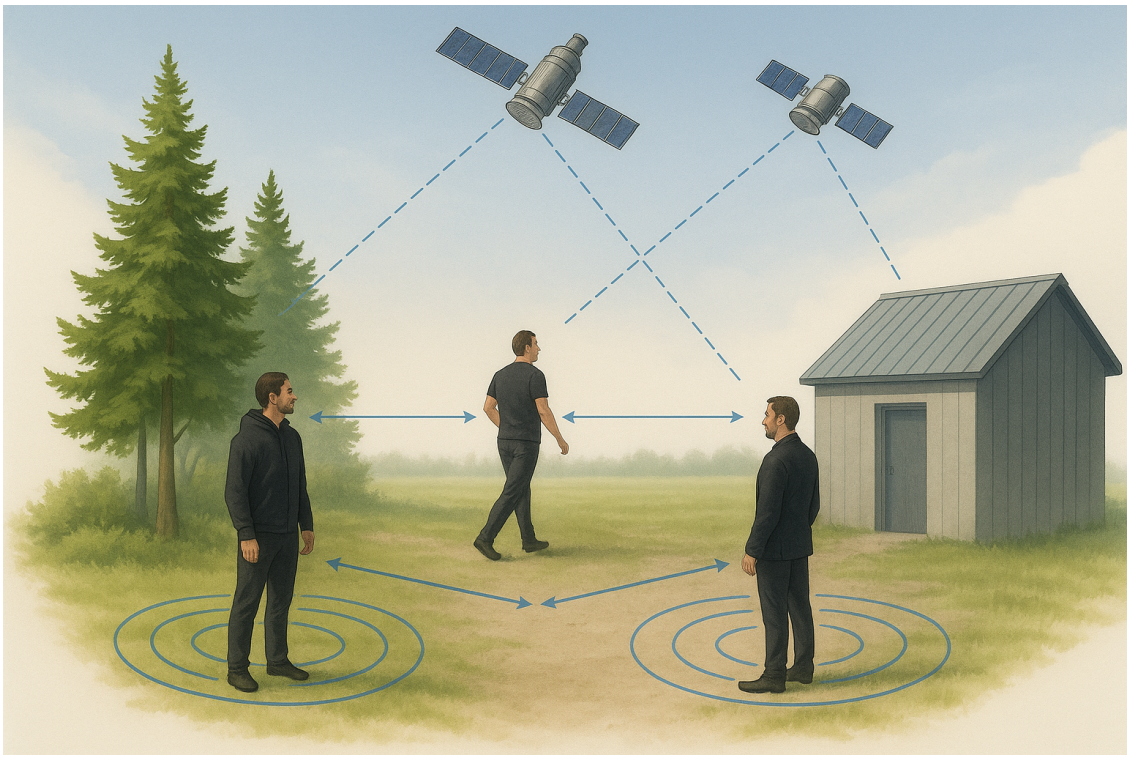

Better Player Position in Rough/Urban Terrain

How sensor fusion of GNSS, UWB, IMU could improve the positioning in urban or rough terrain.

Master's thesis in Systems, Control & Mechatronics

GUSTAV MALMSTRÖM

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2025
www.chalmers.se

Master's thesis 2025

Better Player Position in Rough or Urban Terrain

How sensor fusion of GNSS, UWB, IMU could improve the positioning in urban/rough terrain.

GUSTAV MALMSTRÖM

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Systems and Control Division
Chalmers University of Technology
Gothenburg, Sweden 2025

Better Player Position in Rough or Urban Terrain
How sensor fusion of GNSS, UWB, IMU could improve the positioning in urban/rough terrain.
GUSTAV MALMSTRÖM

Supervisors:
Oscar Gunnarsson, Saab Training & Simulation
Jayadev Naram, Department of Electrical Engineering

Examiner: Henk Wymeersch, Department of Electrical Engineering

Master's Thesis 2025
Department of Electrical Engineering
Systems and Control Division
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Saab AB
Business Area Dynamics
Training & Simulation
SE-561 85 Huskvarna
Sweden
Telephone: +46-(0)36 388 000

Cover: Concept image of multi-sensor positioning using GNSS, IMU, and UWB in varied outdoor environments.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2025

Better Player Position in Rough or Urban Terrain
How sensor fusion of GNSS, UWB, IMU could improve the positioning in urban/rough terrain.
GUSTAV MALMSTRÖM
Department of Electrical Engineering
Systems and Control Division
Chalmers University of Technology

Abstract

This thesis investigates the development of a scalable and robust positioning system designed for use in training environments such as forests and urban terrain, where traditional Global Navigation Satellite Systems (GNSS) often suffer from limited reliability due to signal blockage and multipath interference. To overcome these limitations, a system was designed combining GNSS, Inertial Measurement Units (IMUs), and Ultra-Wideband (UWB) technologies using advanced sensor fusion algorithms. A prototype was implemented and tested in multiple environments using stationary and dynamic test scenarios. The results show that while each sensor has specific strengths and limitations, sensor fusion significantly enhances positioning accuracy and stability, particularly in environments where GNSS data alone proves unreliable. The system was evaluated using real-world data, and its performance was benchmarked against high-accuracy reference systems such as Real-Time Kinematic (RTK). The study concludes that sensor fusion provides a cost-effective and scalable way to improve positional awareness in demanding environments.

GNSS, IMU, UWB, Sensor Fusion, Kalman Filter, Urban Positioning, RTK, Dead Reckoning, Localization, Multisensor Integration

Acknowledgements

First, I would like to express my sincere gratitude to the System Engineering team at Saab for providing the resources, technical feedback, and practical advice that made this project possible. I would especially like to thank my supervisor Oscar Gunnarsson for his continued support, insightful guidance, and constructive discussions throughout all stages of the thesis work.

I would also like to thank my examiner, Professor Henk Wymeersch, for giving me the opportunity to carry out this project and for his valuable feedback and guidance on maintaining the scientific focus of the work. I am equally grateful to my academic supervisor, Jayadev Naram, for his support, constructive input, and advice throughout the thesis.

Finally, I want to thank my family and friends for their encouragement, patience, and support during this entire process.

Gustav Malmström, Gothenburg, June 2025

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

API	Application Programming Interface
CKF	Cubature Kalman Filter
DGNSS	Differential GNSS
EKF	Extended Kalman Filter
GNSS	Global Navigation Satellite System
IMU	Inertial Measurement Unit
I^2C	Inter-Integrated Circuit (serial communication protocol)
MEMS	Micro-Electro-Mechanical Systems
NRTK	Network Real-Time Kinematic
PPP	Precise Point Positioning
RTK	Real-Time Kinematic
ToF	Time of Flight
TWR	Two-Way Ranging
UKF	Unscented Kalman Filter
UWB	Ultra-Wideband

Nomenclature

Below is the nomenclature of indices, sets, parameters, and variables that have been used throughout this thesis.

Indices

p	Index for satellite
k	Index for receiver
i	Index for sigma point or particle
t	Index for time step

Parameters

c	Speed of light
Q	Process noise covariance matrix
R	Measurement noise covariance matrix
S	Innovation (residual) covariance matrix
Σ_{anchor}	Anchor position covariance matrix
Σ_{meas}	Measurement covariance matrix (range/angle)
Σ	Generic covariance matrix
W_i^m	Weight for the i-th sigma point in mean calculation
W_i^c	Weight for the i-th sigma point in covariance calculation
Δt	Time step between filter updates

Variables

$\mathbf{x}$	State vector $[e, n, v, a, \varphi]^T$
e	Position in Easting direction

n	Position in Northing direction
v	Scalar velocity
a	Scalar linear acceleration
φ	Heading angle (yaw)
P	State covariance matrix
z	Measurement vector
y	Measurement residual (innovation)
H	Observation matrix or measurement Jacobian
K	Kalman gain
χ	Sigma points
$\hat{x}$	Predicted state estimate
$\hat{y}$	Predicted measurement
ρ_k^p	Geometric range between satellite p and receiver k
Δt_k, Δt^p	Clock bias of receiver and satellite
τ_k, τ^p	True reception and transmission times
T_k^p	Tropospheric delay
I_k^p	Ionospheric delay
P_k^p	Pseudorange measurement
L_k^p	Carrier-phase measurement
B_k^p	Carrier-phase integer ambiguity

Contents

List of Figures

1
Introduction

When the urbanization continues to expand, the cities are becoming increasingly dense and natural landscapes are transforming into urban environments. This evolution has led to a growing demand for autonomous systems with robust and precise positioning capabilities. Using traditional Global Navigation Satellite Systems (GNSS) in such environments leads to significant challenges due to high elevation angles, multipath interference, and inherently weak signal strength [1]. These limitations result in unreliable positioning performance, making GNSS susceptible to errors such as signal degradation, multipath distortion and even spoofing attacks.
Without advanced error mitigation algorithms or complementary sensor systems the GNSS accuracy typically ranges from 2 to 10 meters [2]. This level of precision is not good enough for applications requiring highly accurate positioning like autonomous systems, military training simulations and human tracking in complex terrains. With this in mind there is a need for innovative positioning solutions that integrate multiple sensor technologies to enhance accuracy and reliability in challenging environments such as dense urban areas and forests.

To overcome these limitations there are advanced techniques such as Real-Time Kinematic (RTK) positioning that have been developed which is offering centimeter-level accuracy [3]. However, RTK has significant drawbacks which includes high infrastructure costs and susceptibility to signal propagation issues near large buildings or dense foliage. These limitations make RTK impractical for large-scale systems such as training environments with hundreds of soldiers and multiple vehicles, where scalability and robustness are crucial.
There is a growing need to enhance existing positioning solutions to meet increasing demands for scalability, adaptability, and precision. Integrating multiple sensing technologies through sensor fusion, combining GNSS, Inertial Measurement Units (IMUs), and Ultra-Wideband (UWB) technologies, offers a promising path to improve accuracy, reliability, and real-time positioning performance in challenging environments. This thesis investigates the feasibility of such a system through a concept study in collaboration with Saab AB.

Saab AB Training and Simulation specializes in developing advanced training solutions across air, land, naval, and civil security sectors. These solutions are designed to provide military personnel with realistic training platforms that replicate the complexities of real-world combat scenarios. By simulating these situations, soldiers can train under conditions that closely mirror actual operations. This enables them to build experience, improve decision-making skills and respond instinctively under

pressure. Saab's goal is to optimize training efficiency by minimizing the time and resources required to achieve operational readiness. To maintain their leadership in this field, Saab AB Training & Simulation continuously explores innovative technologies that are more compact, precise and cost-effective, ensuring their training systems remain at the forefront of modern defence capabilities.

1.1 Purpose

The purpose of this thesis is to investigate, through a concept study in collaboration with Saab, whether it is possible to create a robust, scalable, and cost-effective positioning system using advanced algorithms and a limited set of sensors. The focus is on enhancing the accuracy of a standard GNSS receiver by leveraging sensor fusion techniques and integrating it with Inertial Measurement Units (IMUs) and Ultra-Wideband (UWB) technology. This system aims to address improved accuracy and reliability in challenging environments such as dense urban areas and forests.

The project emphasizes improving positioning accuracy through advanced algorithmic approaches using sensor fusion frameworks. By combining GNSS with IMUs and UWB, this study explores how these technologies complement each other to achieve more reliable positioning. Special attention is given to challenges such as multipath interference, signal attenuation, and dynamic environmental conditions that affect accuracy.

The ultimate goal is to deliver insights into the design and implementation of a functional prototype capable of addressing the unique demands of urban and forested terrains. This research will contribute to the development of advanced training systems for Saab, ensuring the scalability, cost-effectiveness, and operational efficiency of positioning technologies in realistic combat training environments.

1.2 Scope

The scope of this thesis encompasses the development and evaluation of a robust positioning system tailored for challenging environments, with a specific focus on training applications in urban and forested terrains. The project aims to enhance positioning accuracy and reliability by integrating GNSS, IMUs, and UWB technologies within a cohesive sensor fusion framework. The study will prioritize scalability and cost-effectiveness while addressing critical limitations in traditional positioning systems. Some goals and boundaries of the project are outlined below:

- **Focus on Training Environments:** The primary application of the developed system is for training scenarios, excluding other uses such as commercial or recreational positioning systems.
- **Technology Integration:** The project will exclusively consider GNSS, IMU, and UWB technologies, excluding other sensors or techniques that may further improve accuracy but add complexity or cost.
- **Testing Conditions:** Testing and validation will be conducted in urban and

forested terrains to reflect real-world challenges, while other environments like open sea or extreme altitudes are beyond the scope of this study.
- **Algorithm Development:** The scope includes the design and implementation of sensor fusion algorithms without delving into the hardware-level development of sensors or GNSS receivers.
- **Scalability Evaluation:** While scalability and system performance will be assessed, the study will not provide an exhaustive analysis of cost optimization for large-scale deployments.
- **Time-Constrained Development:** The project was carried out within a limited timeframe of approximately 20 weeks, which influenced the number of test cases, depth of analysis, and scope of system integration and optimization.

This focused scope ensures the project remains achievable within the given time constraints while delivering meaningful insights into the potential of integrating GNSS, IMUs, and UWB technologies for advanced positioning systems in complex terrains.

1.3 Related Work

In recent years, the integration of cooperative sensing using multiple sensors, such as GNSS, UWB, and IMU, through fusion frameworks has gained significant attention as a method to improve positioning accuracy and robustness, especially in environments where GNSS performance is limited.

Kailun Song et al. [4] propose a tightly coupled indoor-outdoor positioning system that integrates GNSS, UWB, and IMU sensors. Their system uses a multi-anchor UWB setup for high-accuracy indoor positioning, while the IMU helps compensate for transient UWB outages. In transition zones between indoor and outdoor environments, an adaptive weighting strategy is employed within an Extended Kalman Filter (EKF) to dynamically adjust the influence of each sensor. Although the results show strong improvements in accuracy and continuity, the system relies on fixed infrastructure, which may not be practical for large-scale outdoor deployments involving mobile units.

Retscher et al. [5] investigate a GNSS/UWB fusion approach based on clustering and weighted least squares (WLS) estimation. Their work focuses on GNSS-challenged environments, where UWB range observations are combined with GNSS measurements to improve the overall positioning solution. A key contribution is the development of a clustering method to identify measurement locations without requiring manual input, followed by three WLS models for comparison: UWB only, GNSS only, and combined GNSS/UWB. Their results demonstrate that sensor fusion significantly improves positioning accuracy within the anchor area, although accuracy declines for points outside that area due to poor anchor geometry.

These studies provide important insights into multi-sensor positioning system design. In contrast to fixed-anchor or infrastructure-heavy setups, the current thesis focuses on a scalable, low-cost, and mobile solution suitable for outdoor training

environments. The system is designed to operate on human-mounted platforms, in challenging terrains, without relying on pre-installed infrastructure.

2
Theory

2.1 Positioning Systems

The collective name Global Navigation Satellite Systems (GNSS) is a satellite-based navigation system used worldwide for both real-time and stationary applications [1]. For example, in transportation, which is a real time usage of GNSS, the positioning system enables navigation for cars, rail and air travel with supporting route planning, real-time tracking and precision landings particularly in low-visibility conditions. Stationary GNSS applications demand high-precision accuracy, particularly in disciplines such as surveying, construction, geology, cartography, and archaeology, where millimeter-level precision is often required. Moreover, GNSS plays a critical role in monitoring tectonic plate movements, evaluating environmental impacts, and supporting a wide range of scientific research. These capabilities underscore the system's versatility and importance beyond conventional navigation purposes.

2.1.1 GNSS

Global Navigation Satellite Systems (GNSS) are space-based systems that provide positioning, navigation, and timing information anywhere on Earth. The GNSS include satellite constellations like the United States' GPS, Russia's GLONASS, the European Union's Galileo, and China's BeiDou. These systems operate using a constellation of satellites that typically ranges between 24 and 30, ensuring that a receiver can access signals from at least four satellites at any time for accurate position calculation [6].

Each GNSS satellite transmits signals on multiple frequencies. The GPS system, for example, uses the L1 (1575.42 MHz) and L2 (1227.60 MHz) bands, both in the L-band of the Ultra high frequency (UHF) spectrum. Newer satellites also transmit on the L5 band (1176.45 MHz), allowing for improved accuracy through multi-frequency observations [6, 7]. These frequencies can help correct for atmospheric errors, particularly in the ionosphere, by enabling dual-frequency techniques.

2.1.2 GNSS Error Sources

Although GNSS is a globally available and reliable positioning technology, its performance is affected by several sources of error. These include inaccuracies related to satellite systems, atmospheric interference, and local environmental conditions near

the receiver. Understanding these error sources is essential for improving positioning accuracy, particularly in environments where high precision is required.

Satellite-related errors comes primarily from imperfections in the satellite clocks and deviations in their orbital paths. Even though the atomic clocks onboard GNSS satellites are highly precise, minor timing errors can still introduce noticeable positioning inaccuracies. Similarly, different types of discrepancies in satellite orbit predictions, often caused by gravitational anomalies or solar radiation pressure, may result in the broadcast ephemeris deviating from the satellite's true trajectory [8]. These errors can be mitigated through relative positioning methods or by using refined orbit and clock data in post-processing.

Atmospheric disturbances also play a major role in GNSS signal degradation. The ionosphere, located in the upper atmosphere, contains charged particles that delay signal propagation. This delay is frequency-dependent, which allows for compensation using dual-frequency measurements, such as combining L1 and L2 signals into an ionosphere-free observation [8]. Ionospheric effects are especially significant in polar and equatorial regions, where solar activity causes increased variability.

The atmosphere layer below the ionosphere is the troposphere, which introduces further delay due to changes in humidity, temperature, and pressure. Unlike the ionosphere, the troposphere is non-dispersive that means it the effects cannot be removed by dual-frequency techniques. Instead these delays must be solved in other ways like modelling them, using functions that account for satellite elevation angles and regional meteorological data [8].

Multipath errors occur when GNSS signals reflect off nearby surfaces before reaching the receiver. These reflected signals interfere with the direct line-of-sight signal, distorting measurements and leading to errors in the calculated position. Multipath is especially common in urban environments or near dense vegetation. While modern GNSS receivers often include algorithms to reduce multipath interference, it remains a significant challenge in real-world conditions [8].

Receiver hardware and internal processing also contribute to the total error. Lower-quality GNSS receivers are more susceptible to noise, suffer from poorer multipath rejection, and may introduce delays in signal timing. Hardware differences between signal paths, such as between L1 and L2 frequencies, can result in additional systematic biases unless properly calibrated [8, 10].

In addition, relativistic effects—arising from the combination of satellite velocity and gravitational field differences between the satellite and receiver—can slightly shift the timing of GNSS signals. Although these time shifts are typically on the order of nanoseconds, they translate into position errors of several meters if left uncorrected. Fortunately, modern GNSS systems include relativistic corrections in their satellite signal models [6].

In total these error sources impact the accuracy and stability of GNSS-based positioning. For applications that demand high precision—such as autonomous naviga-

tion, surveying, or military training systems—these limitations highlight the need for advanced error modelling or the integration of complementary technologies such as DGNSS, RTK, or PPP.

2.1.3 GNSS Algorithm

Positioning is achieved by calculating the distance between the satellite and the receiver, referred to as the pseudorange. This is determined by measuring the signal travel time from the satellite to the receiver and multiplying it by the speed of light:

$$P_k^p = (t_k - t^p) \cdot c \tag{2.1}$$

Here, P_k^p is the pseudorange measurement between satellite p and receiver k, t_k is the time at which the signal is received (receiver clock time), t^p is the time at which the signal was transmitted by the satellite, c is the speed of light.

Since both the satellite and receiver clocks are not perfectly synchronized, the actual transmission and reception times include clock biases. The measured times can be expressed in terms of the true times and clock errors as:

$$t_k = \tau_k + \Delta t_k, \quad t^p = \tau^p + \Delta t^p \tag{2.2}$$

where τ_k and τ^p are the true signal reception and transmission times, respectively, Δt_k and Δt^p are the receiver and satellite clock biases.

These biases must be estimated and corrected in the positioning algorithm. The pseudorange equation, once corrected for these biases, serves as the foundational measurement model in GNSS-based positioning systems.

This raw distance also includes several error sources, such as satellite and receiver clock offsets, ionospheric delay, and tropospheric delay, and is commonly modeled as:

$$P_k^p = \rho_k^p + (\Delta t_k - \Delta t^p) \cdot c + I_k^p + T_k^p \tag{2.3}$$

where ρ_k^p is the true geometric range between satellite p and receiver k, I_k^p is the ionospheric delay, T_k^p is the tropospheric delay [10].

