement require the calculated NIS value to be below a threshold of 50, and the misalignment covariance must be below 0.0001 rad^2. Shown below in Figure 7.38 is the estimated misalignment angle covariance crossing this specified threshold. Around 134 seconds into the route, the estimated misalignment angle covariance drops below the required threshold and remains below the threshold for the remainder of the route. From this point on, if the NIS value is calculated to be above a threshold of 50, then a faulty magnetometer measurement is detected and the magnetometer measurement is rejected for that epoch. Figure 7.39 shows the calculated NIS value for each axis when none of the measurements are rejected. From 134 seconds and beyond, any axis that computes a NIS value greater than 50 will be rejected. With this in mind, magnetometer measurements will start being rejected along the second straightaway, rather than the fourth.

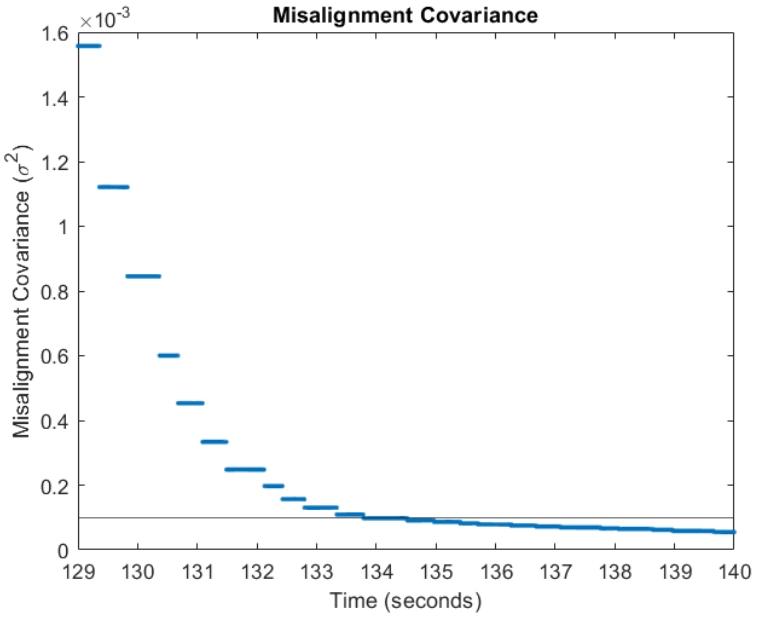


Figure 7.38: Misalignment Covariance with Constant GNSS Availability

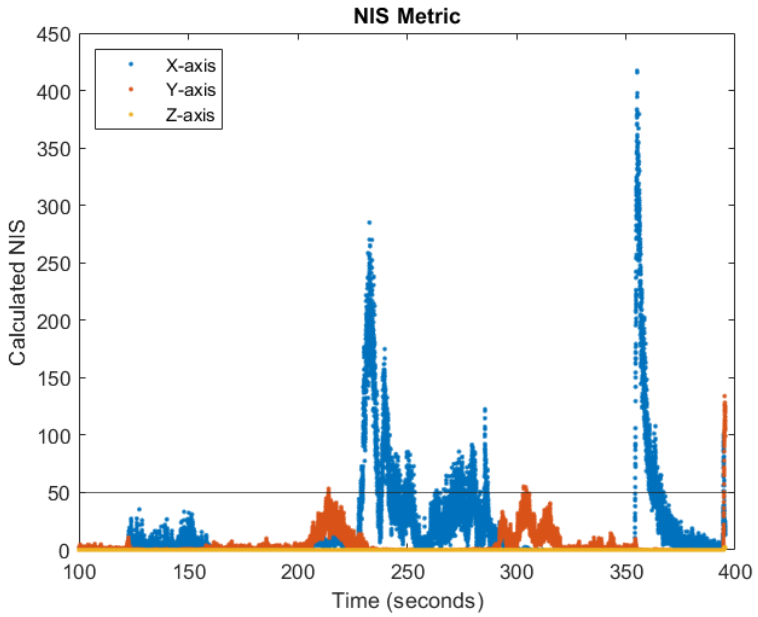


Figure 7.39: Calculated NIS value for each axis ($NIS_{TH} = 50$)

For constant GNSS availability, the results of the estimated heading are shown in Figure 7.40. With the magnetic fault detection algorithm implemented, the estimated heading is much closer to the trends shown by the GNSS Course measurements during the route and $9,803$ faulty magnetometer measurements were detected. However, during the final straightaway from 300 seconds to 350 seconds, a considerable amount of misalignment is still present. This is an indication that the magnetic disturbance is still not fully accounted for.

Figure 7.41 shows the misalignment angle estimation for constant GNSS availability. For the ideal case, the misalignment angle estimate would converge to a relatively constant value and not deviate even in the presence of the magnetic disturbance.

