**CZECH TECHNICAL UNIVERSITY IN PRAGUE**

**F3**

Faculty of Electrical Engineering
Department of Control Engineering

**Master's Thesis**

# Indoor localization system for automated vehicles based on Ultra-Wideband technology

**Bc. Jitka Hodná**
**Cybernetics and robotics**

**May 2021**

Datavision s. r. o.
Supervisor: Ing. Tomáš Novák

# MASTER'S THESIS ASSIGNMENT

## I. Personal and study details

Student's name: **Hodná  Jitka**                    Personal ID number: **439565**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Control Engineering**

Study program: **Cybernetics and Robotics**

Branch of study: **Cybernetics and Robotics**

## II. Master's thesis details

Master's thesis title in English:

**Indoor localization system for automated vehicles based on Ultra-Wideband technology**

Master's thesis title in Czech:

**Interiérový lokalizační systém pro autonomní prostředky s využitím technologie Ultra-Wideband**

Guidelines:

1. Study the state of the art data fusion principles used for pose estimation. Study principles of Inertial navigation systems (INS)
2. Propose a localization system for autonomous vehicles based on fusion of data from Ultra-Wideband (UWB) positioning system and on-board dead-reckoning sensors such as Inertial measurement unit (IMU)
3. Evaluate proposed localization system for use in industrial environments.

Bibliography / sources:

[1] THRUN, SEBASTIAN, WOLFRAM BURGARD, AND DIETER FOX - PROBABILISTIC ROBOTICS, 2005 - Massachusetts Institute of Technology, USA (2005)
[2] GREWAL, MOHINDER S., ANGUS P. ANDREWS, AND CHRIS G. BARTONE - GLOBAL NAVIGATION SATELLITE SYSTEMS, INERTIAL NAVIGATION, AND INTEGRATION - John Wiley & Sons, 2020
[3] KELLY, ALONZO - MOBILE ROBOTICS: MATHEMATICS, MODELS, AND METHODS - Cambridge University Press, 2013
[4] MOORE, THOMAS, AND DANIEL STOUCH - A GENERALIZED EXTENDED KALMAN FILTER IMPLEMENTATION FOR THE ROBOT OPERATING SYSTEM, Intelligent autonomous systems 13. Springer, Cham, 2016. 335-348
[5] HOL, JEROEN D., et al. - TIGHTLY COUPLED UWB/IMU POSE ESTIMATION, 2009 IEEE international conference on ultra-wideband. IEEE, 2009
[6] LI, JIAXIN, ET AL. - ACCURATE 3D LOCALIZATION FOR MAV SWARMS BY UWB AND IMU FUSION, 2018 IEEE 14th International Conference on Control and Automation (ICCA). IEEE, 2018.

Name and workplace of master's thesis supervisor:

**Ing. Tomáš Novák,   Datavision s.r.o.**

Name and workplace of second master's thesis supervisor or consultant:

**Ing. Martin Hlinovský, Ph.D.,   Department of Control Engineering,   FEE**

Date of master's thesis assignment: **15.01.2021**          Deadline for master's thesis submission: **13.08.2021**

Assignment valid until:
**by the end of winter semester 2022/2023**

_____          _____          _____
Ing. Tomáš Novák                prof. Ing. Michael Šebek, DrSc.        prof. Mgr. Petr Páta, Ph.D.
Supervisor's signature          Head of department's signature          Dean's signature

# Acknowledgement / Declaration

I hereby declare that I wrote the presented thesis on my own and that I cited all the used information sources in compliance with the Methodical instructions about the ethical principles for writing an academic thesis.

..........................................

Prague, May 21, 2021

# Abstrakt / Abstract

Tato práce představuje inovativní přístup v oblasti interiérové lokalizace pro autonomní pozemní vozidla se zaměřením na její dlouhodobou přesnost a spolehlivost.

Systém je založený na fúzi ultra wide-band technologie, inerciální měřící jednotky a odometrie. Realizace je uskutečněna jako inerciální navigační systém zpětnovazebně řízený rozšířeným Kalmanovovým filtrem s oddylkovým modelem.

Navržený systém je implementován na robotickou platformu CART2 a experimentálně ověřen na několika testovacích scénárií ve dvou různých vnitřních prostředích. Výsledky ukazují, že systém dosahuje přesnosti v pozici až 10 cm a v orientaci jednotek stupňů. Systém splňuje požadavky pro jeho využití v praktických nasazeních, kde je vyžadována vysoká přesnost a spolehlivost.

**Klíčová slova:** ultra-wideband, imu, ins, localizace, vnitřní prostředí, ekf, rozšířený kalmanův filtr, odchylkový model rozšířeného kalmanova filtru

**Překlad titulu:** Interiérový lokalizační systém pro autonomní prostředky s využitím technologie Ultra-Wideband

This thesis proposes a novel approach in indoor real-time localization of autonomous ground vehicles and aims to ensure its accuracy, repeatability and reliability in the long term.

The system is based on Ultra Wide-Band technology, an Inertial Measurement Unit and odometry. An integration scheme using an Error state Extended Kalman filter in a closed-loop fashion is employed for the realization of this approach.

Evaluation is based on real data obtained from various testing scenarios in two different environments using the proposed aided inertial navigation system integrated on the CART2 robotic platform. The results show that the system can achieve a positioning precision within 10 cm and a few degrees in attitude. The system meets the requirements for practical conditions and seems viable for indoor localization with high accuracy and precision demands.

**Keywords:** ultra-wideband, imu, ins, localization, indoor, kalman-filter, ekf, indoor localization, error state extended kalman filter

# Contents /

# Tables / Figures

# Chapter 1
## Introduction

Indoor real-time localization is a crucial component in autonomous mobile robotics, and nowadays, the interest for precise localization is growing due to the fourth industrial revolution influencing all industries. There are existing approaches and technologies to deal with indoor localization in the industry but do not fully meet all of the requirements of the fourth industrial revolution.

One of the requirements is to keep as few manual interventions as possible for the technology to work reliably for a long period of time. An example is autonomous ground vehicles (AGV) manufactured by the company Ceit in Škoda factories, which are localized by continuous magnetic tapes physically mounted on the factory's ground. The AGV follows these magnetic tapes, which often can be damaged by moving AGVs or people and can not be autonomously replaced or modified.

Another possible requirement is the modularity of the factory environment. Today's trend is to create a factory that consists of modular parts, and the entire production process is assembled according to its current needs. Therefore, approaches based on localization relying heavily on landmarks or contours from cameras or lidars may fail. These approaches must address long-term sustainability, and this topic is not straightforward and can lead to difficulties.

The demand is also for a localization that will be universally applicable to various robotic platforms, whether ground vehicles of different shapes and constructions or flying drones. These requirements are currently being met in the outdoor environment by the Global Navigation Satellite System (GNSS), which unfortunately is not suitable for use in indoor environments. The use of external beacons seems to be a reasonable solution because it has similar properties as GNSS. These beacons can be based, for example, on ultrasound or radio waves as ultra wide-band or Wi-Fi.

A single technology cannot meet all of these requirements, but an appropriate fusion of carefully chosen approaches can.

## 1.1 Aims and requirements

This thesis is assigned by the Czech company Datavision s. r. o. and is a part of a project called Guidance and Localization upgrade creating Autonomous Mobile Robots. The abbreviation for this project is REX, and it is also used in this thesis.

REX aims to create fleet management of autonomous mobile robots, including their localization, control, navigation and planning. The project is co-financed by the Technology Agency of the Czech Republic (TACR) under the TREND Programm FW03010020 and aims to satisfy the fourth industrial revolution requirements.

This thesis aims to propose an indoor real-time localization system, which includes both the position and orientation of AGV. This aim is closely connected with the specification of localization in the REX system but is simplified into 2D. The fundamentals need to be applicable in 3D with a few modifications.

The essential aim is to design a localization system based on ultra wide-band technology and onboard dead reckoning sensors, which should improve the UWB localization itself.

## 1.2 Structure of the thesis

This thesis is organized into six major parts. The following Chapter 2 describes the fundamentals of the sensors used and the concept of inertial navigation systems. Chapter 3 discusses existing data fusion algorithms for pose estimation algorithms and their pros and cons. The proposed localization and its implementation are described in Chapter 4. In Chapter 5, the proposed localization is experimentally evaluated in two environments. The thesis is concluded in Chapter 6, where the usage of this system in the industrial environment is given, followed by a summary of the thesis output and a few proposals for improving and extending the work.

# Chapter 2
## Sensors

In Chapter 2, an overview of the sensors used and their properties is given. The main aim is to describe localization methods based on ultra-wideband technology(UWB), an inertial measurement unit (IMU) followed by inertial navigation systems (INS), and odometry.

The section dedicated to UWB localization briefly introduces the UWB signals follows with three most used localization techniques: Two-Way Ranging, Time Difference of Arrival, and Reverse Time Difference of Arrival.

The IMU section mainly focuses on the unit overview, a short description of gyroscopes and accelerometers used in it, the errors of these sensors, its analysis and outcomes for the localization unit. This section is closely related to the INS section, where principles of INS are explained and a strap-down implementation example is given.

The last section includes basic information about odometry and its benefits as well as which odometry is used during the experimental verification of the proposed localization system.

## 2.1 Localization based on ultra-wideband

**Ultra-wideband** is an emerging wireless personal area network (PAN) radio technology with a wide range of uses. The most promising usage in the field of robotics is accurate indoor localization because its benefits include

- high data rates,
- high time resolution,
- low power consumption,
- multipath immunity,
- low costs,
- small size
- and simultaneous ranging communication [1].

The UWB signal is defined as a signal with an absolute bandwidth ($B$) of at least 500 MHz, defined as

$$B = f_H - f_L, \tag{1}$$

where $f_H$ is the upper frequency and $f_L$ is the lower frequency, or with a fractional bandwidth of larger than 20 % given by

$$B_{frac} = \frac{B}{f_c} = \frac{B}{\frac{f_H+f_L}{2}}[2]. \tag{2}$$

Since the bandwidth of the signal is wide, the power spectral density is low. It means that even if the UWB signals share a spectrum with some narrowband signals

3

such as WiFi, it essentially behaves as environmental noise and does not affect any other narrowband signals much [2]. Also, since the length of each pulse is small, the possibility of overlapping the original pulse is reduced. Thus, it should be robust against multipath problem [2].

For precise communication, a direct line of sight should be established between the transmitter and the receiver. However, as UWB signals consist of many frequencies, some of them can reflect well off of some objects, while others can penetrate through them [2].

A few **localization techniques exist based on UWB** signals which are exchanged between a tag and several reference anchors with known positions [2]. The more accurate results in the line of sight environments are based on measuring the signal's time of flight from several devices, namely Two-Way Ranging (TWR), Time Difference of Arrival (TDoA) and Reverse Time Difference of Arrival (RTDoA). The overview of these technologies follows.

**Two-Way ranging** (TWR) is a simple method where the tag and anchors exchanges message in both ways. Thus, the updated rate of the tag's position is limited and decreased with a higher number of tags asking for their positions. The synchronization of the messages is provided with one anchor declared as an initiator of the network. The communication between the tag and anchors is illustrated in Figure 2.1. This method is the most used nowadays, as it is the simplest [2].
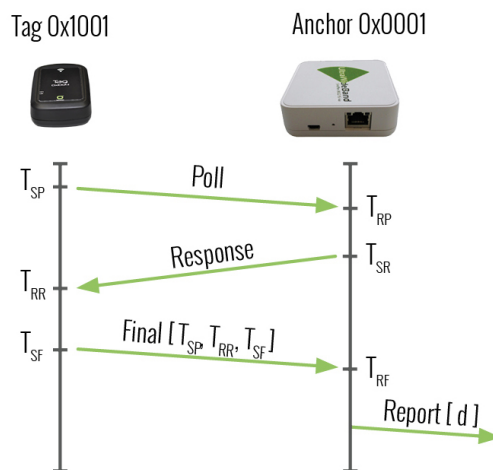


**Figure 2.1.** Two way ranging communication [3].

In **Time Difference of Arrival** (TDoA) technology, the tag only transmits information and the anchors only receive data. The tag sends a message to all available anchors. They estimate the time difference of the delivered messages, and the tag's position is calculated accordingly. With this technology, a higher update rate can be achieved even with more tags than TWR [2].

In **Reverse Time Difference of Arrival** (RTDoA), the tag listens and anchors transmit into the environment. The position is directly computed in the tag. This technology has no upper limit for tags and also promises the highest update rates. Thus, it is well suited for flying drones or for similar applications where this high update rate and low latency are needed [2]. This mode is comparable to the localization algorithm used in GNSS. Unfortunately, this technology is not yet commonly available on the market.

This article [4] proves the similarity of the usage of GNSS and UWB. The authors demonstrate the possibility of switching from GNSS to UWB while a pedestrian user moves from an outdoor space to an indoor environment.

Despite all of the mentioned benefits, in practice, UWB localization faces errors caused by surrounding factors, leading to coordinate jitters or outlying results [5].

## 2.2 Inertial measurement unit

An inertial measurement unit (IMU) is a device that utilizes measurement systems such as gyroscopes and accelerometers to estimate the relative position, velocity and acceleration of a vehicle in motion [6]. The unit is typically integrated with an onboard computational system and may contain more sensors acting as a magnetometer or thermometer.

The gyroscopes measure angular velocities and accelerometers specific forces which can be easily transformed into linear accelerations [6]. The IMU typically contains three orthogonal accelerometers and three orthogonal gyroscopes. Because of that, it can measure angular velocities and specific forces in each axis to maintain a 6-DOF estimate of the pose of the vehicle (position $(x, y, z)$ and orientation ($roll$, $pitch$, $yaw$). The process of the computation can be seen in Figure 2.2.
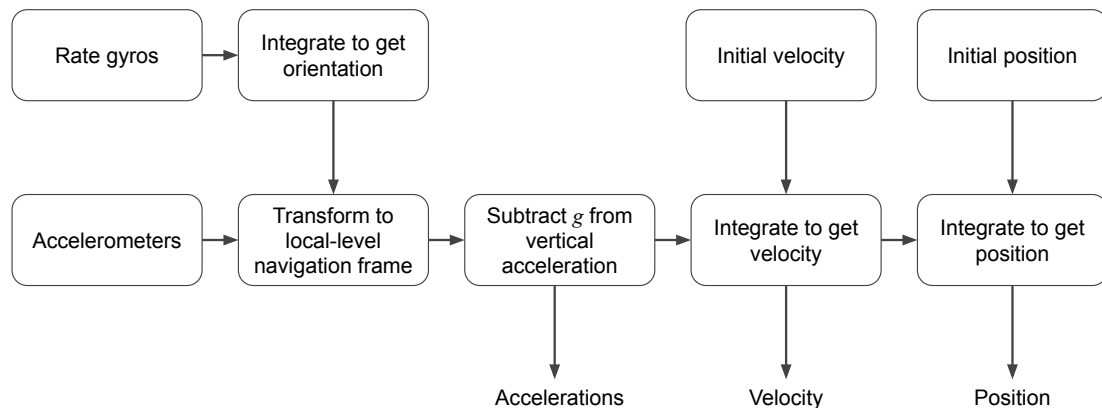


**Figure 2.2.** IMU block diagram [6].

There are two basic ways to mount the IMU to a vehicle, also called mechanization architectures [6–7].

- In **gimbaled systems**, the IMU is attached to a stabilized platform that maintains its inertial orientation as the vehicle maneuvers.
- In **strap-down systems**, it is rigidly attached to the vehicle.

The mechanization determines the conversion between measurements of IMU and estimation of linear accelerations and angular velocities of the vehicle. It means the transformation of the IMU body frame to the local frame. The conversion is closely related to inertial navigation systems described in Section 2.3.

IMU's are extremely sensitive to measurement errors. The way in which the gyroscopes and accelerometers are mounted, along with their properties, play a major factor in how accurate their results are. As the data are once or twice integrated, any error in measurement causes a linear or quadratic error in the pose estimation. Even with a small measurement error, the IMU's drift becomes significant, and it needs to

be externally compensated. The IMU provides a short-term stable solution, which is not affected by the external environment [7], and it has a high data rate (100 Hz - 200 Hz). That makes the IMU measurement complementary to the UWB localization measurement.

IMU's are extremely sensitive to measurement errors given by properties of used gyroscopes, accelerometers and their mounting. As the data are once or twice integrated, any error in measurement causes a linear or quadratic error in the pose estimation. Even with a small measurement error, the IMU's drift becomes significant and it needs to be externally compensated for. The IMU provides a short-term stable solution, which is not affected by the external environment [7] and it has a high data rate (100 Hz - 200 Hz). This is what makes the IMU measurement complementary to the UWB localization measurement.

### ◼ 2.2.1 Accelerometers

Accelerometers can measure external forces acting on the vehicle. They measure a specific force relatively to a non-rotating inertial space in a specific direction. They are sensitive to all forces, including gravity and fictitious forces [6].

**Mechanical accelerometers** use a spring-mass-damper system. When a force acts on the mass, it causes a displacement of the spring. The system is limited by the physical properties of the actual spring.

**Microelectromechanical systems (MEMS) based accelerometers** are made of at least three components, namely a proof mass, a suspension to hold the mass and a pickoff, which relays an output signal to the induced accelerations [8]. MEMS accelerometers are then classified by converting the mechanical displacement of the proof mass to an electrical signal. The most common principles belong to piezoresistive, capacitive sensing, piezoelectric, optical sensing and tunneling current sensing. Unlike the others, the piezoelectric MEMS sensors can not be used for navigation because their output rate is too low [8].

The current accelerometers use technology according to an application that is summarized in Figure 2.3.

### ◼ 2.2.2 Gyroscopes

Gyroscopes are used for estimating a rotational motion of a body. Each gyroscope measures angular rate $\omega$ (inertial angular rotation) relatively to a non-rotating inertial space in one axis. There are three main categories of gyroscopes [6].

**Mechanical gyroscopes** have a mass spinning steadily with respect to a free movable axis, they are not often used anymore, but they can be found in very old submarines.

**Optical gyroscopes** are based on the Sagnac effect, which states that frequency/phase shift between two waves counter-propagating in a rotating ring interferometer is proportional to the loop angular velocity. As a light source, a laser is typically used. Currently, this technology gives the best performance. Examples can be ring laser gyroscopes (RLG) or fibre optic gyroscopes (FOG).

**Vibrating gyroscopes** are based on the Coriolis effect that induces a coupling between two resonant modes of a mechanical resonator. Typically, vibrating gyroscopes are based on MEMS technology [8], and they play a significant role in robotics because of their simplicity. They are small, cheap, have no rotating parts and have low power consumption.