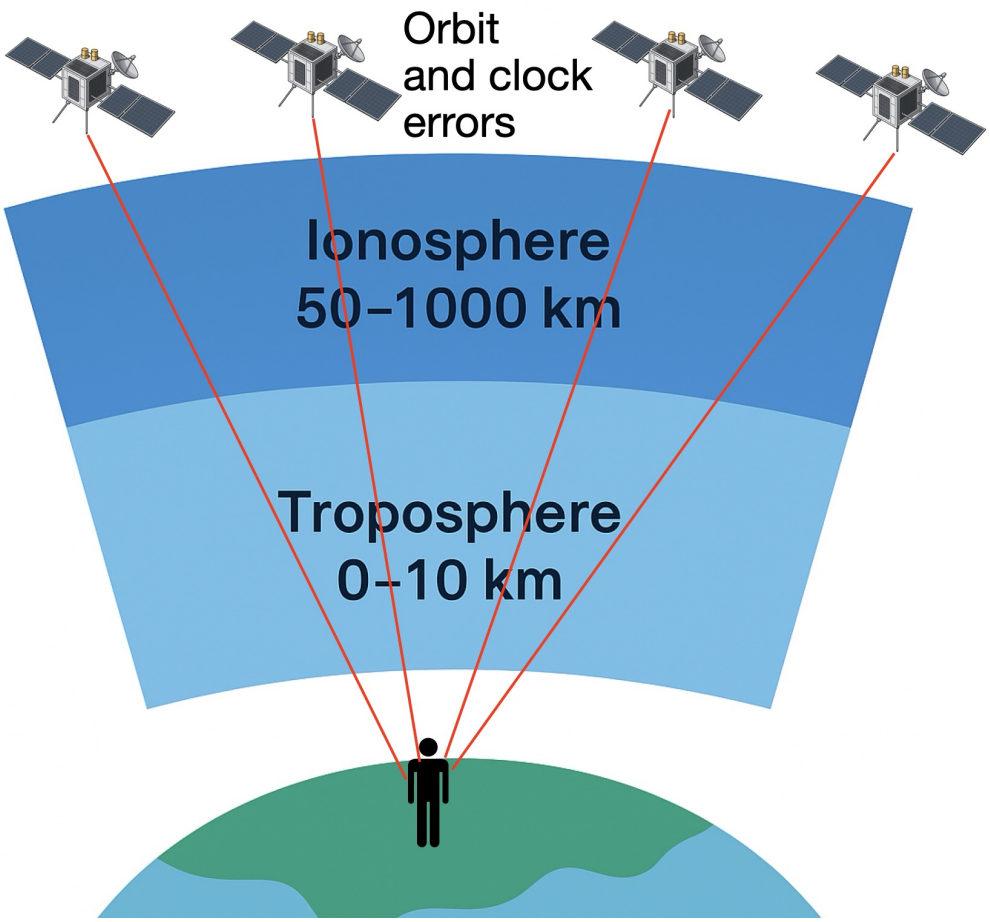

Figure 2.1: Illustration of GNSS signal propagation through the troposphere and ionosphere, showing common error sources such as orbit, clock, and atmospheric delays.

For higher positioning precision, GNSS can use carrier-phase measurements, which track the phase of the satellite's carrier signal rather than the modulated code. These measurements are much more precise, on the order of millimeters, but include an integer ambiguity, B_k^p, representing the unknown number of full signal wavelengths between the satellite p and receiver k.
The carrier-phase measurement is typically modeled as:

$$L_k^p = \rho_k^p + (\Delta t_k - \Delta t^p) \cdot c - I_k^p + T_k^p + B_k^p \tag{2.4}$$

where L_k^p is the measured carrier phase converted to range (in meters), ρ_k^p is the geometric range between satellite and receiver, B_k^p is the integer ambiguity term (in meters).
The sign of the ionospheric delay $-I_k^p$ differs from that in the pseudorange equation because the carrier phase advances as it passes through the ionosphere, whereas the code signal is delayed. This distinction is important when combining code and carrier measurements for precise positioning.
When the integer ambiguity B_k^p is correctly resolved, carrier-phase techniques can achieve centimeter- or even millimeter-level positioning accuracy [9].

The geometric distance between a GNSS satellite p and the receiver at time t is given by:

$$\rho_t^p = \sqrt{(x_t^p - x_t^k)^2 + (y_t^p - y_t^k)^2 + (z_t^p - z_t^k)^2} \tag{2.5}$$

where (x_t^p, y_t^p, z_t^p) is the satellite position and (x_t^k, y_t^k, z_t^k) is the receiver position, all expressed in an Earth-Centered Earth-Fixed (ECEF) frame. To solve for the unknown receiver coordinates, a Taylor series expansion or linearization of the pseudorange equations can be applied. This allows the construction of a system of linear equations, which can be solved iteratively to estimate the receiver position.

The GNSS module provides a position estimate derived from such geometric relationships. This estimate is used directly as a measurement input in the sensor fusion algorithm and is modeled as a noisy observation of the true receiver position:

$$\mathbf{z}_{\text{GNSS},t} = \mathbf{x}_t + \mathbf{v}_{\text{GNSS},t}, \quad \mathbf{v}_{\text{GNSS},t} \sim \mathcal{N}(0, R_{\text{GNSS}}) \tag{2.6}$$

where $\mathbf{z}_{\text{GNSS},t}$ is the measured position at time t, $\mathbf{x}_t$ is the true position, and $\mathbf{v}_{\text{GNSS},t}$ is zero-mean Gaussian noise with covariance R_{GNSS}.

2.1.4 DGNSS, RTK, and PPP

To improve the positioning accuracy of GNSS, several augmentation techniques have been developed. Among the most widely used are Differential GNSS (DGNSS), Real-Time Kinematic (RTK), and Precise Point Positioning (PPP). These methods apply corrections to reduce the effects of GNSS error sources such as satellite clock drift, orbit errors, and atmospheric delays.

DGNSS is a relative positioning technique that uses the code measurements from the satellites and a stationary reference receiver with a known position [11]. This base station continuously calculates the difference between its actual position and the one calculated via GNSS. The resulting corrections are transmitted to a rover receiver in real time. Since both the base and rover receivers are affected similarly by satellite and atmospheric errors (as long as they are within about 10–30 km of each other), DGNSS allows the rover to correct its own positioning solution based on the known discrepancies observed by the base station.

Real-Time Kinematic (RTK) builds upon DGNSS by incorporating not only pseudorange corrections but also carrier-phase measurements [12]. This allows RTK to achieve centimeter-level accuracy in real time. RTK requires a data link between the base and rover, and to reach this level of precision, the system must correctly resolve the integer ambiguity. While RTK offers superior precision, it is sensitive to signal disruptions and requires a stable communication link and infrastructure, making it less suitable for environments with frequent signal blockages such as urban canyons or forests.

Another way to get rid of most of the error sources is to use Precise Point Positioning (PPP) that is an absolute positioning method that does not rely on a local base

station [13]. Instead, PPP uses precise satellite clock and orbit data provided by external services, often from global GNSS reference networks. PPP also incorporates carrier-phase data, and through sophisticated modelling of atmospheric delays and other errors, it can achieve decimeter to centimeter accuracy. On the other hand PPP typically has a long convergence time—often several minutes—before it reaches high precision, which limits its usefulness in highly dynamic or short-duration applications [10].

Each of these correction methods presents trade-offs between accuracy, infrastructure requirements, cost, and real-time capability. DGNSS provides meter-level corrections and is relatively easy to implement, making it suitable for many applications. RTK offers real-time centimeter-level accuracy but demands local infrastructure and stable communication links. PPP offers global coverage and high precision without a local base station but suffers from long initialization times and is best suited for static or slow-moving use cases.

2.2 Inertial Measurement Units (IMUs)

An Inertial Measurement Unit (IMU) is an electronic device used to measure motion and orientation. While IMUs can be built using various technologies, most modern, low-cost IMUs are based on MEMS (Micro-Electro-Mechanical Systems) sensors. These typically consist of a tri-axial accelerometer, a tri-axial gyroscope, and sometimes a tri-axial magnetometer. These sensors individually measure a body's specific acceleration, angular rate, and magnetic field. Together, they provide the data needed to estimate the object's orientation, velocity, and movement in space [14].

IMUs are essential in applications where external positioning signals like GNSS may be unavailable or degraded. By integrating accelerometer and gyroscope data over time, IMUs enable a process known as dead reckoning — estimating the position of an object relative to a known starting point. This makes them particularly useful in GNSS-denied environments such as tunnels, urban canyons, dense forests, or indoors. With recent research, IMUs have been shown to play a critical role in maintaining accurate vehicle positioning in such environments, particularly when combined with lane detection and additional sensors [15]. They also enable navigation systems to continue operating indoors or in areas where satellite signals are blocked, highlighting their importance for continuous positioning even in GNSS-denied environments [16].

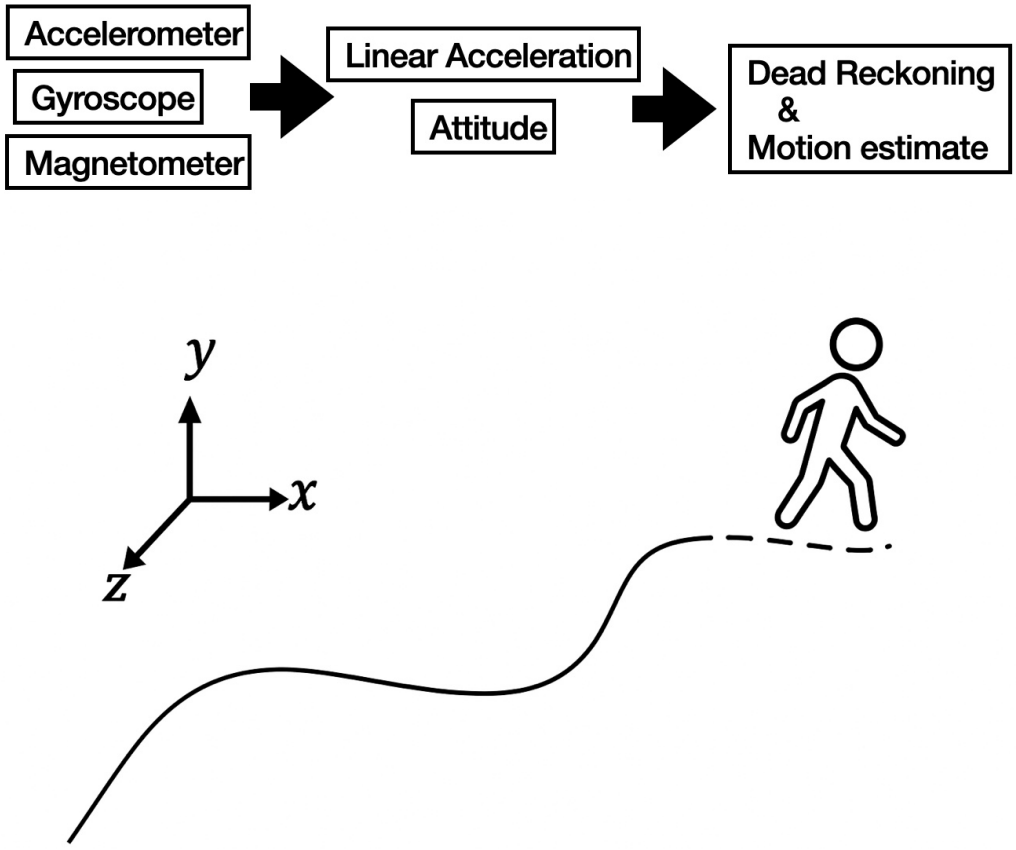

Figure 2.2: Overview of IMU sensor outputs and their use in dead reckoning. The accelerometer, gyroscope, and magnetometer contribute to linear acceleration and attitude estimation, which are integrated to estimate motion.

The nature of inertial integration does however cause some problems, even small errors in measurement can lead to significant drift over time. These errors arise from various sources:

- **Bias instability:** A slow, time-dependent offset in sensor readings.
- **Noise:** Fluctuations in signal that introduce uncertainty.
- **Scale factor error:** Imperfect calibration that distorts the magnitude of readings.
- **Temperature sensitivity:** Performance degradation under changing environmental conditions.

To mitigate these limitations, IMUs are often fused with absolute positioning systems like GNSS or UWB through algorithms such as the Kalman filter. In such systems, the IMU contributes high-rate motion updates and short-term continuity, while GNSS or UWB corrects for long-term drift.

IMUs are widely used in various sectors, including aerospace, automotive, robotics, defense, and mobile electronics. High-performance MEMS-based IMUs, such as those developed by Analog Devices [14], offer low bias instability, low noise, and high bandwidth, making them suitable for precision applications like inertial navigation systems, stabilization platforms, and autonomous systems.

The measurements provided by the IMU sensors, accelerometer, gyroscope, and magnetometer, can be described by the following models, adapted from [17]:

$$\mathbf{a}_{\text{meas},t} = \mathbf{a}_t + \mathbf{b}_a + \mathbf{v}_{a,t}, \quad \mathbf{v}_{a,t} \sim \mathcal{N}(0, R_a) \tag{2.7}$$

$$\boldsymbol{\omega}_{\text{meas},t} = \boldsymbol{\omega}_t + \mathbf{b}_\omega + \mathbf{v}_{\omega,t}, \quad \mathbf{v}_{\omega,t} \sim \mathcal{N}(0, R_\omega) \tag{2.8}$$

$$\mathbf{m}_{\text{meas},t} = \mathbf{m}_t + \mathbf{b}_m + \mathbf{v}_{m,t}, \quad \mathbf{v}_{m,t} \sim \mathcal{N}(0, R_m) \tag{2.9}$$

where:

- $\mathbf{a}_{\text{meas},t}$, $\boldsymbol{\omega}_{\text{meas},t}$, and $\mathbf{m}_{\text{meas},t}$ are the measurements from the accelerometer, gyroscope, and magnetometer respectively,
- $\mathbf{a}_t$, $\boldsymbol{\omega}_t$, and $\mathbf{m}_t$ are the true linear acceleration, angular velocity, and magnetic field,
- $\mathbf{b}_a$, $\mathbf{b}_\omega$, and $\mathbf{b}_m$ are the sensor biases,
- $\mathbf{v}_{a,t}$, $\mathbf{v}_{\omega,t}$, and $\mathbf{v}_{m,t}$ are zero-mean Gaussian noise terms with covariances R_a, R_ω, and R_m respectively.

2.3 Ultra-Wideband (UWB) Positioning

Ultra-Wideband (UWB) is a short-range radio communication technology that operates over a wide frequency spectrum that is often ranging from 3.1 GHz to 10.6 GHz [20]. Compared to narrowband radio systems like Wi-Fi or Bluetooth the UWB technology transmits very short-duration pulses, which enables precise measurement of the signal's time of flight (ToF). This makes UWB highly suitable for high-accuracy positioning applications, especially in GNSS-degraded environments such as indoors, dense forests, or urban areas.

In positioning systems, Ultra-Wideband (UWB) is commonly used to determine distances between mobile units, called tags, and fixed infrastructure devices, called anchors, by measuring the time characteristics of the signal's propagation. UWB supports multiple techniques for location detection, including Time of Flight (ToF), Time Difference of Arrival (TDoA), and Two-Way Ranging (TWR) [19]. Among these, TWR is widely used in real-time systems, where a signal is sent from the tag to the anchor and back, allowing the round-trip time to be measured. This enables accurate distance estimation with centimeter-level precision [18]. Positioning is typically achieved through trilateration by combining range measurements from at least three anchors for 2D positioning, or four for full 3D localization.

UWB offers several advantages compared to other wireless positioning technologies:

- **High accuracy:** UWB can achieve real-time position estimates with an error margin as low as 10–30 cm under optimal conditions.
- **Robustness to multipath:** The use of ultra-short pulses makes UWB resistant to multipath interference, which is a common problem in urban or indoor environments.

- **Low power consumption:** UWB devices consume less energy, making them suitable for mobile and wearable applications.
- **High update rate:** Systems such as Pozyx's RTLS support update rates of up to 5000 positions per second, enabling real-time tracking [18].

The UWB has a lot of benefits but also has certain limitations. The technology typically requires line-of-sight between devices to achieve peak accuracy, and its effective range is relatively short, usually up to 100 meters. UWB systems also depends on dedicated infrastructure, such as well-positioned and synchronized anchors, which can limit their scalability in large or open environments [20].

Beyond its role in high-precision positioning, Ultra-Wideband (UWB) has emerged as a versatile technology in a wide range of modern applications, including consumer electronics, smart infrastructure, and the automotive industry. Its ability to measure distances with high temporal resolution makes it suitable not only for industrial positioning but also for secure access control, device-to-device communication, and context-aware interactions. In mobile devices, UWB enables features such as digital keys for vehicles and buildings, secure proximity-based payments, and location services for finding lost items or navigating crowded environments. In automotive systems, UWB supports functionalities like passive entry and start, valet parking automation, vehicle-to-infrastructure communication, and even collision avoidance for pedestrians and cyclists. Additionally, UWB is gaining traction in smart cities, where it enables indoor navigation, tolling systems, and safety monitoring in densely populated or obstructed environments [20, 18]. These evolving use cases underline UWB's importance not only as a high-accuracy positioning technology but also as a secure, low-power, and future-oriented solution for real-time spatial awareness.

Ultra-Wideband (UWB) systems estimate distance between a mobile tag and fixed anchors using a Two-Way Ranging (TWR) protocol. In this method, the tag sends a Poll message to the anchor, which replies after a short processing delay. The total loop time T_{loop} and the known reply delay T_{reply} are used to calculate the one-way Time of Flight (ToF) of the signal:

$$\text{ToF} = \frac{T_{\text{loop}} - T_{\text{reply}}}{2} \tag{2.10}$$

Assuming the signal propagates at speed c, the distance between the tag and the anchor is given by:

$$d = \text{ToF} \cdot c \tag{2.11}$$

A visual representation of this process is shown in Figure 2.3. A detailed explanation can be found in [21].

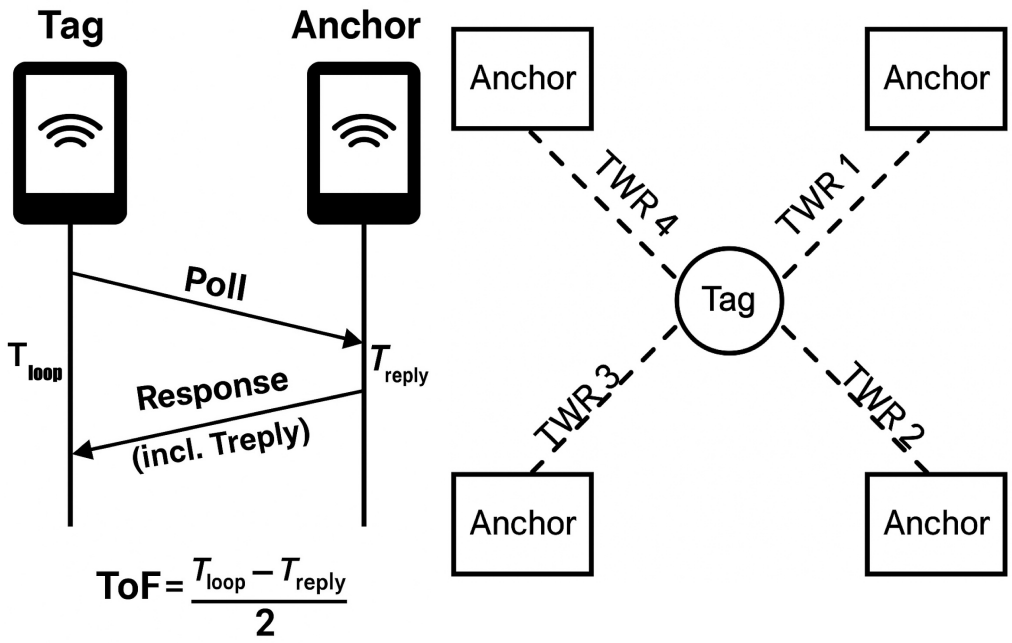

Figure 2.3: Two-Way Ranging (TWR) in UWB systems. Left: Time of flight (ToF) calculation using T_{loop} and T_{reply}. Right: Tag measuring distances to multiple anchors via TWR.