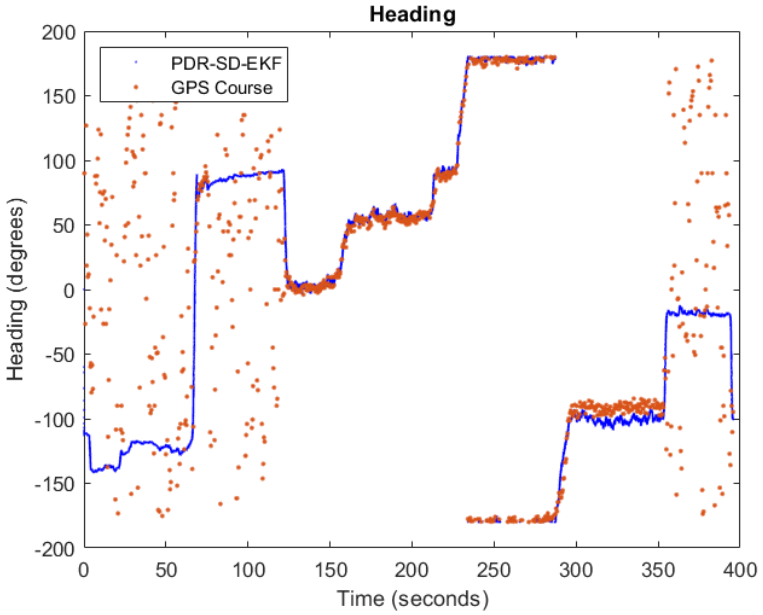


Figure 7.40: Heading Estimates and GNSS Course with Fault Detection

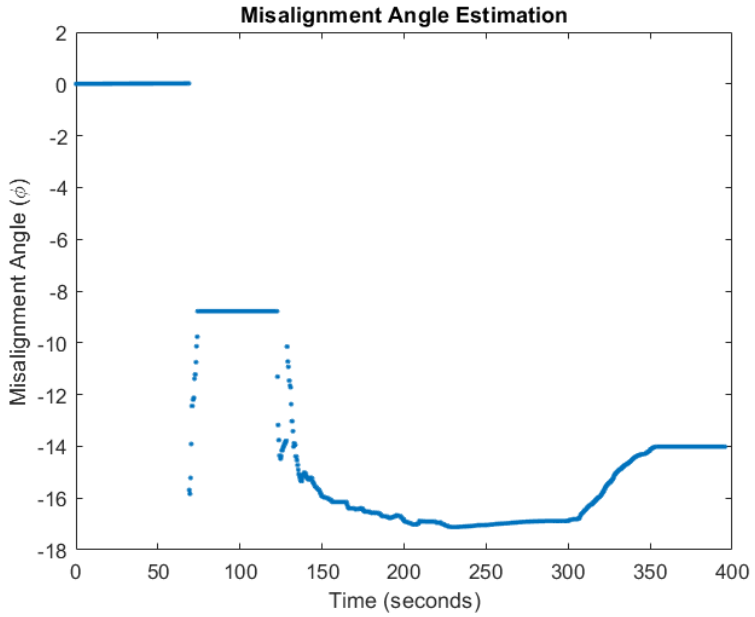


Figure 7.41: Misalignment Estimation With Magnetometer Fault Rejection

To attempt to obtain the desired misalignment estimation response, a NIS threshold of 30 is examined. Lowering the NIS threshold indicates that magnetometer measurements with a smaller amount of error with the estimated orientation from the gyroscope will be rejected. The misalignment angle estimation covariance will remain unchanged at $\sigma_{TH}^2 = 0.0001$ rad^2. Figure 7.42 shows the results of the heading estimation with a NIS threshold of 30. With the NIS threshold lowered to 30, the estimated heading matches the heading from GNSS Course much closer than when the NIS threshold was set at 50 and $13,507$ faulty measurements were detected.

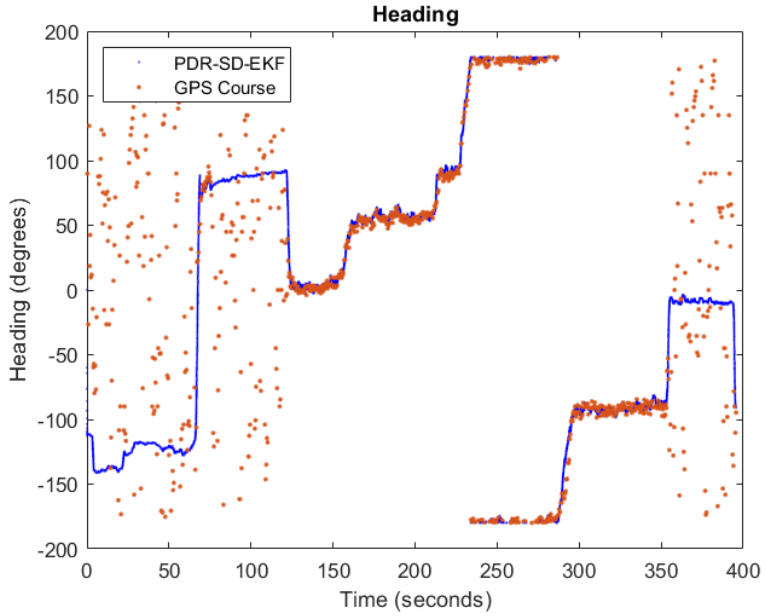


Figure 7.42: Heading Estimates With Fault Detection ($\text{NIS}_{TH} = 30$)

Figure 7.43 shows the misalignment angle estimation when the NIS threshold is adjusted to 30. The final misalignment angle estimate is calculated to be $-17.5°$ with much less deviation from the true misalignment angle to produce results much closer to the ideal case. Due to the preferred performance of the magnetic fault detection when the NIS threshold is set to 30, the remaining tests will use the new threshold for fault detection. The next scenario will examine the same environmental condition when GNSS signal becomes unavailable during the magnetic disturbance.