The performance and application of each technology are demonstrated in Figure 2.4.
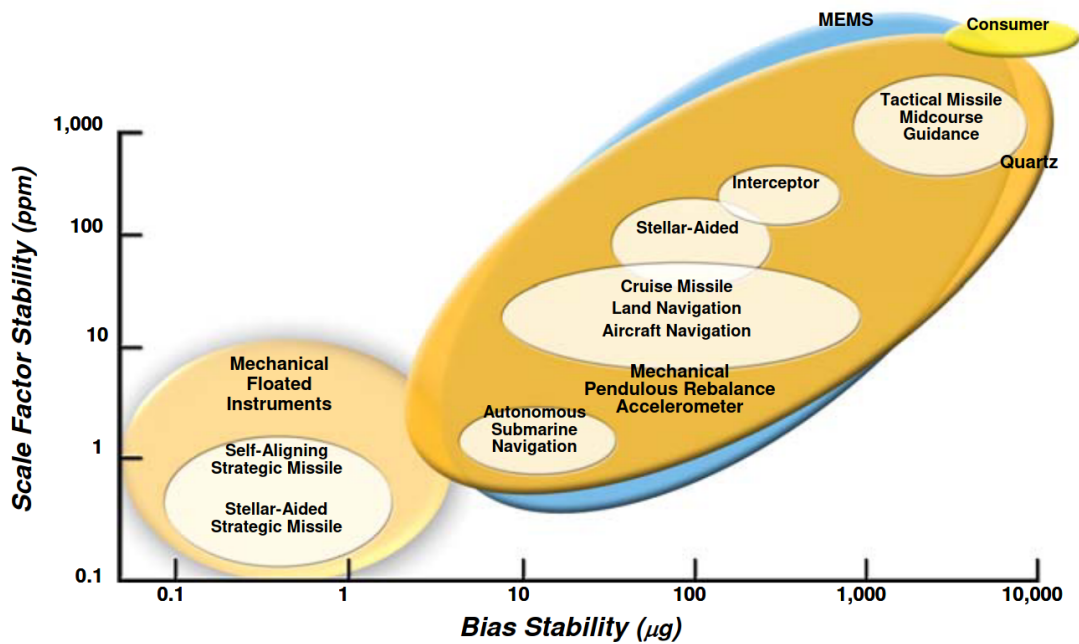
**Figure 2.3.** Accelerometers technology plotted by bias instability and scale factor stability [9].
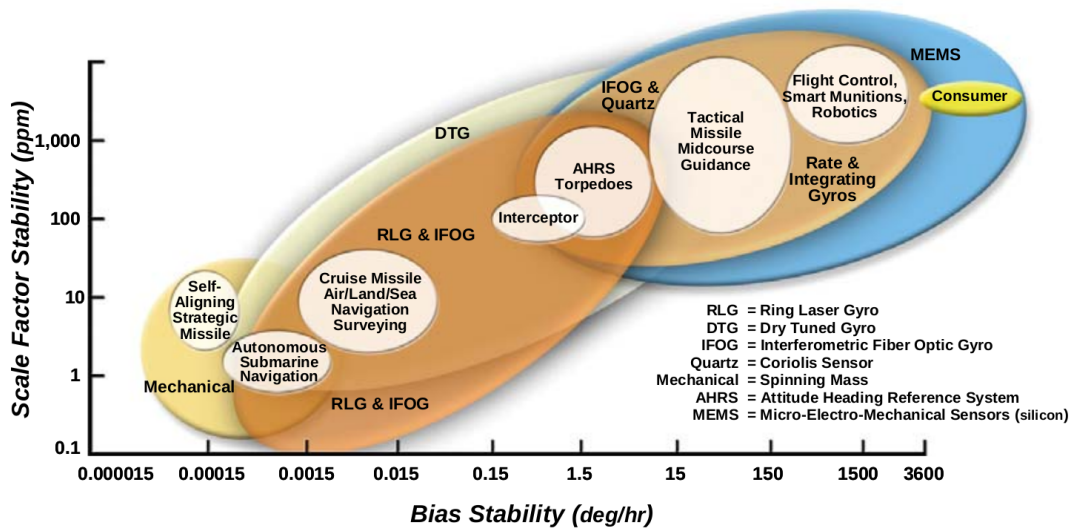


**Figure 2.4.** Gyroscopes technology plotted by bias instability and scale factor stability [9].

### ▣ 2.2.3   IMU's errors and Allan variance analysis

**IMU errors** IMUs faces several error sources which are always related to the specific sample unit and its technology. In this thesis, the main focus is given to MEMS-based IMUs as they are used in the experiments. These sensors are typically small and inexpensive. This section summarises the most significant errors for MEMS sensors and analyzes these errors with the Allan variance. This analysis is applied to a specific IMU which is subsequently used during the experiments.

Errors can be divided into two categories [8]

- stochastic errors, which can be described as random processes,
- and systematic errors, which are caused by manufacturing imperfections or handling issues with IMU. These errors can be corrected by proper calibration.

Nevertheless, errors need to be analyzed and reduced according to the application's requirements. The next following subset of errors is the most significant according to the topic of this thesis.

*Biases* of accelerometers and gyroscopes used in IMU are examples of systematic errors and can be divided into

- bias instability[1], which represents drift of the sensor over time,
- and initial bias[2] a static offset, which can vary when starting up the device each time, but during the run, it remains static.

Biases are typically represented in $°/hr$ or $°/s$ for gyroscopes and $mg$ for accelerometers.

*A scale factor* and *a misalignment error*, both systematic errors, could also prove to be significant. The scale factor is connected to imperfections while converting the real measurement input value and output value. The nonorthogonality of all sensors gives the misalignment error in IMU, and it is caused during production.

*Angle* or *velocity random walks* belong to stochastic errors. The measurement of gyroscopes and accelerometers is subject to white noise (the noise represented by Gaussian distribution). During the estimation of angles and velocities, integration needs to be done. Then the white noise starts to manifest itself by angle or velocity random walk, $(°/s/\sqrt{Hz})$ and $(m^2/s/\sqrt{Hz})$ respectively.

**Allan variance(AVAR)** is widely used to analyze a random error of inertial sensors in the time domain and is the most common time-domain measure of frequency stability. AVAR's brief introduction and important outcomes are given in [10].

The AVAR $\sigma_A^2(\tau)$ is a function of the averaging time $\tau$, computed as

$$\sigma_A^2(\tau) = \frac{1}{2(N-1)} \sum_{i=1}^{N-1} (\overline{y}_\tau(i+1) - \overline{y}_\tau(i))^2, \tag{3}$$

where $N$ represents the number of clusters in the dataset ($N = floor(M/n)$), $n$ is the number of samples in the cluster, $M$ is the total number of samples in the dataset, $\tau$ is the time length of the cluster ($\tau = m \times T_s$), $T_s$ is the sampling period, $\overline{y}_\tau(i+1)$ and $\overline{y}_\tau(i))$ are mean values of certain cluster of $i+1$-th and $i$-th cluster respectively [11].
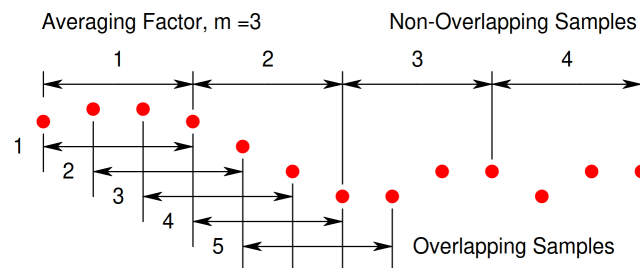


**Figure 2.5.** The difference between non-overlapping and overlapping sample [12].

---

[1] also called in-run bias
[2] repeatability bias

The samples in a cluster can be both non-overlapping and overlapping. The difference is illustrated in 2.5. The overlapping samples improve the confidence of the resulting estimate. That is why this method is generally the most common for measuring time-domain frequency stability [10].

The process of measuring AVAR consist of collecting 24-48 hour long datasets when the inertial sensor is not moving and the environment is not vibrating (no trains or subways that would cause vibration). The sampling values are angular rates or accelerations.

If the dataset is valid and the AVAR is correctly computed, the plot copies the example plot seen in Figure 2.6. It is typically plotted on a log/log scale. A different slope of the graph describes each noise component by that the chart can be easily divided into specific parts.
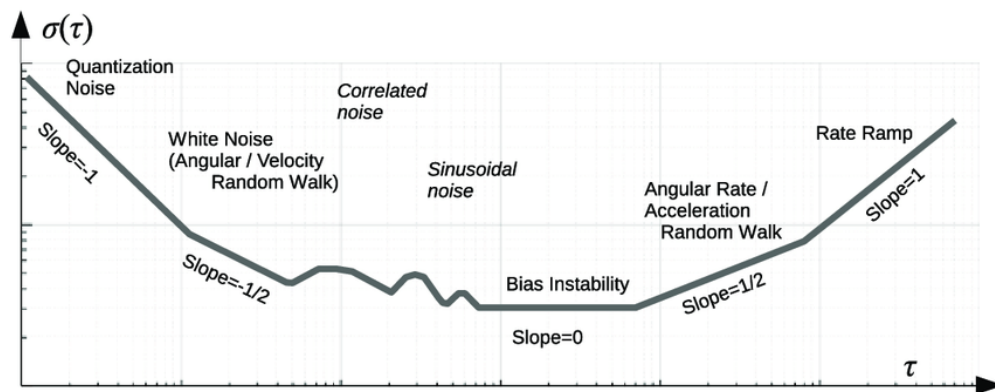


**Figure 2.6.** An example of Allan variance plot [13].

The most significant outcome for navigation is when the bias instability is reached (slope is zero). At this time, the sensor model contains only a white (Gaussian) noise [14]. After that period, an external reset needs to be done.

### 2.2.4  Performance of IMUs according to their application

IMUS can be used in various applications, which differs by IMUs performance. Figure 2.7 summarises the overview of each sensor's precision for a given application is summarized in Figure 2.7.

## 2.3  Inertial navigation systems

The fundamental idea behind Inertial navigation systems (INS) is integrating a linear acceleration into a position. Because of that, this topic is closely connected with the IMU, as it measures current linear acceleration [16]. The integration of IMU measurement is given by **navigation equations**.

Since INS is typically used for navigating aircraft, the principle of INS is introduced in this example. The plane is moving in a navigation coordinate frame. This frame can be specified as

- a local-level frame (as North-East-Down or East-North-Up),
- as a reference to a specific point on planet Earth
- or an Earth-fixed frame as ECEF [8], and these frames are demonstrated in Figure 2.8.
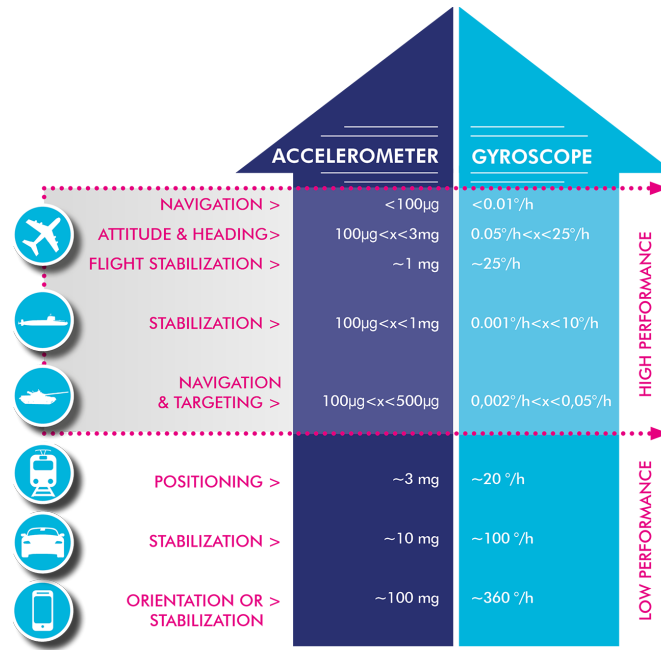
**Figure 2.7.** A performance of IMU per application [15].
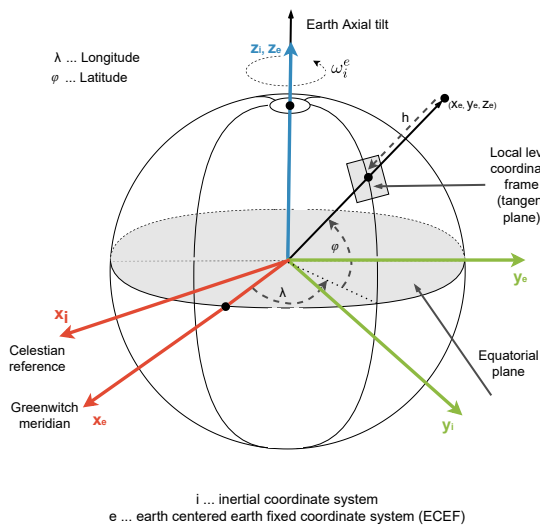


**Figure 2.8.** Example of coordinate frames used in INS[3].
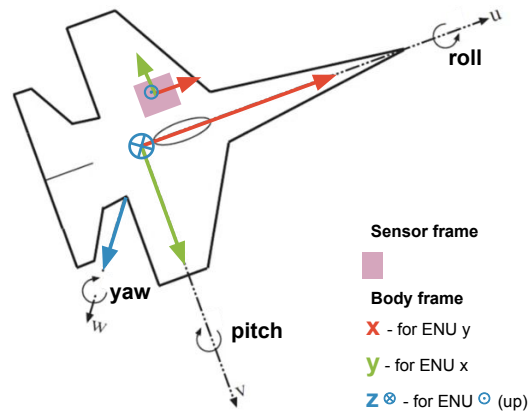


**Figure 2.9.** Example of body and sensor frame.

An IMU is mounted on an aircraft and its gyroscopes and accelerometers measure in its sensor frame. The aircraft's frame is called the body frame, so the measurement needs to be transformed into the body frame (see Figure 2.9). The output of the INS is given in the navigation frame. Thus, the last transformation is from the body frame into a navigation frame.

The navigation equations describe exactly what the transformation is between the sensor and navigation frames as well as the integration of IMU measurements. For example, navigation signal processing in strap-down INS can be seen in Figure 2.10.

INPUTS    SENSORS                    PROCESSING                    OUTPUTS
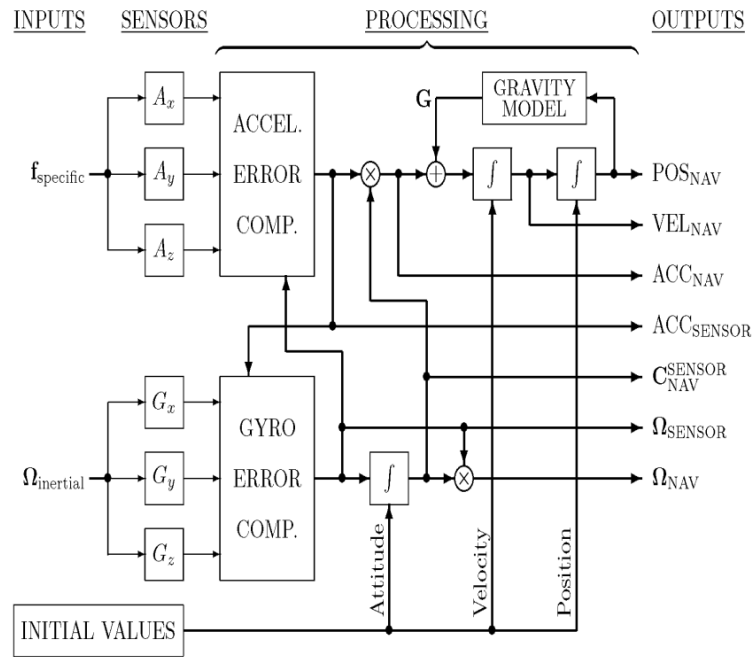


**Figure 2.10.** Schema of strapdown INS [17].

## 2.4 Odometry

Odometry is an example of a dead reckoning system. It estimates the pose and velocity of a device based on internal relative measurements of its motion. It can be obtained from various sources as IMU, lidars, cameras or wheel encoders [6].

Both IMU and wheel encoders are used in the suggested localization system since they can counter each other's negative characteristics since wheel encoders drift over traveled distance and IMU drift over time [6]. In this thesis, the term odometry is used to refer to wheel encoders unless stated otherwise.

The details of odometry information vary with vehicle design, and during experiments, the differential type is used (illustrated in Figure 2.11).
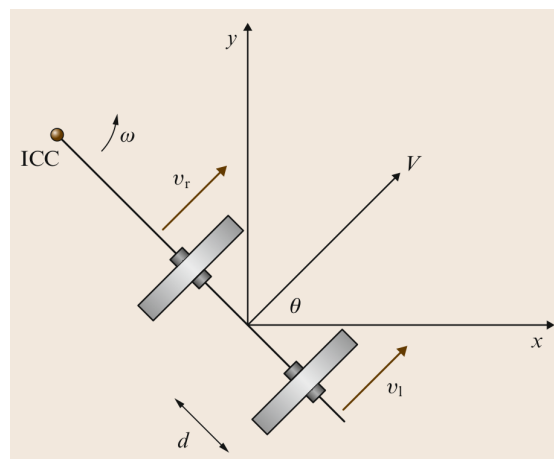


**Figure 2.11.** A differential drive kinematics scheme [6].

# Chapter 3
## Data fusion principles in pose estimation

The localization is estimating a robot's coordinates in an external reference frame from the sensor data. This is called state. In the state estimation algorithms, more states from different sensors are fused to counter sensors' negative attributes and help support the positive ones.

There are many approaches to data fusion. In particular, probabilistic methods are typically more robust in the face of sensor limitations, sensor noise or environment dynamics [18]. Moreover, they often scale much better to complex and unstructured environments, where the ability to handle uncertainty is of even greater importance [18]. Thus, this Chapter is dedicated to probabilistic methods for state estimation based on the Bayes filter [18].

The first section discusses the difference between the Kalman and Particle filter for state estimation. The following section introduces advanced concepts derived from the Kalman filter algorithm, such as the extended Kalman filter and the unscented Kalman filter. The third and fourth sections go deeply into the Error state Extended Kalman filter and introduce the benefits of using it, as it is the core of the localization system of this thesis.

## 3.1 Kalman and particle filter

Both Kalman and particle filters are the first implementations of Bayes filters in continuous time [18], and in both filters, the state is represented by belief, which corresponds to a distribution. For the Kalman filter, it is a multivariate normal distribution, but for the particle filter, the distribution is represented by all particles [18–19].

Both algorithms work with a prediction and correction step, which works with the system and sensor model, respectively. Firstly, it predicts the state based on the internal system model, and secondly, it corrects itself by external measurements and sensor model. Kalman and particle filter algorithms for localization are well described in a referenced Probabilistic robotics [18].

There are a few limitations for both algorithms. For the Kalman filter, the state transitions and measurements must be linear with added Gaussian noise, and the initial state must have normal distribution [18]. There is no such requirement for linearity for the particle filter, and it works fine with nonlinear or multi-modal systems too [18]. But the algorithm can be more computation demanding as a high number of particles needs to be generated in each sample time for a robust estimation [18].

I decided to use an algorithm based on the Kalman filter for several reasons. First, the localization system will be used for the navigation of vehicles. For these kinds of tasks, the update rate has to be relatively high. Second, the state transition and measurements are approximately linear. And third, these algorithms are typically used in the fusion of IMU and GNSS [20–23].

## 3.2 Algorithms based on the Kalman filter

There have been many modifications and extensions of the standard Kalman filter since the 1950s, when the filter was first invented. The linear system and sensor model assumptions with added Gaussian noise are rarely fulfilled in practice [18]. In these cases, the state transition or sensor model are described by nonlinear functions.

There exist many techniques for linearizing nonlinear functions. The most popular tool, called the Extended Kalman filter, use (first-order) the Taylor expansion [18]. This approximation has its limitations, which correspond to a degree of nonlinearity of the functions and a degree of uncertainty. The higher these degrees are, the further the approximation deviates from true belief. In general, the Extended Kalman filter has its benefits in simplicity, optimality and robustness [18], but in practice, it is reliable for the system, which is almost linear in one-time step [24]. The usage of this algorithm in similar cases to this thesis is illustrated in [25–26]. In [25], the EKF is used for estimating 3D position of drones, and in [26] to track a test subject in 2D walking in a room.

The Unscented Kalman filter is a tool that appears superior to the EKF linearization [18, 24]. Carefully selected sample points give the linearization from nonlinear functions. Also, this approach does not assume that the distribution of noise source is Gaussian [24].

In conclusion, the degree of nonlinearity of the system is critical for the state estimation by algorithms based on the Kalman filter. Thus, the relatively recent but promising tool Error state Extended Kalman filter was introduced. In this concept, the error of the state is estimated as it is more likely correctly modeled by a linear function [27–29].

## 3.3 Error state Extended Kalman filter

Error state Extended Kalman filter belongs to a group of Indirect Kalman filters because it does not estimate the state itself, but the error of the state [27].

The main idea is that the true state, which should be the output of the system, is computed as a suitable composition of nominal state and the error state

$$\mathbf{x_t} = \mathbf{x_n} \bigoplus \boldsymbol{\delta x} \tag{1}$$

where
- $\bigoplus$ is a suitable composition as linear sum or matrix product,
- $\mathbf{x_t}$ is the true state,
- $\mathbf{x_n}$ is the nominal state
- and $\boldsymbol{\delta x}$ is the error state.

The nominal state is considered a large signal that can be integrated in its nonlinear form and the error state as a small signal that is a nearly linear function, ideal for Extended Kalman filtering [28].