Once the distance is estimated, the measurement from anchor i to the tag at time t is modeled as:

$$z_t^i = \left\| \mathbf{x}_t - \mathbf{p}^i \right\| + v_t^i, \quad v_t^i \sim \mathcal{N}(0, \sigma_{\text{UWB}}^2) \tag{2.12}$$

where:

- z_t^i is the measured range,
- $\mathbf{x}_t$ is the position of the tag at time t,
- $\mathbf{p}^i$ is the known position of anchor i,
- v_t^i is zero-mean Gaussian noise with variance σ_{UWB}^2.

In multi-sensor positioning systems, UWB can be integrated with GNSS and IMU data through sensor fusion frameworks such as Kalman filtering. While GNSS provides global coverage and IMUs track motion continuously, UWB adds high-precision short-range updates. This combination is particularly effective in GNSS-challenged environments, where UWB can help bridge positioning gaps and correct for IMU drift.

2.4 Sensor fusion

2.4.1 Introduction: The Power of Sensor Fusion

Sensor fusion is a powerful technique that combines data from multiple sensors using algorithms to estimate a more accurate or reliable value of a parameter or system state. This is especially useful when the sensors involved have different

characteristics and can complement each other. For example, one sensor might have high precision but low frequency, while another might provide frequent but noisy measurements. By combining them, the strengths of each sensor can be utilized while minimizing their weaknesses [22, 23].

The idea behind sensor fusion is based on statistical reasoning and is commonly used in systems that require robust and continuous estimations, such as in navigation, robotics, and autonomous vehicles. It allows for greater fault tolerance, improved accuracy, and more stable estimates even when individual sensors fail or produce uncertain data.

2.4.2 Theoretical Foundation: Bayesian Inference

Sensor fusion algorithms are fundamentally based on statistical reasoning, and many of them rely on Bayesian inference. The purpose of Bayesian inference is to provide a systematic mathematical framework for handling uncertainty and updating knowledge about a system as new information becomes available. This is done using probability distributions and the rules of probability calculus [22].

Let x be a random variable with a probability distribution $p(x)$, which may for example be a Gaussian distribution with mean μ and variance σ^2. This distribution is referred to as the *prior*, and it represents our initial beliefs about the state x before any measurements are made.

The measurement process is modeled by a likelihood function $p(y|x)$, which describes the probability of observing a measurement y given the state x. This is known as the *measurement model*.

Using Bayes' rule, the prior and the measurement model are combined to compute the *posterior distribution*:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}, \tag{2.13}$$

where $p(y)$ is a normalizing constant ensuring that the posterior is a valid probability distribution. The posterior $p(x|y)$ contains all the updated information about the state x after incorporating the measurement y.

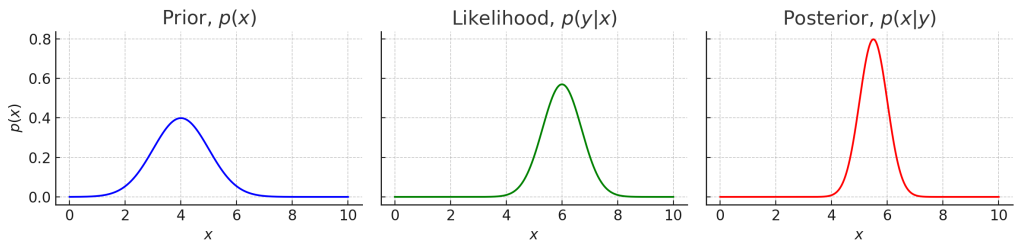

Figure 2.4: Bayesian Inference with prior, likelihood and posterior

In Bayesian inference, the parameters or states are treated as random variables. This contrasts with classical (frequentist) statistics, where parameters are assumed to be fixed and only data is considered random. Bayesian methods therefore allow for natural incorporation of uncertainty and prior knowledge in estimation problems.

When applying Bayesian estimation in practice, one often needs to select a point estimate from the posterior distribution. Common choices include the *maximum a posteriori (MAP)* estimate, which selects the mode of the posterior, and the *minimum mean squared error (MMSE)* estimate, which uses the posterior mean. The choice of point estimate depends on the assumed loss function, such as quadratic loss or absolute error loss [22].

2.4.3 Overview of Sensor Fusion Algorithms

As a practical application of Bayesian estimation, the Kalman Filter (KF) is one of the most widely used algorithms for sensor fusion in linear systems. It is a well suited algorithm for dynamic systems where the true state is uncertain, and the measurements are a bit noisy. The filter provides a recursive solution to the discrete-data linear filtering problem, producing optimal state estimates under the assumption of Gaussian noise and linear dynamics [22].

The Kalman filter consists of two main steps: the prediction step and the update step [24]. These steps are executed sequentially at each time step to estimate the current state of the system. Depending on the frequency of incoming measurements, the prediction step may be performed multiple times before a new update step is applied.

Prediction Step: In this step, the filter predicts the next state of the system using the state-space model, which typically incorporates the previous state and any control input:

$$\mathbf{x}_t = \mathbf{A}\mathbf{x}_{t-1} + \mathbf{B}\mathbf{u}_{t-1} \tag{2.14}$$

where $\mathbf{x}_t$ is the predicted state, $\mathbf{A}$ is the state transition matrix, $\mathbf{B}$ is the control-input matrix, and $\mathbf{u}_{t-1}$ is the known control vector at the previous time step.
The uncertainty in this prediction is modeled with a covariance matrix:

$$\mathbf{P}_t = \mathbf{A}\mathbf{P}_{t-1}\mathbf{A}^\top + \mathbf{Q} \tag{2.15}$$

where $\mathbf{P}_t$ is the predicted covariance matrix and $\mathbf{Q}$ is the process noise covariance matrix.

Update Step: The update step refines the predicted state using incoming measurement data. The measurement residual is calculated as:

$$\mathbf{y}_t = \mathbf{z}_t - \mathbf{H}_t\mathbf{x}_t \tag{2.16}$$

where $\mathbf{z}_t$ is the measurement at time t, and $\mathbf{H}_t$ is the observation matrix.
The residual covariance is computed by:

$$\mathbf{S}_t = \mathbf{H}_t \mathbf{P}_t \mathbf{H}_t^\top + \mathbf{R} \tag{2.17}$$

where $\mathbf{R}$ is the measurement noise covariance.
The Kalman gain is then calculated as:

$$\mathbf{K}_t = \mathbf{P}_t \mathbf{H}_t^\top \mathbf{S}_t^{-1} \tag{2.18}$$

The updated state and covariance matrix are then:

$$\mathbf{x}_t = \mathbf{x}_t + \mathbf{K}_t \mathbf{y}_t \tag{2.19}$$
$$\mathbf{P}_t = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t)\mathbf{P}_t \tag{2.20}$$

The Kalman filter is optimal for linear systems with Gaussian noise and forms the foundation for more advanced nonlinear filters such as the Extended Kalman Filter (EKF).

2.4.4 Nonlinear Kalman Filter

The EKF extends the Kalman filter to systems with nonlinear process and also measurement models if it is also nonlinear by linearizing them around the current estimate. This is done by computing Jacobians of the nonlinear functions, which approximate how small changes in the state affect the system dynamics and measurements. These Jacobian matrices replace the constant matrices used in the standard Kalman filter.
In the prediction step, the state transition function $f(\cdot)$ and process noise covariance are used to propagate the previous state estimate and its uncertainty:

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}, \mathbf{u}_{t-1}) \tag{2.21}$$
$$\mathbf{P}_t = \mathbf{F}_{t-1} \mathbf{P}_{t-1} \mathbf{F}_{t-1}^\top + \mathbf{Q} \tag{2.22}$$

Here, $f(\cdot)$ is the nonlinear process model, and $\mathbf{F}_{t-1}$ is the Jacobian of f with respect to $\mathbf{x}_{t-1}$.
In the update step, the measurement function $h(\cdot)$ is used to predict the measurement and compute the residual, the residual covariance, and finally the Kalman gain:

$$\mathbf{y}_t = \mathbf{z}_t - h(\mathbf{x}_t) \tag{2.23}$$
$$\mathbf{S}_t = \mathbf{H}_t \mathbf{P}_t \mathbf{H}_t^\top + \mathbf{R} \tag{2.24}$$
$$\mathbf{K}_t = \mathbf{P}_t \mathbf{H}_t^\top \mathbf{S}_t^{-1} \tag{2.25}$$

where $h(\cdot)$ is the nonlinear measurement model, and $\mathbf{H}_t$ is its Jacobian with respect to the state.
The updated state and covariance are:

$$\mathbf{x}_t = \mathbf{x}_t + \mathbf{K}_t \mathbf{y}_t \tag{2.26}$$

$$\mathbf{P}_t = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t)\mathbf{P}_t \tag{2.27}$$

The EKF is widely used in practice due to its simplicity and effectiveness for systems that are only mildly nonlinear. It retains much of the computational efficiency of the linear Kalman filter, but the linearization can introduce errors if the system is highly nonlinear or poorly modeled.

The Unscented Kalman Filter (UKF) was developed to improve estimation in systems with stronger nonlinearities, without requiring the explicit computation of Jacobians. Instead of linearizing the functions, the UKF uses a set of deterministically chosen points, called sigma points, to approximate the mean and covariance of the state distribution. These sigma points are propagated through the nonlinear functions, and the posterior statistics are then reconstructed from the transformed points. This method provides better accuracy than the EKF in many nonlinear systems, especially when the nonlinearity is significant or difficult to model analytically. The UKF is particularly advantageous when the system dynamics are smooth but complex, as it can handle curvature in the state distribution more effectively [22].
To compute the predicted mean and covariance, the UKF generates a set of $2n + 1$ sigma points $\chi^{(i)}$ from the prior state estimate $\hat{x}$ and its covariance matrix P where n is the number of states. The sigma points are defined as:

$$\chi^{(0)} = \hat{x} \tag{2.28}$$

$$\chi^{(i)} = \hat{x} + \sqrt{\frac{n}{1 - W_0}} \left[P^{1/2}\right]_i, \quad i = 1, \ldots, n \tag{2.29}$$

$$\chi^{(i+n)} = \hat{x} - \sqrt{\frac{n}{1 - W_0}} \left[P^{1/2}\right]_i, \quad i = 1, \ldots, n \tag{2.30}$$

Here, $\left[P^{1/2}\right]_i$ denotes the i-th column of the matrix square root of P, satisfying:

$$P^{1/2} \left(P^{1/2}\right)^\top = P \tag{2.31}$$

This square root matrix is typically computed using a Cholesky decomposition. The weights W_i are used to compute the weighted mean and covariance of the transformed sigma points. For Gaussian distributions, a common choice is $W_0 = 1 - \frac{n}{3}$.
Each sigma point $\chi_t^{(i)}$ is propagated through the nonlinear process model $f(\cdot)$:

$$\chi_{t|t-1}^{(i)} = f(\chi_{t-1}^{(i)}) \tag{2.32}$$

The predicted state mean $\hat{x}_{t|t-1}$ and covariance $P_{t|t-1}$ are then computed as:

$$\hat{x}_{t|t-1} = \sum_{i=0}^{2n} W_i^{(m)} \chi_{t|t-1}^{(i)} \tag{2.33}$$

$$P_{t|t-1} = Q_{t-1} + \sum_{i=0}^{2n} W_i^{(c)} (\chi_{t|t-1}^{(i)} - \hat{x}_{t|t-1})(\chi_{t|t-1}^{(i)} - \hat{x}_{t|t-1})^\top \tag{2.34}$$

Here, $W_i^{(m)}$ and $W_i^{(c)}$ are the weights for the mean and covariance, respectively, and Q_{t-1} is the process noise covariance.
For the measurement update, the sigma points are passed through the nonlinear measurement function $h(\cdot)$:

$$y_t^{(i)} = h(\chi_{t|t-1}^{(i)}) \tag{2.35}$$

The predicted measurement mean $\hat{y}_t$, the innovation covariance S_t, and the cross-covariance P_{xy} are calculated as:

$$\hat{y}_t = \sum_{i=0}^{2n} W_i^{(m)} y_t^{(i)} \tag{2.36}$$

$$S_t = R_t + \sum_{i=0}^{2n} W_i^{(c)} (y_t^{(i)} - \hat{y}_t)(y_t^{(i)} - \hat{y}_t)^\top \tag{2.37}$$

$$P_{xy} = \sum_{i=0}^{2n} W_i^{(c)} (\chi_{t|t-1}^{(i)} - \hat{x}_{t|t-1})(y_t^{(i)} - \hat{y}_t)^\top \tag{2.38}$$

where R_t is the measurement noise. The Kalman gain, updated state, and covariance are then given by:

$$K_t = P_{xy} S_t^{-1} \tag{2.39}$$

$$\hat{x}_t = \hat{x}_{t|t-1} + K_t(z_t - \hat{y}_t) \tag{2.40}$$

$$P_t = P_{t|t-1} - K_t S_t K_t^\top \tag{2.41}$$

In contrast to the EKF, the UKF does not require the computation of Jacobians. Instead, it captures the effects of nonlinearity more accurately by propagating the full sigma point set through the process and measurement models.

An alternative approach is the Cubature Kalman Filter (CKF), which also avoids linearization but instead uses a third-degree spherical-radial cubature rule to approximate the Gaussian integrals involved in the Bayesian update. The CKF selects a minimal set of integration points on the surface of a unit sphere and transforms them to capture the shape of the underlying distribution.

Like the UKF, the CKF handles nonlinear transformations more accurately than the EKF while keeping the computational cost relatively low. In practice, the performance of UKF and CKF is often similar, and the choice between them may depend on implementation complexity or system-specific characteristics [22].

2.4.5 Particle Filter (PF)

The Particle Filter (PF), also known as the Sequential Monte Carlo (SMC) method, is a non-parametric Bayesian filtering technique used to estimate complex and non-Gaussian distributions. Unlike Kalman-based filters, the PF does not assume the state distribution is Gaussian or even unimodal. Instead, it represents the state distribution with a set of discrete samples, or particles, each with an associated weight.

Each particle represents a possible realization of the state, and the entire set approximates the posterior distribution. The algorithm consists of three main steps: sampling, weighting, and resampling. In the prediction step, particles are propagated through the system dynamics. In the update step, each particle is weighted based on how well it explains the new observation. Finally, in the resampling step, particles with higher weights are duplicated, while those with low weights are discarded, helping to focus computational effort on the most likely regions of the state space.

Because the PF makes no assumptions about linearity or Gaussian noise, it is especially useful for systems with strong nonlinearities or multi-modal distributions. However, its main limitations are high computational cost and the problem of particle degeneracy, where after several iterations only a few particles carry significant weight [22].

2.4.6 Sensor Fusion Framework

In multi-sensor systems, different sensors contribute information at varying rates and in different mathematical forms. To effectively combine them, a state-space model is defined, consisting of a process model and measurement models. The system is typically implemented within an Extended Kalman Filter (EKF) or Unscented Kalman Filter (UKF) framework.

The general system formulation consists of a state vector $\mathbf{x}_t$, which evolves over time through a nonlinear process model:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t) + \mathbf{w}_t, \quad \mathbf{w}_t \sim \mathcal{N}(0, Q) \tag{2.42}$$

Measurements from various sensors are modeled as:

$$\mathbf{z}_t = h(\mathbf{x}_t) + \mathbf{v}_t, \quad \mathbf{v}_t \sim \mathcal{N}(0, R) \tag{2.43}$$

Different sensors are integrated into the system as follows:

- **GNSS:** GNSS provides direct observations of the position and velocity of the system. Since these are linear with respect to the state, the GNSS measurement model can be written as:

$$\mathbf{z}_{\text{GNSS}} = H_{\text{GNSS}}\mathbf{x}_t + \mathbf{v}_{\text{GNSS},t}, \quad \mathbf{v}_{\text{GNSS},t} \sim \mathcal{N}(0, R_{\text{GNSS}}) \tag{2.44}$$

  where H_{GNSS} is a linear observation matrix that selects the observed components from the state vector, and R_{GNSS} is the measurement noise covariance.

- **UWB:** UWB sensors provide range measurements to anchors, which are nonlinear functions of the system's position. These measurements are modeled as:

$$\mathbf{z}_{\text{UWB}} = h_{\text{UWB}}(\mathbf{x}_t) + \mathbf{v}_{\text{UWB},t}, \quad \mathbf{v}_{\text{UWB},t} \sim \mathcal{N}(0, R_{\text{UWB}}) \tag{2.45}$$

  where $h_{\text{UWB}}(\cdot)$ is the nonlinear measurement function, and R_{UWB} is the corresponding noise covariance matrix.
- **IMU:** IMU sensors provide high-rate measurements of linear acceleration and angular velocity. In typical 2D systems, the accelerometer is used to model changes in velocity and position, while the gyroscope provides the yaw rate (change in heading). These measurements are incorporated into the process model to predict the system state between updates:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{a}_{\text{meas},t}, \dot{\theta}_{\text{meas},t}) + \mathbf{w}_t \tag{2.46}$$

  Due to their high sampling rate and smooth temporal behavior, IMUs are well suited for short-term motion prediction. However, they accumulate drift over time and must be corrected using absolute measurements from sensors like GNSS or UWB.

3
Methodology and design

3.1 Overall Approach

During the initial phase of the project, a literature review was conducted to understand existing approaches and evaluate suitable sensor technologies. This review informed the selection of GNSS, IMU, and UWB modules that balanced performance, cost, and ease of integration for the intended application.

The project was carried out in four main stages. The first stage focused on component selection and sensor evaluation. In the second stage, system design and data acquisition architecture were developed, including the hardware layout and interfaces required to combine the selected sensors into a functional system.

Once the system design was established, the third stage centered on algorithm development. In this phase, data from the various sensors was fused using sensor fusion techniques to produce more robust and accurate estimates. Data collection and testing were carried out iteratively throughout this phase to verify functionality, identify problems, and implement improvements.

The fourth stage of the project involved a series of test scenarios designed to evaluate the system in practical environments. Data was collected using the complete sensor setup and then processed through the developed sensor fusion algorithm to assess performance under realistic conditions. Rather than performing traditional simulations with synthetic data or modeled systems, the collected data was used as input to the fusion algorithm in an offline manner to simulate system behavior. This allowed for verification of the filter design and identification of potential improvements.

This structured and modular approach enabled iterative development throughout the project and ensured that each subsystem could be tested, validated, and refined before full system integration.

3.2 Hardware and System Design

To collect real-world data, the project required careful hardware selection to ensure both ease of integration with a microcontroller and the measurement accuracy needed for realistic testing scenarios. The chosen microcontroller platform was the Raspberry Pi, using models 4B and 5, due to their support for common communi-

cation interfaces such as I^2C and USB.

All sensor modules used in this project were provided as development kits with integrated communication interfaces, simplifying hardware integration.

3.2.1 Components

GNSS Module: The GNSS module provides global positioning using multiple satellite constellations. In this project, a standard GNSS module was selected to keep the system cost-effective and maintain focus on improving positioning through sensor fusion.

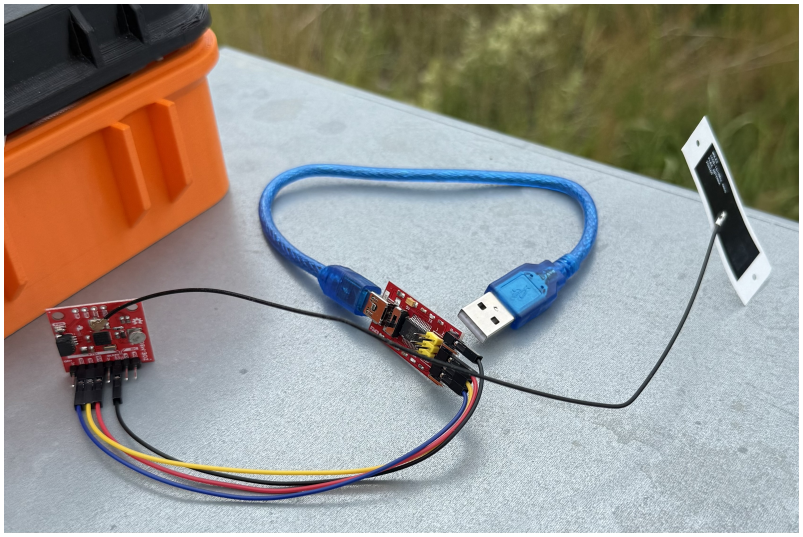

Figure 3.1: GNSS module

The module chosen was the u-blox ZOE-M8Q, a compact and affordable GNSS receiver costing only a few hundred Swedish kronor. This module supports all major GNSS constellations, including GPS, Galileo, GLONASS, and BeiDou, and offers an update rate of up to 10 Hz while maintaining reliable performance. According to the manufacturer, the typical positioning accuracy of the ZOE-M8Q is around 2.5 to 3 meters under open-sky conditions. [25].