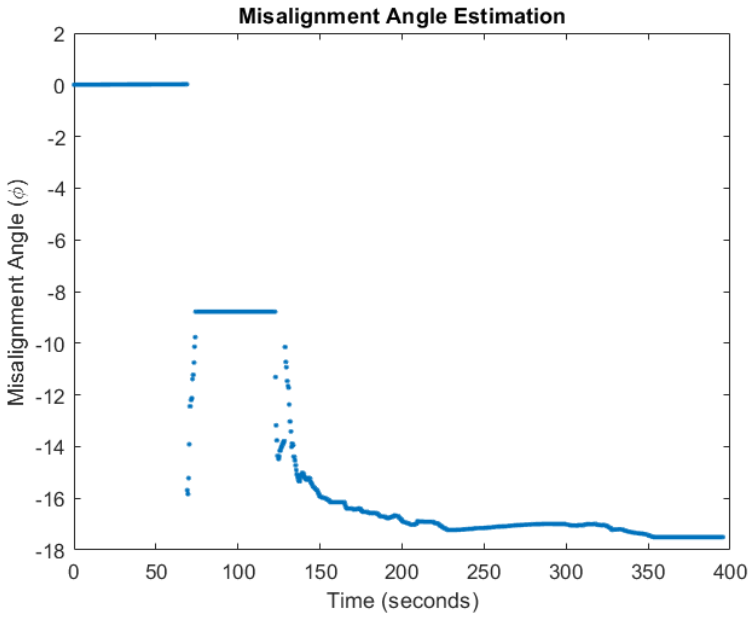


Figure 7.43: Misalignment Estimation ($\text{NIS}_{TH} = 30$)

7.4.2 GNSS Loss During Magnetic Disturbance

The next scenario for the specified environmental condition simulates a loss of GNSS signal during the period of the magnetic disturbance. The goal of this scenario is to evaluate the system's ability to reject faulty magnetometer measurements after the misalignment angle has converged to the final estimate using GNSS. For this scenario the loss of GNSS signal will occur at about 273 seconds into the route when the pedestrian is walking South and the magnetic disturbance has already begun.

Shown below in Figures 7.44 and 7.45 are the heading estimation and misalignment estimation results when a total loss of GNSS is simulated after 273 seconds.

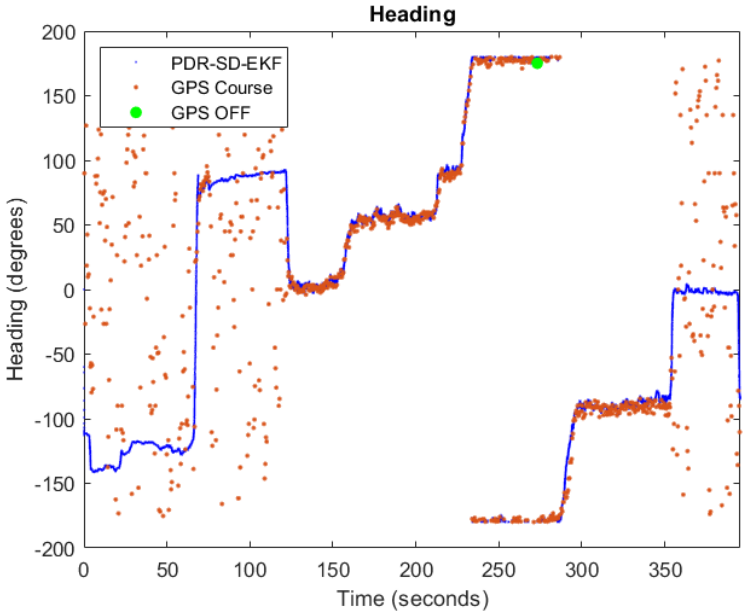


Figure 7.44: Heading Estimation with GNSS Loss at 273 seconds

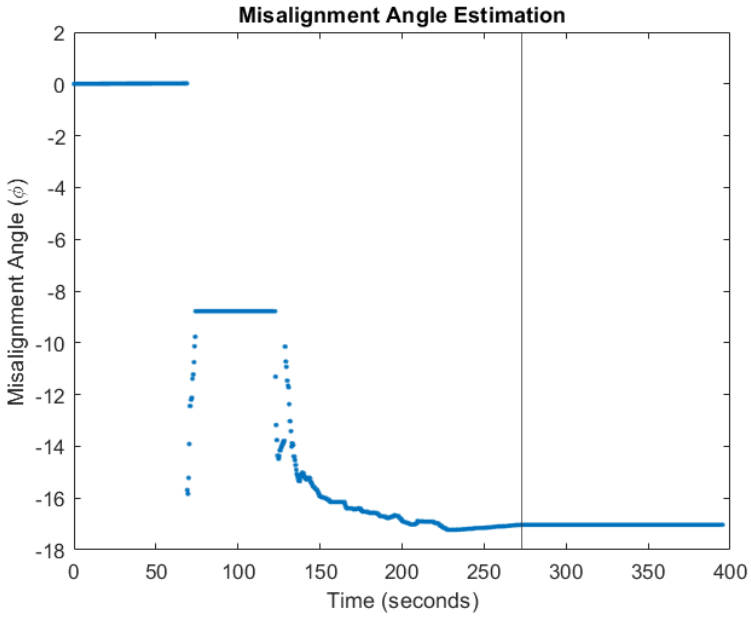


Figure 7.45: Misalignment Estimation with GNSS Loss at 273 seconds

The total number of detected magnetometer faults is $17,356$, which is almost $4,000$ more than when GNSS was constantly available. The final misalignment angle estimate converges to $-17.03°$ and remains there for the remainder of the route. Again, this implementation uses a NIS threshold of 30, and a misalignment estimation covariance threshold of 0.0001 rad^2. Recall from Section 6.4.2 that the additive process noise of the misalignment angle is given by

$$w_{\dot{\psi}_o} \sim N(0, 0.00001 \text{ rad}^2) \tag{7.9}$$

Once the loss of GNSS is simulated the misalignment angle covariance grows by 1.2×10^{-10} rad^2 and still remains below the threshold. With this in mind, the rejection of the magnetic disturbances is entirely dictated by the NIS condition. The final scenario will simulate a loss of GNSS before the magnetic disturbance begins.