The algorithm can be illustrated in the following equations in prediction, correction, injection and resetting steps. In **the prediction step**, the nominal state and its covariance is estimated by following the equation [27–28]

$$\begin{aligned} \mathbf{x_t} &= f(\mathbf{x_{t-1}}, \mathbf{u_t}) \\ \mathbf{P_t} &= \mathbf{F}\mathbf{P_{t-1}}\mathbf{F^T} + \mathbf{B}\mathbf{Q}\mathbf{B^T} \end{aligned} \tag{2}$$

where

13

- $f(\mathbf{x_{t-1}}, \mathbf{u_t})$ is the nonlinear function that describes the current state of the system based on the previous state and current input,
- $\mathbf{x_t}$ is the nominal state in time $t$,
- $\mathbf{x_{t-1}}$ is the true state in time $t-1$,
- $\mathbf{u_t}$ is the input of the system,
- $\mathbf{P_t}$ is the state covariance matrix (also called system covariance),
- $\mathbf{F}$ is the state transition matrix given by $F = \frac{\delta f(x_{t-1}, u_t)}{\delta x_{t-1}}$,
- $\mathbf{B}$ is the input matrix,
- and $\mathbf{Q}$ is the input noise covariance matrix.

In **the correction step**, the state is corrected based on measurements according to equations [27–28]

$$\mathbf{K} = \mathbf{P_{t-1}}\mathbf{H^T}(\mathbf{HPH^T} + \mathbf{R})^{-1}$$
$$\mathbf{e_t} = \mathbf{K}(\mathbf{y_t} - h(\mathbf{x_{t-1}}))$$
$$\mathbf{P_t} = (\mathbf{I} - \mathbf{KH})\mathbf{P_{t-1}}(\mathbf{I} - \mathbf{KH})^T + \mathbf{KRK}^T$$

(3)

where
- $\mathbf{K}$ is the Kalman gain,
- $\mathbf{H}$ is the measurement transition matrix given by $\mathbf{H} = \frac{\delta h(\mathbf{x_t})}{\delta \mathbf{x_t}}$,
- $\mathbf{R}$ is the covariance of the measurement,
- $\mathbf{e_t}$ current estimated error of the state,
- $\mathbf{y_t}$ is the measurement in time $t$,
- and $h(\mathbf{x_{t-1}}$ is the measurement model based on the previous state $\mathbf{x_{t-1}}$.

In **the injection and resetting step**, the state is compensated by error, and the error state and state covariance matrix need to be reset [27–28]. The injection of the error is given by

$$\mathbf{x_t} = \mathbf{x_{t-1}} + \mathbf{e_t}$$

(4)

and the resetting can be illustrated

$$\mathbf{e_t} = g(\mathbf{e_t})$$
$$\mathbf{P_t} = \mathbf{GP_t}\mathbf{G^T}$$

(5)

where
- $g(\mathbf{e_t})$ is the resetting function of the error state
- and $\mathbf{G}$ is the Jacobian matrix defined as $\mathbf{G} = \frac{\delta g(\mathbf{e_t})}{\delta \mathbf{e_t}}$.

## 3.4 Chosen approach for the state estimation

This thesis aims to design a prototype of the localization system of vehicles based on the fusion of the UWB positioning system and onboard dead-reckoning sensors. As I already mentioned in Chapter 1, the system should be used for the localization of indoor vehicles in an industrial environment. However, each such vehicle (and terrain, where it is moving) can differ in its dynamics. Because of that, I decided to use an approach that can replace these dynamics. The inertial navigation system(INS) meets these requirements and gives us the state independent of a specific vehicle and specific terrain where it is moving. The estimation is accurate for a short time (see Section 2.2). Thus, it needs to be corrected via other measurements using the UWB localization system and odometry. I decided to estimate the error of the INS in the Error state Extended Kalman filter for the following reasons.

- The correction of INS can be done at a lower rate than the state estimation for localization purposes itself [27–28],
- The unscented Kalman filter or Particle filter algorithms have a high computational load that usually prevents them from being used in a real-time system [30].
- The model for estimation of the error is near-linear, as the error is
  - close to the origin,
  - and small, so higher orders can be neglected [27–28],
- numerical stability of the solution [27],
- even if the error estimation has a temporary computer failure, the INS is not affected, and some emergency procedures can come into account [27].

To summarize, the state estimation is given by INS, and the ES-EKF estimates the errors in the state using the difference between the INS and external sources of data, which are odometry and the UWB positioning system. The INS can estimate high-frequency motions of the vehicle accurately, so these dynamics are not in the filter explicitly modeled. The ES-EKF uses a model of error propagation in INS, which is at low frequency and very well modeled by linear functions [27, 31, 29]. This approach is also known as Aided Navigation systems and is briefly described in Aided Navigation: GPS with high rate sensors [29]. This algorithm is usually used to fuse INS and GNSS, but this article [32] shows its usage for INS and UWB fusions. In the article [32], they sucessfully tracked the position xy of a man walking in the room.

The following Chapter 4 introduces the localization system design and its implementation in detail.

# Chapter 4

## Localization system design and implementation

In this chapter, the design of the architecture is described along with the tools used in its solution.

The first part proposes the system architecture design and the system kinematics equations follow in the second part. In the third part, the equations for the Error state extended Kalman filter are introduced. The fourth part is dedicated to injecting the estimated error into the estimated state and resetting the injected error. In the fifth and final part, the implementation tools are discussed.

To sum up, this chapter gives the reader a detailed introduction to the proposed localization system with all of the equations and tools used in the final implementation.

## 4.1 System architecture design

Various approaches for state estimation were introduced in Chapter 3. The chosen approach is the Error state extended Kalman filter (ES-EKF). According to ES-EKF, the states' error is estimated using a Kalman filter rather than the state itself. The benefits of this approach are briefly summarized in Chapter 3.

The system consists of three crucial steps. The first is the inertial navigation system (INS), where the state is estimated based on IMU measurements. This state estimation leads to a dead-reckoning system, where the drift grows with time and needs to be corrected.

The second step is the ES-EKF itself. The error of the state is calculated and is then corrected using measurements from UWB localization and odometry. Measurements from UWB localization and odometry observe the error. The UWB localization gives the absolute position, which can reduce the drift in step one.

The third part is injecting the error into INS estimation and resetting the ES-EKF while the injection is done. Finally, the output of the whole system is given by the INS solution. The system requires initial states with covariances for INS and ES-EKF set up first. The simplified architecture is illustrated in Figure 4.1 and described in the following section in detail.

For navigation purposes, the usual rate of pose estimation is hundreds of Hertz [33]. The INS provides a full state estimate with the IMU update rate, which usually satisfies this requirement. Furthermore, it gives us the state estimation independent of external factors, such as wheel slippage [33].

UWB localization and odometry measurements cannot give us a much higher rate than tens Hertz, and they are used only in the correction step.

In other words, the most dynamic part of the estimation is somehow independent of Kalman filtering. On the one hand, the state estimation in INS is fast and straightforward. On the other hand, the error estimation can be more computationally demanding as the computation of Jacobians needs to be done. Therefore, the
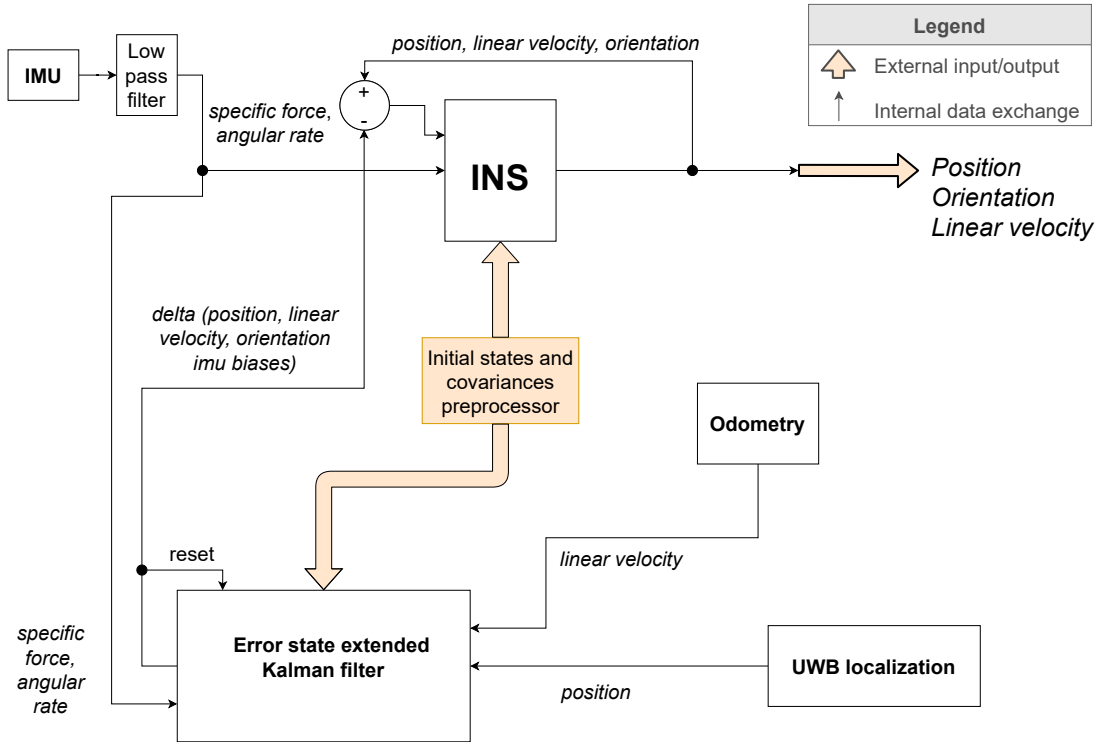
**Figure 4.1.** The proposed architecture of the localization system.

separation of the state estimation and the error estimation makes the calculation of the state sufficiently fast enough for our purposes. The error state is estimated separately in ES-EKF and is injected into the state only if another measurement other than IMU comes in. The only requirement is that the correction must be applied before non-Gaussian noise in IMU measurement is significant. That correction compensates for the drift of the dead-reckoning system.

In conclusion, the benefit of this architecture is state estimation at a high rate, independent of external events. The state is corrected at a lower rate but faster than the non-Gaussian noise becomes significant in state estimation. That brings the best aspects of all of the types of sensors, which are used in the architecture.

In the following sections, the output of the system is called the navigation state (i.e., position, linear velocity, and attitude).

## 4.2 System kinematics

For a more detailed introduction to system architecture, the system kinematics equations need to be announced. Equations are taken from [28], but instead of using GNSS, it uses UWB localization and linear velocity from odometry.

But before I enter that itself, let me describe an important topic, which represents attitude and rotation in 3D space.

### 4.2.1 Representation of 3D attitude and rotation in space

There are many ways to represent 3D attitude and rotation in space. The most commonly used representations in the field of robotics are
- rotation matrices,
- Euler angles,

17

- axis-angle
- and quaternions [6].

To not go into too much detail, each representation has its pros and cons and applications, where it has its purpose. The rotation matrix is chosen as the internal representation of orientation and the quaternion as an output.

There are several reasons to pick this representation [34–35]. Firstly, quaternions and rotation matrices do not suffer from singularities as Euler and fixed angles do [34]. Secondly, quaternion gives us a compact representation. And finally, these two are the most recommended representation in ROS standard rep-103 [35]. As quaternions have many internal models in different libraries (Eigen library in C++ [36], geometry messages library [37] or the transform library tf2 in ROS [38]) and the representation is not easy to imagine, I decided to use the quaternions only as an output and rotation matrix as the internal representation.

### 4.2.2 The kinematic equations in continuous time

The kinematics formulas in continuous time, that relate the inertial sensor measurements to the **true navigation state**, are well-known [28–29, 8, 7]. Therefore, I did not have to derive equations myself and used the one derived in [28] equation 235.

The only difference is that orientation is in the rotation matrix and not quaternion. Equations are

$$
\begin{aligned}
\dot{\mathbf{p}}_\mathbf{t} &= \mathbf{v}_\mathbf{t} \\
\dot{\mathbf{v}}_\mathbf{t} &= \mathbf{R}_\mathbf{t}(\mathbf{a}_\mathbf{m} - \mathbf{a}_\mathbf{bt} - \mathbf{a}_\mathbf{n}) + \mathbf{g}_\mathbf{t} \\
\dot{\mathbf{R}}_\mathbf{t} &= \mathbf{R}_\mathbf{t}(\Omega_\mathbf{t}) \\
\dot{\mathbf{a}}_\mathbf{bt} &= \mathbf{a}_\mathbf{w} \\
\dot{\boldsymbol{\omega}}_\mathbf{bt} &= \boldsymbol{\omega}_\mathbf{w} \\
\dot{\mathbf{g}}_\mathbf{t} &= \mathbf{0},
\end{aligned}
\tag{1}
$$

where

- $\mathbf{p}_\mathbf{t}$ is the true position in 3D $[m]$,
- $\mathbf{v}_\mathbf{t}$ is the true linear velocity in 3D $[m \cdot s^{-2}]$,
- $\mathbf{R}_\mathbf{t}$ is the true rotation matrix of orientation,
- $\mathbf{a}_\mathbf{m}$ is the specific force given by accelerometers $[m \cdot s^{-2}]$,
- $\mathbf{a}_\mathbf{bt}$ is the true accelerometer bias $[m \cdot s^{-2}]$,
- $\mathbf{a}_\mathbf{n}$ is the accelerometers white Gaussian noise $[m \cdot s^{-2}]$,
- $\mathbf{a}_\mathbf{w}$ is the white Gaussian noise accelerometers bias $[m \cdot s^{-2}]$,
- $\mathbf{g}_\mathbf{t}$ is the true gravity vector $[m \cdot s^{-2}]$,
- $\Omega_\mathbf{t} = [(\boldsymbol{\omega}_\mathbf{m} - \boldsymbol{\omega}_\mathbf{bt} - \boldsymbol{\omega}_\mathbf{n})_\times] = \begin{bmatrix} 0 & -(\omega_{m_3}-\omega_{bt_3}-\omega_{n_3}) & \omega_{m_2}-\omega_{bt_2}-\omega_{n_2} \\ \omega_{m_3}-\omega_{bt_3}-\omega_{n_3} & 0 & -(\omega_{m_1}-\omega_{bt_1}-\omega_{n_1}) \\ -(\omega_{m_2}-\omega_{bt_2}-\omega_{n_2}) & \omega_{m_1}-\omega_{bt_1}-\omega_{n_1} & 0 \end{bmatrix}$ is
  the true skew-symmetric matrix (a tensor of angular velocity) $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{m}$ is the angular rate given by gyroscopes $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{bt}$ is the true bias of gyroscopes $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{n}$ is the gyroscope's white Gaussian noise $[\frac{rad}{s}]$,
- and $\boldsymbol{\omega}_\mathbf{w}$ is the white Gaussian noise gyroscope's bias $[\frac{rad}{s}]$.

The state $\mathbf{x}_\mathbf{t}$, is governed by IMU noisy reading $\mathbf{u}_\mathbf{m}$ and perturbed by the white Gaussian noise $\mathbf{w}$, defined by

$$
\begin{aligned}
\mathbf{x}_\mathbf{t} &= [\mathbf{p}_\mathbf{t}, \mathbf{v}_\mathbf{t}, \mathbf{R}_\mathbf{t}, \mathbf{a}_\mathbf{bt}, \boldsymbol{\omega}_\mathbf{bt}, \mathbf{g}_\mathbf{t}]^T \\
\mathbf{u}_\mathbf{t} &= [\mathbf{a}_\mathbf{m} - \mathbf{a}_\mathbf{n}]^T \\
\mathbf{w}_\mathbf{t} &= [\mathbf{a}_\mathbf{w}, \mathbf{w}_\mathbf{w}]^T.
\end{aligned}
\tag{2}
$$

The output of the localization system is **navigation state** (also called nominal), which corresponds to the system kinematics, but does not take into account the noise terms $w_t$ and other possible model imperfections (see equation 237 in [28], hence it is simplified to

$$
\begin{aligned}
\dot{\mathbf{p}} &= \mathbf{v} \\
\dot{\mathbf{v}} &= \mathbf{R}(\mathbf{a_m} - \mathbf{a_b}) + \mathbf{g} \\
\dot{\mathbf{R}} &= \mathbf{R}(\Omega) \\
\dot{\mathbf{a_b}} &= \mathbf{0} \\
\dot{\boldsymbol{\omega}_\mathbf{b}} &= \mathbf{0}, \\
\dot{\mathbf{g}} &= \mathbf{0},
\end{aligned}
\tag{3}
$$

where
- **p** is the position in 3D $[m]$,
- **v** is the linear velocity in 3D $[m \cdot s^{-2}]$,
- **R** is the rotation matrix of orientation,
- $\mathbf{a_m}$ is the specific force given by accelerometers $[m \cdot s^{-2}]$,
- $\mathbf{a_b}$ is the accelerometer bias $[m \cdot s^{-2}]$,
- **g** is the gravity vector $[m \cdot s^{-2}]$,
- $\Omega = [(\boldsymbol{\omega}_\mathbf{m} - \boldsymbol{\omega}_\mathbf{b})_\times] = \begin{bmatrix} 0 & -(\omega_{m_3}-\omega_{b_3}) & \omega_{m_2}-\omega_{b_2} \\ \omega_{m_3}-\omega_{b_3} & 0 & -(\omega_{m_1}-\omega_{b_1}) \\ -(\omega_{m_2}-\omega_{b_2}) & \omega_{m_1}-\omega_{b_1} & 0 \end{bmatrix}$ is the skew-symmetric matrix (a tensor of angular velocity) $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{m}$ is the angular rate given by gyroscopes $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{b}$ is the bias of gyroscopes $[\frac{rad}{s}]$.

The linearized dynamics (see equation 238 in [28]) of the **error state** are

$$
\begin{aligned}
\dot{\boldsymbol{\delta}\mathbf{p}} &= \boldsymbol{\delta}\mathbf{v} \\
\dot{\boldsymbol{\delta}\mathbf{v}} &= -\mathbf{R}[\mathbf{a_m} - \mathbf{a_b}]_\times \boldsymbol{\delta}\Theta - \mathbf{R}\boldsymbol{\delta}\mathbf{a_b} + \boldsymbol{\delta}\mathbf{g} - \mathbf{R}\mathbf{a_n} \\
\dot{\boldsymbol{\delta}\Theta} &= -[\boldsymbol{\omega}_\mathbf{m} - \boldsymbol{\omega}_\mathbf{b}]_\times \boldsymbol{\delta}\Theta - \boldsymbol{\delta}\boldsymbol{\omega}_\mathbf{b} - \boldsymbol{\omega}_\mathbf{n} \\
\dot{\boldsymbol{\delta}\mathbf{a_b}} &= \mathbf{a_w} \\
\dot{\boldsymbol{\delta}\boldsymbol{\omega}_\mathbf{b}} &= \boldsymbol{\omega}_\mathbf{w} \\
\dot{\boldsymbol{\delta}\mathbf{g}} &= \mathbf{0},
\end{aligned}
\tag{4}
$$

where
- $\boldsymbol{\delta}\mathbf{p}$ is the position error in $[m]$,
- $\boldsymbol{\delta}\mathbf{v}$ is the linear velocity error in $[m \cdot s^{-2}]$,
- $\boldsymbol{\delta}\Theta$ is the orientation error,
- $\boldsymbol{\delta}\mathbf{a_b}$ is the acceleration bias error $[m \cdot s^{-2}]$,
- $\boldsymbol{\delta}\boldsymbol{\omega}_\mathbf{b}$ is the gyroscope bias error $[\frac{rad}{s}]$,
- $\boldsymbol{\delta}\mathbf{g}$ is the gravity vector error $[m \cdot s^{-2}]$,
- **R** is the rotation matrix given by the nominal state,
- $\mathbf{a_m}$ is the specific force given by accelerometers $[m \cdot s^{-2}]$,
- $\mathbf{a_b}$ is the accelerometer bias $[m \cdot s^{-2}]$,
- $\mathbf{a_n}$ is the accelerometers white Gaussian noise $[m \cdot s^{-2}]$,
- $\mathbf{a_w}$ is the white Gaussian noise accelerometers bias $[m \cdot s^{-2}]$,
- $\boldsymbol{\omega}_\mathbf{m}$ is the angular rate given by gyroscopes $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{b}$ is the bias of gyroscopes $[\frac{rad}{s}]$,
- $\boldsymbol{\omega}_\mathbf{n}$ is the gyroscopes white Gaussian noise $[\frac{rad}{s}]$,
- and $\boldsymbol{\omega}_\mathbf{w}$ is the white Gaussian noise gyroscopes bias $[\frac{rad}{s}]$.