The module was connected to the Raspberry Pi via a USB-to-UART converter, allowing serial communication using the u-blox proprietary UBX protocol, which offers higher performance and lower data overhead compared to standard NMEA 0183 output. This protocol provides access to detailed positioning data and system status, making it suitable for sensor fusion applications.

An external Molex 206560 GNSS active antenna was used to improve reception performance in both open-sky and partially obstructed environments. The antenna is designed for multi-constellation GNSS applications and supports GPS, GLONASS, Galileo, and BeiDou signals, providing improved signal quality over integrated chip antennas [26].

Although the module supports differential GPS (DGPS) via RTCM 2.3 correction messages, enabling this feature proved more challenging than expected due to compatibility issues with the transmitted RTCM data.

The module was configured to operate at a 5 Hz update rate to achieve a balance between measurement frequency and reliability.

IMU Module: With its ability to provide higher update rate data, the IMU module was used to estimate orientation and capture small changes in acceleration of the system. The chosen IMU module was the BNO055, a 9-axis absolute orientation sensor that includes a built-in microcontroller for sensor fusion. This allows it to provide fully processed orientation data without requiring external computational resources [27].

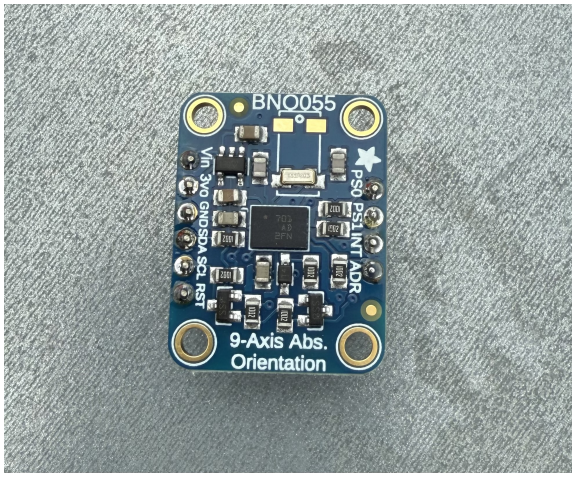

Figure 3.2: IMU Module

The module was connected via I^2C to the Raspberry Pi and was accessed using an existing Python library with a user-friendly API, provided by Adafruit's CircuitPython project [28]. This made data acquisition straightforward and reliable.
For this project, the IMU was configured to operate at 20 Hz. Both the absolute orientation estimate (quaternion or Euler angles) and magnetic field readings were used. The magnetic field measurements were utilized to calculate a yaw bias — the angular difference between the sensor's reported orientation and the magnetic north reference. This bias correction was applied to improve heading estimation.

Additionally, linear acceleration data from the IMU was used to estimate motion. By transforming these acceleration measurements into east and north components using the calculated yaw angle, the system was able to derive movement estimates in the horizontal plane. While the IMU provided valuable short-term motion estimates, it is well known that such sensors are prone to accumulating errors over time due to drift. This makes them unreliable for long-term positioning without correction from external positioning systems.

UWB Module: The UWB modules were used as a local positioning system to estimate the position of a device relative to a device. The technology enables high-precision distance measurements using time-of-flight calculations between devices.

For this project, the Qorvo MDEK1001 Development Kit was selected. It includes multiple UWB modules that can be configured either as anchors (reference points) or tags (mobile devices). One of the main reasons this kit was chosen was its ease of integration with the Raspberry Pi via USB and its high-level communication interface using a pre-configured API. The kit simplifies data access and reduces development time by handling the complex ranging calculations internally.

The MDEK1001 system achieves typical positioning accuracy of approximately 10–30 cm under optimal conditions [29]. However, the system has some limitations. It does not support angle-of-arrival (AoA) or time-difference-of-arrival (TDoA), meaning it cannot estimate heading or orientation between devices—only distance.

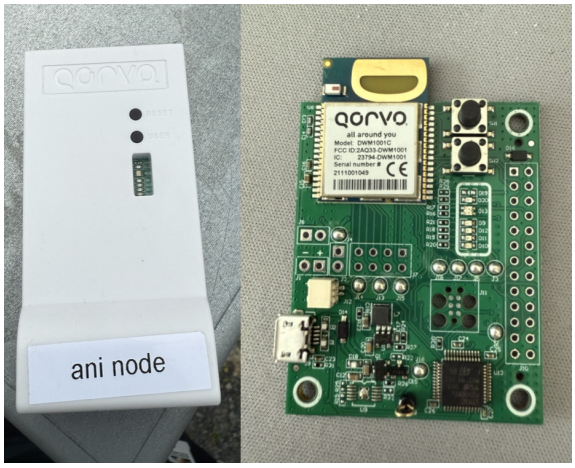

Figure 3.3: UWB Module

Although this project primarily uses a fixed anchor setup, peer-to-peer configurations without anchor infrastructure were also considered as a possible use case. In the development kit used here, tags cannot directly communicate with one another; each mobile tag must be in range of at least one anchor to obtain a position update. This limits the scalability of the system in such peer-to-peer scenarios.

Despite these constraints, UWB offers a valuable supplement to GNSS and IMU data, especially in indoor or GNSS-degraded environments, by providing accurate and robust short-range distance measurements.

3.2.2 Reference System

To evaluate the accuracy of the developed sensor fusion method, a high-precision reference system was required. The reference system used in this project was based

on the SWEPOS national GNSS infrastructure, specifically using its Network Real-Time Kinematic (NRTK) service. This system offers centimeter-level positioning accuracy by providing real-time correction data through a network of fixed reference stations across Sweden [30].

To receive and process the NRTK corrections, a multi-frequency GNSS setup was used. The receiver was a UM981-based RTK module mounted on a simpleRTK3B development board, which supports multi-band GNSS reception and real-time RTK correction via NTRIP [31]. An external high-precision GNSS antenna, the SparkFun GNSS Multi-Band L1/L2/L5 antenna (SPK6618H), was used in combination with the receiver to ensure optimal signal quality and compatibility with the SWEPOS service [32].

This configuration is capable of achieving position accuracies down to approximately 1 cm under optimal conditions. It was used as a ground truth reference to evaluate the output of the fused sensor system in both static and dynamic scenarios.

3.2.3 Physical Setup

The physical setup for each device included a standard GNSS module, an IMU, and two UWB modules, one configured as an anchor and the other as a tag. The anchor was responsible for broadcasting its position and accuracy, while the tag measured its distance to and the position of nearby anchors to later estimate its relative position. Each device also included a Raspberry Pi, which served as the central unit for sensor communication and data logging.

This setup allowed data from all three sensors to be collected simultaneously using Python scripts running on the Raspberry Pi, enabling synchronized logging for later offline processing and sensor fusion analysis.

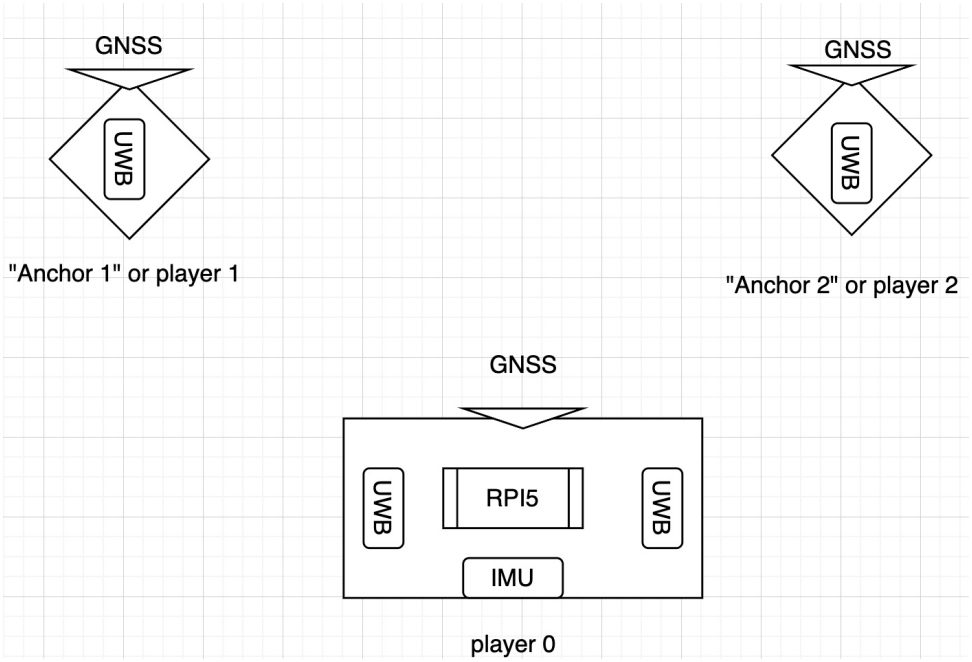

Figure 3.4: System overview showing the integrated mobile unit (player 0) and two GNSS-UWB anchor nodes (players 1 and 2). Each anchor contains a GNSS and UWB module. The mobile unit integrates UWB, GNSS, an IMU, and a Raspberry Pi 5 for logging and processing.

3.3 Sensor Fusion Algorithm

With the different strengths and weaknesses of the sensors used in this project, sensor fusion plays a central role in achieving more accurate and reliable positioning.

The GNSS module provides a global position estimate but suffers from a relatively low update rate, limited accuracy, and frequent signal loss in obstructed environments such as forests or urban areas. The IMU offers high-frequency updates and captures short-term motion well, but it is prone to drift over time, making it unreliable for long-term position estimation on its own.The UWB system contributes precise short-range distance measurements and functions well in local positioning tasks. However, in this project, no angle or heading information was available from the UWB modules, which limited their ability to support orientation-aware positioning.

By combining the complementary strengths of these sensors through sensor fusion, the goal is to create a system that provides stable, continuous, and accurate position estimates even in environments where individual sensors may fail.

3.3.1 Filter Choice and Justification

Because of the nonlinearities involved in the system, such as using angular measurements like heading from the GNSS and IMU, or processing UWB distance and

relative angle data, a nonlinear Kalman filter was selected. A nonlinear Kalman filter was preferred over a particle filter due to its lower computational complexity and better suitability for the available processing resources.

While the Extended Kalman Filter (EKF) is a common choice, it only approximates the system behavior by linearizing around the last known estimate. More advanced nonlinear filters like the Unscented Kalman Filter (UKF) and the Cubature Kalman Filter (CKF) offer improved performance in highly nonlinear systems. For this project, the UKF was chosen because it considers multiple sigma points, including the mean, to better represent the distribution of possible states. This results in a more accurate prediction and update process in the presence of nonlinearities. In contrast, the CKF distributes its sigma points on a spherical surface and does not guarantee that one of them lies at the mean, which can make the estimate less directly interpretable and slightly less stable in some practical applications.

The next step was to define how the prediction step of the filter should behave. It was concluded that the prediction should run at the same update rate as the IMU, allowing continuous integration of high-frequency motion data into the filter's state estimate. while the update step for the measurments of the UWB and GNSS updates at a rate between 2-5 Hz.

3.3.2 State Vector Definition

The main goal of this thesis is to estimate the 2D position of the player using sensor fusion. To support this, the state vector was designed to include not only position but also key motion-related variables that influence position over time and orientation.

The state vector was defined as:

$$\mathbf{x} = [e, n, v, a, \varphi]^T$$

where:
- e, n represent the 2D position in Easting and Northing coordinates,
- v is the scalar velocity (speed),
- a is the scalar linear acceleration,
- φ is the heading angle (yaw), describing orientation in the horizontal plane.

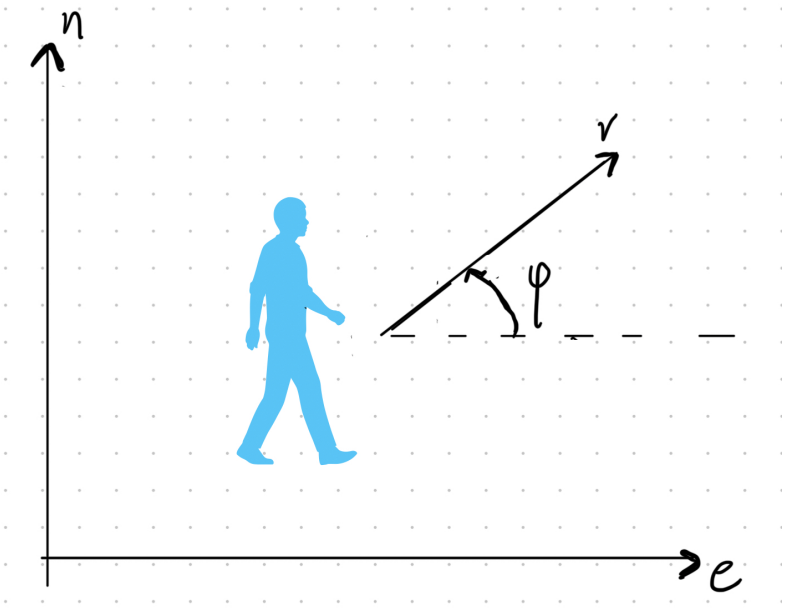

Figure 3.5: State vector diagram

This structure allows the filter to capture both translational and rotational dynamics of the system. The IMU provides acceleration, the GNSS supplies position and heading (course over ground), and the UWB system adds local distance constraints. Including velocity and acceleration enables smoother position estimates, particularly in cases where GNSS data is temporarily unavailable or degraded.

The corrected IMU yaw angle (referred to as `corrected_yaw` in the implementation that comes from the orientation of the IMU) is used to transform the scalar acceleration into directional components within the local East-North coordinate system. This orientation is critical for projecting the forward acceleration vector in the appropriate direction of movement. Separately, the heading angle φ, shown in Figure 3.5, defines the direction of the velocity vector v relative to the east axis. It is used in the motion model to decompose velocity into its easting and northing components. The diagram helps clarify how v and φ are aligned in the global frame, while the IMU-derived heading informs how acceleration contributes to state propagation.

3.3.3 Prediction Step Design

The prediction step of the filter is executed at the same rate as the IMU, which in this project was configured to 20 Hz. This enables the filter to continuously update the estimated state using the most recent sensor input.

The Unscented Kalman Filter (UKF) was used to handle the nonlinearities in the motion model, particularly due to the trigonometric dependency on yaw when updating position. This motion model follows the general nonlinear process model structure introduced in Section 2.4.6, where the state is predicted using sensor inputs and additive process noise. At each prediction step, sigma points are generated and propagated through the motion model function $f(\cdot)$, which uses the current

state and scalar acceleration input to estimate the next state.

In this implementation, the acceleration input is computed from the IMU's east and north linear acceleration components as:

$$a_{input} = \sqrt{a_e^2 + a_n^2}$$

This scalar acceleration is then used to update the velocity in the direction of the current yaw angle, which is stored in the state vector. The yaw angle is carried forward unchanged in the prediction step, and no additional yaw rate model is currently applied.

The motion model performs the following updates:

$$\begin{aligned} e_{t+1} &= e_t + v_t \cdot \Delta t \cdot \cos(\varphi_t) \\ n_{t+1} &= n_t + v_t \cdot \Delta t \cdot \sin(\varphi_t) \\ v_{t+1} &= v_t + a_t \cdot \Delta t \\ a_{t+1} &= a_{\text{input}} \\ \varphi_{t+1} &= \varphi_t \end{aligned}$$

To account for uncertainty in the motion model, a process noise term $\mathbf{w}_t \sim \mathcal{N}(0, Q)$ is added during sigma point propagation. This noise captures model imperfections such as IMU bias, unmodeled yaw drift, and discretization errors. The process noise covariance matrix Q is implemented as a diagonal matrix with empirically tuned values:

$$Q = \text{diag}(0.1,\ 0.1,\ 1,\ 10,\ 1) \tag{3.1}$$

These correspond to the east and north positions, velocity, acceleration, and yaw angle respectively. The relatively high variance on velocity, acceleration and yaw reflects their greater sensitivity to drift and unmodeled dynamics. The values of Q were kept fixed across all test scenarios.

This approach ensures that movement is projected in the current heading direction, while velocity is adjusted based on the new scalar acceleration. During stationary scenarios, where the user is not moving, the filter simply maintains the last known state, and GNSS and UWB are used to constrain position updates.

3.3.4 Update Step Design

The update step of the filter was implemented separately for GNSS and UWB data, as these sources provide different information and require different measurement models.

GNSS Update: The GNSS update corrects the estimated state using absolute measurements of easting, northing, scalar velocity, and course over ground (heading). These four observations are used to update the corresponding elements in the

state vector: position (e, n), velocity v, and yaw angle φ.

This update follows the general linear measurement model introduced in Section 2.4.6, see Equation (2.44), and is instantiated here with the following observation matrix:

$$H_{\text{GNSS}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.2}$$

The measurement noise is modeled using a diagonal covariance matrix:

$$R_{\text{GNSS}} = \text{diag}(\sigma^2_{\text{pos}}, \sigma^2_{\text{pos}}, 2 \cdot \sigma^2_{\text{vel}}, 2 \cdot \sigma^2_{\text{hdg}}) \tag{3.3}$$

where:

- σ^2_{pos} is the position variance (e.g., 1–5 m),
- σ^2_{vel} is the velocity variance (e.g., 0.1–0.5 m/s),
- σ^2_{hdg} is the heading variance in radians.

The position, speed, and heading accuracy values provided by the GNSS module were used to construct the measurement covariance matrix R_{GNSS}. However, to account for observed underestimation in the reported velocity and heading uncertainties, the variances for these two channels were scaled by a factor of two before being used in the filter. These adjusted values were updated at each time step based on the module's internal accuracy estimates.

UWB Update: The UWB update was designed to accommodate the relative nature of the sensor data and was implemented in multiple forms depending on the test scenario. Because the UWB modules used in this work do not support Angle of Arrival (AoA), no directional information was directly measured. Instead, a simulated bearing angle between the device and each anchor was estimated using geometric reasoning and GNSS data. This simulated angle was used alongside the UWB range to improve observability, especially in scenarios with limited GNSS access or weak anchor configurations.

Figure 3.6 illustrates how this simulated angle is computed. The two overlapping blue circles represent range constraints from UWB anchors. Their intersection provides a candidate position for the device. The angle between the anchor and device is then estimated using:

- The angle from the anchor to the circle intersection point:

$$\theta_{\text{intersection}} = \text{atan2}(y_{\text{anchor}} - y_{\text{int}}, x_{\text{anchor}} - x_{\text{int}})$$

- The angle from the anchor to the GNSS-estimated device position:

$$\theta_{\text{GNSS}} = \text{atan2}(y_{\text{anchor}} - y_{\text{device}}, x_{\text{anchor}} - x_{\text{device}})$$

These are averaged to form a stabilized simulated bearing:

$$\theta_{\text{avg}} = \frac{1}{3}(2 \cdot \theta_{\text{intersection}} + \theta_{\text{GNSS}})$$

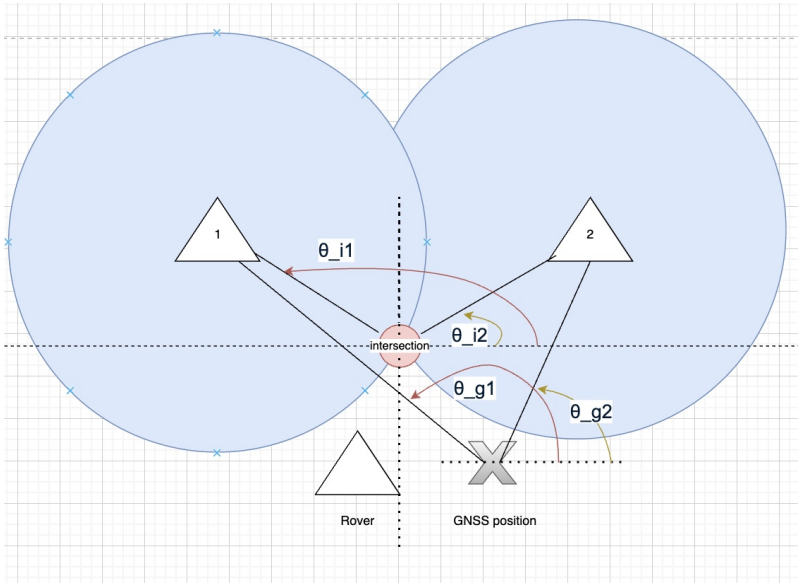

Figure 3.6: Angle estimation from UWB geometry and GNSS reference

This angle is used in two different residual formulations:

- **Range-Angle Residual (Residual 1):** In this method, each UWB update includes two components: a range measurement from the tag to a fixed anchor and a simulated bearing angle between them. Since the UWB module does not support Angle of Arrival (AoA), the bearing angle is estimated using GNSS and geometric reasoning as described above.