7.4.3 GNSS Loss Before Magnetic Disturbance

The final scenario for the specified environmental condition simulates a loss of GNSS signal before the period of the magnetic disturbance. The goal of this scenario is to evaluate the system's ability to reject faulty magnetometer measurements before and after one of the conditions is no longer satisfied. In this case, the loss of GNSS is simulated to occur after 134 seconds into the route. This time was chosen because the misalignment angle covariance drops below the threshold almost half a second prior to this epoch. This will show the converged misalignment angle estimate near the covariance threshold.

Figures 7.46 and 7.47 show the heading estimation and misalignment estimation for a loss of GNSS simulated at 134 seconds.

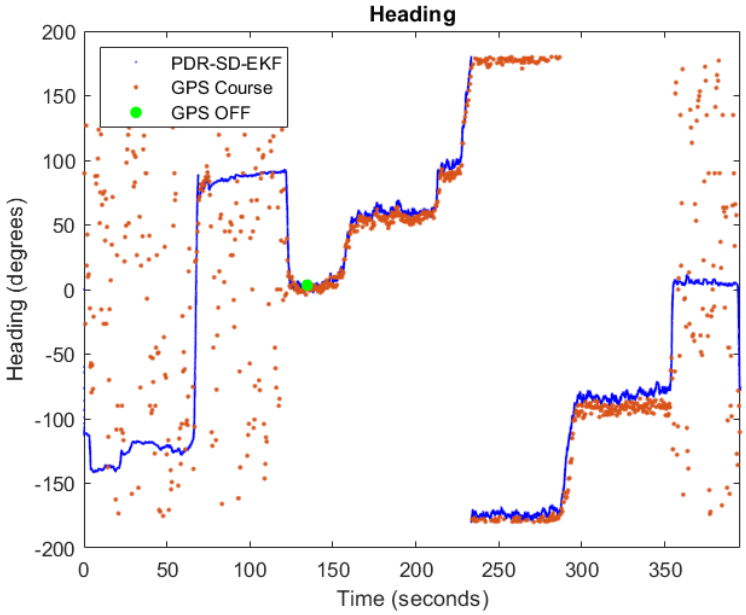


Figure 7.46: Heading Estimation with GNSS Loss at 134 seconds

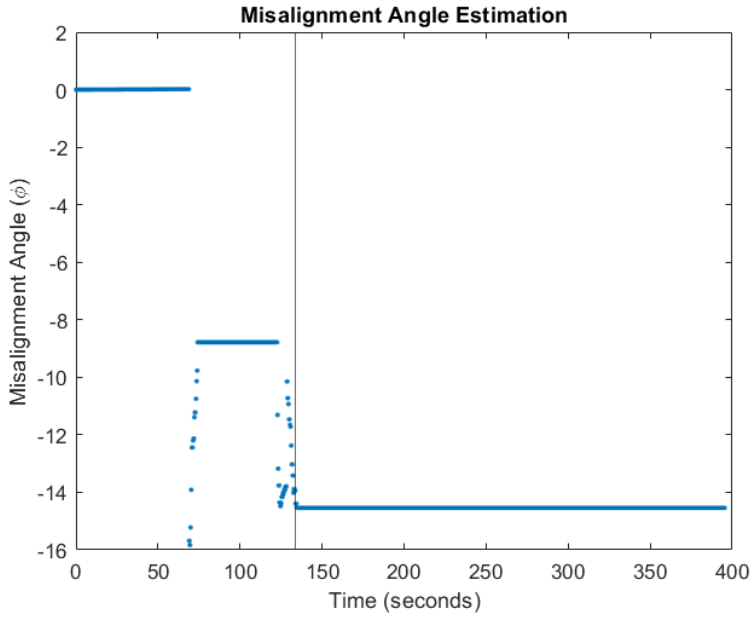


Figure 7.47: Misalignment Estimation with GNSS Loss at 134 seconds

A total of $18,295$ faulty magnetometer measurements were detected and rejected, which is almost $1,000$ more than when GNSS was cut after 273 seconds. The final misalignment angle estimate was calculated to be $-14.5°$. The true misalignment angle is closer to $-17°$ indicating that the misalignment angle is not fully estimated when the covariance crosses the specified threshold.

Chapter 8

Summary and Conclusions

8.1 Summary

Pedestrian navigation has proved to come with challenges that many systems are not subject to, or are more simplistic to solve. Pedestrians are often limited to the amount of sizable hardware that can be carried, as well as the level of comfort desired for mounting such hardware. While many navigation systems for ground, marine, and aerial vehicles can improve the navigation solution with higher quality sensors and devices, the problem of pedestrian navigation is often faced with sacrificing accuracy for comfort and vice versa. Pedestrian's are also faced with challenges of navigating in environments that other systems are not always exposed to and require alternative methods of positioning. In the case of GNSS, this positioning system does not always accommodate the modern environments that a pedestrian would travel in (i.e. indoors, dense foliage, urban canyons, underground). However, the improvements in modern technology of body worn inertial sensors offer an alternative method of navigation through PDR.

PDR is faced with the challenge of initialization information and the unbounded growth of errors over time. Position initialization can be obtained through GNSS receivers, while orientation initialization can be obtained from gravity and magnetic field measurements of the Earth. When attempting to mitigate the unbounded growth of errors over time, several techniques were shown for incorporating knowledge of the pedestrian walking motion with the anatomical location of inertial sensors. To better constrain the growth of errors over

time, the modern methods of integrating PDR with GNSS in a loosely coupled framework were presented. An evaluation of how the PDR solution performed after GNSS was no longer available was also provided.