Note that higher orders in linearization are neglected since the error state is small compared to the navigation state.

During the **filter correction phase**, measurements from UWB localization and odometry come into account. Usual, the sensor delivers measurements that depend on the state, such as

$$\mathbf{y} = h(\mathbf{x_t}) + \boldsymbol{\rho}, \tag{5}$$

where

- $h(t)$ is a general nonlinear function of the system state (the true navigation state)
- and $\boldsymbol{\rho}$ is a white Gaussian noise with covariance. For *UWB localization*, the function is simple as it is

$$\mathbf{y_1} = \mathbf{p_t} + \boldsymbol{\rho_1}, \tag{6}$$

with covariance $\mathbf{R_1}$. But for *odometry*, it is a little bit complicated

$$\mathbf{y_2} = \mathbf{R_t^{-1}v_t} + \boldsymbol{\rho_2} \tag{7}$$

with covariance $\mathbf{R_2}$. This difference is important in the computation of Jacobian for the ES-EKF algorithm.

### 4.2.3 The kinematic equations in discrete time

As the equations in continuous time are derived from book [28], where their representation in discrete time is also presented, I decided to write down only the parts that are different. For more detail, see equations 260 in [28]. The equation 260c is slightly different since I am using a rotation matrix for orientation representation and not quaternion. This equation is changed to

$$\mathbf{R} \leftarrow \mathbf{R} + (\mathbf{R}\Omega)\Delta\mathbf{t}, \tag{8}$$

where

- $\mathbf{R}$ is the rotation matrix of orientation,
- $\Omega = [(\boldsymbol{\omega_m} - \boldsymbol{\omega_b})_\times] = \begin{bmatrix} 0 & -(\omega_{m_3}-\omega_{b_3}) & \omega_{m_2}-\omega_{b_2} \\ \omega_{m_3}-\omega_{b_3} & 0 & -(\omega_{m_1}-\omega_{b_1}) \\ -(\omega_{m_2}-\omega_{b_2}) & \omega_{m_1}-\omega_{b_1} & 0 \end{bmatrix}$ is the skew-symmetric matrix (a tensor of angular velocity) $[\frac{rad}{s}]$,
- $\boldsymbol{\omega_m}$ is the angular rate given by gyroscopes $[\frac{rad}{s}]$,
- $\boldsymbol{\omega_b}$ is the bias of gyroscopes $[\frac{rad}{s}]$.

  This integration is happening in the INS box in Figure 4.1.

## 4.3 Error state extended Kalman filter implementation

The algorithm and equations for the general extended Kalman filter are briefly described in Chapter 3. In this section, these equations are concretized.

The **error state system** is now

$$\boldsymbol{\delta x} \leftarrow f(\mathbf{x}, \boldsymbol{\delta x}, \mathbf{u_m}, \mathbf{i}) = F_x(\mathbf{x}, \mathbf{u_m}) \cdot \boldsymbol{\delta x} + \mathbf{F_i} \cdot \mathbf{i}, \tag{9}$$

where $\mathbf{i}$ is a perturbation vector (usually modeled as white Gaussian noise).

The **Es-EKF prediction part** is given by

$$\begin{aligned} \hat{\boldsymbol{\delta x}} &\leftarrow \mathbf{F_x}(\mathbf{x}, \mathbf{u_m}) \cdot \hat{\boldsymbol{\delta x}} \\ \hat{\mathbf{P}} &\leftarrow \mathbf{F_x P F_x^T} + \mathbf{F_i Q F_i^T} \end{aligned}, \tag{10}$$

where

- **P** is a process covariance matrix,
- **F$_x$** is a transition matrix,
- **F$_i$** is the Jacobian of error state system by impulses,
- **Q** is the covariances of process noise,

The **transition matrix**[1] **F$_x$** is error state Jacobian and it is simply determined by error state kinematics equations $f(\boldsymbol{\delta x_t})$ in discrete time in Section 4.2.3,

$$\mathbf{F_x} = \frac{\partial f(\boldsymbol{\delta x}, \mathbf{u_m})}{\partial \boldsymbol{\delta x}}$$

$$\mathbf{F_x} = \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 & 0 & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{R}[\mathbf{a_m} - \mathbf{a_b}]_\times \Delta t & -\mathbf{R}\Delta t & 0 & \mathbf{I}\Delta t \\ 0 & 0 & \mathbf{R}\{\boldsymbol{\omega_m} - \boldsymbol{\omega_b}\}^\mathsf{T}\Delta t & 0 & -\mathbf{I}\Delta t & 0 \\ 0 & 0 & 0 & I & 0 & 0 \\ 0 & 0 & 0 & 0 & I & 0 \\ 0 & 0 & 0 & 0 & 0 & I \end{bmatrix}. \tag{11}$$

**F$_i$** is given by

$$\mathbf{F_i} = \frac{\partial f}{\partial i}\Big|_{\mathbf{x}, \mathbf{u_m}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ I & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{12}$$

The covariances matrix is given by random impulses applied to the velocity, orientation and bias estimates, modelled by white Gaussian noise [28]

$$\mathbf{Q} = \begin{bmatrix} \boldsymbol{\sigma}_{\mathbf{a_n}}^2 \Delta t^2 \mathbf{I} & 0 & 0 & 0 \\ 0 & \boldsymbol{\sigma}_{\boldsymbol{\omega_n}}^2 \Delta t^2 \mathbf{I} & 0 & 0 \\ 0 & 0 & \boldsymbol{\sigma}_{\mathbf{a_w}}^2 \Delta t^2 \mathbf{I} & 0 \\ 0 & 0 & 0 & \boldsymbol{\sigma}_{\boldsymbol{\omega_w}}^2 \Delta t^2 \mathbf{I} \end{bmatrix}, \tag{13}$$

where

- $\boldsymbol{\sigma}_{\mathbf{a_n}}$ is the standard deviation of accelerometers $[m \cdot s^{-2}]$,
- $\boldsymbol{\sigma}_{\boldsymbol{\omega_n}}$ is the standard deviation of accelerometers $[\frac{rad}{s}]$
- $\boldsymbol{\sigma}_{\mathbf{a_w}}$ is the velocity random walk $[\frac{rad}{s\sqrt{s}}]$,
- $\boldsymbol{\sigma}_{\boldsymbol{\omega_w}}$ is the angular random walk $[\frac{rad}{s\sqrt{s}}]$.

This information can be obtained from the datasheet or AVAR (see Section 2.2.3).

The **ES-EKF correction part** is given by

$$\begin{aligned} \mathbf{K} &\leftarrow \mathbf{PH^\mathsf{T}}(\mathbf{HPH^\mathsf{T}} + \mathbf{R})^{-1} \\ \boldsymbol{\delta x} &\leftarrow \mathbf{K}(\mathbf{y} - h(\hat{\mathbf{x}})) \\ \mathbf{P} &\leftarrow (\mathbf{I} - \mathbf{KH})\hat{\mathbf{P}}(\mathbf{I} - \mathbf{KH})^T + \mathbf{KRK^\mathsf{T}} \end{aligned}, \tag{14}$$

where

- **K** is the Kalman gain,
- **H** is an observation matrix,
- **R** is covariances of observation noise,
- **P** is process covariance,
- **y** is an observation,
- $h(\hat{\mathbf{x}})$ is an observation model,
- **δx** is an error state.

---

[1] Called system matrix in some literature

The **observation matrices** differ for *UWB localization* ($\mathbf{H_1}$) and *odometry* ($\mathbf{H_2}$)

$$\mathbf{H_1} = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\mathbf{H_2} = \begin{bmatrix} 0 & \mathbf{R_t^T} & -\mathbf{R_t^T}[\mathbf{v_t}]_{\times}\mathbf{J_r}(\Theta) & 0 & 0 & 0 \end{bmatrix},$$

$$(15)$$

where
- $\mathbf{R_t}$ is the orientation in navigation state,
- $\mathbf{v_t}$ is the linear velocity in navigation state,
- $\Theta$ is the orientation $\mathbf{R_t}$ in rotation vector form,
- $\mathbf{J_r}$ is the right Jacobian of rotation group $\mathbf{SO(3)}$ (see equation 183 in [28]).

To obtain $\mathbf{H_2}$ from Equation (7), a reader should notice a Jacobian with respect to the rotation vector in section 4.3.4 and equation 188 in [28].

## 4.4 Injection of the error state into the navigation state

While the correction phase is done, the estimated error state comes into account in the navigation state

$$\mathbf{x} \leftarrow \mathbf{x} \bigoplus \boldsymbol{\delta}\mathbf{x},$$

$$(16)$$

where $\bigotimes$ appropriate composition of sums or rotation product.

The equations are

$$\mathbf{p} \leftarrow \mathbf{p} + \boldsymbol{\delta}\mathbf{p}$$
$$\mathbf{v} \leftarrow \mathbf{v} + \boldsymbol{\delta}\mathbf{v}$$
$$\mathbf{R} \leftarrow \mathbf{R} * \mathbf{R}\{\boldsymbol{\delta}\Theta\}$$
$$\mathbf{a_b} \leftarrow \mathbf{a_b} + \boldsymbol{\delta}\mathbf{a_b}$$
$$\boldsymbol{\omega_b} \leftarrow \boldsymbol{\omega_b} + \boldsymbol{\delta}\boldsymbol{\omega_b}$$
$$\mathbf{g} \leftarrow \mathbf{g} + \boldsymbol{\delta}\mathbf{g}$$

$$(17)$$

where $\mathbf{R}\{\boldsymbol{\delta}\Theta\}$ orientation error in rotation matrix.

The injection of the error state is essential, but the resetting of the error state must also be done. The ES-EKF error reset operation is

$$\boldsymbol{\delta}\mathbf{x} \leftarrow \mathbf{0}$$
$$\mathbf{P} \leftarrow \mathbf{G}\mathbf{P}\mathbf{G}^\mathbf{T}$$

$$(18)$$

where $\mathbf{G}$ is the Jacobian matrix defined as

$$\mathbf{G} = \begin{bmatrix} \mathbf{I_6} & 0 & 0 \\ 0 & \mathbf{I} - [\frac{1}{2}\boldsymbol{\delta}\Theta]_{\mathbf{x}} & 0 \\ 0 & 0 & \mathbf{I_9} \end{bmatrix}.$$

$$(19)$$

## 4.5 Implementation tools

This section briefly introduces tools used for implementation: the ROS2 [39] framework, C++ and Python languages.

**ROS2 [39]** is a set of software libraries and tools for building robot applications. It is open-source, and it consists of drivers for hardware, state-of-the-art algorithms, tools for debugging, visualization, simulation and communications over all

of the processes. All applications created in ROS2 are easy to share and are used in the community. It supports the most known and most used programming languages like C++, Python, Java, Lua or Lisp. ROS2 distributions are released to work on operating systems like Ubuntu, MacOs or Windows. Nevertheless, as it is open-source, users usually use it with one Ubuntu distribution, such as 20.04 or 18.04.

The newest version of ROS is ROS2, which was introduced in 2014 at the conference ROSCon 2014 in Chicago [39], but the first distribution was released in May 2019. There are several distributions of ROS2 yet, the localization system and experiments are implemented using Foxy Fitzroy[2] which was released in June 2020[3].

ROS2 has a defined code style and a set of language versions which are recommended to use. The implementation sticks to these rules and uses C++17 and Python3. As the localization needs to be implemented as a real-time application, it is implemented in C++17. Python3 is used for the visualization of experiments results and supporting scripts.

---

[2] The documentation to ROS2 Foxy Fitzroy `https://docs.ros.org/en/foxy/index.html`

[3] The list of all distributions of ROS2 `https://docs.ros.org/en/galactic/Releases.html`

# Chapter **5**
## Experiments

This chapter is dedicated to the experimental evaluation of the proposed localization system. Firstly, the description of evaluation approaches is given. As the system is tested in real experiments, the second section is dedicated to implementing the localization system on the robotic platform CART2. This section contains a brief introduction to specific IMU, UWB network and wheel encoders, and the robotic platform itself.

The following two sections are dedicated to experiments done in two different experimental environments. The first set of experiments, described in the third section, were realized in a lab at Datavision s.r.o. company[1] with an external localization system based on an AprilTag detection with a camera. This AprilTag localization system was set up specifically for evaluation of localization system proposed in this thesis. Additional experiments took place at the Intelligent and Mobile Robotics lab at the Czech Technical University in Prague - Czech Institute of Informatics, Robotics, and Cybernetics, where the Vicon reference system is used for the evaluation. These experiments are summarized in the fourth section of this chapter.

## 5.1 Description of the evaluation

First, it is necessary to determine which variables exhibit the performance of the system well. Then choose the appropriate test scenarios on which to evaluate the behaviour of the individual variables. It is good to select a suitable reference system and perform the entire evaluation on a set of predefined metrics.

The evaluation is focused on the following states

- position x,
- position y
- position in 2D space (xy)
- and angle of rotation (Euler angle yaw).

It is worth noting that the output of the proposed system is the position in a 3D space. However, this work aimed to design a localization in a 2D space, and therefore an accent is put on these four states.

For system evaluations, it is advisable to have an external reference system, which will provide us with reference data with which we can compare the results of the evaluated system. The data from the reference and evaluated systems must have unified timestamps to compare their outputs easily.

Experimental scenarios are chosen in the way that their difficulty increases. The first tests are performed in a smaller area. Experiments start with a stationary test and simple movements at shorter distances (moving in one axis, rotating an one spot). Then, more complex trajectories are approached, such as following

---

[1] To find out more about the company follow link: `https://datavision.software/`

rectangular and infinity shape trajectories. Further experiments are performed in a larger space where more complex trajectories are tested.

Data from both reference and evaluated systems are then processed as follows. Each reference measurement's timestamp is taken according to which the nearest measurement can be found in the evaluated data. This filtering is vital for reference systems that have a lower frequency than the proposed localization.

The criteria that interest us are

- the evolution of individual variables over time,
- the visualization of the trajectories from both systems
- and the errors of variables versus the reference.

These metrics are used to analyze whether the proposed system converges or diverges from reality and whether the system is subject to drift. The root mean square (RMSE), average, median, lower and upper quantile, minimum and maximum metrics define the system's resulting accuracy and precision.

Error is usually defined as a difference between reference and reality measurement. It is pretty straightforward for one variable but can be a little bit tricky. The error in position in 2D space $[x, y]$ is defined as the Euclidian distance between reference $[x, y]_{REF}$ and the output of the proposed localization system $[x, y]$ written in the following equation

$$distance = \sqrt{(x^{REF} - x)^2 + (y^{REF} - y)^2}. \tag{1}$$

The root mean square error (RMSE) is defined as

$$rmse = \sqrt{\frac{\sum_{t=1}^{t=n}(x_t^{REF} - x_t)^2}{n}}, \tag{2}$$

where

- $t$ is the index of the current sample of measurement,
- $n$ is the number of samples in the dataset,
- $x_t^{REF}$ is the current reference measurement
- and $x_t$ is the current measurement.

The rmse for euclidian distances then

$$rmse = \sqrt{\frac{\sum_{t=1}^{t=n}[(x_t^{REF} - x_t)^2 + (y_t^{REF} - y_t)^2]}{n}}. \tag{3}$$

All these testing scenarios and metrics are further applied in two test environments.

## **5.2  Used hardware description**

This section presents an overview of used sensors for the proposed localization and a description of the CART2 robotic platform used during experiments.

**Figure 5.1.** IMU used in experiments Epson M-G365PDF1.



**Figure 5.2.** Experimental setup for static data acquisition of Epson M-G365PDF1.

### 5.2.1 Sensors specification

The onboard sensors of interest are the Inertial Measurement Unit, ultra-wideband localization tag, April tag and encoders on motors.

The **inertial measurement unit** used during the experiments is the Epson M-G365PDF1 (loaner sample). The Epson M-G365 is used in various applications ranging from stabilization systems (as a camera gimbal) to navigation systems.

The IMU has six degrees of freedom and measures angular rates and linear accelerations in three axes. It is factory calibrated and the calibration data are stored in the memory of the unit. Technical specifications of the Epson M-G365 can be found at [40], while the summary is included in Table 5.1

| Specification | Value |
|---|---|
| Triple gyroscopes | $\pm$ 450 °/sec |
| Gyroscopes bias instability | 1.2 °/hr |
| Gyroscopes initial bias error | 0.1 °/s |
| Angular random walk | 0.08 °/√hr |
| Tri-axis accelerometers | $\pm$ 10 G |
| Accelerometers bias instability | 16 $\mu$G |
| Accelerometers initial bias error | 3 mG |
| Velocity random walk | 0.033 (m/s)/√hr |

**Table 5.1.** Technical specifications of Epson M-G365PDF1 [12].

As I already mentioned in Chapter 2.2.3, the AVAR analysis of IMU sensors can give us a brief overview of IMU's specifications. The experimental setup for static data acquisition can be seen in Figure 5.2. The sensor is mounted on two sponges and fixed with cardboard. The static data were recorded for 48 hours at a frequency of 30.0 Hz. The experiment took place in a village without any subways, trams or trains to reduce the external vibrations (to reduce potential outliers) and at standard room temperature (about 23 °C).

For the AVAR computation, I used a python library named AllanTools [41]. For the purposes of this thesis, the overlapping Allan deviation function is used.
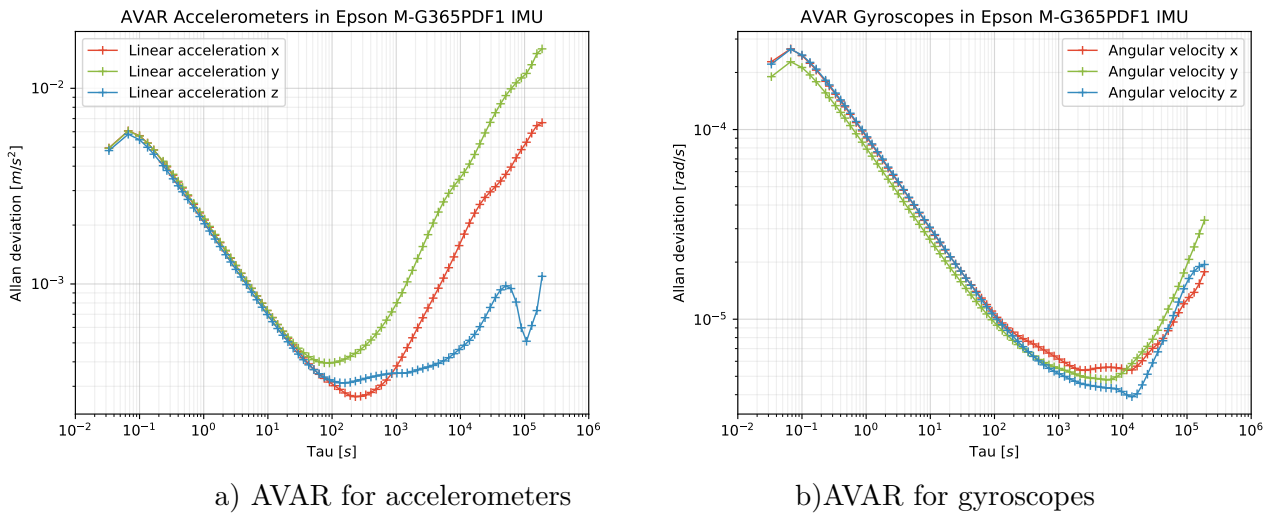
a) AVAR for accelerometers  b)AVAR for gyroscopes

**Figure 5.3.** Overlapping Allan variance plot for Epson M-G365PDF1.

As shown in Figure 5.3, the external reset for the integration of IMU measurement should be performed at least every 100 seconds for the results not to be corrupted by the non-Gaussian noise. The UWB localization system works at 10 Hz, which should be frequent enough.

The IMU is set up for the final experiments to publish the delta angle and the delta velocity at 100 Hz with a moving average filter with tap 64. Because of that, higher frequencies above 10 Hz are filtered, as shown in Figure 5.4.
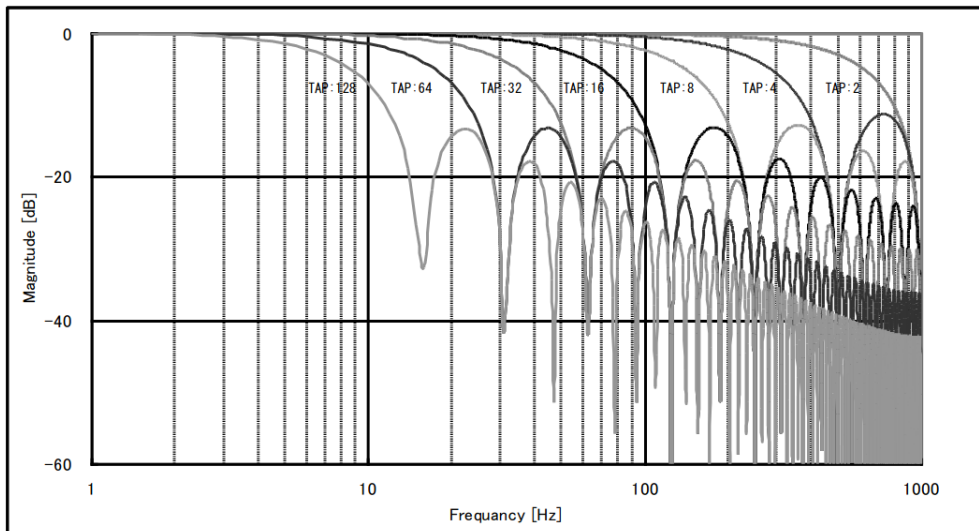


**Figure 5.4.** Moving average filter characteristics for Epson M-G365PDF1 [12].

The **UWB localization system** is provided by Qorvo's MDEK1001 ultra-wideband development kit. This kit includes 12 DWM1001-DEV development boards completely enclosed in plastic, see Figure 5.5.

Each board can be configured as an anchor, tag or bridge node. The system is installed with six fixed anchors that are mounted and one tag which can move. The anchors are higher than the tag. In the network, the initiator anchor provides the synchronization of transmitted data. The example setup of the system can be seen in Figure 5.6.

**Figure 5.5.** DWM1001-DEV development boards [42].



**Figure 5.6.** Positioning of UWB anchors and tags [42].

The anchors should all be mounted at the same height and higher than the operating area of the moving tag. They also should not be mounted close to any metal to get the best accuracy possible. There is a mobile application available to configure the network. The positions of the anchors are estimated manually and set in the network configuration.

Technical specifications of the MDEK1001 and the DWM1001-DEV can be found in documents [42], the summary is listed in Table 5.2.

| Specification | Value |
|---|---|
| Localization technology | Two-way ranging |
| Maximum tag location rate | 10 Hz |
| X-Y location accuracy | < 10 cm |
| Point to point range | up to 60 m in line of sight conditions |
| Scheme range | 25 - 30 m between anchors |

**Table 5.2.** System performance of MDEK1001 [42].

The **odometry** is computed according to the measurement of encoders in Maxon EPOS4 positioning controllers for Maxon brushless DC motors.

**AprilTag** serves as a global reference for the pose of the CART2 platform. The AprilTag detection software computes the precise 3D position, orientation, and identity of the tags relative to the camera [43]. This tag is similar to QR codes (a

type of two-dimensional bar code), but it encodes smaller data payloads (between 4-12 bits), and it can be detected more robustly.

The **camera** used to detect the AprilTag was the Niceboy Stream Pro with Full HD (1920 x 1080) resolution, 30 FPS, 90 ° field of vision and a f/1.8 lens aperture [44].
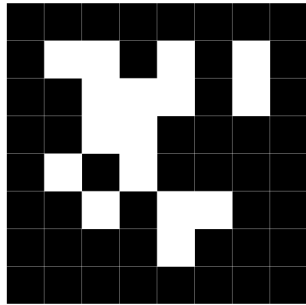


**Figure 5.7.** AprilTag used for detection of CART2 position at Datavision s.r.o.



**Figure 5.8.** Camera used for AprilTag detection Niceboy Stream Pro [44].

### 5.2.2 CART2 platform description and sensors placements

An image of the utilized CART2 platform can be seen in Figure 5.9. The coordinate frame of CART2, called baselink, is illustrated in all Figures 5.9 and 5.10. The CART2 used for various robotic competitions is a differential drive equipped with the ADlink MXE-210 computer.
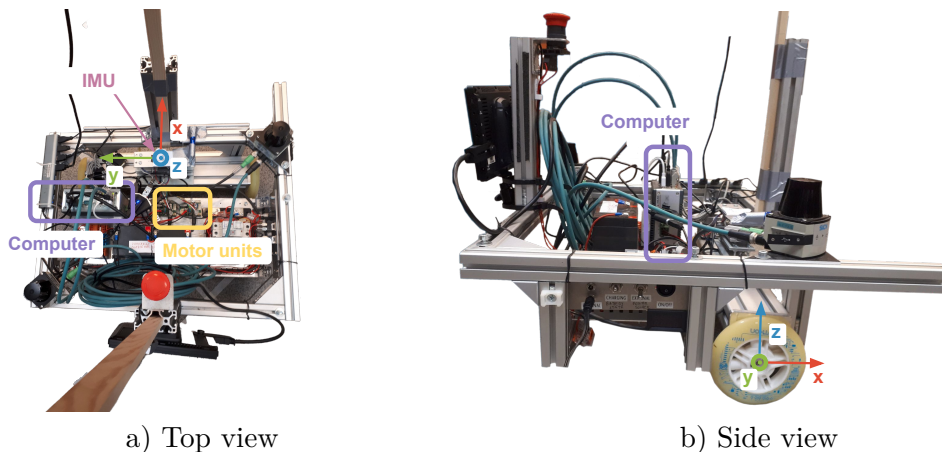


a) Top view

b) Side view

**Figure 5.9.** A photo of the utilized CART2 platform.

The onboard sensor placements are illustrated in Figure 5.10. The vehicle uses a Maxon brushless DC motor (Maxon EC-imotor) controlled by Maxon EPOS4 control units.

The UWB tag is mounted on a wooden stick approximately one meter above the CART2 platform to reduce any reflections of the UWB waves from surfaces and the negative influence of any metal parts. The AprilTag is mounted below the UWB tag not to become a barrier in the UWB wave. When the AprilTag is mounted on top of the CART2 platform, the wooden stick obstructs the camera from detecting the AprilTag. Thus, I decided to mount it as high as possible to reduce any of these situations from happening.
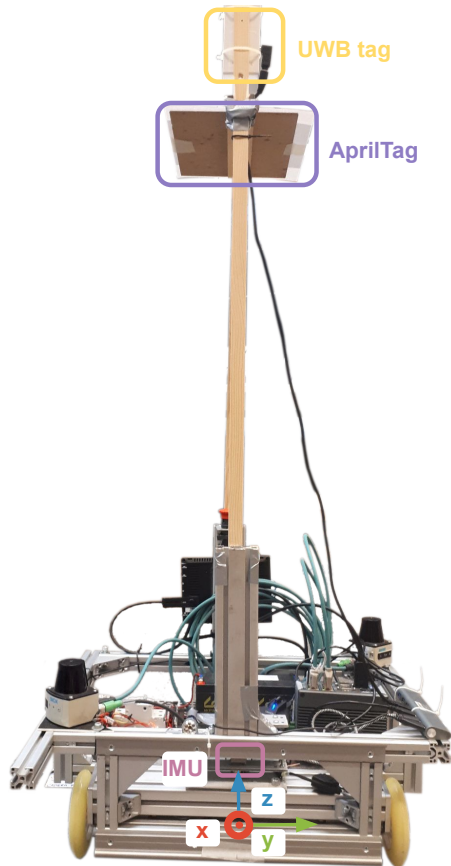
29

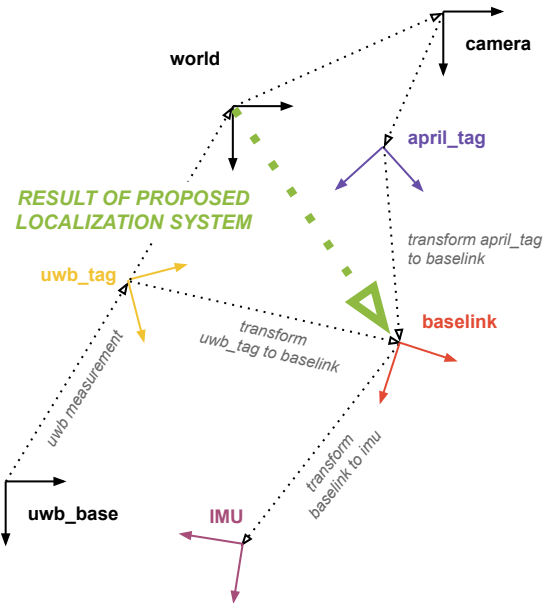**Figure 5.10.** A photo of sensors placements on CART2.



**Figure 5.11.** Illustration of coordinate systems at CART2.

**Coordinate systems of CART2 and the placement of the sensors** are illustrated in Figure 5.11 and are described in Table 5.3. Note that the world, the camera and the uwb_base frames are fixed within the environment and are explained in Section 5.3. This work aims to localize the position and orientation of the baselink, i.e., to estimate the position of the baselink expressed in the world frame and the orientation of the baselink with respect to the world frame.

| Transform | Translation [x, y, z] in [m] | Rotation in quaternion [x, y, z, w] |
|---|---|---|
| uwb_tag to baselink | [0.0, 0.0, 1.192] | [0.0, 0.0, 0.0, 1.0] |
| april_tag to baselink | [0.0, -0.171, 1.071] | [0.707107, -0.707107, 0.0, 0.0] |
| imu to baselink | [0.0, 0.0, 0.05] | [0.0, 0.0, -0.7071068, 0.7071068] |

**Table 5.3.** Summary of transforms for experiments with CART2.

## 5.3 Experiments in the lab at Datavision s.r.o.

### 5.3.1 Experimental lab description

For the first experiments, I created an experimental setup at Datavision s.r.o. with the global reference given by the camera detection from the AprilTag mounted on the CART2 platform. The address of the building is Ukrajinská 1487/2a, 101
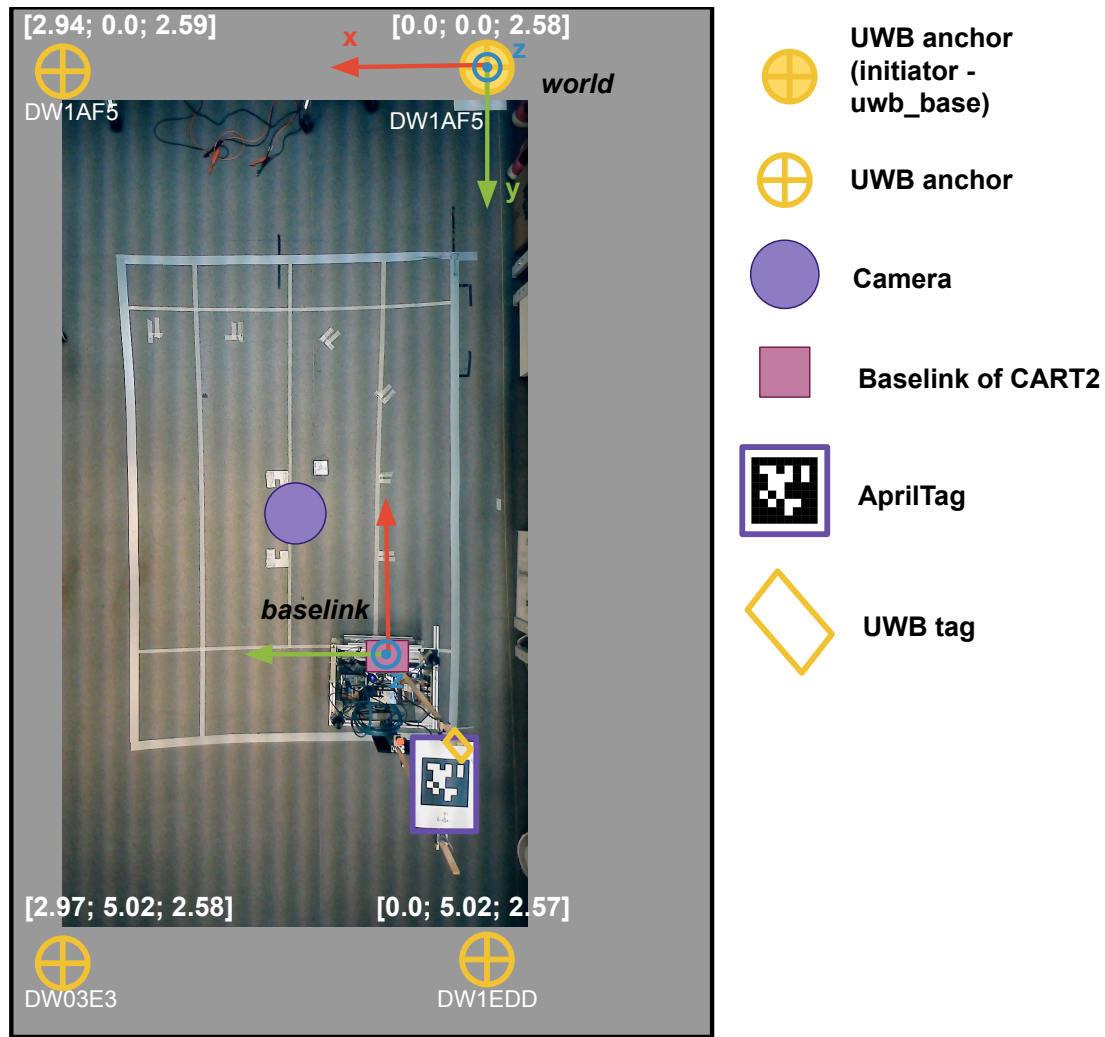
**Figure 5.12.** Experimental setup at Datavision s.r.o..

00 Prague 10 - Vršovice. The dimensions of the room for the experiments are approximately 4 x 6 [m], and the camera view area is approximately 2.5 x 4.5 [m]. The setup can be seen in Figure 5.12.

Firstly, UWB anchors need to be mounted and measure their poses according to the world coordinate system. These poses need to be set in the mobile application for the configuration of the UWB network. The world frame coincides with the uwb_base frame in rotation and only differs in z coordinate in translation.

| Transform | Translation [x, y, z] in [m] | Rotation in quaternion [x, y, z, w] |
|---|---|---|
| world to uwb_base | [0.0, 0.0, 2.58] | [0.0, 0.0, 0.0, 1.0] |
| world to camera | [1.089, 2.024, 2.625] | [0.707107, -0.707107, 0.0, 0.0] |

**Table 5.4.** Summary of transforms for experiments setup at Datavision s.r.o.

Secondly, the camera needs to be mounted and measure its position according to the world coordinate system. It was not possible to estimate transformation between the world and the camera precisely enough to use to evaluate the experiments. Because of that, I decided to use the transformation mentioned in Table

31

5.4 and then computed a homography transformation according to a few positions measured by camera detection and also by hand. The homography defines the transformation between a planar surface (ground) and a camera image plane. The camera homography is then applied to the detected AprilTag pose, which is considered a global reference. These frames are illustrated in 5.11.

### 5.3.2 Description of experiments

The area for experiments is not big. The distance between anchors is 2.5 or 5.0 m, and such setup with nearby anchors significantly downgrades the performance of the UWB localization system. However, I decided that these experiments serve as a proof of concept of the localization idea.

Experiments can be divided based on six simple trajectories of CART2 into
- stationary test (stationary test),
- rotation above 360 degrees in one direction at one place test (rotation test),
- moving in x direction (x test),
- moving in y direction (y test),
- moving in rectangle shape (rectangle test)
- and moving in infinity shape (infinity test).

I picked a specific initial pose for each testing trajectory to repeatedly make these tests in very similar conditions. Also, rotation, rectangle, and infinity tests ran several times in a row without start/stop of the system to see how it behaves in a long term.

Trajectories of these tests and the starting and ending positions of CART2 performing the movement are illustrated in Figure 5.13. CART2 platform was controlled via keyboard and joystick during experiments. The controlling via keyboard has the benefit of control velocity in each direction easily. Thus I used it for constant speed during simple moving in single-axis and simple rotation around a single axis. With that, I controlled all tests, except the infinity test, where the movement is complex. During the rectangle test, CART2 drove forward (moving in single-axis) then stopped and turned at one place above 90 degrees (rotating above single axis). I controlled the CART2 platform with the joystick in infinity shape movement and tried to move similar velocities as in constant movements. The summary of velocities during experiments is given in Table 5.5.

| Test | Description | Speed [m/s] | Turn [rad/s] |
|---|---|---|---|
| stationary | Constant velocities | 0.0 | 0.0 |
| rotation | Constant velocities | 0.0 | 0.1094 |
| x | Constant velocities | 0.0750 | 0.0 |
| y | Constant velocities | 0.0750 | 0.0 |
| rectangle | Constant velocities | 0.0750 or 0.0 | 0.0 or 0.1094 |
| infinity | non constant (control via joystick) | approx. 0.0750 | approx. 0.1094 |

**Table 5.5.** Velocities during experiments in Datavision s.r.o..

### 5.3.3 Evaluation of experiments

Most of the experiments were performed in this environment. However, I select only the most interesting ones for the evaluation in the thesis. The most attractive
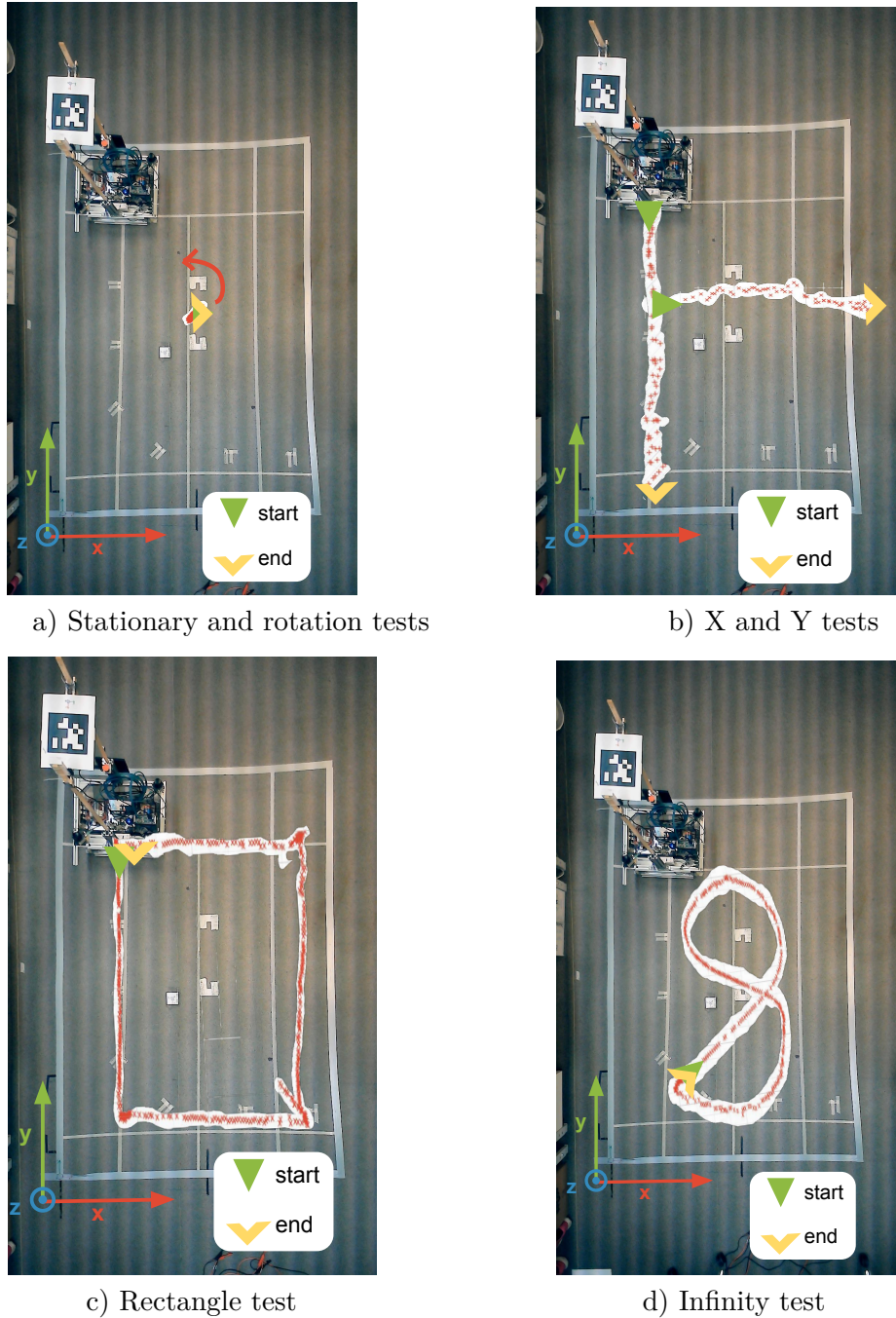
a) Stationary and rotation tests



b) X and Y tests



c) Rectangle test



d) Infinity test

**Figure 5.13.** Trajectories for experiments at Datavision s.r.o..

experiments are stationary, rectangle and infinity shape tests, which are presented bellow.