  The nonlinear measurement function is defined as:

$$h(\mathbf{x}, \mathbf{p}) = \begin{bmatrix} \|\mathbf{x}_{\text{pos}} - \mathbf{p}\| \\ \text{atan2}(p_y - x_y,\ p_x - x_x) \end{bmatrix} \tag{3.4}$$

  where:
  - $\mathbf{x}_{\text{pos}} = [x_x,\ x_y]^\top$ is the position of the state estimate,
  - $\mathbf{p} = [p_x,\ p_y]^\top$ is the known anchor position.

  The residual is formed by comparing the predicted measurement $\hat{\mathbf{z}}$ from the measurement function above with the observed range and simulated angle $\mathbf{z} = [r,\ \theta]^\top$, where θ is the stabilized average of the intersection-based and GNSS-derived angles.

  To account for uncertainty in anchor position Σ_{anchor} and in the range-angle pseudo-measurement Σ_{meas}, the full covariance is computed as:

$$R = H_{\text{anchor}} \Sigma_{\text{anchor}} H_{\text{anchor}}^\top + \Sigma_{\text{meas}} \tag{3.5}$$

  Here, $H_{\text{anchor}} \in \mathbb{R}^{2\times 2}$ is the Jacobian of the measurement function with respect to the anchor position:

$$H_{\text{anchor}} = \begin{bmatrix} \frac{\partial r}{\partial x_a} & \frac{\partial r}{\partial y_a} \\ \frac{\partial \theta}{\partial x_a} & \frac{\partial \theta}{\partial y_a} \end{bmatrix} = \begin{bmatrix} -\frac{dx}{r} & -\frac{dy}{r} \\ \frac{dy}{r^2} & -\frac{dx}{r^2} \end{bmatrix}, \quad dx = x_x - x_a,\ dy = x_y - y_a,\ r = \sqrt{dx^2 + dy^2} \tag{3.6}$$

The anchor position uncertainty Σ_{anchor} is modeled as a diagonal matrix with entries based on GNSS-reported horizontal accuracy at the anchor's location. Each anchor was surveyed using the same GNSS receiver as the mobile unit, and the module's internally reported accuracy. These values were squared to form the diagonal of Σ_{anchor}.

The range and angle measurement uncertainties are similarly modeled with fixed standard deviations:

$$\sigma_r = 0.6 \text{ m}, \quad \sigma_\theta = \sigma_\theta = 7^\circ = \frac{7 \cdot \pi}{180} \text{ rad}$$

These are used to construct the diagonal measurement noise matrix:

$$\Sigma_{\text{meas}} = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_\theta^2 \end{bmatrix} \tag{3.7}$$

This formulation enables the combined effect of anchor position uncertainty and measurement noise to be propagated through the nonlinear measurement model, producing a total innovation covariance matrix R that reflects both environmental and sensor-based uncertainties.

The UKF update step then proceeds using the sigma point transform of the state through $h(\cdot)$, and updates the state and covariance based on the Kalman gain computed from the predicted cross-covariance and the innovation covariance matrix R.

- **Projected Position from Range-Angle (Residual 2):** In this alternative method, the measured range and simulated angle are used to project a pseudo-position for the device relative to the anchor:

$$\begin{bmatrix} x_{\text{projected}} \\ y_{\text{projected}} \end{bmatrix} = \begin{bmatrix} x_a - r \cdot \cos(\theta) \\ y_a - r \cdot \sin(\theta) \end{bmatrix} \tag{3.8}$$

This projected point is treated as a direct measurement of the device's position and compared against the filter's estimated (x, y) state. A standard UKF update is then applied.

Measurement uncertainty is transformed into position space using a Jacobian:

$$H = \begin{bmatrix} 1 & 0 & -\cos(\theta) & r\sin(\theta) \\ 0 & 1 & -\sin(\theta) & -r\cos(\theta) \end{bmatrix} \tag{3.9}$$

where the input vector is $[x_a, y_a, r, \theta]$. The input covariance R_{input} is constructed as a block-diagonal matrix containing Σ_{anchor} and Σ_{meas}. These are extracted as sub-blocks and propagated through the Jacobian to compute the final measurement covariance:

$$R = H \cdot R_{\text{input}} \cdot H^{\top} \tag{3.10}$$

Finally, this linearized measurement is fused with the predicted state using the UKF, treating the projected position as a pseudo-observation of the device's true position.

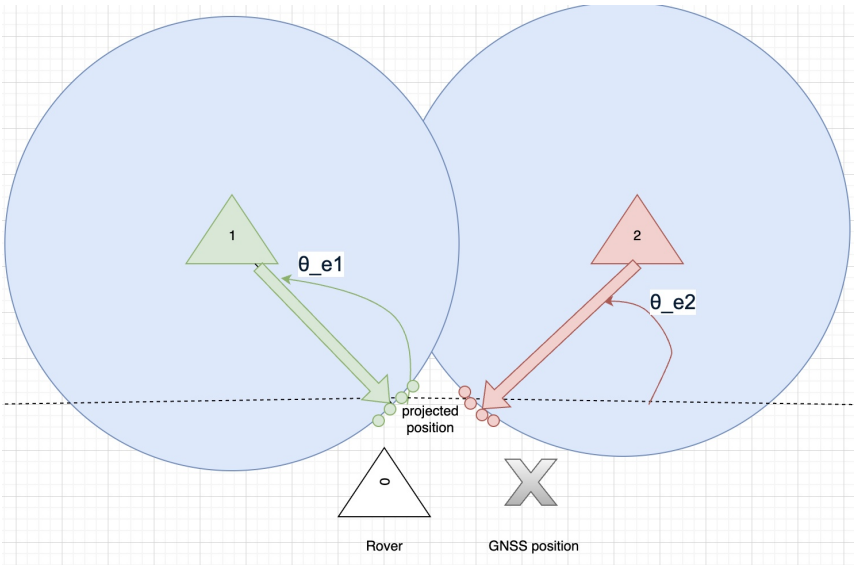

Figure 3.7: Residual 2: Estimating position from range-angle projection.

This projection-based approach allows for a direct update in position space, reducing the nonlinearity compared to residual 1 and simplifying the update when angle and distance are known.

3.3.5 Implementation Details

The sensor fusion algorithm was written in Python using a custom implementation of the Unscented Kalman Filter (UKF). No external libraries for filtering were used, which made it easier to control how the prediction and update steps were handled.

All data was collected in real time on the Raspberry Pi and then processed offline. The prediction step was run at the same rate as the IMU, which was 20 Hz. Both GNSS and UWB updates were added after each measurement cycle, with a frequency of around 2–5 Hz depending on the test and available data. One GNSS update and multiple UWB updates were used at each cycle, based on how many anchors were connected.

The state and the covariance matrix were saved at each timestep to make it possible to analyze and visualize the results after the filtering was done.

3.4 Data Collection and Test Scenarios

Collecting real-world sensor data was an important part of this thesis. It provided a deeper understanding of how each sensor behaves in different environments and revealed practical limitations such as signal degradation, drift, or inconsistent measurement rates.

As the sensor fusion filter was being developed, data was collected continuously in parallel. This allowed the filter to be tested and improved using actual sensor logs instead of relying solely on simulation. Logging in outdoor and semi-controlled environments gave immediate feedback on how well the algorithm handled real-world noise and dynamics.

3.4.1 Test Scenarios

To evaluate the system under realistic conditions, three different environments were tested. In each environment, two types of tests were performed: one with the system stationary at a fixed location, and one where the system was moved dynamically through the area. This allowed both sensor stability and movement tracking to be evaluated in varying signal conditions.

The three test environments were:

- **Open Area:** A car park with clear visibility of the sky. This provided optimal conditions for GNSS reception and served as a baseline for evaluating sensor fusion performance with minimal interference.

Figure 3.8: Open environment

- **Forest / Light Vegetation:** A natural area with moderate tree cover and minimal urban interference. This introduced partial GNSS signal degradation and occasional UWB line-of-sight issues, making it suitable for testing robustness under mild obstruction.

Figure 3.9: Forest environment

- **Obstructed / Urban-like Area:** An area close to buildings or narrow paths between walls, simulating an urban canyon. This scenario challenged GNSS reception due to multipath and signal loss, increasing reliance on UWB and IMU for position estimation.

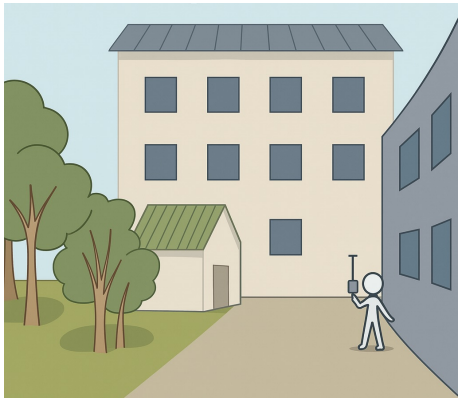

Figure 3.10: Obstructed environment

3.5 Evaluation Metrics and Tools

System performance was evaluated using two main metrics: absolute position error and relative improvement over baseline GNSS. The absolute error was calculated as the Euclidean distance between estimated positions and a measured predefined ground truth point obtained via the SWEPOS NRTK service. Relative improvement was expressed as a percentage of the GNSS-only error to assess the impact of sensor fusion.

Visual inspection of trajectories was also used to assess filter stability and consistency, with trajectories coloured over time to illustrate convergence. All data was collected and processed using custom Python scripts. Plotting and numerical analysis were performed using Python libraries such as NumPy, Matplotlib, and Pandas.

3.6 Simulation

This project did not include simulated environments or artificially generated sensor data. All testing was conducted with physical hardware in real outdoor environments. Filter development was done using recorded logs rather than live streaming data, and performance was validated using post-processing rather than simulation frameworks.

3.7 Limitations

Due to time constraints and GNSS hardware instability during some stages of the project, only one dynamic test was performed. Additional dynamic tests in the open and obstructed areas were initially planned but not completed. While both residual-based measurement functions were implemented in the fusion filter, only two test cases used Residual 2 and the remainder used Residual 1. This was because of the performance and reliably during development, but also due to time constraints.

All testing was conducted offline using logged data. Real-time filtering, live feedback, or adaptive tuning during motion was not implemented.

4

Results

4.1 Overview

This chapter presents the results from the individual sensor evaluations and sensor fusion tests conducted in various environments. First, raw performance of GNSS, IMU, and UWB sensors is evaluated independently under both static and dynamic conditions. These tests establish baseline behavior and highlight the limitations of each sensor when used alone.

Following this, fusion-based position estimates are presented using two different measurement models: one based on UWB angle and distance residuals (Residual 1), and one based on projected UWB positions (Residual 2). The fusion filter is tested across open, forested, and obstructed environments, as well as during a dynamic walking scenario. Comparisons to high-accuracy RTK references and GNSS-only data are used to evaluate positioning accuracy and stability over time.

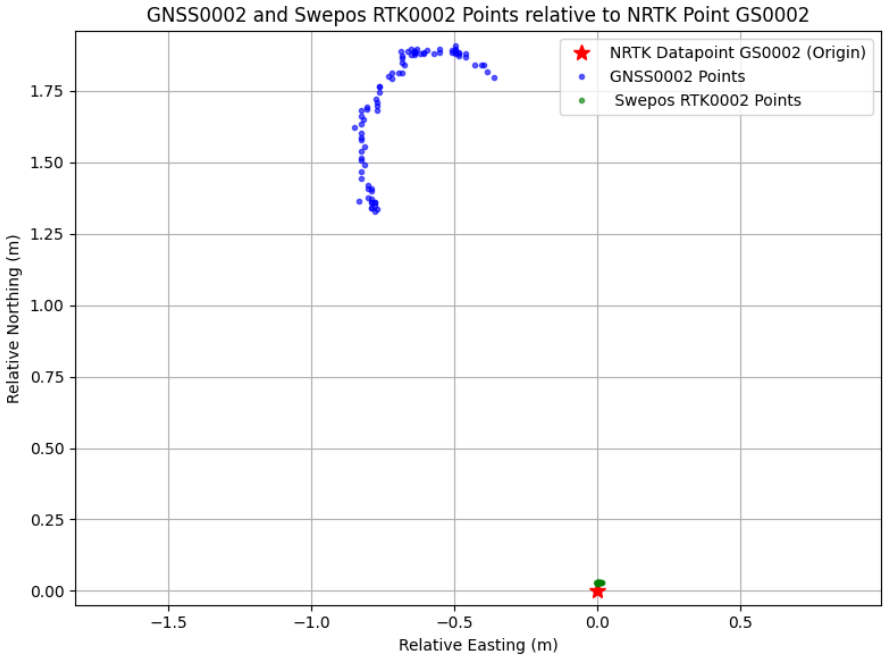

Figure 4.1: GNSS open area

4.2 GNSS-Only Performance

Figure 4.1 shows the GNSS-only measurements collected during the static test in the open area. The red star represents the reference location obtained using a high-end NRTK receiver with the SWEPOS NRTK service (GS0002), which is considered the ground truth. The green point shows the RTK-corrected estimate from a separate professional-grade receiver that also uses the SWEPOS service. Despite not being a high-end unit, this RTK receiver achieves centimeter-level accuracy, making it suitable as a reference in this study. The blue points represent the raw GNSS positions collected from the tested module.
As shown in the figure, the GNSS-only points are consistently offset by approximately 1.5–2 meters to the northwest of the reference position. The spread of the points is relatively small but shows a slight drift over time. This indicates the presence of a consistent bias rather than purely random noise. The accuracy in this environment was roughly within a 2-meter radius, which aligns with the expected performance of consumer-grade GNSS modules without correction support. These measurements serve as a baseline for evaluating the performance of the sensor fusion filter in ideal GNSS conditions.

Figure 4.2 shows the GNSS-only measurements collected during a static test in

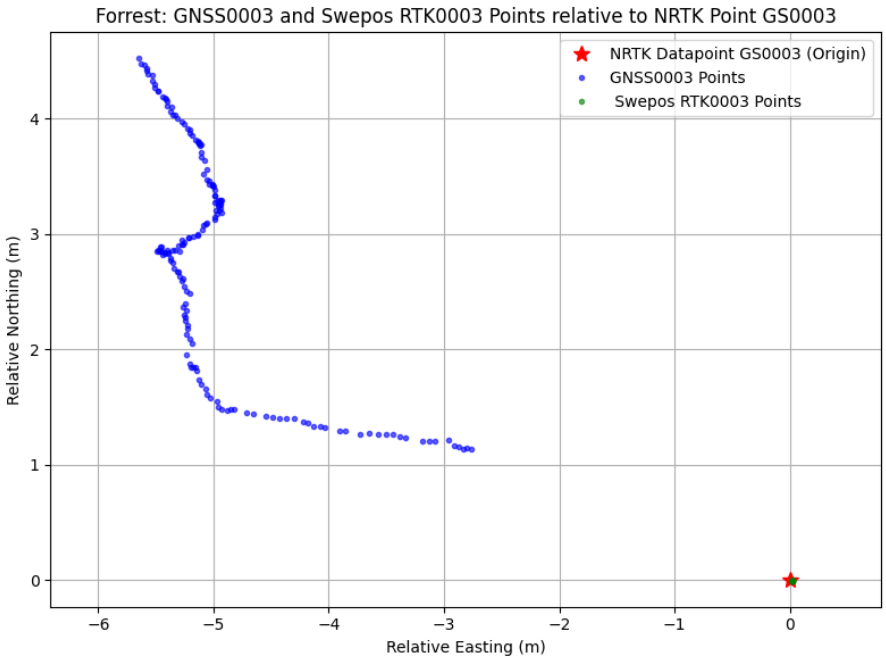

Figure 4.2: GNSS forest area

a forest environment with light to normal vegetation. Unlike the open-area test, no predefined SWEPOS reference point was used. Instead, a reference was estimated using RTK-corrected positions from a high-end receiver connected to the SWEPOS NRTK service, shown as red and green markers.

In this environment, the GNSS-only data (blue) show a significant offset of approximately 3 to more than 10 meters to the northwest of the RTK-derived position. The spread of the points is also wider compared to the open scenario and exhibits increased positional drift.

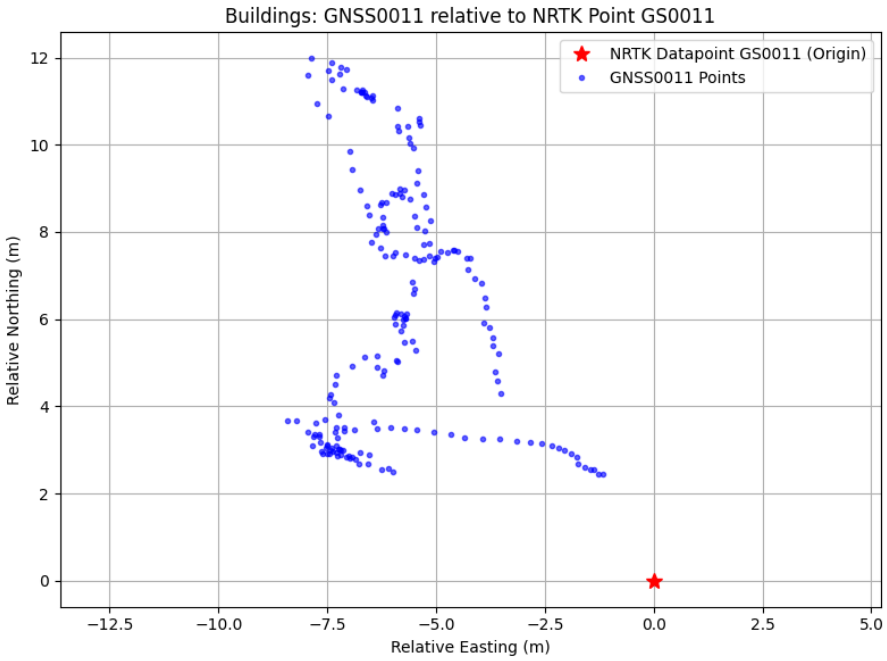

Figure 4.3: GNSS urban area

Figure 4.3 shows the GNSS-only measurements collected in an obstructed urban area with multiple surrounding buildings. The test location was approximately one meter from a building wall, with nearby structures creating potential signal blockage and reflections. The RTK-corrected position (red star) was obtained using a high-end GNSS receiver connected to the SWEPOS NRTK service and serves as the reference.

The GNSS-only points (blue) are offset by approximately 5–15 meters from the reference position, with the cluster biased toward the northwest. The data also show a wider spread and greater point scatter than in the open-area and forest tests.

4.3 IMU Dead Reckoning Performance

Figure 4.4 shows the IMU-only dead reckoning output from a test where the user walked approximately 15 meters forward and then returned along the same path. The position estimates were computed by integrating the measured linear acceleration in the heading direction provided by the IMU.

The left plot shows the full reconstructed path. Although the actual motion was a straight line out and back, the estimated trajectory deviates from linearity and

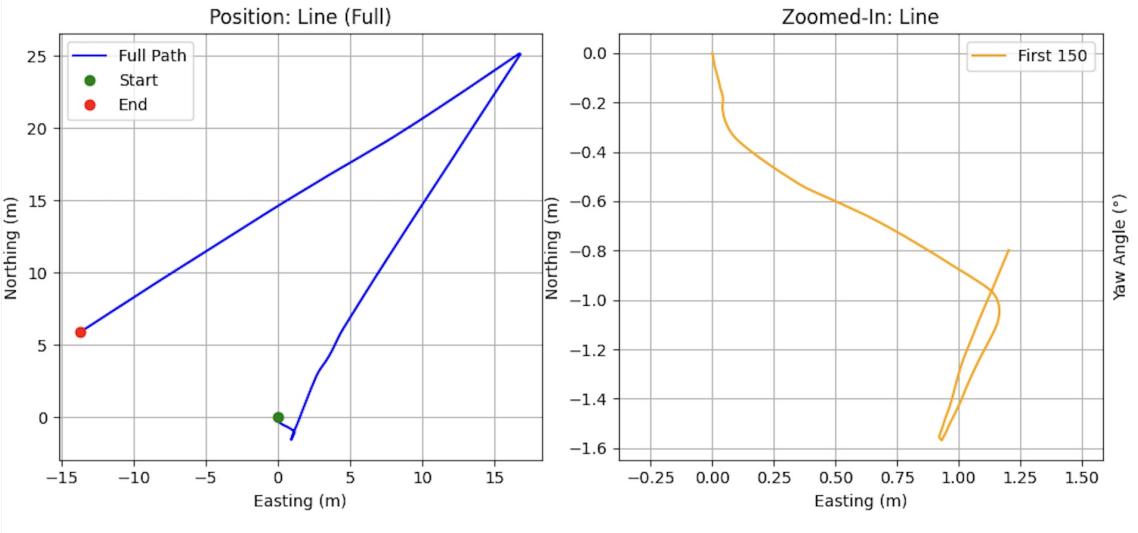

Figure 4.4: IMU Line Test

shows visible drift, especially during the return segment. The loop does not fully close, which indicates accumulated integration errors over the short distance.
The right subplot provides a zoomed-in view of the initial segment of the walk. Small deviations in position appear early, showing that heading uncertainty and acceleration noise affect the trajectory from the beginning.