Several methods of extending the loosely coupled integration of GNSS and PDR for IMUs mounted to the torso was introduced. The first extension of this integration presented a novel step length estimation model along with a method of GNSS pseudo-measurements to discretely measure the pedestrian's step length for model parameter estimation. The next extension presents a method of estimating the misalignment angle between the pedestrian and the magnetometer with GNSS Course measurements. The final extension proposed a method of detecting and rejecting magnetometer measurements that have become corrupted by local disturbances in the magnetic field. The integration of these subsystem extensions with GNSS position and velocity measurements in an overarching EKF framework was introduced to attempt to mitigate errors that would be seen in a more traditional torso mounted PDR navigation system.

The proposed GNSS/PDR-SD-EKF integration was then implemented on real-world data sets that examined different periods of GNSS availability for specific environmental scenarios to better analyze the performance of each subsystem. The step length estimation with GNSS pseudo-measurements subsystem was evaluated by negating the effects of magnetometer misalignment and disturbances in the local magnetic field. Next, the misalignment estimation subsystem was evaluated without the effects of local magnetic field disturbances. Finally, the magnetometer fault detection and misalignment estimation subsystems were evaluated for data that included a disturbance in the local magnetic field.

8.2 Conclusions

This thesis provides a detailed background of the current methods for PDR systems. This focused on the classic PDR-INS methods for foot mounted IMU approaches, and the more recent developments in PDR-SD methods for torso mounted IMU approaches. Both of these approaches and methods were integrated with single-antenna GNSS receivers to reveal their drawbacks. Next, proposed extensions to the loosely coupled GNSS/PDR-SD approach were presented to account for additional sources of error that degrade the PDR solution when GNSS is no longer available. These methods and extensions reduce errors that

result from incorrect step length estimates, heading errors due to misalignment between the inertial sensors and the pedestrian, and heading errors that result from faulty magnetometer measurements.

The first environmental scenario showed that the proposed step length estimation model using position-velocity pseudo-measurements may offer some improvements of step length estimation for variable step sizes when the model leading coefficient is determined to be much less than the following coefficient. However, the results from using velocity pseudo-measurements show that this relationship is not valid and the ratio of coefficient values does not determine the step length model performance. Additionally, no scenarios were examined for when the leading coefficient is larger than the second coefficient. This scenario would need to be examined to identify a relationship between the model coefficients.

The GNSS step length position-velocity pseudo-measurements provide very little benefit for measuring discrete step lengths regardless of the pedestrian's pace and step size. The position-velocity pseudo-measurements are much noisier than the velocity pseudo-measurements and had much fewer valid measurements. The velocity pseudo-measurements are less noisy and are able to show changes in step length more effectively. With this in mind, velocity pseudo-measurements should be considered as a reference step length measurement for a desired model.

The next environmental test scenario proved the improvements of reducing heading errors when misalignment between the magnetometer and the pedestrian is present. The use of an average moving window of course measurements from GNSS velocity are able to quickly determine the pedestrian's true heading that can be utilized to remove any heading errors as a result of hardware misalignment. Accounting for hardware misalignment allows for co-located inertial sensors and magnetometers to be mounted in any position and orientation on the pedestrian's torso.

The last environmental scenario validated the performance of both the misalignment estimation and magnetometer fault detection subsystems when misalignment and local disturbances in the magnetic field are both present. Using the NIS metric to detect faulty magnetometer measurements improved heading estimates when the magnetometer was corrupted. However, the effectiveness of this algorithm is dependent on the ability to estimate the hardware misalignment and requires that the pedestrian be initialized in a disturbance-free area.

8.3 Future Work

Below are potential areas of future work to improve upon the work presented in this thesis. These potential areas include:

- **RTK Corrections**: Real-time kinematic (RTK) positioning provide corrections to GNSS position and velocity measurements that result in a higher accuracy of GNSS measurements. Incorporating RTK for GNSS step length pseudo-measurements may offer improvements in step length model parameter estimation and better capture the true discrete step lengths of the pedestrian. Likewise, the higher fidelity velocity measurements may reduce the time to convergence of the misalignment angle estimation.

- **Lower Quality IMUs**: The real-world data sets used for implementation use MEMS grade IMUs of much higher quality than that of common handheld devices like cellphones and smartwatches. An evaluation of the system performance using lower grade IMUs would provide insight into the performance of the presented PDR-SD approach when inertial measurements become less accurate.

- **Magnetometer Fault Detection Conditions**: The performance of the magnetometer fault detection algorithm is based on the estimation of the misalignment angle, as well as the assumption that the local magnetic field is not disturbed during initialization. Alternative, or additional, conditions for detecting local magnetic disturbances should be examined. A potential method might be in the examination of differentiated heading measurement from the magnetometer with z-axis angular velocity measurements from a gyroscope.

- **Magnetometer Yaw Rate**: While disturbed magnetometer measurements are not ideal when estimating the pedestrian's heading, the discrete differentiation of magnetometer measurements may offer estimation of errors and biases in angular velocity measurements from the gyroscope on the yaw axis. This could increase the stability of estimating heading with dead reckoned gyroscope measurements during periods of magnetic disturbances.

  From the magnetic fault detection results shown in Section 7.4, the calibrated magnetometer measurements were differentiated along each axis. Figure 8.1 below shows a

comparison of the resulting differentiated tri-axial magnetometer measurements during the magnetic disturbance with the tri-axial angular velocity measurements during the magnetic disturbance. The magnetic disturbance begins around 150 seconds into the route and lasts around 200 seconds. In the bottom plot, the angular velocity measurements along the z-axis drift downwards as the magnetic disturbance begins, while the magnetometer measurements remain centered around 0 rad/s.

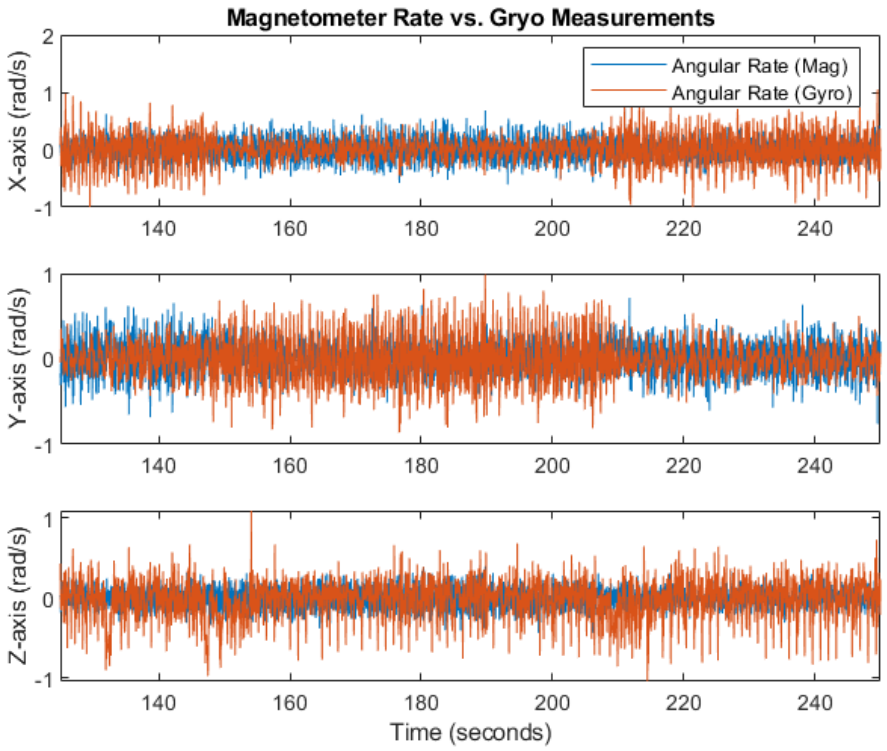


Figure 8.1: Gyroscope Measurements and Differentiated Magnetometer Measurements

Figure 8.2 below shows the estimates of the gyroscope biases along each axis over time. In the bottom plot, the z-axis gyroscope bias fluctuates as the corrupted magnetometer measurements update the orientation estimation. Utilizing the differentiated magnetometer measurements could potentially benefit the estimation of the gyroscope biases and reduce gyroscope drift even when a magnetic field disturbance is present.

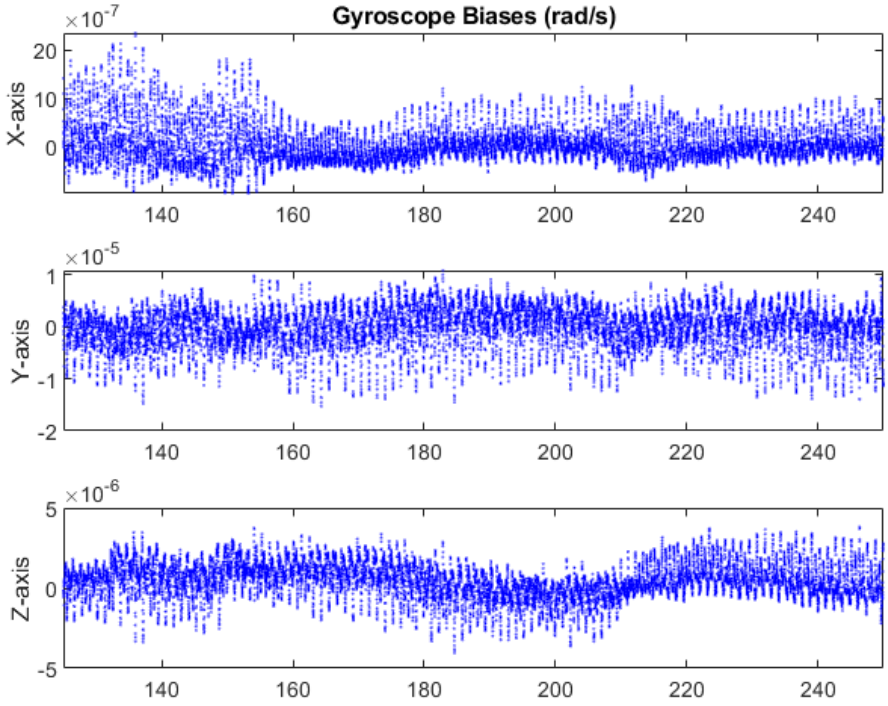


Figure 8.2: Tri-Axial Gyroscope Bias Estimates

References

[1] *Vaka Moana, Voyages of the Ancestors: The Discovery and Settlement of the Pacific*. en-US. June 2020. URL: https://uhpress.hawaii.edu/title/vaka-moana-voyages-of-the-ancestors-the-discovery-and-settlement-of-the-pacific/ (visited on 03/11/2023).

[2] Paul D. Groves. "Principles of GNSS, inertial, and multisensor integrated navigation systems, 2nd edition [Book review]". In: *IEEE Aerospace and Electronic Systems Magazine* 30.2 (Feb. 2015). Conference Name: IEEE Aerospace and Electronic Systems Magazine, pp. 26–27. ISSN: 1557-959X. DOI: 10.1109/MAES.2014.14110.

[3] E. Foxlin. "Pedestrian tracking with shoe-mounted inertial sensors". In: *IEEE Computer Graphics and Applications* 25.6 (Nov. 2005). Conference Name: IEEE Computer Graphics and Applications, pp. 38–46. ISSN: 1558-1756. DOI: 10.1109/MCG.2005.140.

[4] Paul D Groves et al. "Inertial navigation versus pedestrian dead reckoning: Optimizing the integration". en. In: (), p. 14.

[5] A.R. Jimenez et al. "Indoor pedestrian navigation using an INS/EKF framework for yaw drift reduction and a foot-mounted IMU". en. In: *2010 7th Workshop on Positioning, Navigation and Communication*. Dresden, Germany: IEEE, Mar. 2010, pp. 135–143. ISBN: 978-1-4244-7158-4. DOI: 10.1109/WPNC.2010.5649300. URL: http://ieeexplore.ieee.org/document/5649300/ (visited on 04/18/2022).

[6] Vincent Gabaglio. *GPS/INS integration for pedestrian navigation*. Tech. rep. EPFL, 2003.

[7] Dominik Gusenbauer, Carsten Isert, and Jens Krösche. "Self-contained indoor positioning on off-the-shelf mobile devices". In: *2010 International Conference on Indoor*

Positioning and Indoor Navigation. Sept. 2010, pp. 1–9. DOI: 10.1109/IPIN.2010.5646681.

[8] Jeong Won Kim et al. "A Step, Stride and Heading Determination for the Pedestrian Navigation System". en. In: *Journal of Global Positioning Systems* 3.1&2 (Dec. 2004), pp. 273–279. ISSN: 14463156, 14463156. DOI: 10.5081/jgps.3.1.273. URL: http://www.gnss.com.au/JoGPS/v3n12/v3n12p34.pdf (visited on 10/03/2022).

[9] S. H. Shin et al. "Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors". In: *2007 IEEE Sensors Applications Symposium*. Feb. 2007, pp. 1–5. DOI: 10.1109/SAS.2007.374406.

[10] Kai Zhao, Bing-Hao Li, and Andrew G. Dempster. "A New Approach of Real Time Step Length Estimation for Waist Mounted PDR System". In: *2014 International Conference on Wireless Communication and Sensor Network*. ISSN: null. Dec. 2014, pp. 400–406. DOI: 10.1109/WCSN.2014.88.

[11] Quentin Ladetto. "On foot navigation : continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering". en. In: (), p. 6.

[12] Abdelrahman Ali and Naser El-Sheimy. *Low-Cost MEMS-Based Pedestrian Navigation Technique for GPS-Denied Areas*. en. Research Article. ISSN: 1687-725X Library Catalog: www.hindawi.com Pages: e197090 Publisher: Hindawi Volume: 2013. 2013. DOI: https://doi.org/10.1155/2013/197090. URL: https://www.hindawi.com/journals/js/2013/197090/ (visited on 05/13/2020).

[13] Archit Thopay. "Initialization of a Pedestrian Navigation System Using a Transfer Alignment Approach". en. In: (Aug. 2020). Accepted: 2020-08-11T18:03:13Z. URL: https://etd.auburn.edu//handle/10415/7438 (visited on 03/11/2023).

[14] Jeremiah A. Shockley. "Ground Vehicle Navigation Using Magnetic Field Variation". English. ISBN: 9781267626905. Ph.D. United States – Ohio: Air Force Institute of Technology. URL: https://www.proquest.com/docview/1095536512/abstract/89FB37D6A1A04367PQ/1 (visited on 04/28/2022).

[15] Valérie Renaudin and Christophe Combettes. "Magnetic, Acceleration Fields and Gyroscope Quaternion (MAGYQ)-Based Attitude Estimation with Smartphone Sensors for Indoor Pedestrian Navigation". en. In: *Sensors* 14.12 (Dec. 2014), pp. 22864–22890.

ISSN: 1424-8220. DOI: 10.3390/s141222864. URL: http://www.mdpi.com/1424-8220/14/12/22864 (visited on 10/20/2022).

[16] Wonkeun Youn. "Magnetic Fault–Tolerant Navigation Filter for a UAV". In: *IEEE Sensors Journal* 20.22 (Nov. 2020). Conference Name: IEEE Sensors Journal, pp. 13480–13490. ISSN: 1558-1748. DOI: 10.1109/JSEN.2020.3004133.

[17] S Godha, G Lachapelle, and M E Cannon. "Integrated GPS/INS System for Pedestrian Navigation in a Signal Degraded Environment". en. In: (2006), p. 14.

[18] Gianluca Falco et al. "Performance Analysis of Constrained Loosely Coupled GPS/INS Integration Solutions". en. In: *Sensors* 12.11 (Nov. 2012). Number: 11 Publisher: Multidisciplinary Digital Publishing Institute, pp. 15983–16007. ISSN: 1424-8220. DOI: 10.3390/s121115983. URL: https://www.mdpi.com/1424-8220/12/11/15983 (visited on 09/02/2022).

[19] A. Rehman et al. "Accurate and Direct GNSS/PDR Integration Using Extended Kalman Filter for Pedestrian Smartphone Navigation". en. In: *Gyroscopy and Navigation* 11.2 (Apr. 2020), pp. 124–137. ISSN: 2075-1087, 2075-1109. DOI: 10.1134/S2075108720020054. URL: http://link.springer.com/10.1134/S2075108720020054 (visited on 01/04/2021).

[20] Johan Perul and Valerie Renaudin. "HEAD: smootH Estimation of wAlking Direction with a handheld device embedding inertial, GNSS, and magnetometer sensors". en. In: *NAVIGATION* 67.4 (2020). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/navi.389, pp. 713–726. ISSN: 2161-4296. DOI: 10.1002/navi.389. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/navi.389 (visited on 08/27/2021).

[21] Manon Kok, Jeroen D Hol, and Thomas B Schon. "Using Inertial Sensors for Position and Orientation Estimation". en. In: (), p. 95.

[22] Moti Ben-Ari. "A Tutorial on Euler Angles and Quaternions". en. In: (), p. 22.

[23] Joan Solà. "Quaternion kinematics for the error-state Kalman filter". In: *arXiv:1711.02508 [cs]* (Nov. 2017). arXiv: 1711.02508. URL: http://arxiv.org/abs/1711.02508 (visited on 06/04/2020).

[24] Per K. Enge. “The Global Positioning System: Signals, measurements, and performance”. en. In: *International Journal of Wireless Information Networks* 1.2 (Apr. 1994), pp. 83–105. ISSN: 1572-8129. DOI: 10.1007/BF02106512. URL: https://doi.org/10.1007/BF02106512 (visited on 10/06/2022).

[25] Greg Welch and Gary Bishop. *An introduction to the Kalman filter*. Tech. rep. 1995.

[26] Robert H. Rogne et al. “MEMS-based Inertial Navigation on Dynamically Positioned Ships: Dead Reckoning”. en. In: *IFAC-PapersOnLine* 49.23 (2016), pp. 139–146. ISSN: 24058963. DOI: 10.1016/j.ifacol.2016.10.334. URL: https://linkinghub.elsevier.com/retrieve/pii/S2405896316319218 (visited on 04/11/2022).

[27] Ashutosh Kharb et al. “A review of gait cycle and its parameters”. In: *IJCEM Int J Comput Eng Manag* 13 (Jan. 2011).

[28] Francisco J. Zampella et al. “Simulation of foot-mounted IMU signals for the evaluation of PDR algorithms”. In: *2011 International Conference on Indoor Positioning and Indoor Navigation*. ISSN: null. Sept. 2011, pp. 1–7. DOI: 10.1109/IPIN.2011.6071930.

[29] Daniel Pierce. “Incorporation of a Foot-Mounted IMU for Multi-Sensor Pedestrian Navigation”. en. In: (Apr. 2016). Accepted: 2016-04-29T19:38:32Z. URL: https://etd.auburn.edu//handle/10415/5074 (visited on 03/11/2023).

[30] *VectorNav's VN-100 IMU/AHRS, the world's most trusted surface mount solution*. en. URL: https://www.vectornav.com/products/detail/vn-100 (visited on 10/14/2022).

[31] *ZED-F9P module*. en-US. Apr. 2018. URL: https://www.u-blox.com/en/product/zed-f9p-module (visited on 10/14/2022).

[32] C. Randell, C. Djiallis, and H. Muller. “Personal position measurement using dead reckoning”. en. In: *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings.* White Plains, NY, USA: IEEE, 2003, pp. 166–173. ISBN: 978-0-7695-2034-6. DOI: 10.1109/ISWC.2003.1241408. URL: http://ieeexplore.ieee.org/document/1241408/ (visited on 09/27/2022).

[33] Neil Zhao. “Full-Featured Pedometer Design Realized with 3-Axis Digital Accelerometer”. en. In: (2010), p. 5.

[34] Robert Harle. “A Survey of Indoor Inertial Positioning Systems for Pedestrians”. In: *IEEE Communications Surveys & Tutorials* 15.3 (2013). Conference Name: IEEE Communications Surveys & Tutorials, pp. 1281–1293. ISSN: 1553-877X. DOI: 10.1109/SURV.2012.121912.00075.

[35] Qinghua Zeng et al. “Seamless Pedestrian Navigation Methodology Optimized for Indoor/Outdoor Detection”. In: *IEEE Sensors Journal* 18.1 (Jan. 2018). Conference Name: IEEE Sensors Journal, pp. 363–374. ISSN: 1558-1748. DOI: 10.1109/JSEN.2017.2764509.

[36] Inge Bylemans, Maarten Weyn, and Martin Klepal. “Mobile Phone-Based Displacement Estimation for Opportunistic Localisation Systems”. In: *2009 Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*. Oct. 2009, pp. 113–118. DOI: 10.1109/UBICOMM.2009.23.

[37] Harvey Weinberg. “Using the ADXL202 in Pedometer and Personal Navigation Applications”. en. In: (), p. 8.

[38] Wiebren Zijlstra and At Hof. “Displacement of the pelvis during human walking: Experimental data and model predictions”. In: *Gait & Posture - GAIT POSTURE* 6 (Dec. 1997), pp. 249–262. DOI: 10.1016/S0966-6362(97)00021-0.

[39] Jasper Jahn et al. “Comparison and evaluation of acceleration based step length estimators for handheld devices”. In: *2010 International Conference on Indoor Positioning and Indoor Navigation*. Sept. 2010, pp. 1–6. DOI: 10.1109/IPIN.2010.5646888.

[40] Fan Li et al. “A reliable and accurate indoor localization method using phone inertial sensors”. en. In: (), p. 10.

[41] Young Soo Suh. “Orientation Estimation Using a Quaternion-Based Indirect Kalman Filter With Adaptive Estimation of External Acceleration”. In: *IEEE Transactions on Instrumentation and Measurement* 59.12 (Dec. 2010). Conference Name: IEEE Transactions on Instrumentation and Measurement, pp. 3296–3305. ISSN: 1557-9662. DOI: 10.1109/TIM.2010.2047157.

[42] A. Rehman et al. “Accurate and Direct GNSS/PDR Integration Using Extended Kalman Filter for Pedestrian Smartphone Navigation”. en. In: *Gyroscopy and Navigation* 11.2 (Apr. 2020), pp. 124–137. ISSN: 2075-1109. DOI: 10.1134/S2075108720020054. URL: https://doi.org/10.1134/S2075108720020054 (visited on 08/27/2021).

[43] Adrian Waegli and Dr Jan Skaloud. “Assessment of the Integration Strategy between GPS and Body-Worn MEMS Sensors with Application to Sports”. en. In: (), p. 10.

[44] Zhaozhong Chen et al. “Weak in the NEES?: Auto-tuning Kalman filters with Bayesian optimization”. In: *2018 21st International Conference on Information Fusion (FUSION)*. IEEE. 2018, pp. 1072–1079.