### Stationary test

During the stationary test, the CART2 is not moving, and the experiment lasts about 16 minutes. As shown in Figure 5.14, the precision of the position does not change significantly over time. The position in the proposed system is determined mainly by the absolute localization of the UWB, but it can be seen that our system can further refine the UWB localization. There is a significant drift in the orientation angle yaw caused by the INS. The orientation error is corrected by the

a) Position in 2D space



b) Yaw angle in time



c) Position x in time



d) Position y in time
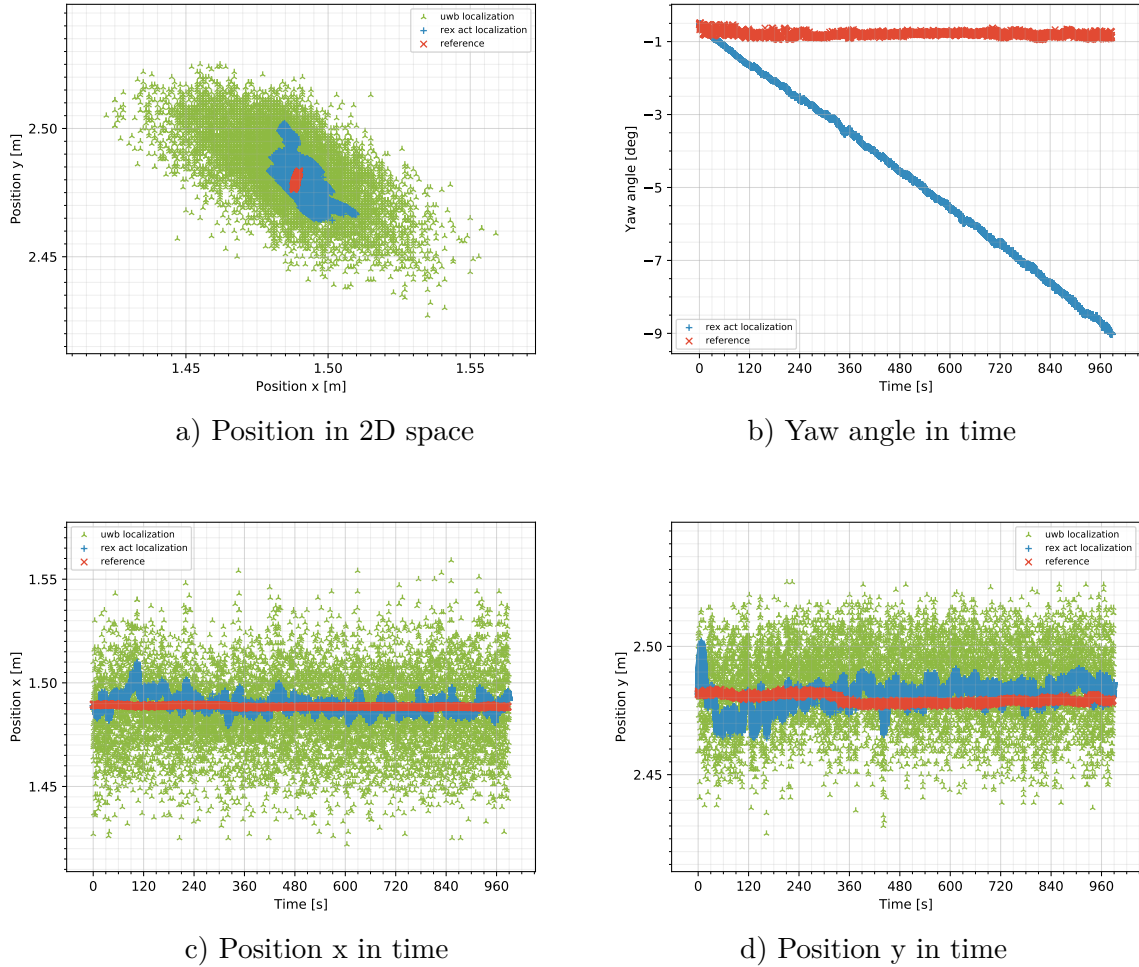
**Figure 5.14.** Stationary test, evaluation of trajectory at Datavision s.r.o.

linear velocities given by odometry measurement, but because the CART2 is not moving, the error is not corrected, and the drift slowly grows.

In general, box plots show the five-number summary of a set of data: including the minimum, first (lower) quartile, median (interquantile), third (upper) quartile, and maximum. I added a visualisation of the RMSE, to keep these information together.

In Figures 5.15 and 5.16 are all these metrics for the position in 2D error and the orientation error during the stationary test shown. Sufficient information from Figure 5.16 is that maximum orientation error during 16 minutes long stationary test is not bigger than ten degrees. The position error is small and does not increase in Figure 5.15. It is worth noticing that the external reference frequency is not high (1-2 Hz). When filtering the corresponding measurement samples for external reference timestamp, closer values may be favoured if the localization fluctuates. Thus, there may be a slight deformity here.

**Rectangle-shaped test**

During the experiment, nine repetitions of the rectangle-shaped trajectory were performed to define the localization's accuracy, precision, and repeatability. The course of one such passage is shown in Figure 5.17. The waves in the left part of Figure 5.17a can be caused by reflections of the UWB signal as CART2 moves close
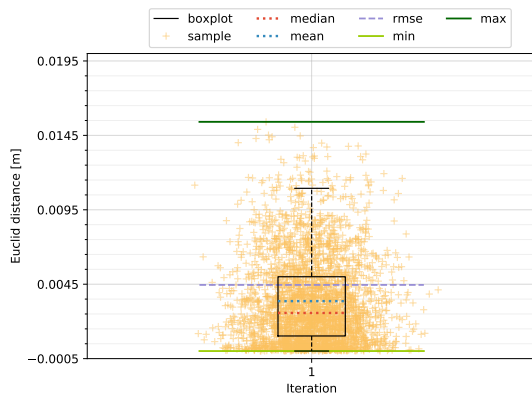
34

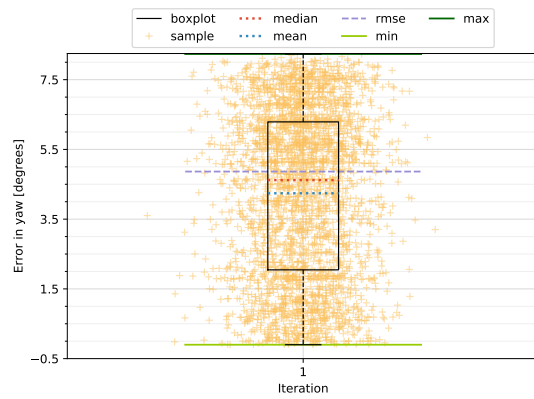**Figure 5.15.** Stationary test: boxplot, rmse, min and max analysis of the position 2D error, at Datavision s.r.o.
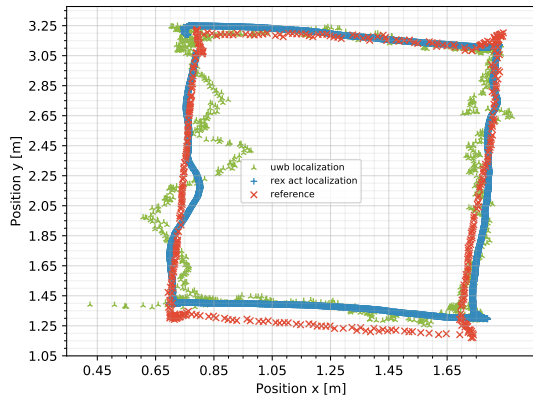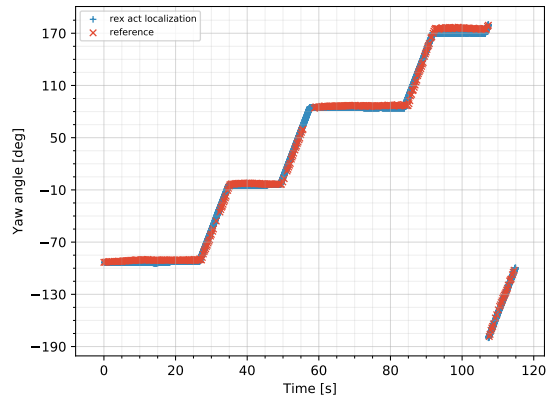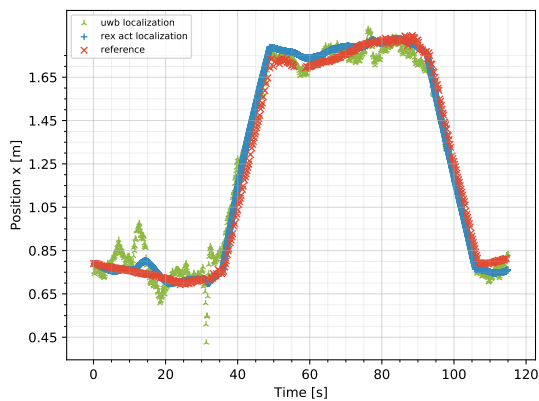
**Figure 5.16.** Stationary test: boxplot, rmse, min and max analysis of the orientation error, at Datavision s.r.o.
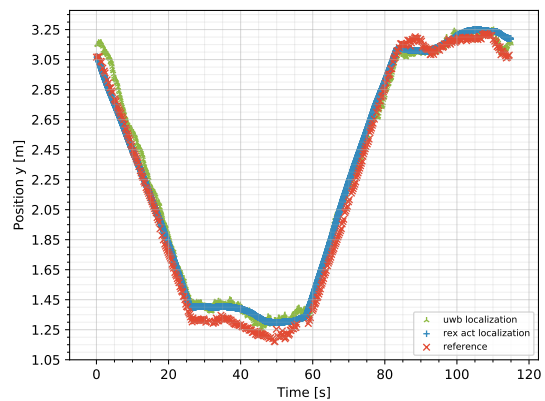


a) Position in 2D space

b) Yaw angle in time

c) Position x in time

d) Position y in time

**Figure 5.17.** Rectangle-shaped test, analysis of trajectory, Datavision s.r.o.

to different furniture and boxes. This phenomenon can be seen in all reproductions of testing scenarios. However, despite this fact, the system manages to locate CART2 with sufficient precision. The static offset at the bottom of Graph 5.17a
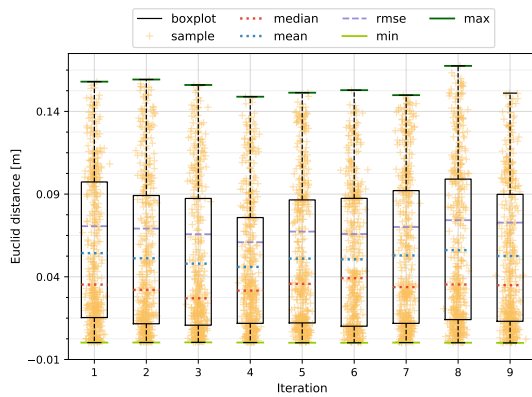
**Figure 5.18.** Rectangle-shaped tests: box-plot, rmse, min and max analysis of the position 2D error, at Datavision s.r.o.
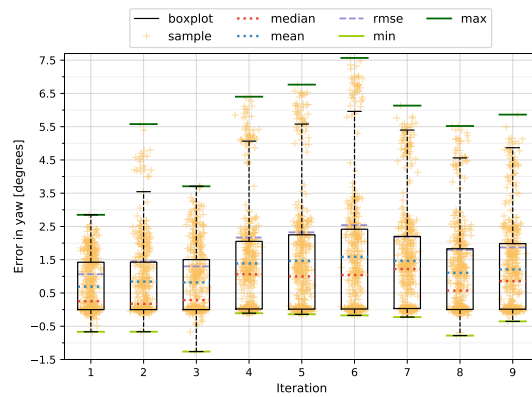


**Figure 5.19.** Rectangle-shaped test: box-plot, rmse, min and max analysis of the orientation error, at Datavision s.r.o.

is possibly given by a short line of sight between the localization anchors or the transformation inaccuracies in the external reference.

Figures 5.18 and 5.19 show that the localization results are well repeatable. RMSE in position is held at 7 cm and in the orientation is not higher than 3 degrees. The maximum errors in the position are about 15 cm and in the orientation 8 degrees. More than 50 % of the errors are kept below 10 cm in the position and 3 degrees in the orientation.

### Infinity-shape test

Evaluating the infinity-shaped trajectory brings the most complex view because the CART2 here moves simultaneously in linear and angular velocity (Figure 5.20). Ten repetitions of such a trajectory demonstrate how the system behaves in the most realistic scenario.
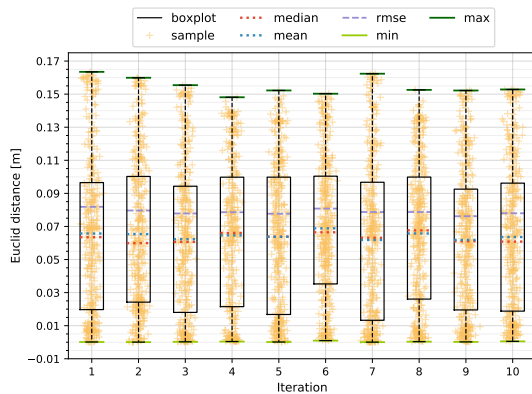


**Figure 5.21.** Infinity-shaped: boxplot, rmse, min and max analysis of the position 2D error, at Datavision s.r.o.
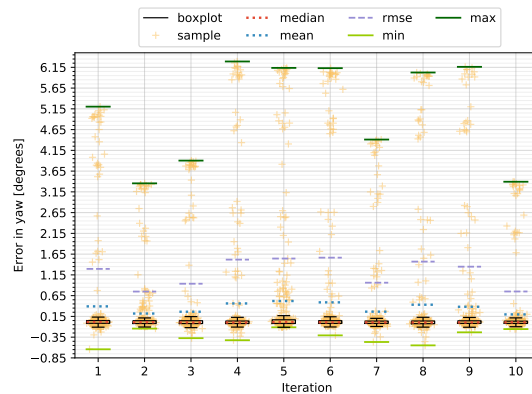


**Figure 5.22.** Infinity-shaped: boxplot, rmse, min and max analysis of the orientation error, at Datavision s.r.o.

As can be seen in Figures 5.21 and 5.22, the errors are small. RMSE of position is lower than 10 cm, and RMSE of orientation is below 2 degrees. There are maximum errors of 16 cm in position and 7 degrees in orientation, but most do not exceed 10 cm and 1 degree, respectively.

Notice that the precision of the proposed localization system meets the limits of the precision of the external reference system.
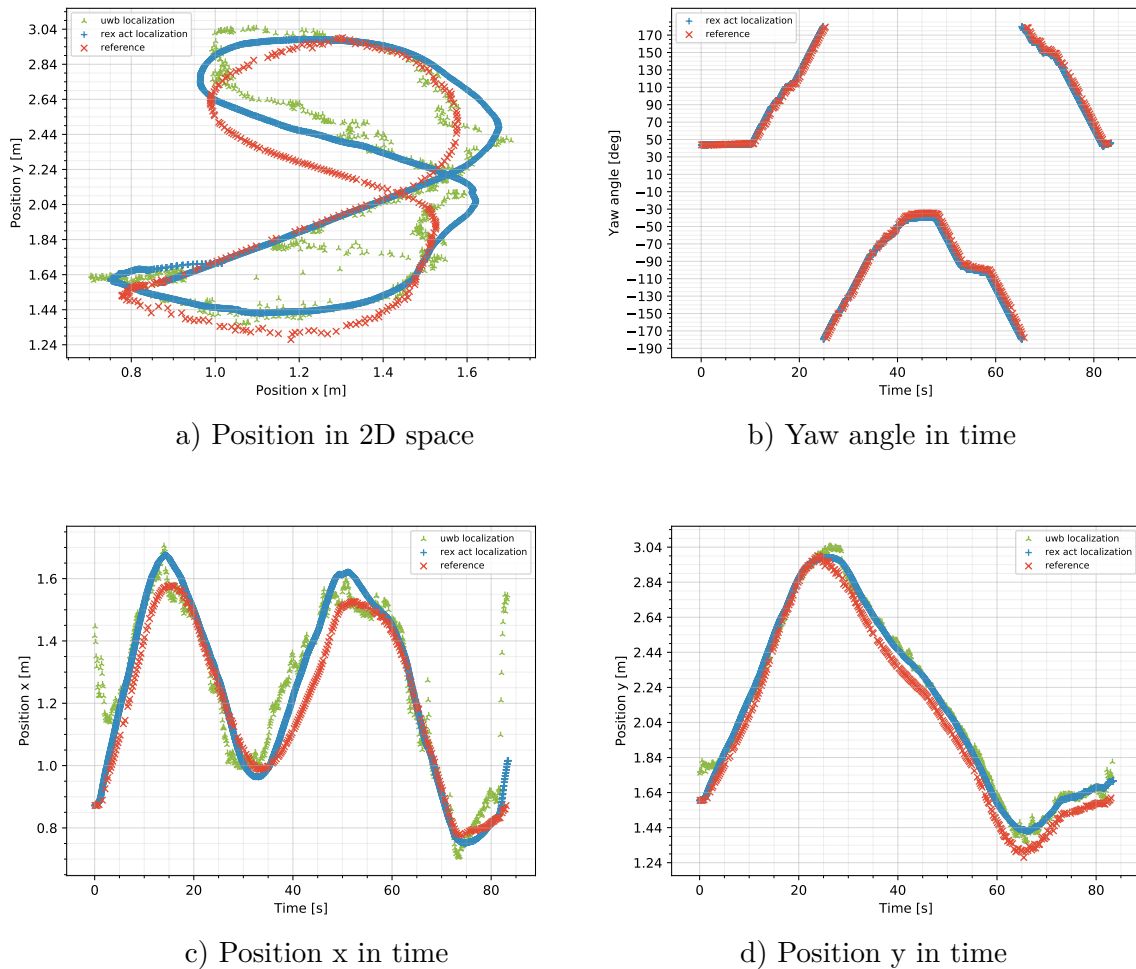
a) Position in 2D space



b) Yaw angle in time



c) Position x in time



d) Position y in time

**Figure 5.20.** Infinity-shaped test, analysis of trajectory, Datavision s.r.o.

## 5.4 Experiments at CIIRC

### 5.4.1 Experiments at CIIRC description

The next experimental environment was placed at the Czech Technical University in Prague – Czech Institute of Informatics, Robotics, and Cybernetics (CIIRC) in the Intelligent and Mobile Robotics lab. The localization system was evaluated using the VICON external camera localization system [45]. The VICON defines the world frame. UWB localization is installed with six anchors and their positions are defined in a world frame. The setup is illustrated in Figure 5.23. The benefit of this environment is the size of the laboratory. The minimal distance between two anchors is larger than 3,5 meters; thus, the accuracy of the UWB localization should be increased.

Experiments here are more complex because they are longer both in terms of duration and distance. These experiments promised to evaluate possible drift caused by using two relative localizations based on IMU and odometry. As I already mentioned in previous chapters, IMU's drift increases over time and odometry's drift grow larger over distance travelled. Performed experiments can be divided based on three simple trajectories of CART2 into
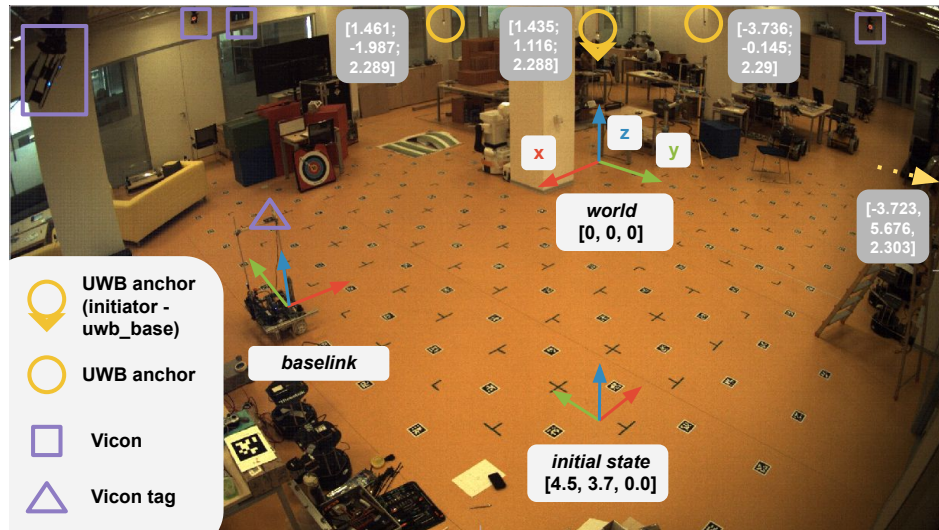
a) View from camera 1
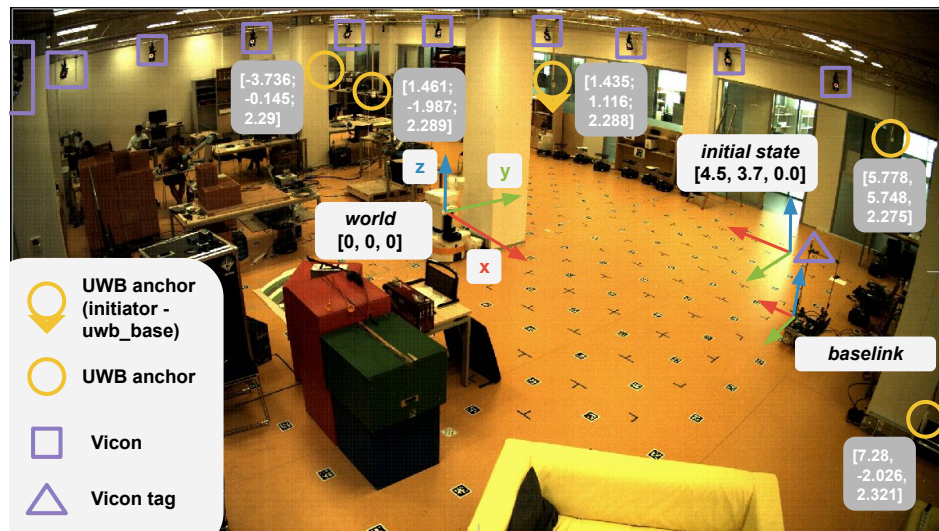


b) View from camera 2

**Figure 5.23.** Experimental setup at the Intelligent and Mobile Robotics lab (CIIRC).

- moving in the shape of a rectangle (rectangle test),
- moving in the shape of an infinity symbol (infinity test)
- and so-called joy ride, where CART2 tries to move as long-distance as possible without crossing its trajectory (joy ride test).

I picked a specific initial pose for each testing trajectory to repeatedly make these tests in very similar conditions. Also, the rotation, the rectangle and the infinity tests ran several times in a row without starting or stopping the system to see how it behaves under longer time durations.

### 5.4.2 Evaluation of experiments at CIIRC

**Rectangle-shaped test**

In Figure 5.24, there is no major problem with estimating the position, which is close to the external reference even at greater distances travelled. There is no drift in the position on a longer trajectory, and the filter continuously converges to the correct solution. It can be seen that the position estimation accuracy is

closely related to the positioning accuracy from the UWB localization, and thus when CART2 passes very close to the initiator anchor, the position estimation get worse. Unfortunately, in all experiments, there was a problem with determining the orientation. The results do not very much correspond to the results of the experiments in the Datavision s.r.o. Laboratory, I propose that the problem may have occurred during the actual measurement in the IMR CIIRC during the setup of the CART2 platform. But unfortunately, I was not able to analyze it properly by the deadline for submitting this thesis.
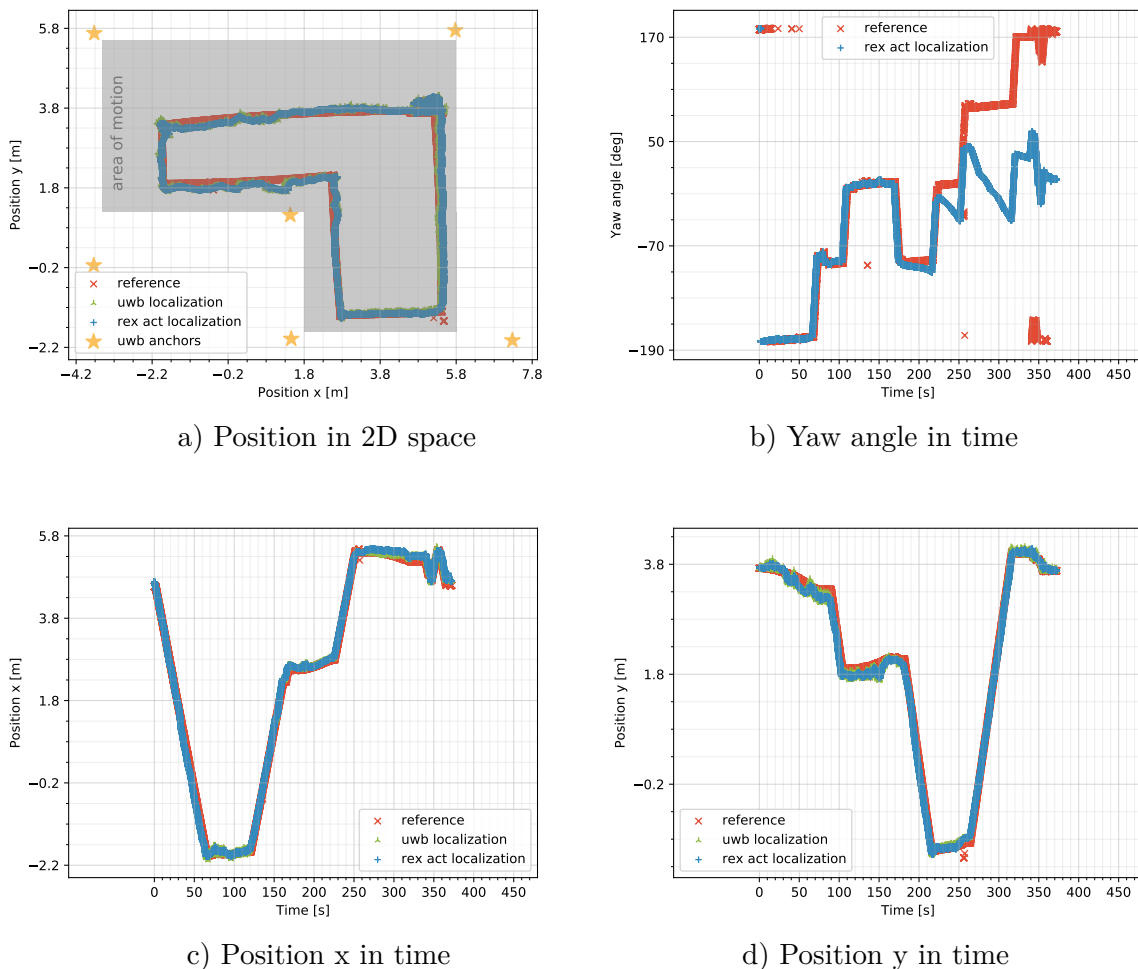


a) Position in 2D space

b) Yaw angle in time

c) Position x in time

d) Position y in time

**Figure 5.24.** Rectangle-shaped test, analysis of trajectory, at IMR CIIRC

The experiment was repeated a total of three times. Analysis of error is given in Figure 5.25. At the beginning of each iteration, the system rebooted. The system's errors turn out to be

- repeatable, as the error values do not differ significantly from each other,
- and accurate, as the RMSE was less than 10 cm for all three passes.

The maximum values here could exceed 40 cm. However, these extremes are caused by outliers in the external reference.

Another interesting experiment was to go through several iterations of the same rectangle-shaped trajectory without shutting down the system. As can be seen in
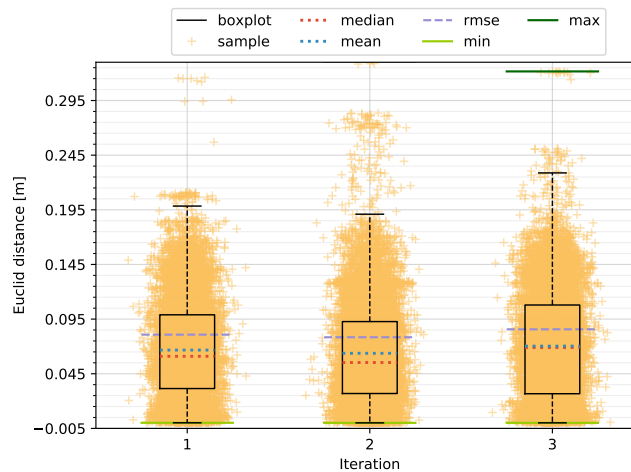
39

**Figure 5.25.** Rectangle-shaped tests: boxplot, rmse, min and max analysis of the position error, at IMR CIIRC

the Figure 5.26, the individual iteration does not differ from each other. It can therefore be seen that there is no significant drift in the position estimation.

**Infinity-shaped test**

During the infinity-shaped test in Figure 5.27, the position estimation is also sufficient. However, the detection from the global reference sometimes failed and created a few outliers. Thus, the results are a bit skewed.

**Joy ride test**

As shown in Figure 5.28, the position estimation worsens if the CART2 manipulates close to the initiator anchor. Also, it is worth to mention, that there is a wide column at [1.8, 1.0] m, which can cause signal reflections.

**Sumarize of experiments errors**

Comparison of position errors in individual experiments in Figure 5.29 shows interesting conclusions. Errors are similar for different test scenarios, not bigger than 30 cm, but usually, RMSE is close to 10 cm. Based on the result of the experiments, the proposed localization system seems viable for indoor localization with high accuracy and precision demands.
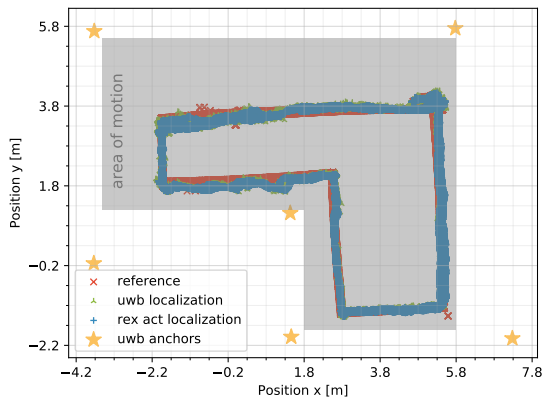
a) Position in 2D space
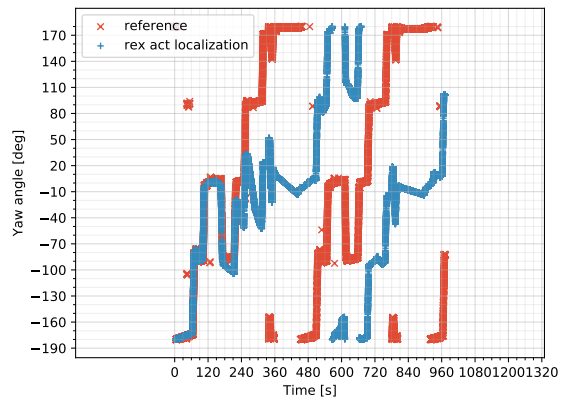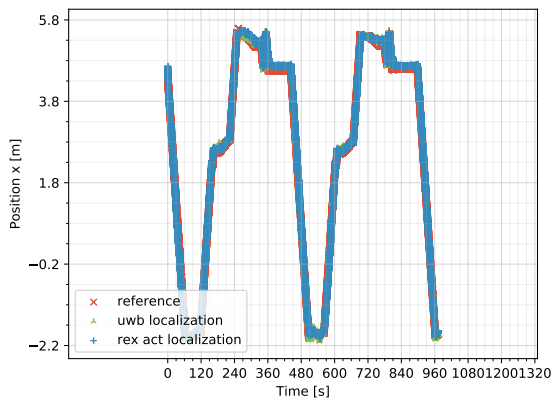
b) Yaw angle in time

c) Position x in time

d) Position y in time

**Figure 5.26.** Multiple rectangle-shaped test without shut down, analysis of trajectory, at IMR CIIRC
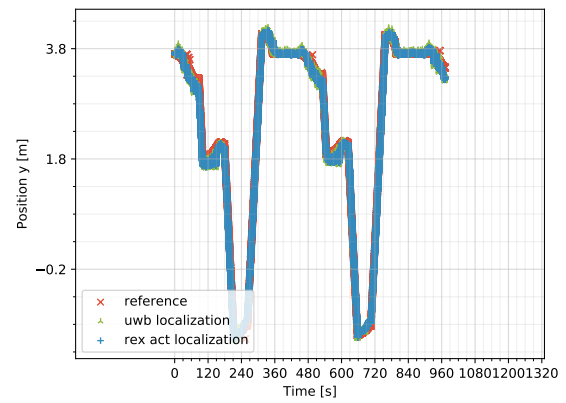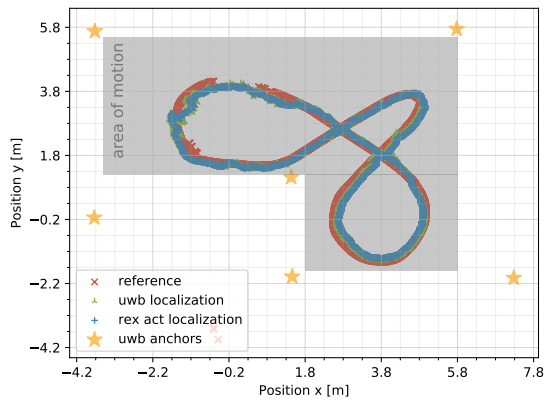
a) Position in 2D space
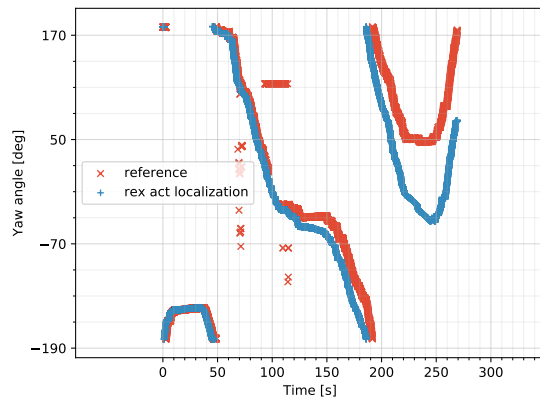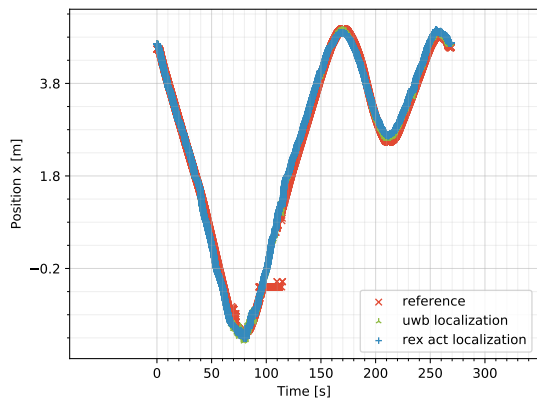
b) Yaw angle in time

c) Position x in time

d) Position y in time

**Figure 5.27.** Infinity-shaped test, analysis of trajectory, at IMR CIIRC

a) Position in 2D space
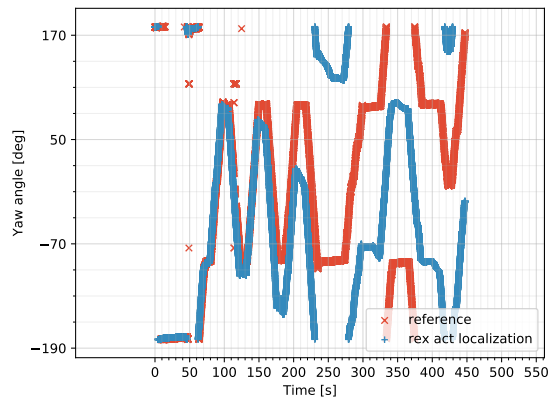


b) Yaw angle in time



c) Position x in time



d) Position y in time

**Figure 5.28.** Joy ride test, analysis of trajectory, at IMR CIIRC
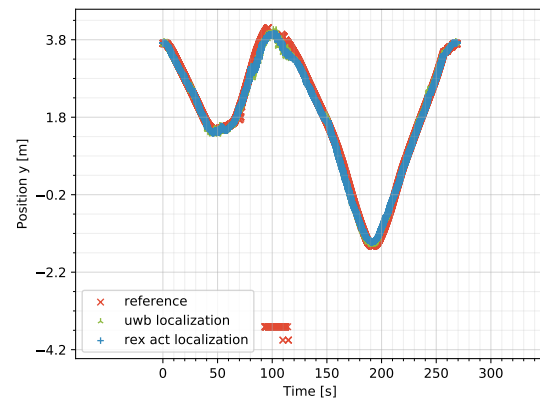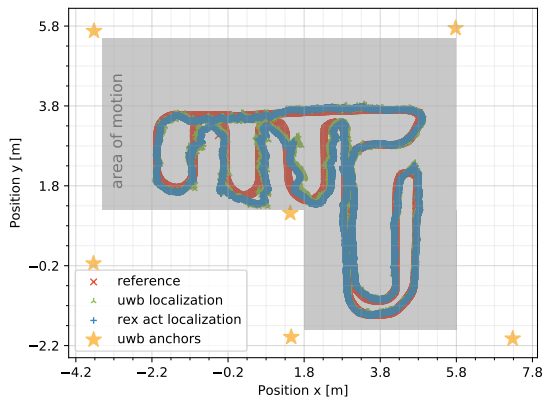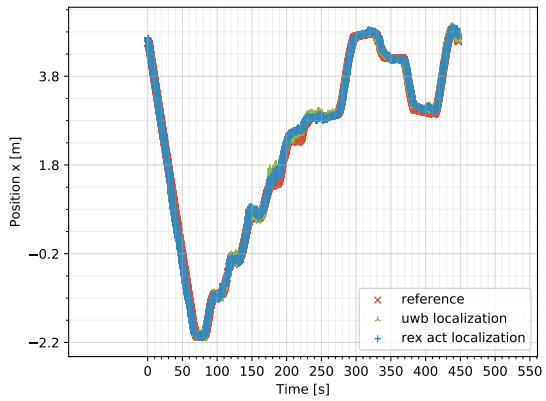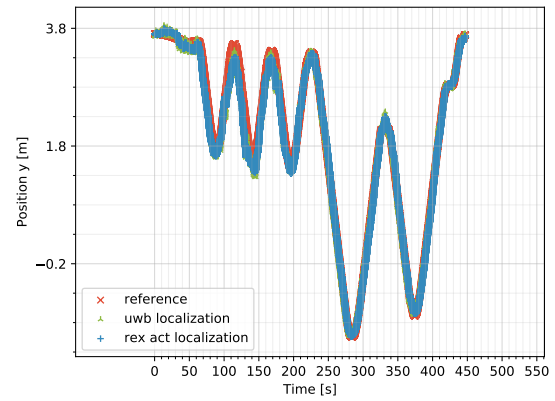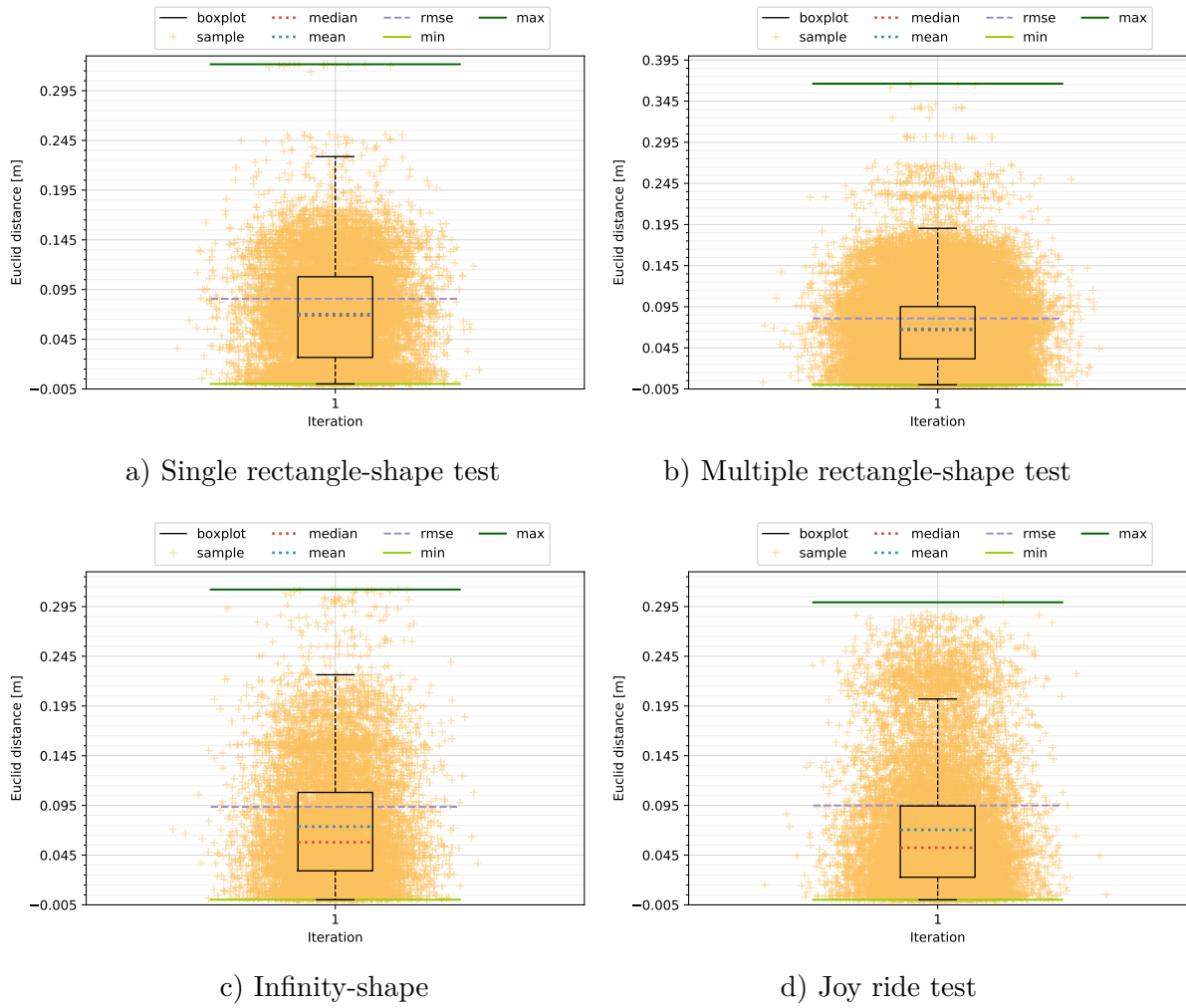
a) Single rectangle-shape test       b) Multiple rectangle-shape test

c) Infinity-shape       d) Joy ride test

**Figure 5.29.** Summary of errors in the position during experiments at IMR CIIRC

# Chapter **6**

## Conclusion and future work

## 6.1  Usage of the system in the industry

As I already mentioned in Chapter 1, the key features for industrial usage of localization are

- reliability of the system in the long term while keeping the number of manual interventions at a minimal,
- reduce the influence of the industrial environment on the system,
- ability for the system to be deployed on different robotic platforms,
- precision and repeatability.

The localization system proposed in this thesis is designed as an aided navigation system. The main component is the inertial navigation system which provides short term accurate relative location but is affected by drift. The estimated INS error removes the drift based on the absolute localization from the UWB beacon and the linear velocity from the AGV's odometry.

The proposed implementation reduces the influence of the structure of the vehicle or the morphology of the ground. Also, the localization principle is not based on any external landmarks or environment contours, reducing the requirement for a static environment to make the localization reliable in the long term without any manual interventions. From this point of view, the system architecture satisfies the industrial requirements quite well.

The discussion about accuracy is directed mainly to UWB performance because the accuracy of the UWB localization defines the accuracy of the whole system. UWB localization is relatively accurate and functions well when the robot is moving. On the contrary, when standing still, it is necessary to reduce the effect of UWB localization on the results. This effect can be very well ensured by the so-called virtual sensors when an artificially created signal enters the filtering at a suitable moment. For example, when the vehicle is not moving, the virtual sensor produces zero velocity. This approach is suggested in a possible future improvement of the system.

The accuracy of the UWB system is also reduced in No-Line-of-Sight environments, and therefore, it is necessary to have a relatively dense beacon network so that such phenomenas do not occur. The location of the beacon and the precise estimation of their positions are significant for the entire system's resulting accuracy.

Another important aspect is the actual mounting of sensors on the robotic platform. The placement of the UWB tag especially affects the resulting precision of UWB localization. Two positions of the UWB tag on the CART2 platform were evaluated during the experiments. The first position is near the metal parts of the robot and 0.2 m above the ground. The second is 1.0 m above the robotic platform without any metal part around. The 6.1 graph shows that possible signal reflections from the ground or metal parts on the robot distort the UWB localization.
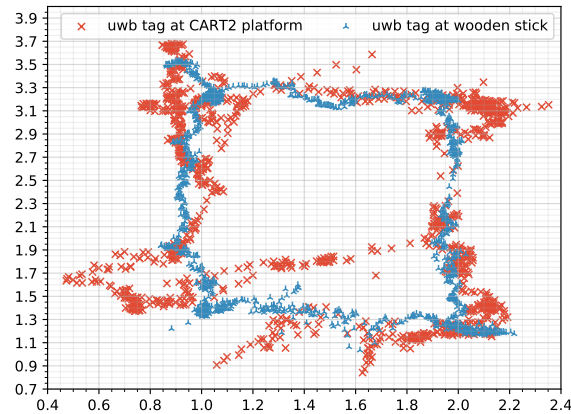
**Figure 6.1.** Comparison of the influence of two different mounted UWB tags on UWB localization.

Experiments should be performed in a larger industrial environment to evaluate the precision and repeatability of the system properly. Unfortunately, it was not possible to provide such an environment because installing anchors themselves and providing an external localization system is costly. However, the accuracy and repeatability of the localization are illustrated in Figures 5.18, 5.19, 5.25 and 5.29, where it can be seen that similar results are obtained with repeated passes from the same starting position and with the same control displays.

As shown in this section, this system meets the latest requirements for use in industry, and therefore, it would be appropriate to test it in more realistic scenarios directly in an industrial environment.

## 6.2 Conclusion

This thesis aims to design an indoor real-time localization system for autonomous ground vehicles based on ultra wide-band technology.

To properly understand the topic, it was necessary to study the properties of the sensors used to select individual components for implementation on the AGV platform CART2. Namely, ultra wide-band technology, inertial measurement unit, very closely related inertial navigation systems and odometry.

Furthermore, it was necessary to study the latest algorithms commonly used to estimate state based on the sensors mentioned above, such as the Kalman and the Particle filter algorithms. The whole state fusion is then based on the Error state Extended Kalman filter, which seems most suitable.

The proposed system's architecture is based on an inertial navigation system aided with measurements from absolute localization given by the UWB localization system and odometry of the AGV. Results show a significant decrement of the positioning error compared to the UWB localization and reliable attitude and position estimation without significant drift. The system was evaluated following a standardized testing method, considering the horizontal position error and the yaw angle as the primary performance metrics. The experimental results reported in this thesis demonstrate the possibility to employ the proposed localization system in industrial indoor environments.

The whole system was implemented in ROS2 and C++ and deployed on a real robotic platform, CART2. For a successful deployment, it was necessary to focus on the specific sensors used, such as the UWB localization system MDEK1001 from Qorvo, the IMU in the M-G365PDF1 model from Epson the odometry read from the Maxon Epos4 wheen encoder.

The system was then experimentally validated in two test environments using two different external references. The test scenarios were appropriately selected to verify the behavior of the system in all possible situations.

And the last part summarizes the possible use of the system in the industry, where it was shown that the system's design meets the latest requirements and then presented several proposals for improving and expanding the existing system.

## 6.3 Future work

This thesis shows that the proposed localization system can be successfully used to estimate the pose of autonomous ground vehicles. Future work is divided into two main categories.

- Possible improvements of the algorithms and hardware specifics
- and testing the system in more complex scenarios.

### 6.3.1 Possible improvements of the system

Multiple possible improvements have been identified during work on this thesis, namely

- using Reverse Time Difference of Arrival localization mode,
- implementing virtual sensors
- and suggest a suitable calibration process.

The first possible improvements are aimed at using **UWB localization in RTDoA mode**. This mode offers a higher update rate and the ability to use the system on multiple AGVs at once, as mentioned in section 2.1.

Another option is to use **virtual sensors** that improve the precision of the entire system and are commonly used in similar applications [46–47].

For a long-term reliable localization system, it is also possible to design **calibration processes**. The calibration enables obtaining the most precise data from sensors and is valid for sensors such as IMU or odometry. In present systems, calibrations are typically applied before the entire system is switched on and during the ride. A suitable calibration process should be part of the final localization product.

### 6.3.2 Experimental verification in the complex testing scenario

The initial experiments evaluated in this work look promising. The next step is to verify the system in a more complex test environment. The test scenarios should include

- testing the system on *multiple AGVs at once*,
- experimentally *verifying* whether *the proposed system* works for localization *in 3D*, especially *on flying drones*,
- verifying functionality *in an industrial environment* under the demand of industrial partners,
- finding out the possible influence of *moving people in the environment* (the human body could be a significant obstacle for UWB signals).

Extensive testing is planned to evaluate localization system performance thoroughly. This will allow the final product to be robust and reliable for use in the target environment.

# References

[1] V. Lakkundi. Ultra Wideband Communications: History, Evolution and Emergence. *Acta Polytechnica.* 2006, 46 DOI 10.14311/844.

[2] Zafer Sahinoglu, Sinan Gezici, and Ismail Gvenc. *Ultra-Wideband Positioning Systems: Theoretical Limits, Ranging Algorithms, and Protocols.* USA: Cambridge University Press, 2011. ISBN 0521187834.

[3] Sewio Networks. *Two way ranging illustration.* https://cdn.sewio.net/wp-content/uploads/2016/04/TWR-Scheme.jpg.

[4] Vincenzo Di Pietra, Paolo Dabove, and Marco Piras. Loosely Coupled GNSS and UWB with INS Integration for Indoor/Outdoor Pedestrian Navigation. *Sensors.* 2020, 20 (21), DOI 10.3390/s20216292.

[5] Qigao Fan, Biwen Sun, Yan Sun, and Xiangpeng Zhuang. Performance Enhancement of MEMS-Based INS/UWB Integration for Indoor Navigation Applications. *IEEE Sensors Journal.* 2017, 17 (10), 3116-3130. DOI 10.1109/JSEN.2017.2689802.

[6] Bruno Siciliano, and Oussama Khatib. *Springer Handbook of Robotics.* Springer-Verlag, 2007.

[7] Peter Teunissen, and Oliver Montenbruck. *Springer handbook of global navigation satellite systems.* Springer, 2017.

[8] Priyanka Aggarwal. *MEMS-based integrated navigation.* Artech House, 2010.

[9] N. Barbour, and G. Schmidt. Inertial sensor technology trends. *IEEE Sensors Journal.* 2001, 1 (4), 332-339. DOI 10.1109/7361.983473.

[10] William Riley, and David Howe. *Handbook of Frequency Stability Analysis.* 2008. https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=50505.

[11] Shuvra S Bhattacharyya, Ed F Deprettere, Rainer Leupers, and Jarmo Takala. *Handbook of signal processing systems.* Springer, 2018.

[12] Seiko Epson Corporation. *Inertial Measurement Unit (IMU) : M-G365.* https://global.epson.com/products_and_drivers/sensing_system/imu/g365/.

[13] Agnieszka Szczęsna, Przemysław Skurowski, Ewa Lach, Przemyslaw Pruszowski, Damian Pęszor, Marcin Paszkuta, Janusz Słupik, Kamil Lebek, Mateusz Janiak, Andrzej Polanski, and Konrad Wojciechowski. Inertial Motion Capture Costume Design Study. *Sensors.* 2017, 17 612. DOI 10.3390/s17030612.

[14] Wikipedia®. *Wikipedia - white noise definition.* https://en.wikipedia.org/wiki/White_noise.

[15] © Thales group. *Performance of IMU per application.* https://www.thalesgroup.com/sites/default/files/database/d7/

`assets/images/thales_topaxyz_imu_infographie_copyright_thales_l`
`ight_0.png`.

[16] B. Barshan, and H.F. Durrant-Whyte. Inertial navigation systems for mobile robots. *IEEE Transactions on Robotics and Automation.* 1995, 11 (3), 328-342. DOI 10.1109/70.388775.

[17] *Inertial Navigation Systems.* In: *Global Positioning Systems, Inertial Navigation, and Integration.* John Wiley and Sons, Ltd, 2007. 9. ISBN 9780470099728.
`https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470099728.`
`ch9`.

[18] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics.* Cambridge, Mass.: MIT Press, 2005 . ISBN 0262201623 9780262201629.

[19] Nak Yong Ko, and Tae Gyun Kim. *Comparison of Kalman filter and particle filter used for localization of an underwater vehicle.* In: *2012 9th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI).* 2012. 350-352.

[20] Mushfiqul Alam, Ginés López, Martin Sipos, and Jan Rohac. *INS/GNSS Localization Using 15 State Extended Kalman Filter.* In: 2016.

[21] M. Petovello. *Real-time integration of a tactical-grade IMU and GPS for high-accuracy positioning and navigation.* In: 2003.

[22] Gianluca Falco, Marco Pini, and Gianluca Marucco. Loose and Tight GNSS/INS Integrations: Comparison of Performance Assessed in Real Urban Scenarios. *Sensors.* 2017, 2017 27. DOI 10.3390/s17020255.

[23] Shubham Godha. Performance Evaluation of Low Cost MEMS-Based IMU Integration with GPS for Land Vehicle Navigation Application. 2006,

[24] Simon J. Julier, and Jeffrey K. Uhlmann. *New extension of the Kalman filter to nonlinear systems.* In: Ivan Kadar, eds. *Signal Processing, Sensor Fusion, and Target Recognition VI.* SPIE, 1997. 182 – 193.
`https://doi.org/10.1117/12.280797`.

[25] Jiaxin Li, Yingcai Bi, Kun Li, Kangli Wang, Feng Lin, and Ben Chen. *Accurate 3D Localization for MAV Swarms by UWB and IMU Fusion.* In: 2018. 100-105.

[26] Jeroen D. Hol, Fred Dijkstra, Henk Luinge, and Thomas B. Schon. *Tightly coupled UWB/IMU pose estimation.* In: *2009 IEEE International Conference on Ultra-Wideband.* 2009. 688-692.

[27] S.I. Roumeliotis, G.S. Sukhatme, and G.A. Bekey. *Circumventing dynamic modeling: evaluation of the error-state Kalman filter applied to mobile robot localization.* In: *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C).* 1999. 1656-1663 vol.2.

[28] Joan Solà. Quaternion kinematics for the error-state KF. 2015,

[29] Jay Farrell. *Aided navigation: GPS with high rate sensors.* McGraw-Hill, Inc., 2008.

[30] Alessandro Benini, Adriano Mancini, and Sauro Longhi. An IMU/UWB/Vision-based Extended Kalman Filter for Mini-UAV Localization in Indoor Environment using 802.15.4a Wireless Sensor Network. *Journal of Intelligent and Robotic Systems.* 2013, 70 DOI 10.1007/s10846-012-9742-1.

[31] Venkatesh Madyastha, Vishal Ravindra, Srinath Mallikarjunan, and Anup Goyal. *Extended Kalman Filter vs. Error State Kalman Filter for Aircraft Attitude Estimation.* In: 2011. ISBN 978-1-60086-952-5.

[32] Lukasz Zwirello, Xuyang Li, Thomas Zwick, Christian Ascher, Sebastian Werling, and Gert Trommer. *Sensor data fusion in UWB-supported inertial navigation systems for indoor navigation.* In: 2013. 3154-3159. ISBN 978-1-4673-5641-1.

[33] Jay A. Farrell, and Paul F. Roysdon. Advanced Vehicle State Estimation: A Tutorial and Comparative Study. *IFAC-PapersOnLine.* 2017, 50 (1), 15971-15976. DOI https://doi.org/10.1016/j.ifacol.2017.08.1751. 20th IFAC World Congress.

[34] M. D. Shuster. Survey of attitude representations. *Journal of the Astronautical Sciences.* 1993, 41 (4), 439-517.

[35] Mike Purvis ROS, Tully Foote. *Standard Units of Measure and Coordinate Conventions.*
`https://www.ros.org/reps/rep-0103.html`.

[36] Gaël Guennebaud, Benoît Jacob, and others. *"Eigen", a C++ template library for linear algebra: matrices, vectors, numerical solvers, and related algorithms.* http://eigen.tuxfamily.org. 2010.

[37] Michel Hidalgo ROS, Tully Foote. *ROS geometry messages package.*
`http://wiki.ros.org/geometry_msgs`.

[38] Tully Foote. *tf: The transform library.* Open-Source Software workshop. 2013.

[39] Dirk Thomas, William Woodall, and Esteve Fernandez. *Next-generation ROS: Building on DDS.* In: *ROSCon Chicago 2014.* Mountain View, CA: Open Robotics, 2014.
`https://vimeo.com/106992622`.

[40] Seiko Epson Corporation. *Datasheet of Epson M-G365.*
`https://global.epson.com/products_and_drivers/sensing_system/`
`download_hidden/pdf/m-g365pd_datasheet_e_rev20201007.pdf`.

[41] Anders E. E. Wallin Revision dd987cf3 © Copyright 2014-2019. *Allantools python library.*
`https://allantools.readthedocs.io/en/latest/index.html`.

[42] Inc © 2021 Qorvo. *Qorvo's MDEK1001 ultra-wideband (UWB) development kit.*
`https://www.qorvo.com/products/p/MDEK1001`.

[43] Edwin Olson. *AprilTag: A robust and flexible visual fiducial system.* In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA).* IEEE, 2011. 3400-3407.

[44] Niceboy®. *niceboy ® STREAM PRO camera.*
`https://niceboy.eu/en/products/stream-pro`.

[45] Pierre Merriaux, Yohan Dupuis, Rémi Boutteau, Pascal Vasseur, and Xavier Savatier. A Study of Vicon System Positioning Performance. *Sensors.* 2017, 17 (7), DOI 10.3390/s17071591.

[46] Yueyang Ben, Guisheng Yin, Wei Gao, and Feng Sun. *Improved filter estimation method applied in zero velocity update for SINS.* In: *2009 International Conference on Mechatronics and Automation.* 2009. 3375-3380.

[47] Rahul P Suresh, Vinay Sridhar, J Pramod, and Viswanath Talasila. *Zero Velocity Potential Update (ZUPT) as a Correction Technique*. In: *2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU)*. 2018. 1-8.

# Appendix A
## Abbreviations and symbols

## A.1  A list of abbreviations

All abbreviations used in this thesis are listed below.

| | |
|---|---|
| AGV | Autonomous ground vehicles. |
| AVAR | Allan variance. |
| CIIRC | Czech Institute of Informatics, Robotics and Cybernetics. |
| CTU | Czech Technical University in Prague. |
| DOF | Degrees of freedom. |
| ES-EKF | Error state Extended Kalman filter. |
| GNSS | Global Navigation Satellite System. |
| IMU | Inertial measurement unit. |
| INS | Inertial navigation system. |
| MEMS | Microelectromechanical systems. |
| ROS2 | Robot Operating System 2. |
| TACR | Technology Agency of the Czech Republic. |
| TWR | Two-way ranging. |
| TDoA | Time Difference of Arrival. |
| RMSE | Root mean square error. |
| RTDoA | Reverse Time Difference of Arrival. |
| UWB | Ultra-wideband. |

## A.2  A list of symbols

$\oplus$  is a suitable composition as linear sum or matrix product.