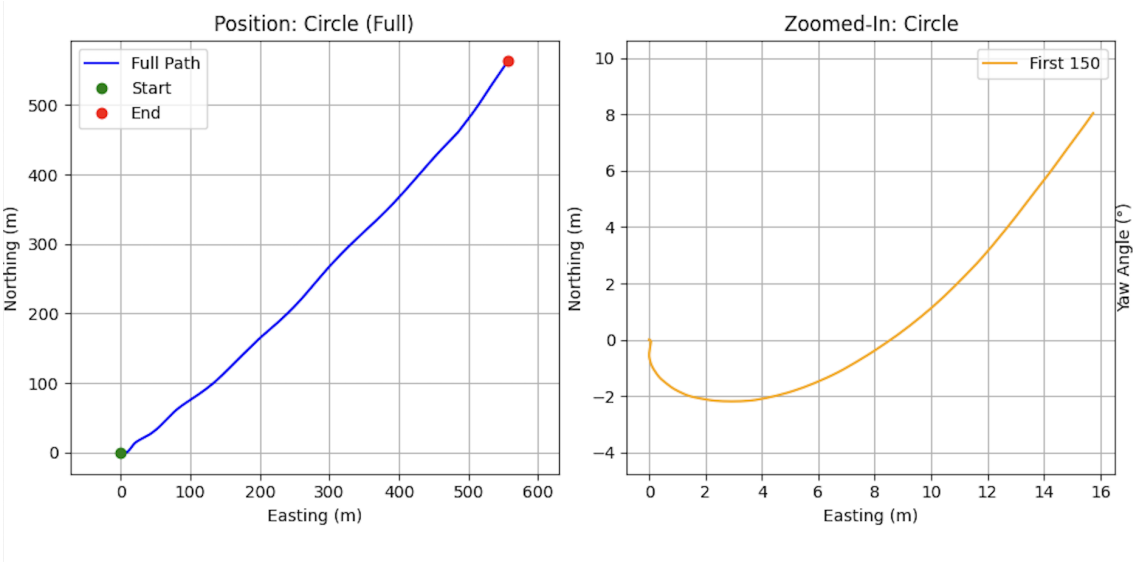

Figure 4.5: IMU Circle Test

Figure 4.5 shows the IMU-based dead reckoning performance during a circular walking test. The user attempted to walk in a circle with an approximate diameter of 10 meters for several laps. The estimated path was computed by integrating linear acceleration projected using the IMU's heading.

The left plot shows the full reconstructed trajectory. Instead of forming a closed circular loop, the estimated path stretches outward with increasing displacement,

highlighting the compounding integration error.

The right plot provides a zoomed-in view of the initial segment. While the path begins to curve in the intended shape, this curvature is quickly lost as drift accumulates.

4.4 UWB Ranging and Positioning Results

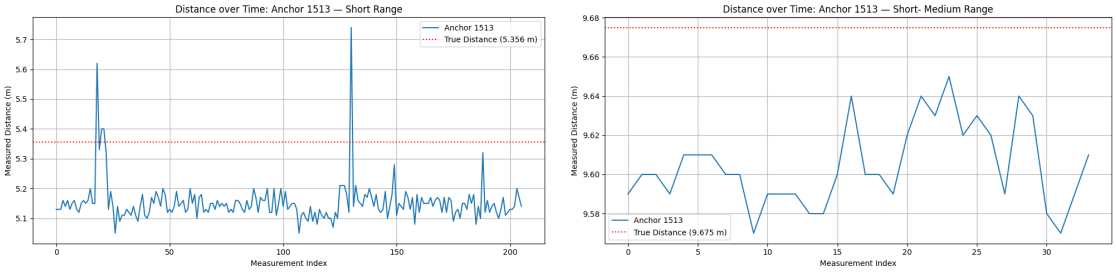

Figure 4.6: UWB Range test Short and Short-Medium

Figure 4.6 shows UWB distance measurements to a fixed anchor (Anchor 1513) at two distances: approximately 5.36 meters (short range, left) and 9.68 meters (short-medium range, right). The true distances are indicated by the horizontal red dashed lines.

At the short range, the measured values fluctuate slightly around the true distance, with deviations typically within ±10–30 cm. Occasional spikes up to 5.6 meters are visible but isolated. Overall, the measurements stay close to the ground truth with low variance.

At the short-medium range, the measured distances remain consistently below the true distance, with values tightly clustered around 9.60 meters compared to the reference of 9.675 meters. The variance is relatively low, and the offset is approximately 7–10 cm. Despite the systematic underestimation, the stability of the readings indicates good repeatability at this distance.

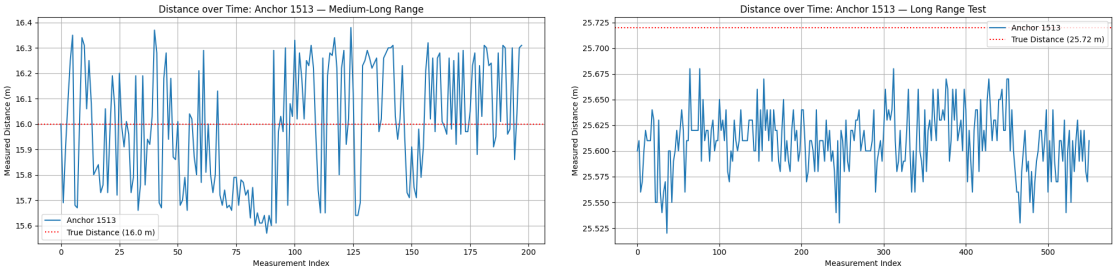

Figure 4.7: UWB Range test Medium-Long and Long

Increasing the range, Figure 4.7 shows the UWB distance measurements for medium-long and long range tests. In the medium-long range test (left), with a true distance

of 16.0 meters, the measured values fluctuate around the reference line with an average offset of ±30-40 cm. The variation increases compared to shorter ranges, and the measurements occasionally drift beyond ±20 cm.

In the long-range test (right), with a true distance of 25.72 meters, the offset becomes slightly larger but consistent. Most values fall between 25.55 and 25.68 meters, indicating an underestimation of around 15-30 cm. The spread is slightly more irregular than at shorter ranges, suggesting increased noise with longer distances, but still within a consistent operating margin.

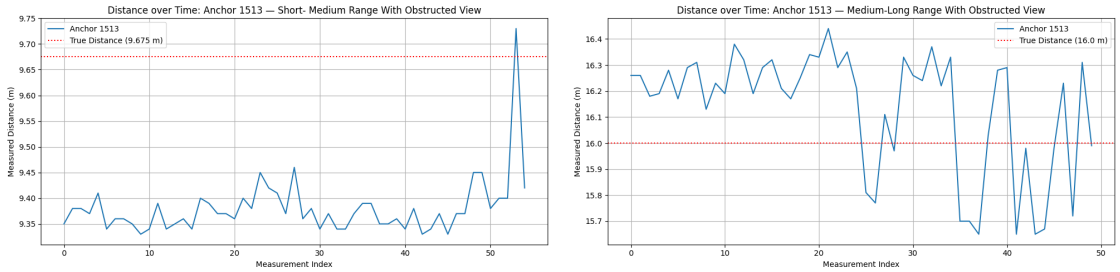

Figure 4.8: UWB Range test Short-Medium and Long-Medium with obstructed view

Introducing more "harder terrain" the Figure 4.8 shows the UWB ranging performance under partial line-of-sight obstruction. In the short–medium range test (left), the true distance was 9.675 meters. The measured values remained mostly below the ground truth, with a consistent negative offset of about 30-40 cm and generally low variation. A single outlier appears near the end of the measurement series.

In the medium–long range test (right), the effect of obstruction is more pronounced. The true distance was 16.0 meters, but the measured values show both higher variance and stronger deviations, ranging from slight overestimation to underestimations of more than 30 cm. Several measurements fluctuate rapidly, indicating unstable signal quality caused by the obstructed view.
Figure 4.9 shows the UWB-based position estimates using two anchors placed at known positions measured with RTK. Each anchor was used to project the estimated position of the moving device using its measured distance and an estimated angle toward a predefined target point.

The red and green circles indicate the fixed RTK-measured anchor positions, while the corresponding red and green triangles show the projected position estimates derived from UWB ranging and angle estimation. The cyan stars represent predefined reference points on the field.

The spread of projected positions from both anchors shows a general alignment along the correct direction but with visible deviation from the reference line. The distribution of points indicates some consistency, but with a moderate spread and offset relative to the true position. The pattern is similar for both anchors, suggesting symmetric measurement behavior.

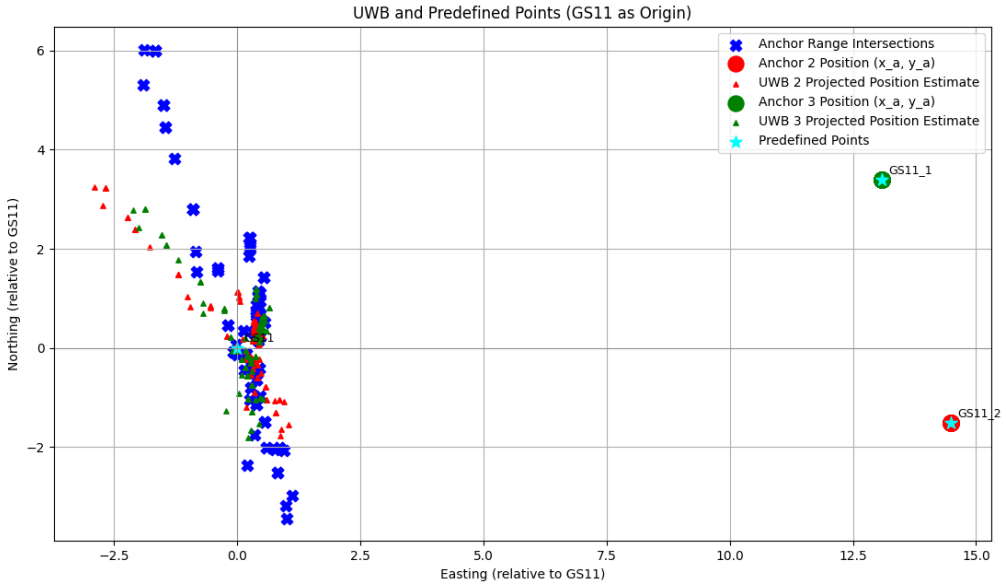

Figure 4.9: UWB positioning with intersection

Figure 4.9 also shows the intersection of range measurements from two anchors. Each blue cross represents a possible device position derived from solving the geometric intersection of two circles defined by the measured distances to each anchor.

Compared to the single-anchor projection, the intersection approach shows that using two anchors with RTT constrains the estimated positions more tightly along the region where the two range circles overlap. The blue crosses indicate possible positions based purely on the measured ranges, and show a narrower spread than the individual projections. However, the data still exhibit a noticeable offset and clusters around the true position.

4.5 Sensor Fusion Performance

4.5.1 Stationary tests

4.5.1.1 Residual 1 - angle and distance open area

A stationary sensor fusion test was performed in an open area using both GNSS and UWB measurements, as illustrated in Figure 4.10. The same GNSS module was mounted on one of the UWB anchors to estimate its position, which was then used in the UWB projection calculation.

The fusion filter in this test uses a measurement function based on angle and distance residuals. The black crosses represent raw GNSS measurements, while the red and green triangles show position estimates projected from each UWB anchor. The cyan star indicates the predefined NRTK reference location for this test.

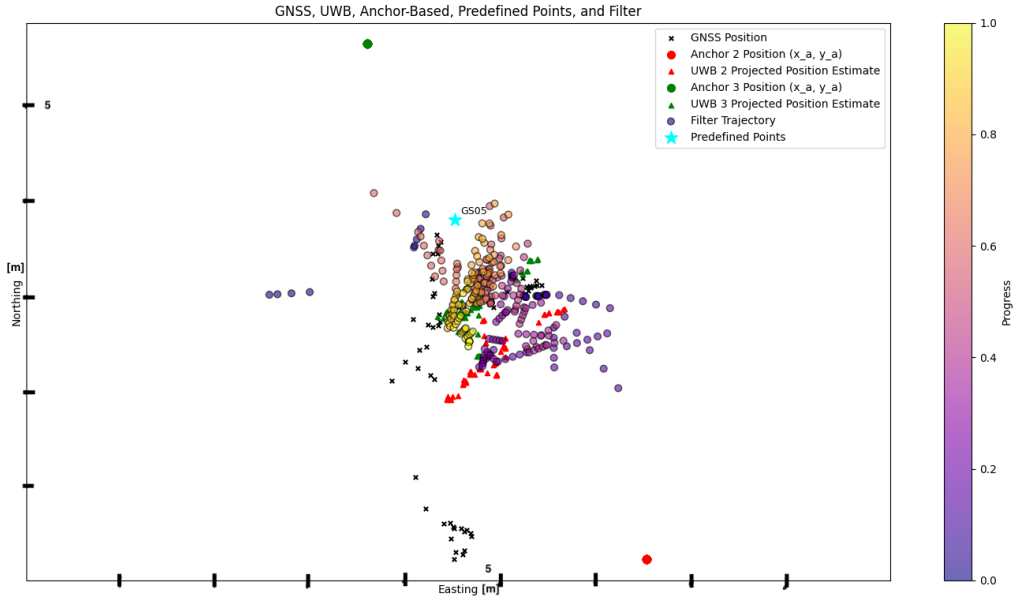

Figure 4.10: Open area, residual 1

The coloured dots represent the filter trajectory over time, with colour fading from purple (start) to yellow (end) to indicate progress. The fused position estimates begin with some spread but gradually stabilize and cluster closer to the reference area over time.

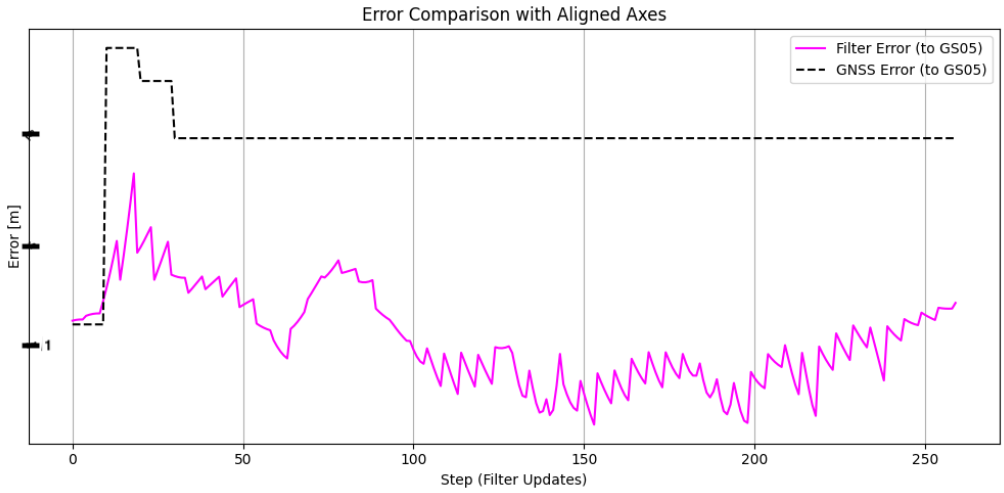

Figure 4.11: Open area total error, residual 1

The development of position error throughout the stationary open-area test is shown in Figure 4.11. This plot complements the previous fusion result in Figure 4.10 by quantifying the Euclidean distance to the predefined NRTK reference point at each update step.

The magenta line represents the error of the fusion filter's estimated position, while the black dashed line shows the static error level of the raw GNSS measurements

relative to the same reference. The filter error initially increases, then stabilizes and decreases as the filter converges, eventually maintaining a lower error than the standalone GNSS.

This plot highlights the temporal behavior of the fusion filter and provides a direct comparison of performance relative to uncorrected GNSS data.

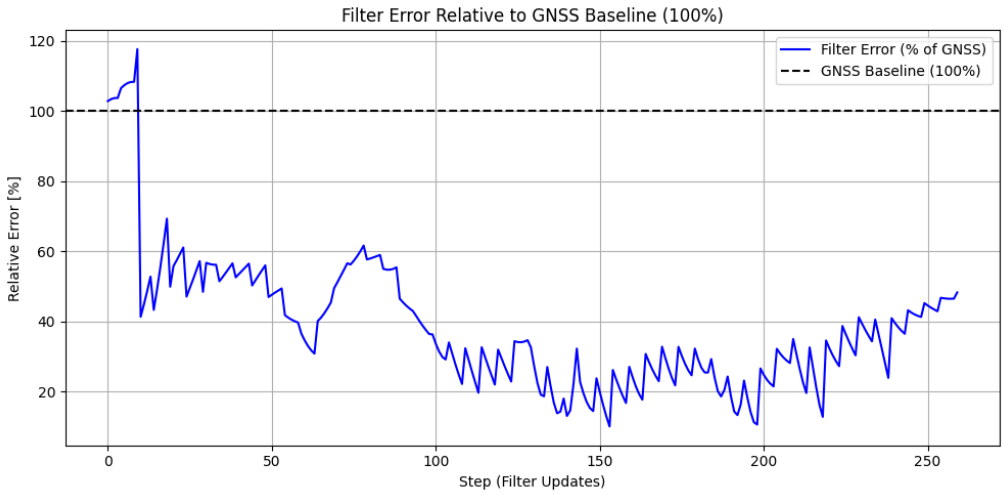

Figure 4.12: Open area error percent, residual 1

To better visualize the filter performance relative to GNSS, Figure 4.12 presents the filter error as a percentage of the static GNSS baseline error. The black dashed line marks 100%, representing the GNSS error used as a reference.

The blue line shows how the filter's error compares over time. After an initial rise, the relative error drops below 100% and continues decreasing, eventually stabilizing between 30% and 50% for a significant portion of the test. The filter error remains below the GNSS baseline for the majority of the measurement period, as indicated by values consistently under the 100% reference line.

4.5.1.2 Residual 2 - projected UWB position open area

A second stationary fusion test was run in the open area using a different measurement function based on projected UWB positions (residual 2). Figure 4.13 shows the resulting filter trajectory, with progress colored from purple to yellow.

The black crosses and triangle markers show the GNSS and UWB projections as before, but in this test, the filter uses the directly projected UWB positions instead of angle and distance as residuals. The trajectory remains centered near the reference area, with a visibly different evolution compared to the first residual.

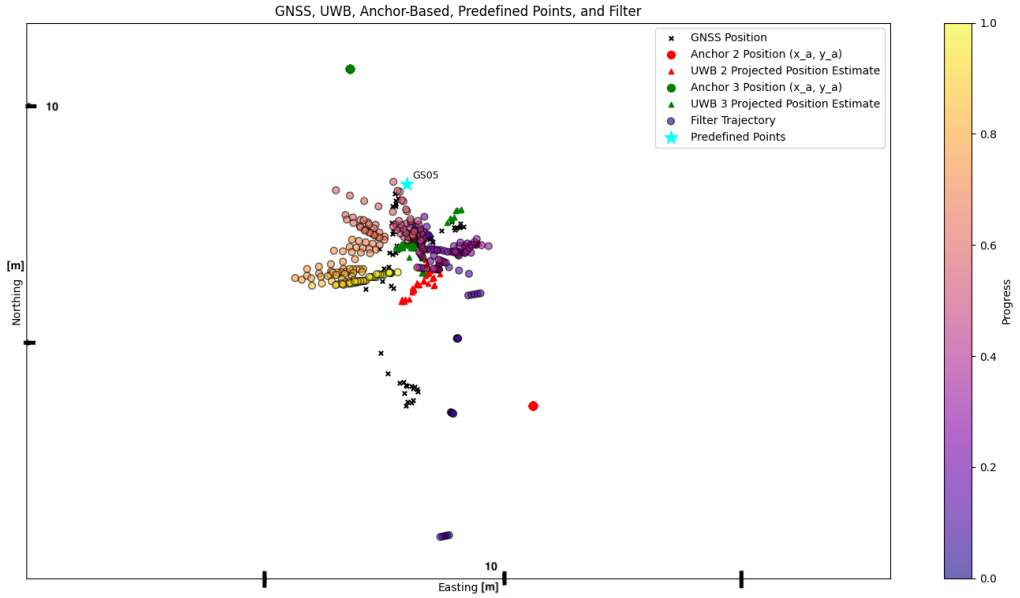

Figure 4.13: Open area, residual 2

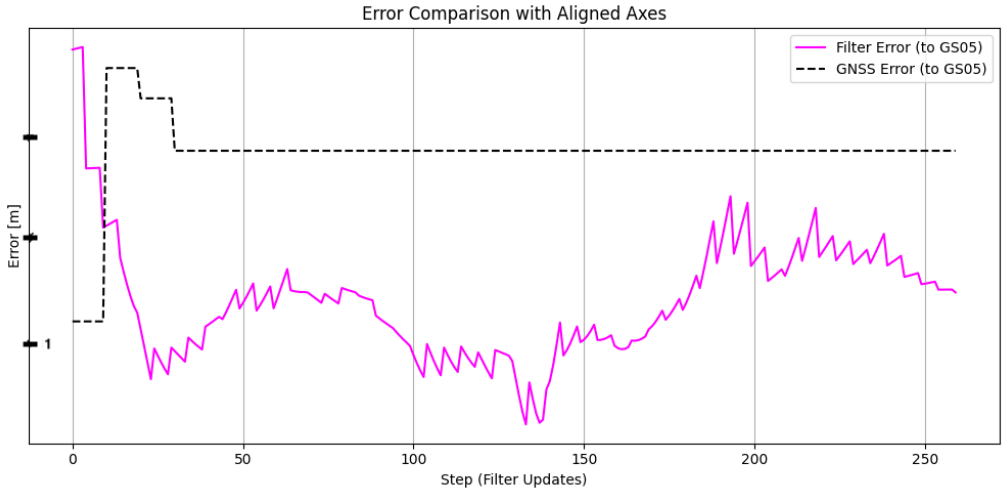

Figure 4.14: Open area total error, residual 2

Figure 4.14 presents the total positioning error for the filter using residual 2, as shown in the corresponding trajectory in Figure 4.13. The error is computed as the Euclidean distance to the NRTK-defined reference point.

The magenta line tracks the filter error at each update step, while the dashed black line indicates the static GNSS baseline. The filter error varies over time, initially decreasing and later oscillating around a lower level than the GNSS reference throughout most of the test duration.

To complement the absolute error view in Figure 4.14, Figure 4.15 shows the filter error using residual 2 expressed as a percentage of the GNSS baseline. The dashed black line at 100% represents the average GNSS error.

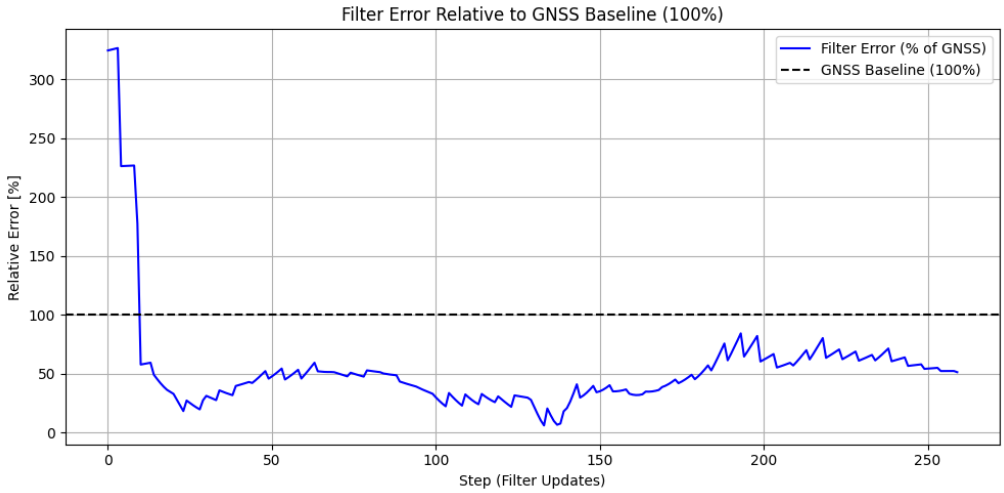

Figure 4.15: Open area error percent, residual 2

The blue line shows how the filter's relative error changes over time. After a brief initial peak, the error drops rapidly and remains well below the GNSS reference for the majority of the test. Throughout most of the updates, the filter error stays between 20% and 70% of the GNSS error level.

4.5.1.3 Forest area results

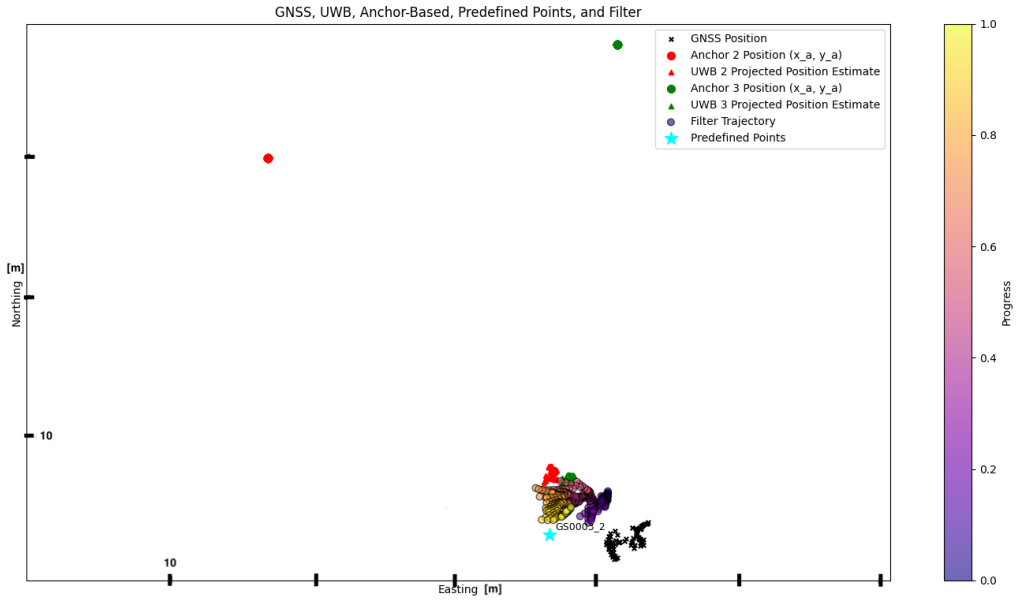

Figure 4.16: Forest filter

A stationary sensor fusion test was also conducted in a forest environment with moderate vegetation. Figure 4.16 shows the resulting filter trajectory using GNSS and UWB measurements, with color indicating progress from purple (start) to yellow (end).

As in previous tests, raw GNSS positions are shown as black crosses, while red and green triangles represent projected UWB estimates based on anchor positions. The predefined reference point is marked in cyan. The UWB anchors were placed in a slightly more open area, and their positions were measured using the same standard GNSS receiver as used for the moving unit.

Despite the reduced signal quality typical in forested areas, the filter trajectory remains centered around the reference location and shows a compact cluster of estimates with relatively limited spread.

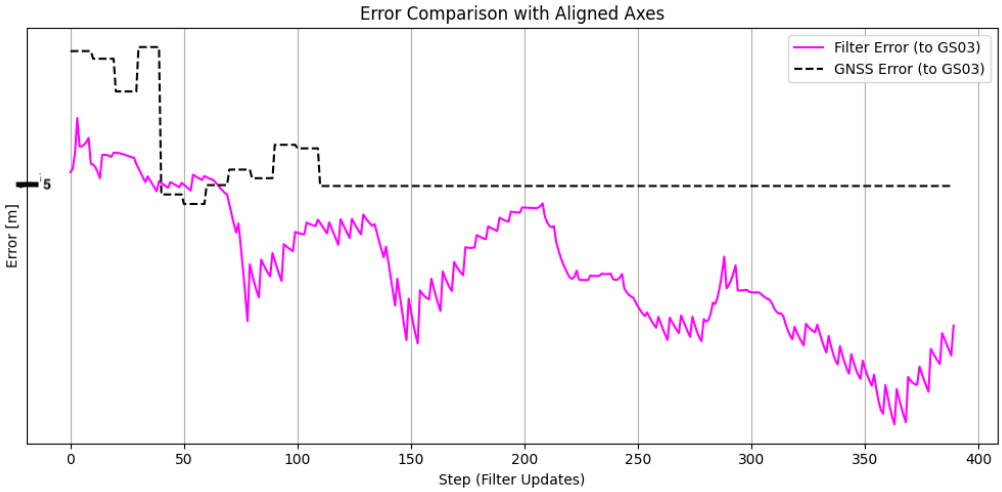

Figure 4.17: Forest total error

Figure 4.17 presents the absolute positioning error over time from the stationary fusion test in the forest environment. The error is calculated as the Euclidean distance between the filter's estimated position and the predefined NRTK reference point.

The magenta line shows the filter error across update steps, while the black dashed line represents the average GNSS error for comparison. The filter error fluctuates but generally stays below the GNSS baseline throughout the test.

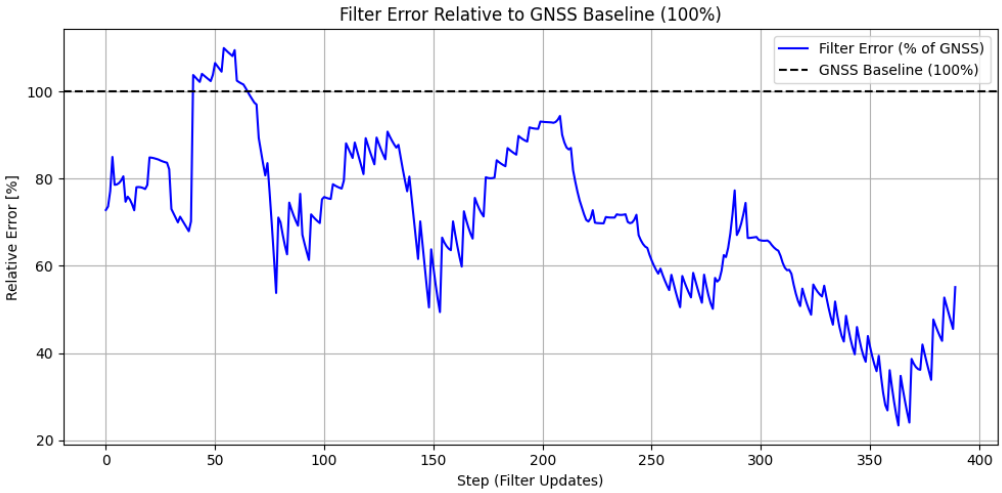

Figure 4.18: Forest error percent

To compare the filter performance relative to GNSS in percentage terms, Figure 4.18 shows the filter error as a percentage of the GNSS baseline. The dashed black line at 100% marks the GNSS error reference.

The blue curve illustrates how the filter error evolves over time. After initial variation, the error remains below the GNSS baseline for most of the measurement period, often fluctuating between 50% and 90% of the reference level.

4.5.1.4 Buildings area results

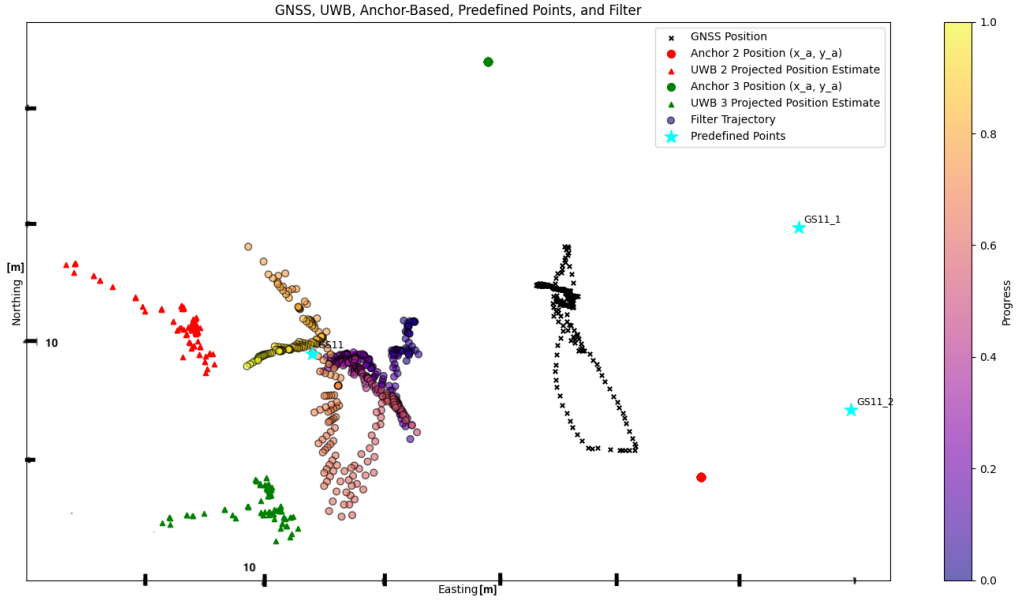

Figure 4.19: Buildings filter

A final stationary fusion test was performed near a building wall to simulate a signal-obstructed environment. Figure 4.19 shows the filter trajectory alongside GNSS and

UWB measurements, with color progression from purple to yellow representing time.

The receiver was placed close to the building and moved slightly back and forth along the wall during the test. The UWB anchors were positioned in adjacent open areas, and their coordinates were measured using the same standard GNSS receiver applied throughout the study. The predefined points GS11_1 and GS11_2 mark the true locations of the UWB anchors in this setup. Raw GNSS positions (black crosses) exhibit a large offset and erratic pattern, while UWB projections (red and green triangles) remain closer to the expected region. The filter manages to stabilize toward the predefined reference zone, although with more spread and variation compared to open or forested environments.

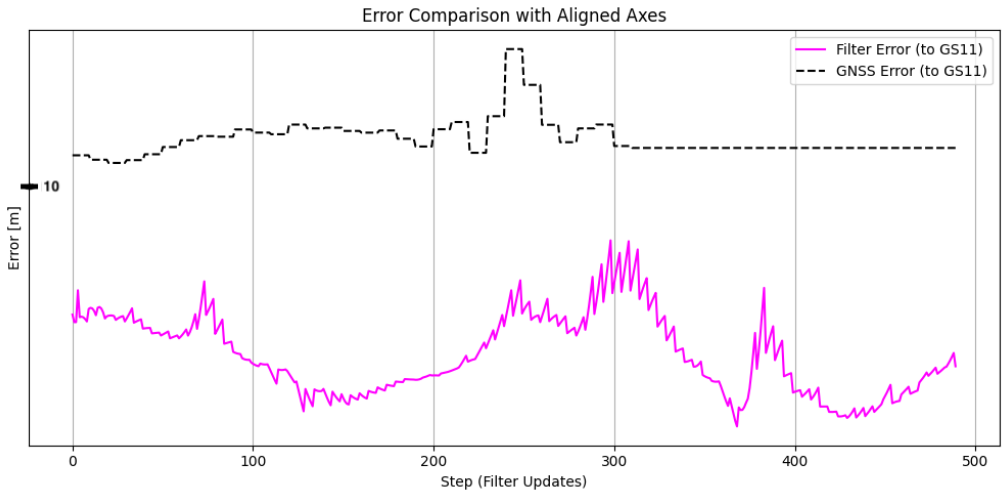

Figure 4.20: Buildings filter total error

Figure 4.20 presents the absolute positioning error from the building-side test, showing both filter error (magenta line) and GNSS baseline error (black dashed line).

The filter error fluctuates across the test but consistently remains lower than the GNSS error for most of the update steps, despite the challenging environment and slight motion of the receiver during the test.

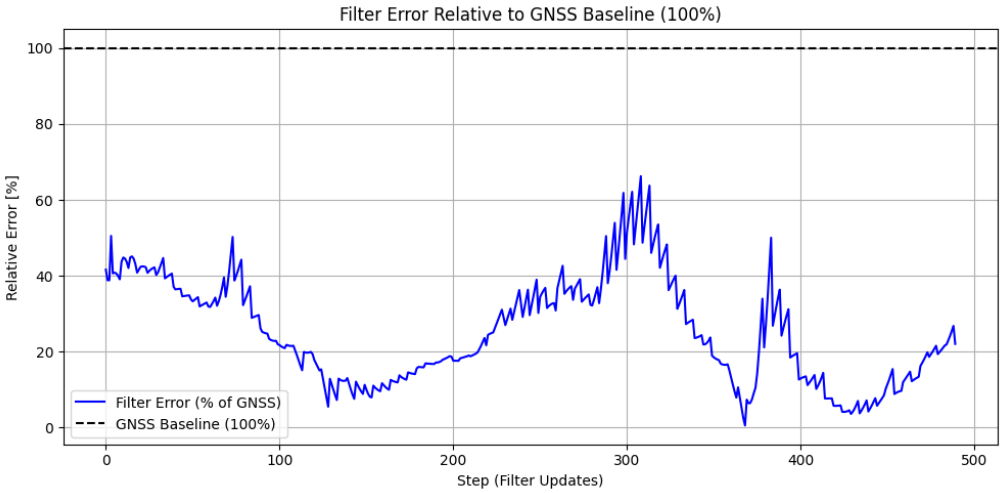

Figure 4.21: Buildings filter percentage error

To express the results relative to GNSS performance, Figure 4.21 shows the filter error as a percentage of the GNSS baseline. The blue line remains below the 100% reference line for the entire test duration.

Despite being positioned in a partially obstructed area and undergoing minor movement, the filter output maintained error levels mostly under 60% of the GNSS baseline.

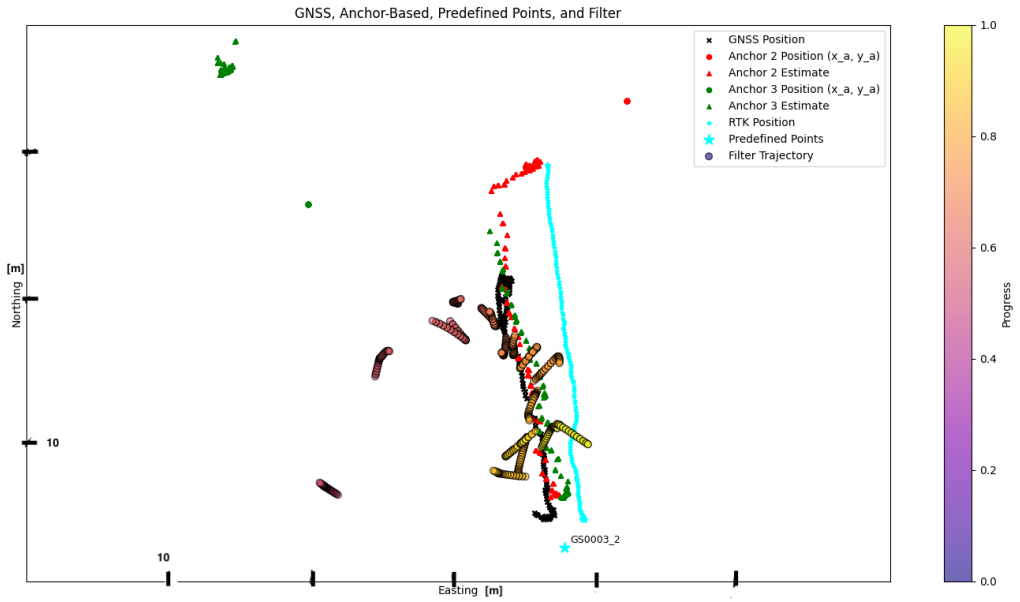

Figure 4.22: Dynamic forest test GNSS

4.5.2 Dynamic test

The first dynamic test was conducted in a forested area while walking along a predefined path. Figure 4.22 presents the results using GNSS data only. The filter

trajectory (colored points) drifts away from the RTK path (cyan) due to GNSS noise and limited accuracy in partially obstructed conditions. Anchor projections and predefined point GS0003_2 are also shown for reference, but were not used in this configuration.

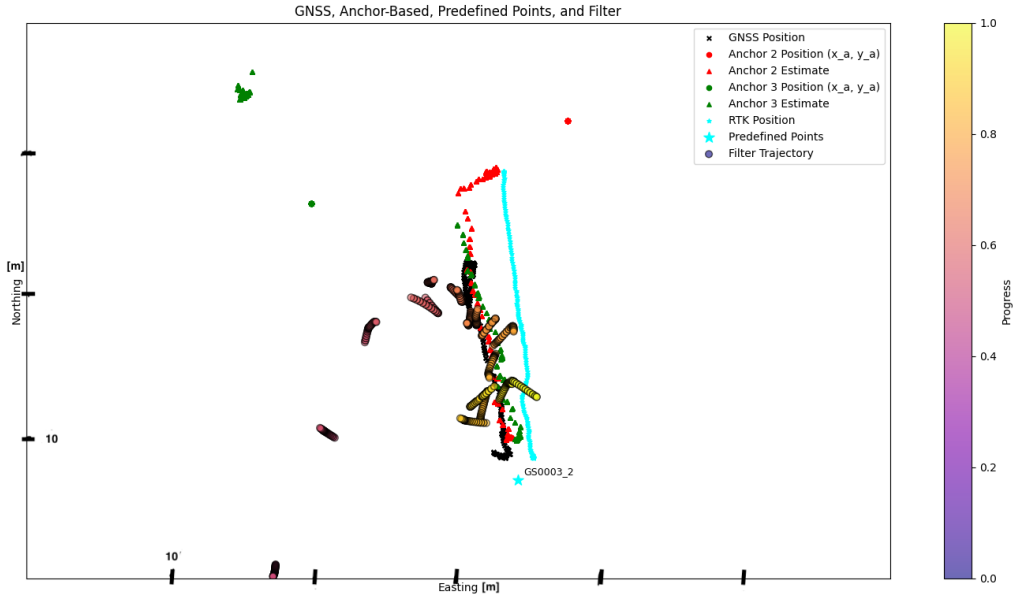

Figure 4.23: Dynamic forest test GNSS and IMU

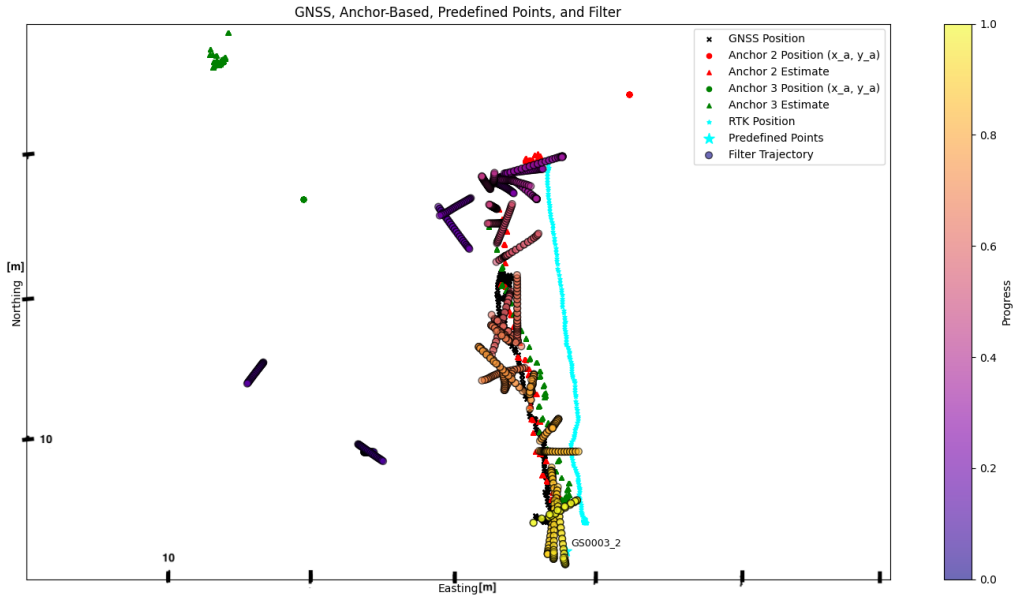

Figure 4.24: Dynamic forest test all sensors, Residual 1

In the second setup, GNSS data was fused with IMU measurements to improve short-term motion tracking. Figure 4.23 illustrates the resulting trajectory. The addition of IMU data leads to a more stable and smoother estimate, especially in segments where GNSS data quality degrades. The filter follows the intended path more closely, though some deviation from the RTK reference still occurs.

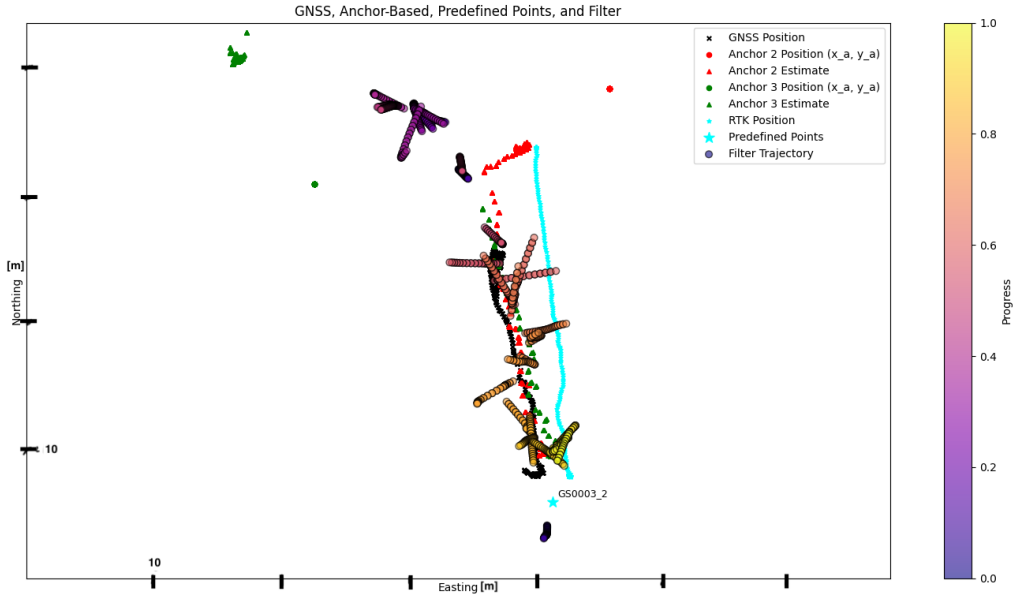

Figure 4.25: Dynamic forest test all sensors, residual 2

Figure 4.24 shows the final result using all available sensor inputs: GNSS, IMU, and UWB, with the measurement function using angle and distance residuals (Residual 1). The filter trajectory aligns more tightly with the RTK path, especially through GNSS-challenged zones. UWB estimates (red and green triangles) provide complementary local positioning data that help correct drift, while the IMU maintains continuity between measurements. Initially, some projected positions from one anchor appear offset in the wrong direction, but as the filter progresses, the projections improve and align more accurately with the expected path.
The final dynamic test result in Figure 4.25 includes all sensors, GNSS, IMU, and UWB, using the projected UWB positions as the measurement function (Residual 2). Compared to the previous filter configuration with angle and distance residuals, this trajectory shows a more dispersed pattern, particularly during the early and middle segments of the walk.

Although the filter output is more scattered, it still generally follows the overall trajectory defined by the RTK path. The progression from purple to yellow indicates a forward motion consistent with the test, and the filter maintains coverage around the intended route. The increased spread reflects greater variability in the projected UWB inputs, but the result remains aligned with the main motion path.

4.6 Result Summary

This chapter presented results from both static and dynamic positioning tests across various environments, using different combinations of GNSS, IMU, and UWB sensors. The GNSS-only tests established a performance baseline, while IMU and UWB results illustrated the limitations and strengths of each technology in isolation. Sensor fusion was then applied using two different measurement functions,

showing improvement in both accuracy and trajectory stability, particularly under degraded GNSS conditions. Tests were conducted in open, forested, and obstructed environments to evaluate real-world performance.

4.7 Limitations Observed During Testing

Several limitations affected the testing scope and balance of sensor configurations in this project.

Although both Residual 1 (angle and distance) and Residual 2 (projected position) measurement functions were implemented early in the development process, only two test cases were evaluated using Residual 2. This was due to performance trade-offs observed during filter tuning, where Residual 1 showed better results after improvements in angle estimation. Given time constraints, most testing and evaluation focused on Residual 1.

Dynamic testing was limited to a single environment—the forest scenario. Additional dynamic tests in open or obstructed areas were initially planned but could not be completed due to time limitations and issues with the GNSS hardware used for data collection. These problems affected both positioning stability and logging consistency during movement.

All sensor fusion results were produced using post-processed logs. Real-time implementation and tuning were outside the scope of this thesis, which limited feedback during data collection and constrained adaptive calibration.

5
Discussion and Future Work

5.1 Summary of Findings

5.1.1 GNSS

In areas with more vegetation, as seen in the forest test in Figure 4.2, the GNSS module performs significantly worse. The data shows greater drift and variation, likely due to signal loss, multipath effects, and other environmental disturbances. While correction methods might improve accuracy slightly in these conditions, they are unlikely to reduce the error to the centimeter or millimeter level required for high-precision applications.

In obstructed urban scenarios, shown in Figure 4.3, the performance degrades further compared to open or forested areas. Operating close to buildings — for example, about one meter from a wall and surrounded by other structures — the GNSS-only data show systematic offsets of roughly 10–15 meters with consistent clustering in one direction. This spread and bias are mainly caused by signal blockage and reflections from surrounding surfaces, which shift the apparent position and increase measurement noise. This result highlights that in dense urban environments, standard GNSS without additional correction or fusion cannot reliably deliver the accuracy needed for precision applications.
Overall, the findings indicate that a standard GNSS receiver cannot be relied on alone for precision positioning, as it generally cannot deliver better than 1–2 meter accuracy without external correction support — and can show much larger errors in obstructed or GNSS-challenged conditions.

5.1.2 IMU

When evaluating the IMU in standalone dead reckoning tests—such as the line test and the circle test shown in Figures 4.4 and 4.5, it becomes clear that the IMU cannot reliably estimate position on its own by simply integrating acceleration data. Without a global reference or position correction, the estimation quickly drifts due to sensor noise and bias.

In addition, the IMU requires a reliable heading estimate to project acceleration into horizontal motion. While it is possible to estimate velocity by integrating acceleration over time, changes in heading are much harder to detect from acceleration alone, especially during gradual or rotational motion. For example, in the

circle test, the acceleration magnitude remained relatively consistent, but without accurate heading input, the integrated position quickly diverged from the intended path. This illustrates that even if velocity is roughly correct, incorrect heading leads to substantial position error.

In human movement, this challenge is further amplified. Unlike a wheeled vehicle that typically moves in the direction it faces, a person can rotate or sidestep without a corresponding change in forward acceleration. As a result, heading estimation becomes a critical weakness in IMU-only positioning for pedestrian applications. If a reliable heading estimate were available continuously, the IMU-based position would likely follow the intended path more closely. Therefore, meaningful use of IMU data in such scenarios would require additional mechanisms, such as magnetic correction, step orientation detection, or external sensor fusion to maintain directional awareness.

5.1.3 UWB

The UWB system performs well as a local positioning technology, with accuracy consistently within the decimeter range, as demonstrated in Section 4.4. This makes it highly effective in short-range applications where GNSS performance is degraded or unavailable. One of the key limitations of UWB, however, is its reliance on accurately placed anchors. As shown in Figure **??** shows that when the anchors have well-defined positions, UWB can project positions with relatively high accuracy.

The follow-up intersection results in Figure 4.9 show that using two anchors with RTT measurements further constrains the possible positions to the overlapping region of the range circles, which reduces geometric ambiguity compared to the single projection method. However, the remaining spread and offset demonstrate that even small range measurement noise can lead to significant shifts in the intersection point, resulting in clusters and drift relative to the true position.

This approach still relies on having at least some devices in the system with precise absolute positions, typically provided by GNSS or another external reference. In scenarios where all units are mobile or no absolute reference is available, UWB alone cannot provide globally meaningful coordinates and instead acts as a relative positioning system. In that sense, if all devices already had accurate absolute positions, UWB would offer limited added value for standalone positioning. Nonetheless, it remains highly effective for improving short-term local accuracy, especially in obstructed or GNSS-denied environments, and serves as a valuable constraint in sensor fusion frameworks.

5.2 Sensor Fusion Performance

The main focus of the thesis was to improve position estimation using sensor fusion between the sensors described in the sections above. Overall, the fusion approach improved the position estimate in all tested scenarios. In the open area test case,

where both the anchors and the rover used the same standard GNSS receiver, the filter reduced the positioning error by almost 50% during the test period, as seen in Figure 4.12 (Open area error percent, Residual 1). This improvement is likely due to the drift observed in the GNSS measurements, which the filter was able to suppress by incorporating local UWB-based distance and angle estimates.

In the second test, conducted in a forest with light vegetation, the filter also improved the estimated position, though not to the same extent as in the open area. After a short convergence period, the filter maintained an error level below 60%, as shown in Figure 4.18 (Forest error percent). The fusion output remained in a consistent area and exhibited significantly less drift compared to the standard GNSS receiver.

The final stationary test, performed near buildings in an obstructed environment, revealed the limitations of GNSS in more complex scenarios, where multipath effects play a significant role. In this case, the filter helped stabilize the position estimate and maintain it relatively close to the reference location. As shown in Figure 4.21 (Buildings filter percentage error), the error often remained below 40% compared to the raw GNSS error.

In the dynamic test scenario, the filter's performance was more affected by heading estimation, as discussed earlier in the IMU section. With GNSS only, or GNSS combined with IMU, the filter tended to drift during the prediction stages. However, when UWB data was included, particularly in the Residual 1 configuration—the filter produced a smoother and more reliable estimate of the position.

Comparing Residual 1 and Residual 2 in the filter, Residual 1, which uses estimated angle and distance as input, consistently performed better than Residual 2, which projects a full position using distance and angle. This difference may be due to how the filter handles the estimated variances associated with each residual formulation and how it weighs the measurement updates. If the UWB system had supported direct Angle of Arrival (AoA) or Time Difference of Arrival (TDoA) measurements, the angle estimates used in Residual 1 could have been more accurate, potentially improving the overall filter performance even further.

5.3 Future Work

Several potential improvements and extensions can be explored in future work to enhance the system's positioning performance and reliability:

- **GNSS Enhancement:** The current setup relies on standard GNSS without correction services. Future implementations could include support for DGNSS or real-time correction methods such as RTK or PPP using dual-frequency L-band GNSS receivers, which could significantly reduce positioning error, especially in open environments.
- **Improved Heading Estimation:** Heading estimation played a critical role in fusion accuracy. Instead of relying solely on GNSS-derived headings, fu-

ture systems could estimate heading using IMU orientation data fused with UWB directionality or relative movement history. This would be particularly beneficial in low-speed or dynamic human motion scenarios.
- **Pedestrian Motion Modeling:** A basic step detection and direction estimation algorithm could be implemented using IMU data to improve dead reckoning performance during walking. By estimating step frequency and stride direction, the system could maintain more accurate position updates during GNSS dropouts or low UWB coverage.
- **Real-Time Integration:** This study relied on post-processing logged data. A future goal could be the implementation of the full sensor fusion system in real-time, allowing for live evaluation, feedback, and adaptive filter tuning during operation.

5.4 Conclusion

This thesis investigated the feasibility of improving positioning accuracy using sensor fusion between GNSS, IMU, and UWB in outdoor environments with varying levels of signal obstruction. Tests were conducted in open, forested, and obstructed scenarios to evaluate the strengths and limitations of each sensor, both individually and in combination.

The results showed that standard GNSS receivers offer limited accuracy, typically around 1–2 meters in open areas, and degrade significantly in forested or obstructed conditions. IMU data alone was not sufficient for position estimation due to integration drift and unreliable heading, especially during human motion. UWB demonstrated high accuracy in short-range local positioning but required fixed, accurately known anchor positions to be effective.

Sensor fusion significantly improved positioning performance across all environments. The fusion filter reduced GNSS error by up to 50% in open conditions and maintained greater stability in forested and obstructed tests. In dynamic testing, the filter performed best when GNSS, IMU, and UWB data were combined, particularly using Residual 1, which relied on distance and angle rather than full position projections. This configuration consistently delivered the most reliable and stable position estimates.

Overall, the study confirms that combining low-cost sensors through fusion can substantially enhance positioning accuracy and robustness in realistic, GNSS-challenged environments.

Bibliography

[1] Maciuk, K. (2018). The applications of GNSS systems in logistics. *Budownictwo i Architektura*, 17(3), 181–188.

[2] ArduSimple. (n.d.). RTK Explained – Everything You Didn't Dare To Ask. Retrieved from `https://www.ardusimple.com/rtk-explained/`

[3] Global GPS Systems. (n.d.). RTK GPS: Understanding Real-Time Kinematic GPS Technology. Retrieved from `https://globalgpssystems.com/gnss/rtk-gps-understanding-real-time-kinematic-gps-technology/`

[4] Song, K., Li, J., Ren, M., & Liu, S. (2024). Indoor and outdoor fusion positioning system based on UWB/GNSS/IMU multi-sensor technology. In *Proceedings of the 2024 International Conference on Autonomous Driving and Intelligent Sensing Technology (ADIST)*, Zhuhai, China, June 28–30, 2024. ACM. https://doi.org/10.1145/3696021.3696030

[5] Retscher, G., Kiss, D., & Gabela, J. (2023). Fusion of GNSS pseudoranges with UWB ranges based on clustering and weighted least squares. *Sensors*, 23(6), 3303. https://doi.org/10.3390/s23063303

[6] Lantmäteriet. (n.d.). GPS och andra GNSS. Lantmäteriet.

[7] Lantmäteriet. (n.d.). GPS Lantmäteriet.

[8] Lantmäteriet. (n.d.).Felkällor vid GNSS-mätning Lantmäteriet.

[9] Lantmäteriet. (n.d.). Avståndsmätning med bärvåg Lantmäteriet.

[10] Blewitt, G. (n.d.). Basics of the GPS technique: Observation equations. Department of Geomatics, University of Newcastle.

[11] Lantmäteriet. (n.d.). Relativ kodmätning, DGNSS Lantmäteriet.

[12] Lantmäteriet. (n.d.).Relativ bärvågsmätning, RTK Lantmäteriet.

[13] Lantmäteriet. (n.d.).Andra GNSS-tekniker Lantmäteriet.

[14] Analog Devices. (2023). MEMS Inertial Measurement Units (IMUs): High Performance for High Precision. Analog Devices, Inc.

[15] Yang, Y., Zhang, J., Hu, J., & Yu, R. (2021). Lane Detection Aided Online Dead Reckoning for GNSS Denied Environments. *Sensors*, 21(21), 7334.

[16] Wikipedia contributors. (2024). Inertial measurement unit. Wikipedia.

[17] Sanket, N. J. (n.d.). Kalman Filter for Attitude Estimation. Retrieved from `https://nitinjsanket.github.io/tutorials/attitudeest/kf`

[18] Pozyx. (n.d.). Pozyx UWB RTLS. Pozyx Labs.

[19] Wikipedia contributors. (2024). Ultra-wideband. Wikipedia.

[20] Qorvo. (n.d.). Ultra-Wideband (UWB) – High precision, secure and real-time location technology. Qorvo, Inc.

[21] Dahl, J. (2013). Evaluation of an Ultra-Wideband system for positioning in harsh environments (Master's thesis, Luleå University of Technology).

Retrieved from https://www.diva-portal.org/smash/get/diva2:645575/FULLTEXT01.pdf
[22] Särkkä, S., & Svensson, L. (2023). *Bayesian Filtering and Smoothing* (2nd ed.). Cambridge University Press.
[23] Linköping University. (n.d.). Sensor Fusion.
[24] Addison, A. (n.d.). Extended Kalman Filter (EKF) with Python Code Example. Automatic Addison.
[25] u-blox. (2019). ZOE-M8Q: u-blox M8 GNSS SiP Module Data Sheet. u-blox AG.
[26] Molex. (2023). Molex 206560 GNSS Active Antenna Datasheet.
[27] Adafruit. (n.d.). Adafruit BNO055 Absolute Orientation Sensor.
[28] Adafruit. (n.d.). Adafruit CircuitPython BNO055 Library.
[29] Qorvo. (2018). MDEK1001 System User Manual, Version 1.1.
[30] Lantmäteriet. (n.d.). SWEPOS – Nationellt referensnät i Sverige.
[31] u-blox. (2022). UM981 – High Precision GNSS RTK Receiver. ArduSimple.
[32] SparkFun. (2020). GNSS Multi-Band L1/L2 Antenna – SPK6618H.

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY