



Eva Reitbauer, BSc.

# **Radio Fingerprinting Optimization Tailored to Vehicles in Parking Garages**

## **MASTER'S THESIS**

to achieve the university degree of

Diplom-Ingenieurin

Master's degree programme: Geomatics Science

submitted to

**Graz University of Technology**

Supervisor

Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Manfred Wieser

Institute of Geodesy

Faculty of Mathematics, Physics and Geodesy

Graz, October 2017

## **AFFIDAVIT**

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

---

Date

---

Signature

# Acknowledgements

I would like to thank my supervisor, Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Manfred Wieser for his expertise and support. Thank you for giving me the opportunity to write this thesis as a student project assistant in your working group.

A big thank you goes to Dipl.-Ing. Thomas Moder. Thank you for all the advice you have given me, for guiding me through almost a year of researching and programming, for your help with the IMU reference trajectory and for always taking the time to answer my questions.

Thank you, Dipl.-Ing. Roman Wilfinger, for all your help during the project. I cannot even express how grateful I am that you assisted me in measuring the whole radio map for the parking garage anew.

Thank you, Fachoberinspektor Bernd Mölg, for conducting the tachymetric measurements for the reference trajectory. A thank you also goes to Dipl.-Ing. Markus Dorn for his advice on the filter design.

Furthermore I want to thank Magna Steyr for giving me the opportunity to work on a research topic that has a practical application, and for providing the test vehicle and access to the parking garage.

I would like to thank my family, especially my parents, for their love and support. Thank you for having enabled me to study what I wanted, and for being my role models. A big thank you goes to my father for proofreading this thesis, for correcting my English and always giving me constructive feedback on how to improve my writing. Thank you for your endless patience and for taking so much time to read through everything I have written.

Thank you Fabian for your love and for always being there for me.

# Abstract

Nowadays, Global Navigation Satellite Systems (GNSS) enable us to determine our position whenever desired in outdoor areas. In spite of the global availability of GNSS, their signals cannot be used in indoor areas for positioning. A widely used indoor positioning method has therefore emerged in the last decade: fingerprinting.

Fingerprinting consists of two phases. There is an offline phase, where a radio map is recorded by measuring the signal strength of signals received in indoor areas (such as Bluetooth low energy or WLAN signals) at reference points. There is also an online positioning phase, where the same signals are measured and compared to the signals in the radio map to estimate a position. As the offline phase is time-consuming, there is a need to optimize fingerprinting.

This thesis focuses on radio fingerprinting optimization tailored to vehicles in parking garages. It evaluates deterministic and probabilistic fingerprinting methods regarding their optimization potential and investigates approaches to the algorithmic and economic optimization as well as an integrated solution with vehicle sensor data.

The algorithms developed were tested in a parking garage near Graz. The algorithmic optimization showed that the best results are achieved using deterministic Weighted K Nearest Neighbour (WKNN) fingerprinting with distance metrics based on the  $L_1$  norm. The economic optimization revealed that the achievable accuracy does not decrease when fingerprints are only recorded in areas where the vehicle is allowed to drive. The integration with vehicle sensor data lowered the Root Mean Square Error (RMSE) of the trajectories to less than 3 m.

# Kurzfassung

Globale Satellitennavigationssysteme (GNSS) ermöglichen heutzutage die weltweite Positionsbestimmung in Außenbereichen. Trotz der globalen Verfügbarkeit können GNSS-Signale in Innenräumen nicht zur Positionsbestimmung herangezogen werden. Eine weitverbreitete Positionierungsmethode in Innenbereichen ist das sogenannte Fingerprinting.

Fingerprinting besteht aus zwei Phasen, der Offline- und der Online-Phase. In der Offline-Phase wird eine Merkmalskarte aufgenommen, indem in Innenräumen verfügbare Signale (wie beispielsweise Bluetooth Low Energy oder WLAN Signale) an Referenzpunkten aufgenommen und gespeichert werden. In der Online-Phase werden dieselben Signale gemessen und mit den in der Merkmalskarte gespeicherten Signalen verglichen, um eine Position zu schätzen. Da die Offline-Phase sehr zeitaufwendig ist, besteht beim Fingerprinting Optimierungsbedarf.

Diese Masterarbeit beschäftigt sich mit der Optimierung von Fingerprinting mit Radiosignalen für Autos in Parkgaragen. Sowohl deterministische als auch probabilistische Fingerprinting Ansätze werden hinsichtlich ihres Optimierungspotentials untersucht. Hierbei werden die Optimierung der Algorithmen, die ökonomische Optimierung sowie die Integration von Fahrzeugtelemetriedaten diskutiert.

Die entwickelten Algorithmen wurden umfangreich in einer Parkgarage in der Nähe von Graz getestet. So zeigte die algorithmische Optimierung, dass sich deterministisches Fingerprinting mit einem gewichteten K-nächste-Nachbarn Ansatz und  $L_1$ -basierten Normen für Fahrzeuge in Parkgaragen am besten eignet. Als Ergebnis der ökonomischen Optimierung wurde festgestellt, dass es ausreicht, wenn nur Referenzpunkte an tatsächlich befahrbaren Bereichen der Garage aufgenommen werden. Durch die integrierte Positionslösung mit Fahrzeugtelemetriedaten konnte die mittlere Abweichung der Trajektorien von der Referenztrajektorie auf unter 3 Meter verringert werden.

# Abbreviations

<b>ABS</b>	Anti-lock Braking System
<b>AoA</b>	Angle of Arrival
<b>AP</b>	Access Point
<b>BLE</b>	Bluetooth Low Energy
<b>BLUE</b>	Best Linear Unbiased Estimator
<b>CAN</b>	Controller Area Network
<b>DR</b>	Dead Reckoning
<b>ECU</b>	Engine Control Unit
<b>EKF</b>	Extended Kalman Filtering
<b>ESP</b>	Electronic Stability Programme
<b>GK</b>	Gauß-Krüger
<b>GNSS</b>	Global Navigation Satellite Systems
<b>GPS</b>	Global Positioning System
<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>IMU</b>	Inertial Measurement Unit
<b>ISM</b>	Industrial, Scientific and Medical
<b>ISO</b>	International Organization for Standardization
<b>LIN</b>	Local Interconnected Network
<b>MAC</b>	Medium Access Control
<b>MEMS</b>	Microelectromechanical System
<b>ML</b>	Maximum Likelihood
<b>MOST</b>	Media Oriented System Transport
<b>PDR</b>	Pedestrian Dead Reckoning
<b>QQ</b>	Quantile-Quantile
<b>RMSE</b>	Root Mean Square Error
<b>RP</b>	Reference Point
<b>RSS</b>	Received Signal Strength

---

<b>SIG</b>	Special Interest Group
<b>SPSC</b>	Signal Processing and Speech Communication
<b>SSID</b>	Service Set Identifier
<b>TDoA</b>	Time Difference of Arrival
<b>ToA</b>	Time of Arrival
<b>UUID</b>	Universally Unique Identifier
<b>UWB</b>	Ultra-Wideband
<b>WAF</b>	Wall Attenuation Factor
<b>WGS84</b>	World Geodetic System 1984
<b>WKNN</b>	Weighted K Nearest Neighbour
<b>WLAN</b>	Wireless Local Area Network

# Contents

Acknowledgements	III
Abstract	IV
Kurzfassung	V
Abbreviations	VI
<b>I Theoretical foundations</b>	<b>1</b>
<b>1 Introduction</b>	<b>2</b>
1.1 Motivation . . . . .	2
1.2 Research aims . . . . .	3
1.3 Outline . . . . .	4
<b>2 Radio signals in indoor navigation</b>	<b>5</b>
2.1 Wireless local area networks . . . . .	5
2.1.1 IEEE standardization . . . . .	6
2.1.2 Indoor positioning using wireless local area networks . . . . .	7
2.2 Bluetooth low energy . . . . .	7
2.2.1 Specification . . . . .	8
2.2.2 Beacons . . . . .	9
2.2.3 Advantages of Bluetooth low energy in indoor positioning . . . . .	10
2.3 Properties of the received signal strength . . . . .	11
<b>3 Fingerprinting</b>	<b>12</b>
3.1 Basic principles . . . . .	12
3.2 Deterministic fingerprinting . . . . .	14

3.3	Probabilistic fingerprinting . . . . .	15
3.3.1	Probabilistic radio map . . . . .	16
<b>II Optimization theory</b>		<b>21</b>
<b>4</b>	<b>Algorithmic optimization</b>	<b>22</b>
4.1	Deterministic fingerprinting . . . . .	22
4.1.1	Weighted k nearest neighbour . . . . .	22
4.1.2	Norms . . . . .	23
4.2	Probabilistic fingerprinting . . . . .	26
4.2.1	Optimal kernel bandwidth . . . . .	26
4.2.2	Weighting . . . . .	28
<b>5</b>	<b>Economic optimization</b>	<b>30</b>
5.1	Sparse fingerprinting . . . . .	31
5.1.1	Radio map interpolation . . . . .	32
5.2	Automatic updating of radio maps . . . . .	34
5.3	Fully automatic creation of radio maps . . . . .	35
<b>6</b>	<b>Integration of vehicle sensor data</b>	<b>38</b>
6.1	Sensor fusion . . . . .	38
6.1.1	Kalman filtering . . . . .	40
6.1.2	Stages of integration . . . . .	42
6.2	Vehicle sensor data . . . . .	46
6.3	Real-time filter . . . . .	48
6.3.1	Pre-processing . . . . .	49
6.3.2	Master filter . . . . .	50
6.3.3	Pseudo-heading update . . . . .	55
<b>III Practical investigations and results</b>		<b>57</b>
<b>7</b>	<b>Practical investigations</b>	<b>58</b>
7.1	RSS distribution . . . . .	58
7.2	Test bed for probabilistic fingerprinting . . . . .	62
7.3	First real-time tests . . . . .	65
7.3.1	Comparison to the reference trajectory . . . . .	67

## CONTENTS

---

7.3.2	Results . . . . .	69
7.4	Recording a new radio map for the Thondorf parking garage . . . . .	70
7.5	Final test measurements . . . . .	72
<b>8</b>	<b>Results</b>	<b>75</b>
8.1	Algorithmic optimization . . . . .	75
8.1.1	Deterministic fingerprinting . . . . .	75
8.1.2	Probabilistic fingerprinting . . . . .	80
8.2	Economic optimization . . . . .	82
8.2.1	Reducing the number of reference points . . . . .	82
8.2.2	Reducing the number of access points . . . . .	89
8.3	Integration of vehicle sensor data . . . . .	91
8.4	Conclusion and outlook . . . . .	97
	<b>Bibliography</b>	<b>XI</b>
	<b>List of Figures</b>	<b>XVIII</b>
	<b>List of Tables</b>	<b>XXI</b>

# Part I

## Theoretical foundations

# Chapter 1

## Introduction

This chapter aims at giving a general introduction to this Master's thesis on *radio fingerprinting tailored to vehicles in parking garages*. Section 1.1 discusses the motivation behind this thesis, explaining in which scenarios fingerprinting is needed, and evaluating why fingerprinting still needs to be optimized. The research aims are then covered in Section 1.2. To conclude this chapter, an overview of the structure of this thesis is given in Section 1.3.

### 1.1 Motivation

This section aims at explaining the relevance of fingerprinting and the need to optimize the process.

For several millennia, travelling the world has been of interest to mankind. Military interests such as the expansion of territory, economic interests, especially trade, and scientific advancements through the exchange of knowledge with other cultures could not have been realized without navigation, and navigation requires the knowledge of one's position.

Nowadays we are used to being able to determine our position whenever desired through Global Navigation Satellite Systems (GNSS). However, as soon as we move indoors, be it as a pedestrian entering a big building or as a driver entering a parking garage, GNSS signals can no longer be used for positioning. In the last years, fingerprinting has become a standard method in absolute indoor positioning.

Fingerprinting makes use of signals of opportunity, i.e. signals which are already present in indoor environments such as magnetic field strength, signals within Wireless Local Area Networks (WLAN) or Bluetooth Low Energy (BLE) signals. If the latter two are used, we speak of radio fingerprinting. In general, radio fingerprinting consists of two phases. In the offline phase or training phase of radio fingerprinting, Received Signal Strengths (RSS) are measured at Reference Points (RP) and stored in a radio map along with the coordinates of the reference point and information from which Access Points (AP) the RSS were recorded. In the online phase or positioning phase, RSS are recorded and matched to the radio map to estimate a position.

Given that fingerprinting has already become a common indoor positioning method, why is there a need for optimization? One of the main drawbacks of fingerprinting is that the offline phase is extremely time-consuming. Radio maps change over time and need to be updated regularly; it is therefore desirable to minimize the time needed to create and update radio maps. Additionally, hardware and hardware maintenance costs can still be minimized. The algorithms used have potential for improvement in order to describe relationship between RSS and position in a more stable manner. If additional data, e.g. relative positioning data, is present to support the algorithm, it can be used to compute an integrated positioning solution.

## 1.2 Research aims

The aim of this thesis is to compare existing algorithms for fingerprinting and evaluate their optimization potential. Both deterministic and probabilistic fingerprinting are investigated in terms of their principles, functioning and potential algorithmic optimization.

Furthermore, sparse fingerprinting as well as the automated interpolation of fingerprints with path loss models are analysed in order to optimize the time needed to create radio maps.

As this thesis focuses on fingerprinting tailored to vehicles, an integrated positioning solution using both fingerprinting and vehicle sensor data is developed for real-time applications. Finally, the evaluated algorithms are tested in a parking garage.

## 1.3 Outline

This thesis is divided into three parts. Part I deals with the theoretical foundations of radio fingerprinting; it includes a chapter on commonly used radio signals in indoor navigation and examines the fundamental principles of fingerprinting. Part II presents the theoretical background of fingerprinting optimization and comprises chapters on algorithmic and economic optimization as well as the combination of fingerprinting with vehicle sensor data. Part III describes the practical investigations carried out during the research project and presents the results as well as an outlook for further research.

# Chapter 2

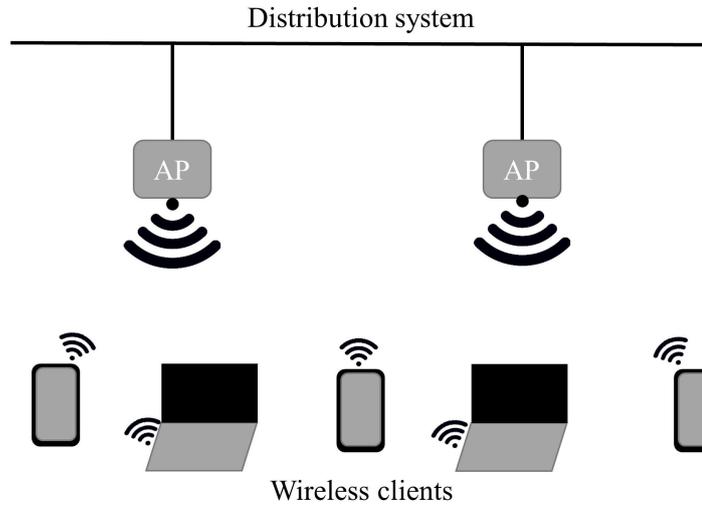
## Radio signals in indoor navigation

This chapter will focus on the two types of radio signals which are commonly used in indoor navigation in combination with fingerprinting, namely Wireless Local Area Network (WLAN) and Bluetooth Low Energy (BLE). Both technologies are based on electromagnetic radio waves and operate in the unlicensed Industrial, Scientific and Medical (ISM) band. The properties of both technologies and their application in indoor positioning will be discussed. Furthermore, the attributes of the Received Signal Strength (RSS) are investigated.

### 2.1 Wireless local area networks

This section will introduce WLAN and give a brief overview of its standardization and application in indoor positioning.

WLAN is a technology used to wirelessly connect computers which are close to each other. WLAN can operate in two modes, *ad-hoc mode* and *infrastructure mode* [7]. In an *ad-hoc* network, two computers communicate peer-to-peer, i.e. they are hierarchically equal. In *infrastructure mode*, two or more devices are connected to the network using a distribution system which is realized by one or more Access Points (AP). A schematic view of the system architecture of WLAN operating in infrastructure mode is given in Figure 2.1. Usually, AP are at a fixed location and wireless clients connect to them to join the WLAN. All AP and clients have a Medium Access Control (MAC) address, which allows identifying them uniquely within the network. Most WLAN operate in infrastructure mode, which is also the mode used in indoor positioning systems.



**Figure 2.1:** System architecture of WLAN in infrastructure mode.

**Table 2.1:** Release year, spectrum and channel width of the most important WLAN protocols according to [40] and [48]

Protocol	Release year	Spectrum [GHz]	Channel width [MHz]
802.11a	1999	5	20
802.11b	1999	2.4	22
802.11g	2003	2.4	20
802.11n	2009	2.4	20/40
		5	20/40
802.11ac	2013	5	20/40/80/160

### 2.1.1 IEEE standardization

WLAN is a standard of the Institute of Electrical and Electronics Engineers (IEEE) 802.11 family, with its first specification published in 1997. The initial specification only defined the use of the 2.4 GHz band with data rates of 1 or 2 Mbit/s. Later the standard was expanded and the working group was split into two task groups working with a 2.4 GHz band (IEEE 802.11b) and a 5 GHz band (IEEE 802.11a) [27]. Ever since then, the development of the standard has led to an increase in performance and at the same time a decrease in costs [42]. With the growth of the market, the standards 802.11g, 802.11n, 802.11ac emerged, allowing data rates of up to 1300 Mbit/s (IEEE 802.11ac at a bandwidth of 80 MHz). Table 2.1 gives an overview of the most important standards and channel bandwidths.

### 2.1.2 Indoor positioning using wireless local area networks

Positioning in WLAN environments is attractive due to the fact that WLAN have become ubiquitous in modern indoor environments and no additional hardware is needed [51], since all smartphones and computers are capable of exchanging data using the IEEE 802.11 standard. Indoor positioning using WLAN is based on beacon signals which are transmitted between the fixed access points and the mobile clients.

According to Kushki et al. (2012), four types of radio signal features in WLAN can be used for positioning:

- **Angle of Arrival (AoA):** The angle of the arriving wave is measured with an antenna. With waves coming from two or more AP with known coordinates, the position of the antenna can be calculated.
- **Time of Arrival (ToA):** The time which the signal needs to travel from the AP to the receiver is used to calculate the distance between them. With three or more distances and the known coordinates of the AP in a two dimensional coordinate system, the position of the client can be computed through circular lateration.
- **Time Difference of Arrival (TDoA):** At two receivers, the ToA is measured and the difference between the two ToAs is calculated. Hyperbolic lateration can be used to calculate the position of the user with three or more TDoA measurements.
- **RSS:** The received signal power at the user's location is measured. As the distance increases, the RSS decreases. However, the RSS is also influenced by other factors, which will be further discussed in Section 2.3.

AoA requires special antennae to detect the direction on the incoming radio wave, which leads to additional hardware costs. For ToA and TDoA, the transmitters must be precisely synchronised, which is not easily achievable in WLAN systems [29]. The most commonly used feature of WLAN radio signals used in indoor positioning is therefore RSS.

## 2.2 Bluetooth low energy

BLE is a technology which emerged in 2009, when the Special Interest Group (SIG) published the Bluetooth standard 4.0. It is also known as *Bluetooth Smart*. In contrast to

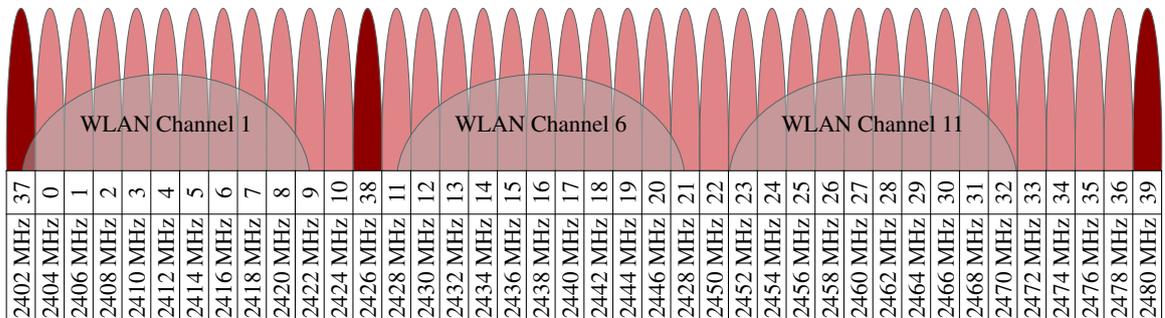
classic Bluetooth, where the data rates were increased with every published standard, BLE was designed for very low power consumption [19]. Fulfilling the low power requirements requires the data rate to be lowered, as can be seen in Table 2.2. BLE can be used for a number of applications in the fitness and health care sectors, in wearable devices, proximity sensing and car electronics [23].

**Table 2.2:** Development of the data rates of the Bluetooth standard according to Heydon (2013)

Version	Data rate
Bluetooth v1.1	1 Mbps
Bluetooth v2.0	3 Mbps
Bluetooth v3.0	54 Mbps
Bluetooth v4.0 (BLE)	0.3 Mbps

## 2.2.1 Specification

Bluetooth and BLE, like 802.11b, 802.11g, and 802.11n WLAN, operate on the 2.4 GHz ISM band. Both BLE and Bluetooth classic use frequency hopping. While classic Bluetooth uses 79 channels with a bandwidth of 1 MHz, BLE operates on 40 channels with a bandwidth of 2 MHz each [4]. Of the 40 channels, BLE only uses three to broadcast its identifier while the other 37 channels are used for data transmission. The three advertisement channels (37, 38 and 39) are centred at 2402 MHz, 2426 MHz, and 2428 MHz, respectively, and are depicted in dark red in Figure 2.2. Their frequencies were chosen to minimize interference with the commonly used WLAN channels 1, 6 and 11 [15], which are also displayed in Figure 2.2.



**Figure 2.2:** BLE spectrum and three WLAN channels in the 2.4 GHz band

As BLE uses different channels than Bluetooth classic, they are not compatible. Therefore, three types of devices have existed since the Bluetooth 4.0 standard was introduced [19]:

- **Single-mode:** Single-mode devices only support BLE. They can communicate with other single-mode devices as well as with dual-mode devices.
- **Dual-mode:** Dual-mode devices are capable of receiving both Bluetooth classic and BLE. They support communication with single-mode, other dual-mode and classic devices.
- **Classic:** Classic devices only support Bluetooth classic and cannot communicate with single-mode devices. However, they can communicate with dual-mode devices.

Since BLE was designed for low power consumption, BLE single-mode devices are small, low-cost and can run on batteries [14]. Modern smartphones, such as the Samsung Galaxy S3, Google Nexus 5 or iPhone 5S and later versions, are equipped with BLE receivers [62].

### 2.2.2 Beacons

BLE access points, or beacons, transmit their advertising data according to a standardized protocol. *iBeacon* is a standard developed by Apple Inc. in 2013 [54] and is now widely used in indoor positioning. It was established to determine whether a user is in a certain region around the beacon [1] and allows estimating the proximity to the beacon.

An iBeacon advertisement consists of a Universally Unique Identifier (UUID) and a Major and Minor, which identify the beacon. The RSS is used to estimate the distance from the beacon.

Various manufacturers produce BLE beacons which transmit data with the iBeacon protocol. For this thesis, beacons by the Spanish *Accent Systems* were used. They run on coin cell batteries and an example is displayed in Figure 2.3.



**Figure 2.3:** BLE beacon by *Accent Systems*

### 2.2.3 Advantages of Bluetooth low energy in indoor positioning

In indoor positioning, the use of WLAN fingerprinting has become a common approach. However, BLE can also be used for fingerprinting, and has a series of advantages compared to WLAN.

Due to the fact that BLE beacons have a very low power consumption, are small and run on batteries, they can easily be placed at many desired positions in indoor environments. WLAN routers, on the other hand, are larger, more expensive and need to be plugged into a socket for power supply, which limits the possibilities of deploying them at positions which are geometrically ideal for positioning.

BLE scans are faster than WLAN scans and enable a higher update rate. With WLAN more than 50 channels are available in the 2.4 and 5 GHz bands. Each access point broadcasts a Service Set Identifier (SSID), and a broadcast interval has a duration of 100 ms. During a passive WLAN scan, a device waits for the SSID to be broadcast, and with so many available bands the scan durations are high [15]. Faragher and Harle (2015) showed that the scan rate of a Samsung Galaxy S4 is 0.25 Hz when scanning 2.4 and 5 GHz WLAN channels<sup>1</sup>. BLE scans, in contrast, are significantly shorter, as a scan cycle only hops over the three advertising channels.

Furthermore, a 2014 study conducted in one floor of a university building [62] showed that BLE can outperform WLAN at 2.4 GHz in indoor positioning in terms of accuracy. The authors attribute these results to the significantly higher sampling rate of BLE, its channel hopping, and its lower transmission power. Channel hopping helps BLE to minimize interferences, whereas the low transmission power minimizes multipath effects. Lohan et al. (2015) conducted multi-floor tests for BLE and WLAN 2.4 GHz as well as WLAN 5 GHz and showed that WLAN 2.4 GHz and BLE achieve similar results. The

---

<sup>1</sup>Note that active WLAN scans can also be used, but they increase network traffic.

results of WLAN 5 GHz were of poor quality, which is attributed to the fact that less AP were able to transmit at this frequency.

## 2.3 Properties of the received signal strength

When using the RSS of WLAN and BLE for indoor positioning, it is vital to understand its characteristics in indoor environments.

RSS is measured in decibel-milliwatts (dBm), a logarithmic power ratio which refers to the power of one milliwatt. In free space, the RSS decreases inversely proportional to the square of the distance between transmitter and receiver [31]. In indoor environments, such as buildings or parking garages, radio channels are affected by interference and noise and the RSS varies both spatially and temporally.

When it comes to the spatial properties of the RSS of radio signals, the signal strength not only decreases with increasing distance, it is also strongly attenuated by walls and propagates asymmetrically. Given that the resonance frequency of water lies at 2.4 GHz [25], radio signals with a frequency of 2.4 GHz, such as BLE and 2.4 GHz WLAN, are absorbed by water. As the human body mainly consists of water, it is a strong attenuator for radio signals at this frequency. This implies that when a radio map is recorded, the operator should not block the signal path. Objects located in indoor environments can cause shadowing or multipath effects due to reflection and scattering of the signal.

The RSS at a fixed location also changes over time when objects in the environment change, e.g. in buildings when furniture is moved or people walk around.

To calculate positions from RSS measurements, parametric and non-parametric methods can be used. Parametric methods are based on path-loss models. These models are used to compute the distances to the AP from RSS. If the coordinates of the AP are known, the distances to the AP can be used to compute the position of the receiver using circular lateration. Non-parametric methods, or fingerprinting-based methods do not use functional relationships to model the relation between RSS and position and will be described in more detail in Chapter 3.

# Chapter 3

## Fingerprinting

In Chapter 2, it was concluded that the RSS is a commonly used feature of radio signals to determine positions indoors. Walls and objects located in indoor environments cause irregular, spatially varying RSS patterns, and fingerprinting exploits exactly this property. This chapter will present the basic principles of fingerprinting and then focus on the two types of fingerprinting methods, deterministic and probabilistic fingerprinting.

### 3.1 Basic principles

Fingerprinting consists of two phases: an offline training phase and an online location estimation phase [12], [33], [56].

In the offline phase, a radio map is built. WLAN or BLE Access Points (AP) are placed in the target environment in a way that full coverage is ensured. At Reference Points (RP), sample measurements of the RSS from the surrounding AP are taken. These RSS patterns are so-called fingerprints and are stored in a database called radio map together with location information, i.e. the coordinates of the fingerprint. Consequently, at each RP the following information is stored:

- The coordinates of the RP,
- RSS measurements received from the AP,
- information on which RSS measurement belongs to which AP.

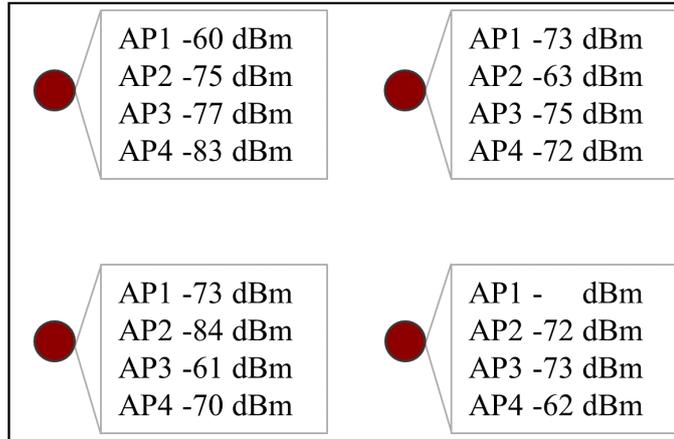
**Table 3.1:** Requirements that, according to Niedermayr (2015), attributes used for fingerprinting should fulfil.

Property	Description
Spatial variation	The attribute of the field has to change significantly with the position.
Temporal stability	The attribute of the field should not change over time or the temporal changes should be predictable with sufficient accuracy.
Observability	The attribute used for fingerprinting has to be measurable.
Robustness	The attribute field should be as robust as possible against changes in the environment.
Unambiguity	The vector of attributes should be unique for one position.
Availability	Generating the attribute field should be easy or the attribute field should already be available.

In the online phase, the user measures RSS at an unknown location. The RSS measured are compared to the fingerprints in the radio map to estimate the user's location.

Note that fingerprinting cannot only be done with radio signal RSS, it can also use other physical quantities, i.e. magnetic field strength. Whichever attribute is used for fingerprinting, it has to fulfil a series of requirements, which are listed in Table 3.1. The first attribute listed, spatial variation, determines the achievable accuracy of the positioning system: the higher the gradient of the attribute field, the more accurately a position can be determined [43]. As temporal stability is not given for radio fingerprinting, radio maps need to be updated regularly. The observability of radio fingerprinting is given through RSS measurements. Radio signals are robust enough in order to be used for fingerprinting. Unambiguity is achieved through the unique propagation characteristics of radio signals in indoor environments. The availability of radio signals in indoor environments can be taken for granted, as WLAN is already ubiquitous. BLE beacons can easily be deployed, and even some light bulbs are already equipped with BLE.

The methods used for fingerprinting can be divided into two groups: deterministic and probabilistic fingerprinting [6]. Sections 3.2 and 3.3 will focus on them in more detail.



**Figure 3.1:** Schematic view of a deterministic radio map. At every reference point (dark red), the RSS received from the surrounding AP are stored as a single deterministic value.

## 3.2 Deterministic fingerprinting

Deterministic fingerprinting techniques use deterministic parameters and observations to estimate locations. They are based on the computation of distances between the observed vector of RSS and the vector of RSS stored at the reference points in the radio map [43].

Let a radio map consist of  $N$  AP and  $M$  RP. In the offline training phase, only one value for the RSS from every accessible AP is stored in the radio map for every reference point. The radio map now contains  $M$  vectors (one for every reference point) containing  $N$  elements each (the measured RSS for every AP). Figure 3.1 shows a schematic representation of a deterministic radio map with four reference points and four access points ( $N = M = 4$ ). As shown in the example, it can also happen that not all AP are visible at every reference point.

During the online phase, a receiver is situated at the unknown position  $\mathbf{p}(x, y)$ . At this position, the RSS received from all access points  $AP_i$  are measured and stored in a vector  $\mathbf{r}_p$

$$\mathbf{r}_p = \begin{pmatrix} \text{RSS}_{AP_1,p} \\ \text{RSS}_{AP_2,p} \\ \vdots \\ \text{RSS}_{AP_N,p} \end{pmatrix}. \quad (3.1)$$

The measured signal pattern  $\mathbf{r}_p$  is now compared to all  $M$  fingerprints stored in the radio map. A distance metric  $D$  is used as a measure of similarity, for which the Euclidean norm is often used [56]:

$$D_i = \|\mathbf{r}_p - \mathbf{r}_i\|_2 = \sqrt{\sum_{n=1}^N [\mathbf{r}_p(n) - \mathbf{r}_i(n)]^2} \quad i = 1 \dots M. \quad (3.2)$$

Formula 3.2 is used to compute the distance to every RP. The list of distances is sorted in ascending order. The RP at the top of the list are the RP which are closest (smallest computed distance) to the position of the receiver. The simplest deterministic method to estimate the unknown position  $P(x, y)$  is called *Nearest Neighbour*. This technique simply selects the RP with the minimum distance and assumes the user is located at this RP. A more sophisticated approach is the *K Nearest Neighbour*, which takes the  $K$  RP with the smallest distances and computes the unknown position by averaging the coordinates of these  $K$  RP.

Apart from Nearest Neighbour methods other deterministic fingerprinting methods have also been developed. They are based on techniques for data classification, like Support Vector Machines [16] and Neural Networks [32]. The investigation and optimization of classification-based schemes in combination with fingerprinting, however, lies beyond the scope of this thesis.

A problem can occur in deterministic fingerprinting when RSS values received from AP are observed which are missing in the reference vectors. An incomplete reference vector will result in a computed distance which cannot be compared to a distance calculated from a full reference vector. In general, the distance is bigger if it is calculated from more observations, as each element contributes positively to the calculated distance (except when the observation equals the reference value, then the contribution is 0). To overcome this problem, Roos et al. (2002) and Hotta et al. (2012) set the missing reference values to a fixed value, e.g. the smallest observable RSS.

### 3.3 Probabilistic fingerprinting

In probabilistic fingerprinting, the RSS vector of observations  $\mathbf{r}$  and the position  $\mathbf{p}$  are seen as multivariate random variables [41]. The variables  $\mathbf{r}$  and  $\mathbf{p}$  are dependent, meaning that if one is known, the other one can be predicted [31].

A probabilistic radio map contains fingerprints which express the probability of measuring an RSS pattern  $\mathbf{r}$  at the position  $\mathbf{p}$ ; this conditional probability can be expressed as

$$P(\mathbf{r}|\mathbf{p}). \quad (3.3)$$

In the online location estimation phase, an RSS vector  $\mathbf{r}$  is observed. The aim is to find the most likely ( $\operatorname{argmax} P$ ) position given our observations [61] :

$$\hat{\mathbf{p}} = \operatorname{argmax}_{\mathbf{p}} P(\mathbf{p}|\mathbf{r}), \quad (3.4)$$

where  $P(\mathbf{p}|\mathbf{r})$  is the conditional probability density function of the state  $\mathbf{p}$  given the measurements  $\mathbf{r}$ . This conditional probability can be rewritten using Bayes' theorem [60], [35]:

$$P(\mathbf{p}|\mathbf{r}) = \frac{P(\mathbf{r}|\mathbf{p})P(\mathbf{p})}{P(\mathbf{r})}. \quad (3.5)$$

$P(\mathbf{r}|\mathbf{p})$  is the likelihood, the conditional probability of the RSS vector at the reference positions.  $P(\mathbf{p})$  is the a-priori distribution, or prior probability.  $P(\mathbf{r})$  is a normalizing constant [21], [28]. The position can therefore be estimated from

$$\hat{\mathbf{p}} = \operatorname{argmax}_{\mathbf{p}} P(\mathbf{r}|\mathbf{p})P(\mathbf{p}). \quad (3.6)$$

Equation 3.6 can be used in combination with particle filtering to estimate the positions along the trajectory of a moving object. Information on the prior probability  $P(\mathbf{p})$  of the position can be extracted from the particle filter. If the prior is unknown or only one position is estimated, the prior is assumed to be uniform [46], and equations 3.4 and 3.6 can be rewritten as

$$\hat{\mathbf{p}} = \operatorname{argmax}_{\mathbf{p}} P(\mathbf{r}|\mathbf{p}). \quad (3.7)$$

The Maximum Likelihood (ML) estimator in Equation 3.7 has one disadvantage: the estimated position will always be at one of the reference points, which limits the positioning accuracy to the density of the RP. To overcome this problem, a *K Nearest Neighbour* estimator can be used which takes the *K* most probable RP and averages their coordinates to obtain the estimated position [31].

### 3.3.1 Probabilistic radio map

While a deterministic radio map only requires one observed RSS to be stored at every RP for each AP, probabilistic radio maps need to store a density function which expresses the likelihood that an RSS pattern  $\mathbf{r}$  is observed at the position of each RP.

The likelihood density  $P(\mathbf{r}|\mathbf{p})$  at every RP is multivariate. A common assumption in radio fingerprinting is that the AP are independent, so that the probability to observe an RSS pattern at a certain position  $\mathbf{p}_i$  can be calculated from the product of the probabilities of observing each RSS sample at this position [61],[60], [21]:

$$P(\mathbf{r}|\mathbf{p}_i) = \prod_{j=1}^N P(r_j|\mathbf{p}_i). \quad (3.8)$$

To obtain the likelihood density, a time series of RSS has to be measured at each RP for every AP. These time series are then used to estimate a probability density using either a non-parametric approach or a parametric approach.

**Parametric approach.** Parametric approaches assume a model, i.e. an underlying distribution, and estimate the parameters of the model. In radio fingerprinting, the density is often assumed to be Gaussian [31]. If Gaussian distribution is assumed, only the mean and the variance of each time series have to be stored in the radio map, as they fully describe Normal distribution.

A schematic representation of a probabilistic radio map with four AP and four RP is portrayed in Figure 3.2. At every reference point, time series of the RSS of the AP were measured and used to estimate a probability density. The density function describes the probability of measuring a certain RSS value at this reference point. To obtain the probability to measure a RSS pattern, the probabilities of measuring every single value in this pattern have to be multiplied according to Equation 3.8.

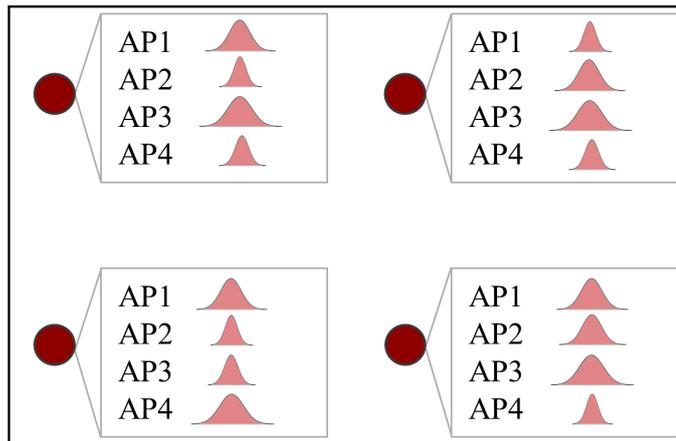
**Non-parametric approaches: the histogram.** Non-parametric methods estimate the density without assuming a specific statistical form of the data. A widely used non-parametric technique to estimate the density is the histogram. Histograms are constructed by counting the frequency of values falling into a certain interval, a so-called bin. The relative frequency of a value is an estimate of the probability of occurrence.

Multiple methods exist to estimate the ideal number of bins,  $k$ , and the bin width,  $b$ . One example is given by [47]:

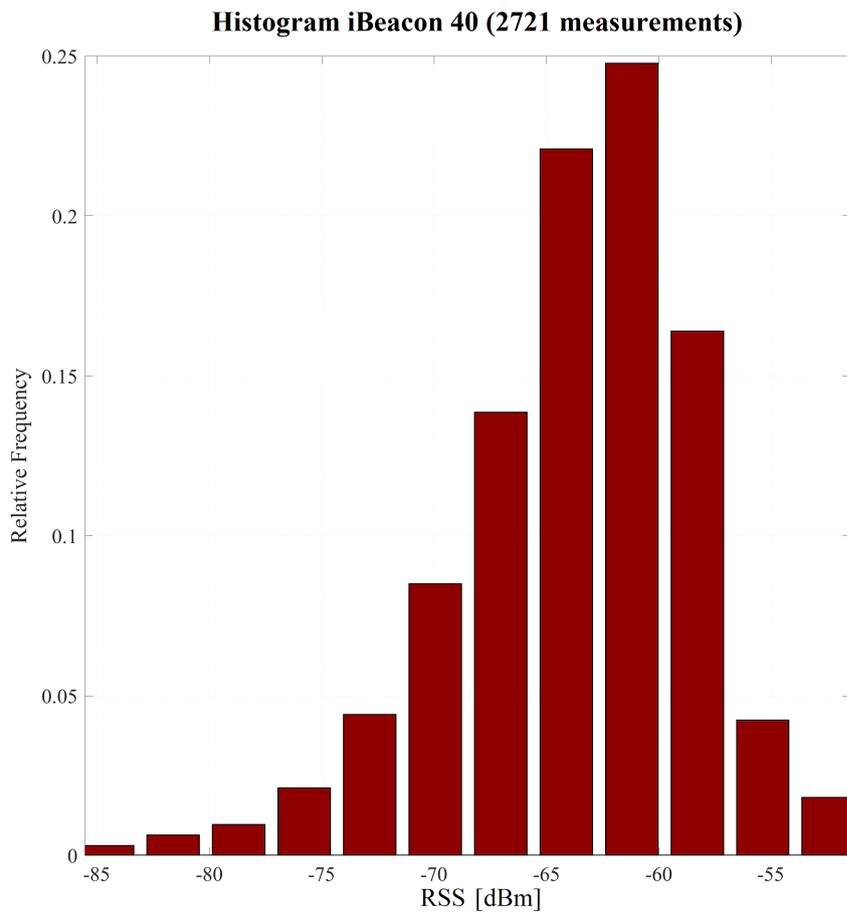
$$k \approx 1 + 3.3 \log_{10} n \quad (3.9)$$

$$b \approx \frac{x_{max} - x_{min}}{\sqrt{n} + 1}. \quad (3.10)$$

Equation 3.9 calculates the number of bins from the number of samples  $n$ . Equation 3.10 yields the bin width, taking into account the maximum and minimum values of the samples  $x_{max}$  and  $x_{min}$  as well as the total number of samples  $n$ .



**Figure 3.2:** Schematic view of a probabilistic radio map. At every reference point (dark red), a probability distribution which describes the likelihood that a certain RSS value will be received from a specific AP, is stored.



**Figure 3.3:** Histogram obtained from 2721 RSS measurements received from a BLE access point.

An example of an RSS histogram is given in Figure 3.3. 2721 RSS measurements received from a BLE access point were taken to construct the histogram, using equations 3.9 and 3.10 to estimate the number of bins and the bin width.

The histogram estimate obtained by taking measurements from one AP at a specific RP is univariate. It yields an estimate of the probability  $P(r_j|\mathbf{p}_i)$  that an RSS value from one AP  $r_j$  is observed at the RP  $\mathbf{p}_i$ . To obtain the multivariate estimate for the probability of observing a signal pattern  $\mathbf{r}$ , the single probabilities have to be multiplied according to Equation 3.8.

Using histograms to estimate the likelihood density has the advantage that no prior statistical form of the data has to be assumed. However, there are also several disadvantages. First, the density estimated from the histogram is zero outside the range of the sample data. At the bin boundaries, the density is discontinuous. Furthermore, a large number of bins need to be stored to construct a radio map, and the likelihood density can also be obtained from other methods which need less storage space.

**Non-parametric approaches: Kernel density estimators.** Kernel density estimators provide an alternative to histograms and are often used in radio fingerprinting [30]. A kernel density estimate results from a superposition of kernel functions which are placed at the measured data points. For a set of  $n$  RSS samples  $\mathbf{r}_i$  at the reference point  $\mathbf{p}_i$ , the kernel density estimate is defined as follows [31]:

$$\hat{f}(\mathbf{r}|\mathbf{p}_i) = \frac{1}{n\sigma_{\mathbf{r}}} \sum_{t=1}^n K\left(\frac{\mathbf{r} - \mathbf{r}_i(t)}{\sigma_{\mathbf{r}}}\right), \quad (3.11)$$

with the kernel function  $K(\cdot)$  and  $\sigma_{\mathbf{r}}$  being the kernel bandwidth.

A  $d$ -dimensional kernel function  $K(\cdot)$  which is used to model the likelihood density must fulfil the following four conditions [31], [49]:

1. The range of the kernel function has to be positive:

$$K(\mathbf{r}) \geq 0, \forall \mathbf{r} \in \mathbb{R}^d. \quad (3.12)$$

2. The kernel function should have unit area:

$$\int_{\mathbb{R}^d} K(\mathbf{r}) = 1. \quad (3.13)$$

3. The mean of the kernel function is zero:

$$\int_{\mathbb{R}^d} \mathbf{r}K(\mathbf{r}) = 0. \quad (3.14)$$

4. The covariance matrix of the kernel corresponds to the identity matrix:

$$\int_{\mathbb{R}^d} \mathbf{r}\mathbf{r}^T K(\mathbf{r}) = \mathbf{I}_{d \times d}. \quad (3.15)$$

Examples of univariate kernel functions are the Epanechnikov kernel, the triangular kernel and the Gaussian kernel. According to [31], the Epanechnikov kernel function is defined as:

$$K_{Epanechnikov}(x) = \begin{cases} \frac{3}{4}(1 - \frac{1}{5}x^2)/\sqrt{5}, & \text{for } x < \sqrt{5} \\ 0, & \text{for } x \geq \sqrt{5} \end{cases}, \quad (3.16)$$

and the triangular kernel is defined as:

$$K_{triangular}(x) = \begin{cases} 1 - |x|, & \text{for } |x| < 1 \\ 0, & \text{for } |x| \geq 1. \end{cases} \quad (3.17)$$

A commonly used kernel in radio fingerprinting is the Gaussian kernel [46]:

$$K_{Gauss}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right). \quad (3.18)$$

The parameter  $\sigma$  in Equation 3.18 defines the kernel bandwidth. The influence of this parameter will be discussed in Section 4.2.1. As with histograms, a multivariate kernel is obtained by multiplying univariate Kernels.

Using Gaussian kernels instead of histograms has the advantage that a continuous likelihood density is obtained and the probability for values outside the range of the sample data does not become zero. However, a likelihood density obtained from kernel density estimation requires a lot of memory space, as every data point needs to be stored. Parametric methods can significantly reduce the storage space, at the disadvantage that prior assumptions regarding the likelihood distribution have to be made.

**Part II**

**Optimization theory**

# Chapter 4

## Algorithmic optimization

In this chapter, approaches to optimizing deterministic and probabilistic fingerprinting algorithms will be discussed. For deterministic fingerprinting, a WKNN approach and various distance metrics will be presented. For probabilistic fingerprinting, the influence of the kernel bandwidth will be discussed, and different weights for a probabilistic WKNN approach will be examined.

### 4.1 Deterministic fingerprinting

This section will focus on the optimization of deterministic fingerprinting algorithms based on Nearest Neighbour techniques.

In the offline phase of deterministic fingerprinting, one RSS value received from every AP is stored at each RP in the radio map. During the online positioning phase, the user measures an RSS pattern  $\mathbf{r}_p$ , which is then compared to the patterns  $\mathbf{r}_i$  stored in the radio map. Using the *K Nearest Neighbour* technique, the  $K$  most similar RP are determined with Equation 3.2 and their coordinates are averaged to obtain an estimate of the user's position  $\hat{\mathbf{p}}$ .

#### 4.1.1 Weighted k nearest neighbour

An algorithmic improvement can be achieved by using the WKNN technique. The WKNN method assumes that the distances  $D_i$  computed from the RSS can be compared to

actual distances between the user and the RP. The user's position  $\hat{\mathbf{p}}$  is computed from a weighted average of the  $K$  most similar RP [56]:

$$\hat{\mathbf{p}}_{WKNN} = \sum_{i=1}^K \frac{1}{D_i} \mathbf{p}_i \cdot \left( \sum_{i=1}^K \frac{1}{D_i} \right)^{-1}. \quad (4.1)$$

The  $K$  most similar points in Equation 4.1 are the  $K$  points with the smallest computed distance metric  $D_i$ . The distances  $D_i$  can be computed using the Euclidean norm (see Equation 3.2). In WKNN fingerprinting, the inverse distances are used as weights, giving the coordinates of a more similar (closer) RP a higher weight than those of a RP where the RSS pattern is less similar and the distance is bigger. The inverse sum of the inverse distances  $(\sum_{i=1}^K \frac{1}{D_i})^{-1}$  is used as a normalizing factor.

The size of  $K$  can be any integer from 1 to the number of RP  $M$ . If only one RP is taken into consideration ( $K = 1$ ), the estimation corresponds to a simple Nearest Neighbour method. If all RP are considered ( $K = M$ ), the estimation might not necessarily yield the best result, as RP which are further away might have a too strong influence. Kushki et al. (2012) showed that the optimal number of AP in WKNN fingerprinting is  $K = 4$ .

### 4.1.2 Norms

Even though the Euclidean norm is commonly used to determine the most similar fingerprints and to compute the weights for WKNN fingerprinting, other metrics might better express the relationship between distance and RSS similarity. Del Corte-Valiente et al. (2009) and Torres-Sospedra et al. (2015) investigated different distance metrics for indoor positioning systems and found that there are alternatives which outperform the Euclidean distance. Out of the 5 metrics tested by Del Corte-Valiente et al. (2009) and out of the 51 metrics proposed by Torres-Sospedra et al. (2015), the most efficient ones will be presented in this thesis.

Recall that we have defined a deterministic radio map with  $N$  AP and  $M$  RP in Section 4.1. At every RP,  $N$  RSS values (one received from every AP) are stored along with the coordinates of the RP. In the online positioning phase, the user measures an RSS pattern  $\mathbf{r}_p$  which is compared to the RSS patterns  $\mathbf{r}_i$  at the RP. Distance metrics  $D_i$  are used as measures of similarity between the stored fingerprints and the measured RSS pattern.

Cha (2007) divides distance norms into subgroups, so-called families. This nomenclature will be adopted for this thesis. The norms presented are taken from [52] and [5].

**The Minkowski  $L_p$  family.** The Minkowski family includes the  $L_p$  norms, such as the Manhattan distance ( $L_1$ ), Euclidean distance ( $L_2$ ) and the Chebyshev distance ( $L_\infty$ ). The most general member of this family, the Minkowski distance  $L_p$ , is calculated as

$$D_{L_p,i} = \sqrt[p]{\sum_{n=1}^N |\mathbf{r}_p(n) - \mathbf{r}_i(n)|^p} \quad i = 1 \dots M. \quad (4.2)$$

All other norms from the Minkowski family can be derived from Equation 4.2. The Euclidean distance was already addressed in Section 4.1 in Equation 3.2. The Manhattan, or City Block distance ( $p = 1$ ), is computed from the absolute value of the difference of the measured RSS  $\mathbf{r}_p(n)$  received from AP  $n$  and the RSS value  $\mathbf{r}_i(n)$  from AP  $n$  stored at the RP  $i$ .

$$D_{Manhattan,i} = \sum_{n=1}^N |\mathbf{r}_p(n) - \mathbf{r}_i(n)| \quad i = 1 \dots M. \quad (4.3)$$

As  $p$  approaches infinity, the Chebyshev distance is derived:

$$D_{Chebyshev,i} = \max_n |\mathbf{r}_p(n) - \mathbf{r}_i(n)| \quad i = 1 \dots M. \quad (4.4)$$

**The  $L_1$  family.** Another family of norms is the  $L_1$  family, whose members are based on the Manhattan ( $L_1$ ) distance. A prominent member of this family is the Sørensen or Bray-Curtis distance, which is calculated as

$$D_{Sørensen,i} = \frac{\sum_{n=1}^N |\mathbf{r}_p(n) - \mathbf{r}_i(n)|}{\sum_{n=1}^N (|\mathbf{r}_p(n)| + |\mathbf{r}_i(n)|)} \quad i = 1 \dots M. \quad (4.5)$$

The Gower norm also belongs to the  $L_1$  family. It is computed from the  $L_1$  norm divided by the number of RSS, which in WKNN methods is equivalent to the Manhattan norm, as it yields the same distances just divided by a constant:

$$D_{Gower,i} = \frac{1}{N} \sum_{n=1}^N |\mathbf{r}_p(n) - \mathbf{r}_i(n)| = \frac{1}{N} D_{Manhattan,i} \quad i = 1 \dots M. \quad (4.6)$$

Another representative of the  $L_1$  family is the Soergel distance,

$$D_{Soergel} = \frac{\sum_{n=1}^N |\mathbf{r}_p(n) - \mathbf{r}_i(n)|}{\sum_{n=1}^N \max(|\mathbf{r}_p(n)|, |\mathbf{r}_i(n)|)} \quad i = 1 \dots M, \quad (4.7)$$

which is similar to Kulczynski:

$$D_{Kulczynski} = \frac{\sum_{n=1}^N |\mathbf{r}_p(n) - \mathbf{r}_i(n)|}{\sum_{n=1}^N \min(|\mathbf{r}_p(n)|, |\mathbf{r}_i(n)|)} \quad i = 1 \dots M. \quad (4.8)$$

While Soergel divides the  $L_1$  by the maximum absolute value of the RSS values compared, Kulczynski uses the minimum absolute value. The Canberra norm might at first resemble Sørensen but it normalizes the Manhattan norm individually for every RSS comparison:

$$D_{Canberra} = \sum_{n=1}^N \frac{|\mathbf{r}_p(n) - \mathbf{r}_i(n)|}{(|\mathbf{r}_p(n)| + |\mathbf{r}_i(n)|)} \quad i = 1 \dots M. \quad (4.9)$$

The Lorentzian norm applies the natural logarithm to the  $L_1$  norm. To avoid negative distances the integer value 1 is added:

$$D_{Lorentzian} = \sum_{n=1}^N \ln(1 + |\mathbf{r}_p(n) - \mathbf{r}_i(n)|) \quad i = 1 \dots M. \quad (4.10)$$

**The squared  $L_2$  or  $\chi^2$  family.** A further group of norms is the squared  $L_2$  or  $\chi^2$  family. Norms belonging to this family are based on the square of the Euclidean distance and include the squared Euclidean, Neyman, or the squared  $\chi^2$  distance. The squared Euclidean is most simple metric belonging to the squared  $L_2$  family. As its name suggests, it is calculated from the square of the Euclidean distance:

$$D_{squaredEuclidean} = \sum_{n=1}^N (\mathbf{r}_p(n) - \mathbf{r}_i(n))^2 \quad i = 1 \dots M. \quad (4.11)$$

The Neyman distance is calculated from the Euclidean distance divided by the RSS value at the RP

$$D_{Neyman,i} = \sum_{n=1}^N \frac{(\mathbf{r}_p(n) - \mathbf{r}_i(n))^2}{|\mathbf{r}_i(n)|} \quad i = 1 \dots M. \quad (4.12)$$

Note that the absolute value is taken in the denominator in Equation 4.12 to avoid negative distances. Another version of the Neyman distance can be obtained by dividing the Euclidean distance by the absolute value of the measured RSS:

$$D_{Neyman2,i} = \sum_{n=1}^N \frac{(\mathbf{r}_p(n) - \mathbf{r}_i(n))^2}{|\mathbf{r}_p(n)|}. \quad i = 1 \dots M \quad (4.13)$$

The distance metric squared  $\chi^2$  is obtained by dividing the Euclidean distance by the sum of the absolute value of the measured RSS and the stored RSS at the RP:

$$D_{\chi^2,i} = \sum_{n=1}^N \frac{(\mathbf{r}_p(n) - \mathbf{r}_i(n))^2}{|\mathbf{r}_p(n)| + |\mathbf{r}_i(n)|} \quad i = 1 \dots M. \quad (4.14)$$

## 4.2 Probabilistic fingerprinting

In probabilistic fingerprinting, likelihood distributions  $P(\mathbf{r}|\mathbf{p})$  expressing the probability of measuring an RSS pattern  $\mathbf{r}$  at the position of the RP  $\mathbf{p}$  are stored in the radio map. These likelihood distributions can be obtained using parametric or non-parametric techniques (see Section 3.3.1). Parametric techniques assume an underlying distribution and estimate the parameters of the distribution function from the given data. Non-parametric methods estimate the likelihood density only from the given data and include histograms and kernel density estimation techniques. The advantage of kernels over histograms is that they yield continuous density functions with a non-zero probability of values outside the sample range. A factor which strongly influences the shape of the resulting likelihood density is the kernel bandwidth, whose optimization will be discussed in Section 4.2.1.

In the online positioning phase of probabilistic fingerprinting, the most likely position given the measured RSS pattern is estimated. To overcome the problem of solely estimating positions located at one of the reference points, WKNN methods can be used. Section 4.2.2 will focus on the weights which can be used in probabilistic fingerprinting.

### 4.2.1 Optimal kernel bandwidth

During the offline phase in probabilistic fingerprinting, time series of the RSS received from the surrounding access points are measured at each reference point. The observed RSS are then used to model a likelihood distribution  $P(\mathbf{r}|\mathbf{p})$ , which describes the probability of measuring an RSS pattern  $\mathbf{r}$  at the position of the RP  $\mathbf{p}$ .

The likelihood distribution described above can be obtained using kernel density estimators. Kernel density estimators place kernel functions at the measured data points and calculate the density estimate from the superposition of these functions. The properties which kernel functions have to fulfil as well as examples of kernel functions are given in Section 3.3.1.

Kernel density estimates are influenced by the chosen kernel function and the kernel bandwidth. In radio fingerprinting, the Gaussian kernel is most commonly used [46] [31]. The influence of the kernel bandwidth will be discussed in this section.

The kernel bandwidth defines the broadness of the region of influence of each measured RSS sample. The bigger the bandwidth, the broader the region of influence of each value becomes. Kushki et al. (2012) explain this influence by demonstrating the effects when the bandwidth approaches extreme values:

- When the bandwidth comes closer to zero, the region of influence shrinks and the kernel density estimate approaches a delta function:

$$\sigma_{\mathbf{r}} \rightarrow 0, \quad K\left(\frac{\mathbf{r} - \mathbf{r}_i(t)}{\sigma_{\mathbf{r}}}\right) \rightarrow \delta(\|\mathbf{r} - \mathbf{r}_i(t)\|). \quad (4.15)$$

- As the bandwidth approaches infinity, the region of influence of each observed RSS value becomes infinitely large and the density estimate becomes a uniform distribution:

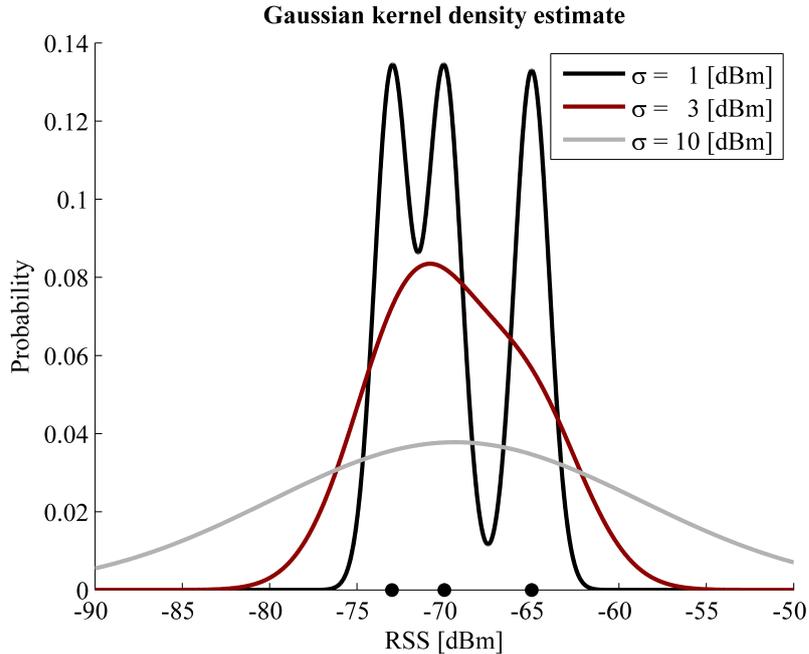
$$\sigma_{\mathbf{r}} \rightarrow \infty, \quad K\left(\frac{\mathbf{r} - \mathbf{r}_i(t)}{\sigma_{\mathbf{r}}}\right) \rightarrow 1. \quad (4.16)$$

The influence of the bandwidth is further illustrated in Figure 4.1. Three RSS values were used as training points to estimate the kernel density: -65 dBm, -70 dBm and -73 dBm. The values of the training points are portrayed as black circles on the horizontal axis of the graph. A Gaussian kernel (see Equation 3.18) was selected to estimate the density, and the bandwidth was varied from  $\sigma = 1$  dBm over  $\sigma = 3$  dBm to  $\sigma = 10$  dBm. The higher the bandwidth, the smoother the density estimate becomes. The density estimate obtained with a bandwidth of 1 dBm shows sharp peaks at the values of the training points, a local structure which does not necessarily have to occur in the actual density. The density estimates resulting from a  $\sigma$  of 3 dBm and 10 dBm are smooth and have a broader region of influence. However, a broad range of RSS values with almost equal probability, which results from the estimate using a bandwidth of 10 dBm, is not ideal for fingerprinting either.

The optimal kernel bandwidth, which should yield a minimal difference between the estimated and real density, depends on the training data. For Gaussian kernels, Scott (1992) found the optimal bandwidth to be

$$\sigma = \left(\frac{4}{d+2}\right)^{\frac{1}{d+4}} \hat{\sigma} n^{\frac{-1}{d+4}}, \quad (4.17)$$

where  $n$  stands for the number of samples, in the case of radio fingerprinting the number of RSS values used to estimate the density;  $d$  is the dimension of the measurements, which in radio fingerprinting is one; and  $\hat{\sigma}$  is an estimate of the standard deviation of the training data [31].



**Figure 4.1:** Influence of bandwidth  $\sigma$  on the kernel density estimate. The training points lie at -65 dBm, -70 dBm and -73 dBm, a Gaussian kernel was used to obtain the density estimate.

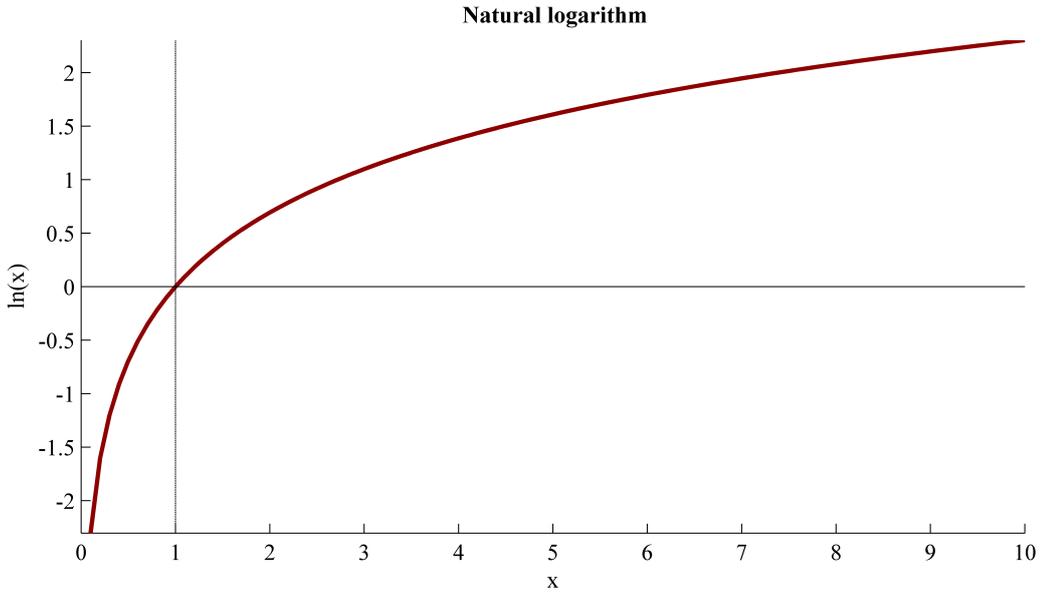
## 4.2.2 Weighting

In the online positioning phase of probabilistic fingerprinting, the observed RSS pattern  $\mathbf{r}$  is compared to the likelihood distributions stored at the reference points in the radio map. Using a maximum likelihood estimator as defined in Equation 3.7, the resulting position estimate will lie at one of the reference points [31]. The achievable positioning accuracy is therefore limited to the distance between the reference points.

Higher accuracies can be achieved by adding more reference points, which is extremely time-consuming in the offline phase and significantly increases the computational complexity in the online phase. An efficient alternative to increasing the number of reference points is the use of WKNN methods.

In WKNN probabilistic fingerprinting, the  $K$  most probable reference points are selected and their coordinates are used to compute a position estimate through weighted averaging. The general form of a WKNN position estimation with weights  $\omega_i$  and coordinates of the selected reference points  $\mathbf{p}_i$  reads as follows:

$$\hat{\mathbf{p}}_{WKNN} = \frac{\sum_{i=1}^K \omega_i \mathbf{p}_i}{\sum_{i=1}^K \omega_i}. \quad (4.18)$$



**Figure 4.2:** Natural logarithm. Domain  $\{x|x \in \mathbb{R}_{>0}\}$ , range  $\{y|y \in \mathbb{R}\}$ .

Different weights may be used. The most simple form,

$$\omega_i = 1, \quad (4.19)$$

assigns equal weights to all  $K$  most probable reference points. A more sophisticated and widely used approach uses the probabilities that the RSS patterns are observed at the RP as weights:

$$\omega_i = P(\mathbf{r}|\mathbf{p}_i). \quad (4.20)$$

Ma et al. (2015) observed that the magnitudes of the probabilities of the  $K$  selected reference points differ greatly. If a position estimate is calculated using weights from Equation 4.20, it will lie at the RP with the highest probability. Ma et al. (2015) therefore suggest using logarithmic weights to achieve a better estimation.

Logarithms, as shown in the example of the natural logarithm  $\ln(x)$  in Figure 4.2, have the set of positive real numbers  $\mathbb{R}_{>0}$  as their domain. For values smaller than one, the values of logarithms are negative, at  $x = 1$  they are zero. Probability density functions have a range from zero to one. To avoid negative and zero weights, the following formulation is suggested for logarithmic weights:

$$\omega_i = \ln(c + P(\mathbf{r}|\mathbf{p}_i)), \quad c > 1. \quad (4.21)$$

# Chapter 5

## Economic optimization

When it comes to the economic optimization of radio fingerprinting, several aspects should be considered:

- AP hardware acquisition and maintenance costs,
- the amount of time needed to create radio maps,
- the amount of time it takes to update a radio map.

As far as hardware costs are concerned, BLE beacons are generally cheaper than WLAN routers. However, more BLE beacons need to be deployed to ensure the same coverage. An example is given in [54], where a 4 800 m<sup>2</sup> parking garage was equipped with 60 BLE access points and 10 WLAN routers. The BLE beacons by *Accent Systems* cost 12.50 € each, the WLAN routers (Linksys E2500) supporting both 2.4 GHz and 5 GHz WLAN cost approximately 60 € per router. The total financial effort to construct the radio map with WLAN routers was smaller (600 € compared to 750 € for BLE beacons), yet the accuracy of the position estimate obtained from the radio map with BLE beacons was significantly higher: a point positioning error of 9-12 m was achieved for WLAN and 3-6 m for BLE in the static case.

Ji et al. (2015) analysed how the positioning accuracy depends on the number of beacons. In a simulation for a 100 m × 100 m floor, they deployed between 10 and 100 BLE beacons both randomly and on a regular grid. Not surprisingly, they found out that the more beacons are deployed, the more accurate the positioning result becomes. However they also discovered that once a certain number of beacons is used (in their case 80), hardly any improvements in accuracy can be made by adding more. There seems to be a threshold or an optimal number of beacons, which can be found through simulations. If

the ideal number of beacons is found, the hardware acquisition costs can be reduced by avoiding buying too many beacons.

Even though hardware costs can be optimized, the biggest optimization potential lies in the expenditure of time needed to create and update radio maps. Chai and Yang (2007) illustrate an example of how labour intensive the offline phase of fingerprinting is. If a radio map is designed with  $N$  AP that construct an RSS sample and  $M$  RP and  $T$  is the sampling time needed to obtain WLAN RSS measurements from one AP, the total sampling time can be computed from  $N \cdot M \cdot T$ . The more AP, the more RP or the higher the sampling time, the more time it takes to construct the radio map. For large indoor areas such as public parking garages or shopping malls, collecting the sample measurements can take more than a day's work. Methods which reduce the time needed to obtain the measurements for the radio map, with a focus on path loss models, will therefore be discussed in Section 5.1.

Given that radio maps change over time, they need to be updated regularly. To avoid having to repeat the tedious procedure of measuring the whole radio map anew, two approaches to how radio maps can be automatically updated will be presented in Section 5.2, one of which is based on crowdsourcing.

Finally, crowdsourcing and path loss models will be combined and presented in an approach which allows obtaining a radio map only from crowdsourced data in Section 5.3.

## 5.1 Sparse fingerprinting

As discussed in the introduction of this chapter, the creation of radio maps in the offline phase is a tedious and time-consuming procedure. The time needed to create the radio map can be reduced by reducing the sampling time or by reducing the number of reference points, though this inevitably deteriorates the achievable positioning accuracy. Chai and Yang (2007) reduced both the sampling time and the number of RP by two-thirds in a testbed for probabilistic WLAN fingerprinting and showed that the accuracy decreased by 26 percent.

Aiming to compensate for the loss of accuracy resulting from a reduced number of RP and reduced sampling times, a number of researches have focused on the area of sparse fingerprinting. Li et al. (2015) propose using a cost function which combines not only the

degree of similarity but also the dimension of the fingerprint, meaning that RP should preferably be selected if they have a large number of AP in common with the measured sample. Furthermore they introduce a spatial constraint, represented by a circle around the last position. The radius of the circle depends on the speed of the moving user and the update interval.

Sorour et al. (2015) and Majeed et al. (2015) propose a machine learning algorithm, Manifold Alignment, which makes use of RSS correlations and calculates a radio map from very few observations.

Khalajmehrabadi et al. (2017) propose interpolation techniques to obtain a finer grid of reference points. A selection of possible interpolation techniques will be covered in the following subsection.

### 5.1.1 Radio map interpolation

Radio map interpolation is an efficient method to obtain a dense radio map from only a few fingerprints measured. Interpolation methods for radio fingerprinting are based on models which express the indoor propagation characteristics of radio signals. Radio propagation models used for interpolation model the path loss in the direction of the direct path from the AP to the receiver.

Ji et al. (2006) divide radio propagation models into three categories, with ascending complexity:

- simple attenuation models,
- partition models and
- site-specific models.

*Simple attenuation models* describe path loss depending on an attenuation exponent  $n$ . An example of a simple attenuation model would be

$$P(d)[dB] = P(d_0)[dB] - 10 \cdot n \cdot \log_{10} \left( \frac{d}{d_0} \right), \quad (5.1)$$

where  $d$  is the distance to the transmitter (i.e. the access point),  $P(d)$  the power and  $P(d_0)$  the power at the reference distance  $d_0$ . The attenuation exponent  $n$  is two in free

space, and bigger than two in indoor environments. For office buildings with numerous floors, the attenuation exponent lies between  $2 \leq n \leq 6$  [31].

*Partition models* are based on simple attenuation models and consider attenuation effects of partitions occurring in indoor environments, such as walls, doors and floors. Bahl and Padmanabhan (2000) introduce a partition model which considers walls with a Wall Attenuation Factor (WAF):

$$P(d)[dB] = P(d_0)[dB] - 10 \cdot n \cdot \log_{10} \left( \frac{d}{d_0} \right) - \begin{cases} n_W \cdot WAF, & \text{for } n_W < C \\ C \cdot WAF, & \text{for } n_W \geq C \end{cases}. \quad (5.2)$$

In Equation 5.2,  $n_W$  is the number of walls,  $WAF$  the Wall Attenuation Factor and  $C$  the highest number of walls where the  $WAF$  still has an influence. Bahl and Padmanabhan (2000) determined the  $WAF$  empirically by first measuring the RSS at a known distance from the AP with no obstructions in the line of sight and then repeating the RSS measurements with walls between the AP and the receiver. By varying the number of walls between the receiver, they found the WAF to lie at 3.1 dBm for their testbed and  $C$  to be four.

*Site-specific models* are more sophisticated than partition models, as they model path loss dependent on parameters which are specific to the site. These parameters can describe the thickness of walls, the materials used and the geometry of objects located in indoor environments.

An example of a site-specific model can be found in Lott and Forkel (2001). The authors derived a multi-wall-and-floor model from ray tracing. The model can be parametrized by

$$P(d)[dB] = P(d_0)[dB] - 10 \cdot n \cdot \log_{10} \left( \frac{d}{d_0} \right) - \sum_{i=1}^I \sum_{k=1}^{K_{wi}} L_{wik} - \sum_{j=1}^J \sum_{k=1}^{K_{fj}} L_{fjk}, \quad (5.3)$$

where  $L_{wik}$  is the attenuation caused by wall type  $i$  and the  $k$ -th passed wall,  $L_{fjk}$  is the loss in signal strength caused by floor type  $j$  and the  $k$ -th passed floor,  $I$  and  $J$  are the number of wall and floor types respectively,  $K_{wi}$  is the number of passed walls belonging to category  $i$  and  $K_{fj}$  is the number of passed floors belonging to category  $j$ .

The more complex the model, the more time needs to be invested in determining the specific parameters of the model. While simple attenuation models only require finding a representative attenuation exponent  $n$ , site-specific models require detailed information about the materials and geometries in indoor environments.

Even though a dense radio map can be obtained using path loss models, a simulated radio map will differ from a measured radio map. Deasy and Scanlon (2007) examined this aspect to find out how the positioning accuracy is affected if measurements are replaced by models. They tested both deterministic and probabilistic fingerprinting in two test bed locations and compared the accuracy of the estimated result when simulated and measured radio maps were used. Even though their propagation model underestimated RSS by up to 15 dB, the accuracy of the estimated positions only decreased by about 30 percent in both deterministic and probabilistic fingerprinting when interpolated radio maps were used.

## 5.2 Automatic updating of radio maps

Radio maps change over time and need to be updated on a regular basis. Wilk et al. (2015) even declare that fingerprinting does not only consist of an offline and an online phase, it also requires a maintenance phase. Radio maps can be updated manually by repeating the offline phase and measuring the RSS samples anew. They can also be updated automatically, which will be described in more detail below.

Yin et al. (2005) introduced adaptive radio maps which relied on receivers which are placed at reference points and measure temporal RSS variations. They used a regression analysis to learn about the temporal relationship between RSS values at the reference points and model the changes for the whole radio map. This approach relies on additional infrastructure.

Another approach is crowdsourcing, which does not require additional hardware and has become attractive, especially for smartphone manufacturers. Wilk et al. (2015) developed a crowdsourcing system to automatically update radiomaps which is based on Pedestrian Dead Reckoning (PDR). They describe the maintenance phase of radio maps as a four-step-procedure:

1. The observed RSS samples are logged together with PDR information and uploaded to a server. The PDR data are provided by an external algorithm which computes speed and heading change from accelerometer and gyroscope measurements obtained from the sensors inside the smartphone.
2. On the server, the fingerprints are processed. The building and floor is identified from the measured RSS samples.

3. In the third step, locations are assigned to the observed fingerprints. This procedure is divided into two smaller steps: first, a relative trajectory is determined using a PDR algorithm, then the locations of the fingerprints are estimated using the PDR trajectory and the RSS similarity to the radio map.
4. Finally, the existing radio map is updated using the measured RSS samples and the locations estimated in the third step.

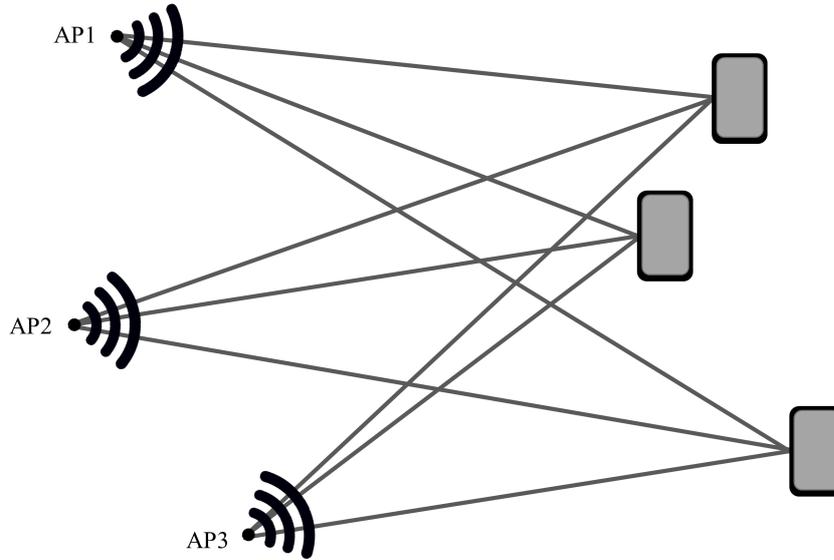
Wilk et al. (2015) found that the positioning accuracy can be improved if the radio map is updated in combination with PDR. If PDR is not used, their maintenance algorithm cannot achieve any improvements and can therefore not directly be used for indoor vehicle positioning. However, if relative car sensor data, such as heading change and covered distance, are available, their algorithm could be adapted for vehicle indoor positioning.

Wu et al. (2015) developed a similar crowdsourcing system to update radio maps. They use a combination of Dead Reckoning (DR) and map-matching to determine the user's position within the building and then update the radio map when the user is not moving. The smartphones are seen as movable reference points which collect a sufficient number of RSS samples during a static phase; these samples are then used to update the radio map.

### 5.3 Fully automatic creation of radio maps

Another crowdsourcing approach, the *EZ Localization* algorithm is able to automatically create radio maps without prior knowledge of the environment and was developed by Chintalapudi et al. (2010). In order for the proposed approach to work, occasional absolute position fixes have to be obtained, either by GNSS near windows or near entrances of the building.

Chintalapudi et al. (2010) assume that users move in an indoor environment and are equipped with devices, such as smartphones or notebooks, which are able to collect RSS samples. The smartphones or notebooks must be connected to a server. The scenario in Figure 5.1 should help to illustrate how the algorithm works: if all 9 distances between three receivers and three access points are known in a 2D plane, their relative positions are fixed. If the coordinates of three non-collinear objects of this constellation can be fixed, the coordinates of the other three objects can be determined. The three position fixes can be achieved by fixing the positions through GNSS close to windows or entrances.



**Figure 5.1:** If the distances between three receivers and three AP are known, the relative geometry of the constellation is rigid.

However, the distances to the AP cannot be measured directly, only RSS values are measured. To overcome this problem, Chintalapudi et al. (2010) use a simple path loss model which describes the distance-RSS relationship:

$$p_{ij} = P_i - 10n_i \log(d_{ij}) + R. \quad (5.4)$$

In the equation above,  $d_{ij}$  is the distance between the  $j$ -th user's position and AP  $i$  and  $p_{ij}$  is the RSS received at the  $j$ -th user's position from AP  $i$ .  $P_i$  is the reference RSS at a distance of 1 m from the AP and  $n_i$  is the attenuation exponent for AP  $i$ . Both  $n_i$  and  $P_i$  are treated as unknowns.  $R$  is a random variable which should account for multipath effects and other RSS variations. Equation 5.4 can be rewritten to obtain the distance:

$$d_{ij} = 10^{\frac{P_i - p_{ij}}{10n_i}}. \quad (5.5)$$

In a 2D coordinate system with  $m$  AP which are visible at  $n$  unknown locations, the number of observations is  $m \cdot n$ .

The number of unknowns can be calculated as follows: each of the  $n$  unknown user locations has two unknowns (the 2D coordinates  $x$  and  $y$ ), which gives  $2n$  unknowns. Each AP has four unknowns, two of which are its coordinates. The others are the parameters  $P_i$  and  $n_i$ , which yields  $4m$  unknowns for all AP. In total, the number of unknown parameters is therefore  $4m + 2n$ .

If the number of observations is bigger than the number of unknowns ( $m \cdot n > 4m + 2n$ ), the system of equations can be solved.

If the coordinates of the AP are estimated together with the path loss coefficient of a simple attenuation model, this simple attenuation model can then be used to interpolate the RSS observations on a grid of reference points.

# Chapter 6

## Integration of vehicle sensor data

For many applications in navigation, the desired accuracy cannot be reached using a single positioning system. To achieve higher accuracy, more robustness and reliability, different sensors can be combined. In the application case of indoor positioning for vehicles in parking garages, radio fingerprinting can be combined with a DR obtained from vehicle sensor data.

This chapter will first give a general introduction to sensor fusion for navigation applications in Section 6.1. It will take a look at the different types of redundancies, possible updating techniques and then explain the principle of Kalman filtering and different stages of integration. Section 6.2 will describe vehicle sensor data which can be used to compute relative position differences, focusing on sensor data which can be retrieved through a Controller Area Network (CAN) bus in real-time. Section 6.3 will then present a real-time filter combining radio fingerprinting with vehicle sensor data, which has been developed in the scope of this thesis.

### 6.1 Sensor fusion

This section deals with the principles of sensor fusion in navigation. It discusses redundancies, updating techniques and Kalman filtering. The following explanations are based on Hofmann-Wellenhof et al. (2003).

**Redundancies.** Sensor fusion is based on redundant information, meaning that more information is present to solve a task than required. Four types of redundancy can be distinguished:

- **Parallel redundancy:** occurs when various identical sensors are used. An example of parallel redundancy would be mounting two BLE antennas on a vehicle.
- **Complementary redundancy:** arises when several sensors with different operation principles and characteristics are used. The sensors are complementary, meaning that the disadvantage of one sensor can be complemented by the advantage of the other sensor. The most well-known example of complementary redundancy is the use of GNSS sensors in combination with inertial navigation. The combination of fingerprinting with DR also falls into this category.
- **Dissimilar redundancy:** occurs when two or more non-identical and non-complementary sensors are used. An example would be the combination of Loran-C and GNSS: both allow computing positions and are based on electromagnetic radio waves, but their signal structures and system architectures differ completely.
- **Analytical redundancy:** can occur when there is additional analytical knowledge which can be used to compute the position. An example would be map aiding: information from a digital map can serve as pseudo-observations. In a scenario where only two GNSS pseudoranges are available due to an obstructed signal path, the position of the receiver cannot be computed (at least four pseudoranges would be needed). If analytical information from a map is present, i.e. if it is known that the receiver is located in a street and the height and direction of the street are known from the map, the position of the receiver can be obtained.

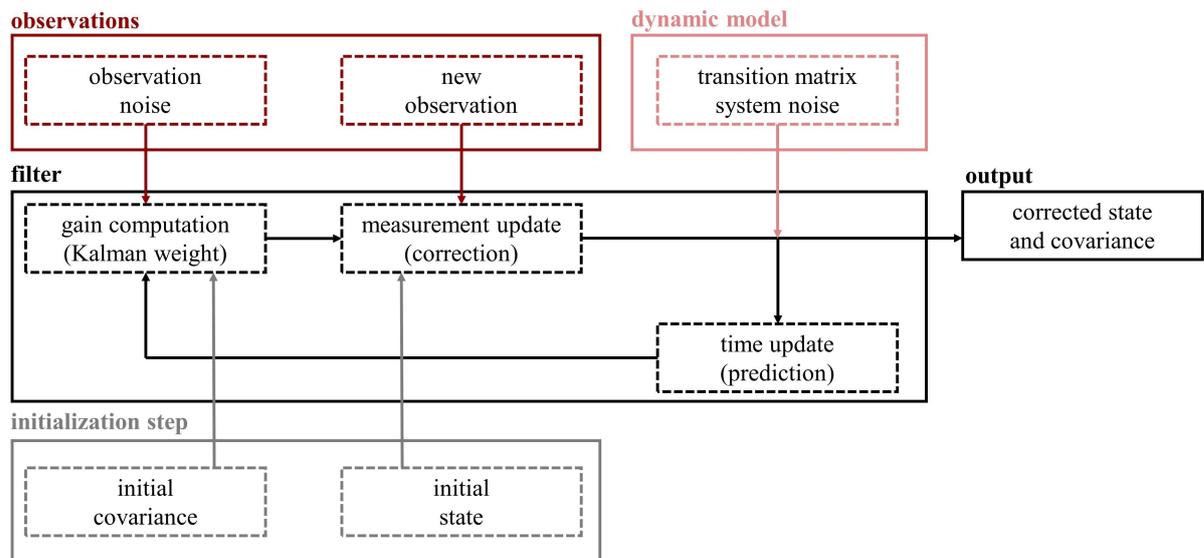
**Updating techniques.** Based on the type of redundancy, different updating techniques can be considered for sensor fusion. Hofmann-Wellenhof et al. (2003) distinguish between three types of updating techniques:

- **Signal blending:** is applied to compute a simple weighted average of the sensor output; it is mostly used for parallel redundancy. It can only be employed when sensors produce the same output [11]. Information on the quality of the output can be used for weighting.
- **Filtering:** is a more sophisticated method which requires a dynamic motion model and uses stationary stochastic models of the system noise.
- **Optimal filtering:** also requires a dynamic motion model. In contrast to conventional filtering, a time-variant stochastic model is used for the system noise.

The updating techniques presented above differ in the complexity and quality of the achievable result. Signal blending is the most simple updating technique; however it can only be used for identical sensors and does not consider dynamic motion. Optimal filtering is the most complex updating technique, yet it yields the best results for navigation applications. Optimal filtering can be achieved with the use of a particle filter or a Kalman filter. The latter was used in this thesis and will now be presented in more detail.

### 6.1.1 Kalman filtering

A Kalman filter is a recursive filter algorithm which can be used to estimate time-varying parameters of a dynamic process. An example of a dynamic process is a moving vehicle. A Kalman filter can be used to estimate the parameters and covariance of the vehicle's state (coordinates, velocity, accelerations and attitude). The filter combines noisy observations with a dynamic motion model to estimate the state and its stochastic behaviour [18].



**Figure 6.1:** Schematic process of Kalman filtering based on Dorn (2014).

An overview of the filtering process is given in the schematic diagram in Figure 6.1. The filter is based on a recursive least squares adjustment. In contrast to a classical least squares adjustment, it considers the state and its stochastic process as time-dependent, nonstationary random processes. The filter algorithm itself consists of three major steps:

gain computation, measurement update and time update. The Kalman filter is a Best Linear Unbiased Estimator (BLUE) for the state vector [20]. *Best* means that the variance of the estimate is lowest [53], *linear* that the parameters of the state can be estimated from a linear combination of the observations, and *unbiased* means that the estimated value corresponds to the expected value.

In the following, the three major steps of the Kalman filtering algorithm will be explained. The notation used is taken from Hofmann-Wellenhof et al. (2003).

1. **Gain computation.** In the first step of the Kalman filter, the Kalman weight  $\mathbf{K}_k$  is calculated. Here the subscript  $k$  stands for the discrete epoch  $t_k$ . The Kalman gain weighs the observations in relation to the prediction of the previous epoch. It is calculated from

$$\mathbf{K}_k = \tilde{\mathbf{P}}_k \mathbf{H}_k^T (\mathbf{H}_k \tilde{\mathbf{P}}_k \mathbf{H}_k^T + \mathbf{R}_k)^{-1}, \quad (6.1)$$

where  $\tilde{\mathbf{P}}_k$  is the predicted covariance matrix at epoch  $t_k$  and  $\mathbf{R}_k$  is the covariance matrix containing the measurement noise.  $\mathbf{H}_k$  is the design matrix which describes the functional relationship between the measurements and the parameters of the state vector. Note that the Kalman gain is only computed from covariance information and the design matrix and not the actual measurements or prediction itself. For the first epoch, the predicted covariance matrix of the parameters does not exist yet and is therefore replaced with the initial covariance matrix.

2. **Measurement update (correction step).** In the second filter step, the predicted state vector  $\tilde{\mathbf{x}}_k$  and predicted covariance matrix  $\tilde{\mathbf{P}}_k$  are updated or *corrected* with the measurements from epoch  $t_k$ . The measurement update is computed from the following formulas:

$$\hat{\mathbf{x}}_k = \tilde{\mathbf{x}}_k + \mathbf{K}_k (\mathbf{z}_k - \mathbf{H}_k \tilde{\mathbf{x}}_k) \quad (6.2)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \tilde{\mathbf{P}}_k. \quad (6.3)$$

In Equation 6.2, the Kalman weight  $\mathbf{K}_k$  weighs the measurements  $\mathbf{z}_k$  in relation to the prediction. If the Kalman weight is high, the measurements are highly precise and given a strong weight. If the measurements have a high standard deviation and are less precise than the prediction, the Kalman weight is close to zero and the measurements are therefore given a low weight. In the latter case, the estimated parameters  $\hat{\mathbf{x}}_k$  are mainly calculated from the prediction  $\tilde{\mathbf{x}}_k$ . In Equation 6.3,  $\mathbf{I}$  stands for the identity matrix. Note that in the first epoch, the initial state vector is used instead of the predicted state vector in Equation 6.2 and the initial covariance matrix replaces the predicted covariance matrix in Equation 6.3.

3. **Time update (prediction step).** During the time update, the previously estimated parameters  $\hat{\mathbf{x}}_k$  and their covariance matrix  $\mathbf{P}_k$  are predicted for the next epoch using a dynamic motion model. It models the behaviour of the moving object between two consecutive epochs. For vehicles, a model of linear motion is often assumed. The state vector is predicted for the epoch  $t_{k+1}$  with

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{\Phi}_k \hat{\mathbf{x}}_k, \quad (6.4)$$

where  $\mathbf{\Phi}_k$  is the transition matrix containing the motion model. The covariance matrix is predicted for the next epoch with

$$\tilde{\mathbf{P}}_{k+1} = \mathbf{\Phi}_k \mathbf{P}_k \mathbf{\Phi}_k^T + \mathbf{Q}_k, \quad (6.5)$$

where  $\mathbf{Q}_k$  is the covariance matrix of the motion model.  $\mathbf{Q}_k$  contains the uncertainties of the dynamic model and is computed from the following variance propagation:

$$\mathbf{Q}_k = \mathbf{N}_k \mathbf{R}_n \mathbf{N}_k^T. \quad (6.6)$$

Equation 6.6 expresses the system noise via  $\mathbf{R}_n$ , a covariance matrix of the parameters neglected by the motion model. If a linear motion model (no accelerations) is assumed,  $\mathbf{R}_n$  contains the covariances of the accelerations.  $\mathbf{N}_k$  expresses the functional relationship between the estimated parameters  $\hat{\mathbf{x}}$  and the parameters neglected by the motion model.

The three steps of the Kalman filter are repeated recursively. For epoch  $t_{k+1}$ , the predicted covariance matrix  $\tilde{\mathbf{P}}_{k+1}$  from the previous time update is used to compute the new Kalman weight  $\mathbf{K}_{k+1}$  together with the covariance matrix  $\mathbf{R}_{k+1}$  of the new measurements. The predicted state vector  $\tilde{\mathbf{x}}_{k+1}$  is then corrected in the measurement update, using the Kalman weight and the new observations  $\mathbf{z}_{k+1}$ . The corrected state vector  $\hat{\mathbf{x}}_{k+1}$  and covariance matrix  $\mathbf{P}_{k+1}$  are then predicted in the time update and the whole process starts anew.

### 6.1.2 Stages of integration

The Kalman filter can be used to process data from one sensor to compute the state vector along a trajectory, but it can also be used to integrate measurements from different sensors by means of optimal filtering.

When data from multiple sensors are processed, centralized and decentralized filters can be considered [20]. A central filter directly processes all sensor data in one step. Decentralized filters use a two-step procedure: first, data from different sensors are preprocessed individually; then a master filter processes the preprocessed data together in the second step.

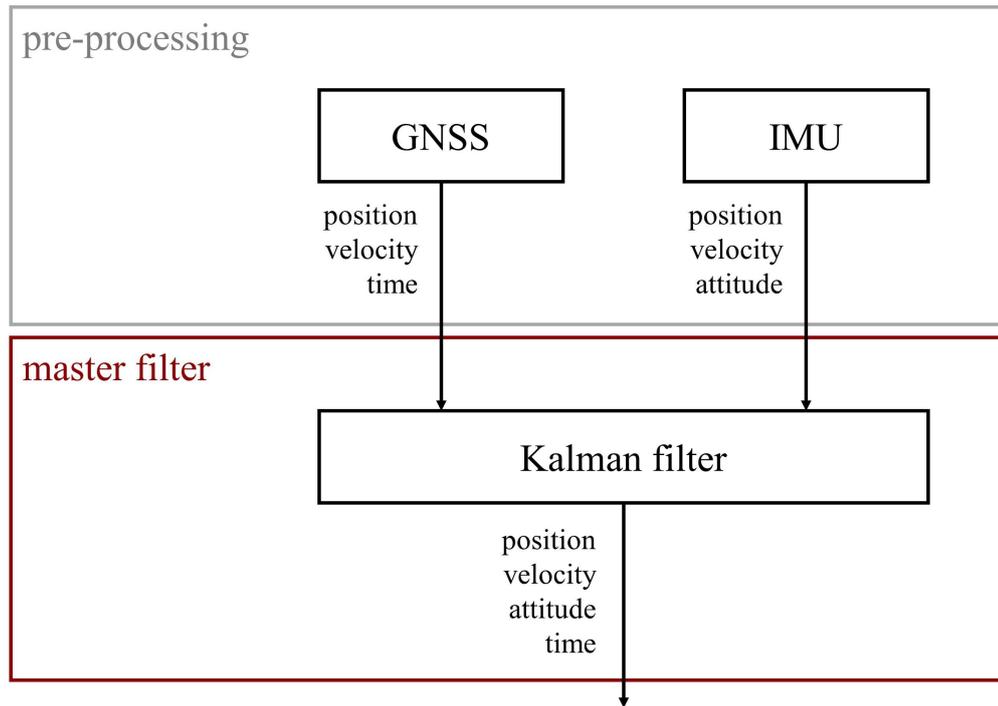
When it comes to the integration of different sensor data with a Kalman filter, three levels of integration can be distinguished, using uncoupled, loosely coupled and tightly coupled filters. Uncoupled and loosely coupled filters are decentralized, while tightly coupled filters are centralized.

In navigation applications, sensors which are complementarily redundant are often integrated. The idea behind this type of integration is that the advantages of one sensor compensate for the disadvantages of the other. The integration is therefore expected to be a fusion of advantages. An example of sensors which are complementarily redundant and often fused is the combination of a GNSS receiver and an Inertial Measurement Unit (IMU). While the IMU compensates for the mediocre short-term stability and vulnerability due to exterior disturbances of GNSS, GNSS makes up for the poor long-term stability of the IMU. The example of a GNSS and IMU sensor fusion is used in the following to illustrate the different stages of integration for filtering.

**Uncoupled integration.** Uncoupled integration, as pictured in Figure 6.2, relies on a two-stage, decentralized filter. In the first filter step, the sensor data are pre-processed. In the case of GNSS, the raw data are processed to obtain position, velocity and time; for an IMU, a strapdown algorithm can be used to calculate position, velocity and attitude. The end products of the pre-processing stage are then integrated in a common master filter.

**Loosely coupled integration.** Loose coupling also uses a two-stage, decentralized filter and is shown in Figure 6.3. Similar to uncoupled integration, the sensor data are first pre-processed and then processed together in a master filter. What differentiates loose coupling from an uncoupled integration is that in a loosely coupled scenario the master filter is used to estimate additional parameters, such as systematic sensor errors. These parameters are then returned as feedback to the filters in the pre-processing stage, where they are used to better process the sensor data.

**Tightly coupled integration.** In the case of tight coupling, a central filter is used, as depicted in Figure 6.4. The central filter directly processes the raw sensor data. If GNSS



**Figure 6.2:** Uncoupled integration of GNSS and IMU.

and IMU are tightly coupled, the master filter processes pseudorange, Doppler and the carrier phase obtained by GNSS together with the angular rates and specific forces of the IMU. The mathematical formulation of a tightly coupled filter is complex because a functional relationship between raw data and the parameters of the state vector has to be found. The advantage of tight coupling is that updates can be computed even if not enough observations are present to obtain an individual position from the sensor data. In the example of GNSS, at least four pseudoranges from satellites are needed to calculate a position (to account for four parameters, the receiver clock error and a three-dimensional coordinate tuple). If less than four satellites are tracked by the receiver, uncoupled and loosely coupled filters discard the GNSS observations, as they are unable to compute a position in the pre-processing stage. A tightly coupled filter can integrate raw data from less than four satellites and use them to improve the IMU-only solution.

When it comes to the integration of fingerprinting with additional sensor data, only uncoupled or loosely coupled filters can be considered. Fingerprinting is a non-parametric method, which means that there is no mathematical function which describes the relation between the raw data (the observed RSS) and the position. A tightly coupled filter can therefore not be used in combination with fingerprinting.

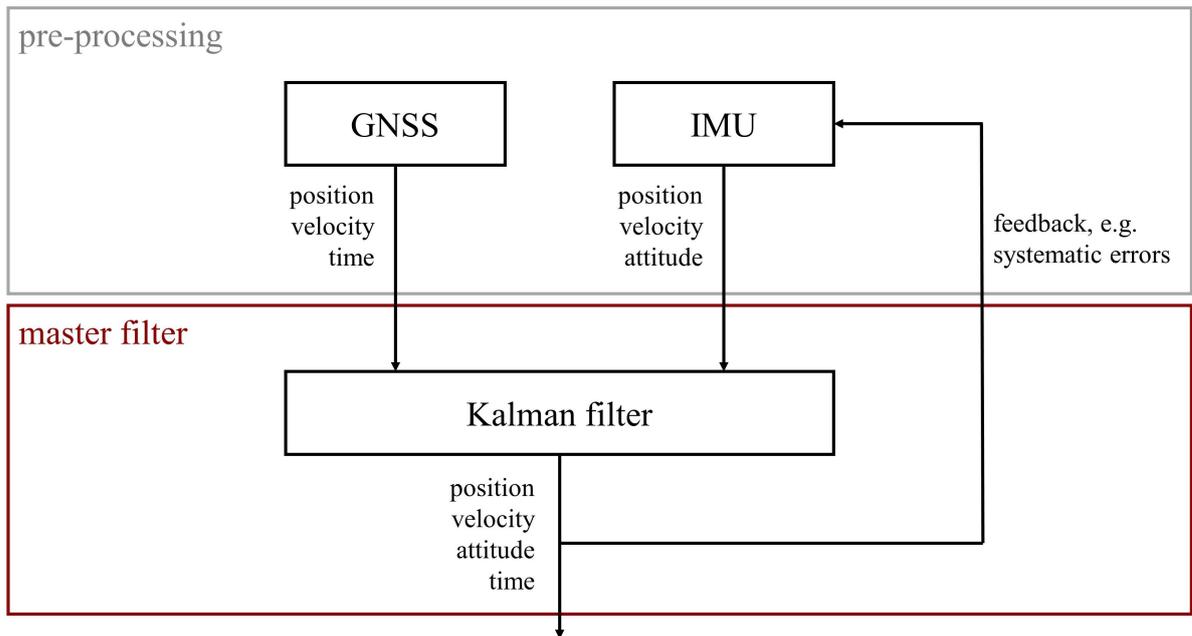


Figure 6.3: Loosely coupled integration of GNSS and IMU.

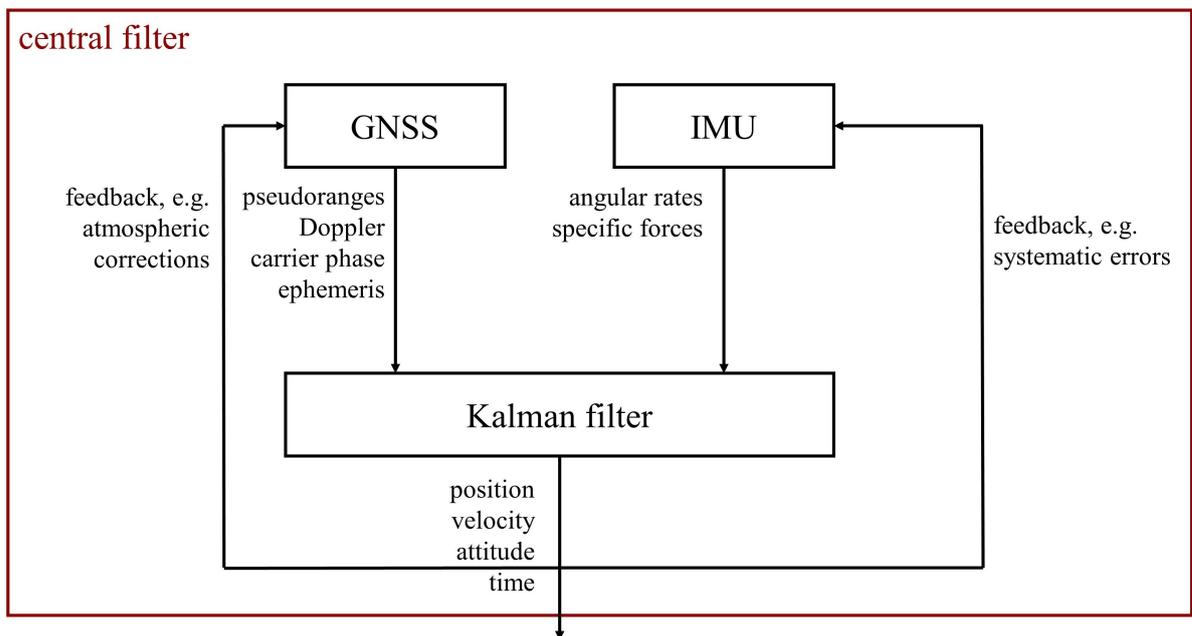


Figure 6.4: Tightly coupled integration of GNSS and IMU.

## 6.2 Vehicle sensor data

Modern vehicles are equipped with a variety of sensors, which are necessary in vehicle control systems such as the Anti-lock Braking System (ABS) or Electronic Stability Programme (ESP). Some of the sensors can also be used for positioning. In this section, an interface which can be used to retrieve vehicle sensor data will be presented along with the sensor data which can be used for positioning.

A variety of bus standards are used nowadays in vehicles, such as Local Interconnected Network (LIN), FlexRay, Media Oriented System Transport (MOST), K-Line, Ethernet and CAN [56]. CAN is a serial bus protocol which was originally developed by Bosch [34] at the beginning of the 1980s. It became a standard protocol as ISO-11898 (high-speed CAN, 1 Mbit/s) and ISO-11519 (low-speed CAN, 125 kbit/s) and is now widely used in the automotive sector [44]. Reasons for the popularity of CAN are its simple protocol, low cost and reduced wiring complexity, as only two cables are needed to communicate over a CAN bus [44] [56].

Via a CAN bus, sensor data can be retrieved from the vehicle. In previous investigations [55] [56], these data included acceleration, wheel speeds, yaw rate, steering wheel rate and angle, yaw rate and various integer flags for vehicle states. The yaw rate and speed are of particular interest for positioning.

**Yaw rate.** *Yaw* is an angle of rotation referring to a turn around a body's  $\mathbf{x}_3$ -axis. The axes of a vehicle body system and the attitude parameters *roll*, *pitch* and *yaw* are depicted in Figure 6.5. The origin of the system lies in the vehicle's center of mass. The  $\mathbf{x}_1$ -axis points towards the front,  $\mathbf{x}_3$  is orthogonal to  $\mathbf{x}_1$  and points upwards. The  $\mathbf{x}_2$ -axis is orthogonal to  $\mathbf{x}_1$  and  $\mathbf{x}_3$ , forming a left-handed three-dimensional Cartesian coordinate frame. The attitude parameter *roll* describes a turn around the vehicle's  $\mathbf{x}_1$ -axis (e.g. when the vehicle is in a tilted position), *pitch* describes a turn around the  $\mathbf{x}_2$ -axis (e.g. when the vehicle is driving upwards or downwards) and *yaw* is a turn around the  $\mathbf{x}_3$ -axis (e.g. resulting from the turn of the steering wheel in a plane).

The yaw rate refers to the change of the parameter *yaw*. Yaw rate sensors measure the angular rate around the  $\mathbf{x}_3$ -axis. The sensor used in previous research [55] [56] was a Microelectromechanical System (MEMS), located in the Engine Control Unit (ECU) together with accelerometers.

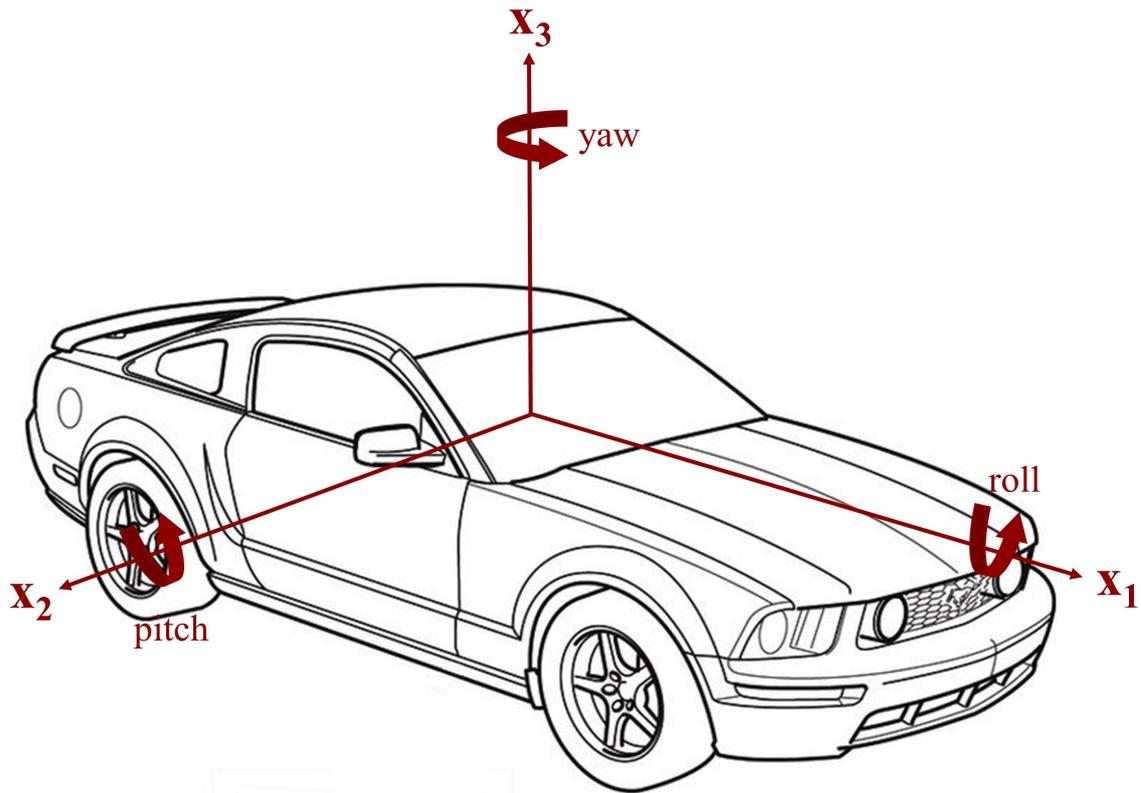


Figure 6.5: Body system of a vehicle.<sup>1</sup>

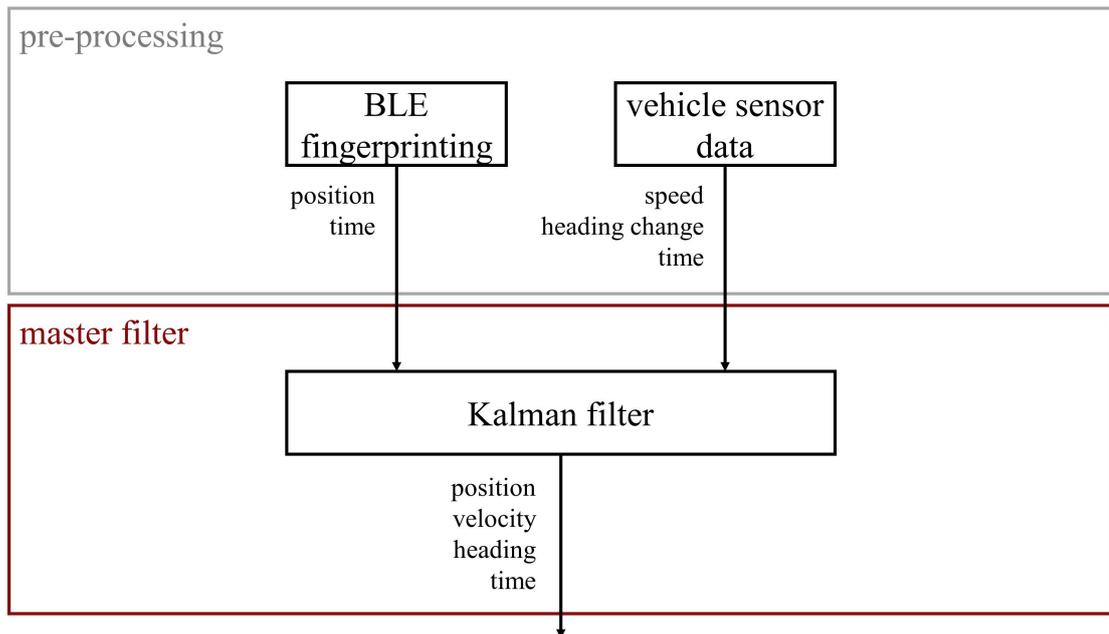
**Vehicle speed.** The speed of the individual wheels can be measured with a Hall sensor [56]. The individual wheel speeds are used in the ABS to detect when a vehicle starts to slip. For positioning, it is not the individual wheel speeds but the vehicle speed which is of interest.

To obtain the vehicle speed, the wheel speeds of the non-propelled wheels are averaged for vehicles with a two-wheel drive. The propelled wheels are not taken into consideration as they might be affected by wheel slip. The average of left and right wheel is taken due to the fact that in curves, the outer wheel covers a larger distance than the inner wheel in the same time interval; therefore the wheel speed of the outer wheel is higher than the one of the inner wheel. By averaging the two speeds, the speed in the middle of the vehicle, where the origin of the body system lies, is obtained.

<sup>1</sup>Picture of the vehicle taken from: <http://diysolarpanelsv.com/images/black-mustang-car-clipart-3.jpg>, last accessed 21.09.2017

### 6.3 Real-time filter

One of the main research aims of this thesis is the development of a filter capable of integrating positions obtained through fingerprinting and vehicle sensor data in real-time. As discussed in Section 6.1.2, only decentralized filters can be taken into consideration for fingerprinting. An uncoupled integration was chosen. A basic overview of the filter architecture is given in Figure 6.6.



**Figure 6.6:** Decentralized filter architecture combining BLE fingerprinting and DR from vehicle sensor data.

The RSS are pre-processed using a fingerprinting algorithm to obtain a two-dimensional position. The vehicle sensor data are pre-processed to obtain the mean vehicle speed and a heading change. Whenever new, pre-processed information comes in, the master filter computes an update. The advantage of combining BLE fingerprinting, an absolute positioning technique, with DR from vehicle sensor data, a relative positioning technique, is their complementary redundancy. Fingerprinting has a poor short-term accuracy due to its noisiness; a DR from vehicle sensor data is affected by drift, as systematic errors are accumulated. However, a high short-term accuracy can be achieved by using vehicle sensor data; and fingerprinting is not affected by drifts, as it is an absolute positioning

method. Integrating both methods should lead to a fusion of advantages, i.e. to a smooth trajectory unaffected by drift.

This section is structured as follows: first, the pre-processing stage will be explained. Afterwards, the filter architecture of the master filter will be presented, where the vehicle sensor data and BLE updates are investigated separately. Lastly, the implementation of a possible pseudo-heading update will be discussed.

### 6.3.1 Pre-processing

In decentralized filters, the observations are pre-processed before they are handed over to the master filter. The flowchart in Figure 6.7 shows the workflow of the pre-processing stage for a filter combining RSS measurements and vehicle sensor data.

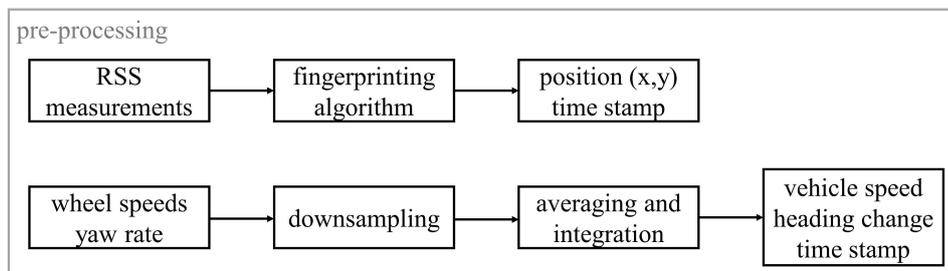
The observations for absolute positioning, RSS, are received from AP. The RSS are used to estimate a position with a fingerprinting algorithm as described in Chapter 3.

The vehicle sensor data used for positioning, wheel speeds and yaw rate, are first sampled down to the desired sampling rate. The wheel speeds of the non-propelled wheels are averaged to obtain the mean vehicle speed  $v$ :

$$v = \frac{v_{left} + v_{right}}{2}. \quad (6.7)$$

The yaw rate  $\dot{h}$  is measured in degrees per second. It has to be converted to radians by multiplying it by the time difference and a conversion factor to obtain the heading change  $\delta h$ . In doing so, it is important to ensure that the time difference which is used refers back to the previous vehicle sensor update:

$$\delta h_{CAN}(t_k) = \dot{h}(t_k) \cdot \Delta t_{CAN} \cdot \frac{\pi}{180} \quad \text{with} \quad \Delta t_{CAN} = t_k - t_{previousCAN}. \quad (6.8)$$



**Figure 6.7:** Pre-processing stage of a filter combining RSS and vehicle sensor data.

### 6.3.2 Master filter

The master filter is a Kalman filter which integrates the pre-processed data: a two-dimensional position obtained from fingerprinting, and the vehicle speed and heading change obtained after pre-processing vehicle sensor data.

First, the motion model and the parameters of the state vector will be explained. Then the measurement updates with the vehicle speed, heading change and position will be presented individually.

**Dynamic model.** The filter developed was designed for vehicles in parking garages. As parking garages can be expected to be plane, the state vector of the vehicle includes two-dimensional coordinates  $(x, y)$ ; moreover it also contains the velocities in the direction of the coordinate axes  $(\dot{x}, \dot{y})$  and the heading  $h$  (oriented direction of movement of the vehicle in the local-level frame):

$$\mathbf{x} = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \\ h \end{pmatrix}. \quad (6.9)$$

For the initialization of the filter, the first coordinate pair is taken from fingerprinting. The heading has to be set manually and the velocity can either be initialized with zero or with the speed obtained from the vehicle sensor data, which has to be brought to the local-level frame.

Taking into consideration that the filter should be capable of dealing with real-time data, the time update has to be the first computation for each epoch. The time difference is calculated by subtracting the time of the previous update from the current time:

$$\delta t = t_k - t_{k-1}. \quad (6.10)$$

During the time update, the estimated parameters of the state vector from the previous epoch  $\hat{\mathbf{x}}_{k-1}$  are predicted for the current epoch. The prediction  $\tilde{\mathbf{x}}_k$  is computed from:

$$\tilde{\mathbf{x}}_k = \mathbf{\Phi}_{k-1} \hat{\mathbf{x}}_{k-1}, \quad (6.11)$$

where  $\mathbf{\Phi}_{k-1}$  is the transition matrix describing the dynamic behaviour of the vehicle from the previous to the current epoch. A model of linear, uniform motion was chosen to

describe the vehicle dynamics. For the state vector given in Equation 6.9, the transition matrix  $\Phi$  for linear motion is defined as follows:

$$\Phi_{k-1} = \begin{pmatrix} 1 & 0 & \delta t & 0 & 0 \\ 0 & 1 & 0 & \delta t & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (6.12)$$

Not only the parameters of the state vector, also the covariance matrix of the measurements is predicted for the current epoch:

$$\tilde{\mathbf{P}}_k = \Phi_{k-1} \mathbf{P}_{k-1} \Phi_{k-1}^T + \mathbf{Q}_{k-1}, \quad (6.13)$$

with  $\mathbf{P}_{k-1}$  being the covariance matrix of the parameters of the estimated state vector in the previous epoch, and  $\mathbf{Q}_{k-1}$  the covariance matrix describing the uncertainty of the dynamic model.  $\mathbf{Q}_{k-1}$  is calculated from the variance propagation in Equation 6.6. The matrix  $\mathbf{N}$  expresses the functional relationship between the parameters of the state vector  $\mathbf{x}$  and the parameters neglected by the motion model  $\mathbf{n}$ :

$$\mathbf{x} = \mathbf{N} \cdot \mathbf{n}. \quad (6.14)$$

For a linear motion model, the accelerations  $(\ddot{x}, \ddot{y})$  and the heading change  $(\dot{h})$  are neglected:

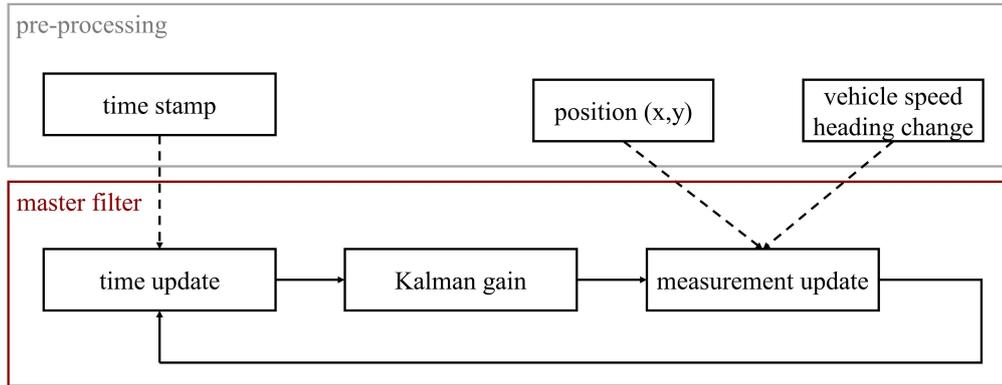
$$\mathbf{n} = \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \dot{h} \end{pmatrix}. \quad (6.15)$$

Taking equations 6.9, 6.14 and 6.15 into consideration, the matrix  $\mathbf{N}$  takes on the following form:

$$\mathbf{N} = \begin{pmatrix} 0.5\delta t^2 & 0 & 0 \\ 0 & 0.5\delta t^2 & 0 \\ \delta t & 0 & 0 \\ 0 & \delta t & 0 \\ 0 & 0 & \delta t \end{pmatrix}. \quad (6.16)$$

Assuming the accelerations and heading change to be uncorrelated, the system noise  $\mathbf{R}_n$  is modelled as follows:

$$\mathbf{R}_n = \begin{pmatrix} \sigma_{\ddot{x}}^2 & 0 & 0 \\ 0 & \sigma_{\ddot{y}}^2 & 0 \\ 0 & 0 & \sigma_{\dot{h}}^2 \end{pmatrix}. \quad (6.17)$$



**Figure 6.8:** Stages of the master filter.

**Measurement update with vehicle sensor data.** After the computation of the time update, the master filter computes the Kalman gain (see Equation 6.1) and the measurement update. An overview of the stages of the master filter is given in Figure 6.8.

In order to compute the Kalman gain and then conduct the measurement update, the functional relationship between the pre-processed sensor data and the parameters of the state vector has to be known. In the case of heading change and vehicle speed, the functional relationship to the parameters of the state vector is nonlinear and therefore Extended Kalman Filtering (EKF) has to be applied [20].

In EKF, the non-linear functional relationships are linearized [2] [13]. If the changes of the parameters of the state vector are small, the behaviour of the functional models between two epochs is almost linear. The predicted values of the previous epoch can be taken as approximations of the state vector, and the linearized models are used to estimate the small changes.

To derive the mathematical formulation of the linearized design matrix  $\mathbf{H}_k$ , assume the following notation for a nonlinear set of observation equations [20] with  $l_k$  observations and  $n$  parameters of the state vector:

$$\mathbf{z}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k. \quad (6.18)$$

In Equation 6.18,  $\mathbf{z}_k$  are the observations (of dimension  $1 \times l_k$ ),  $\mathbf{x}_k$  is the state vector (of dimension  $1 \times n$ ) and  $\mathbf{h}_k$  is a vector containing the observation functions  $h_k(\mathbf{x}_k)$ . The vector  $\mathbf{v}_k$  contains the observation noise, which is assumed to be normally distributed with the null vector as the expected value ( $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ ).

The elements of the design matrix  $\mathbf{H}_k$  (dimension  $l_k \times n$ ) can be obtained by linearizing the observation functions as follows [20]:

$$H_{k(i,p)} = \left. \frac{\partial h_{k(i)}(\mathbf{x})}{\partial x_p} \right|_{\mathbf{x}=\tilde{\mathbf{x}}_k} \quad \text{with} \quad \begin{cases} i = 1, \dots, l_k, \\ p = 1, \dots, n, \end{cases} \quad (6.19)$$

where  $\tilde{\mathbf{x}}_k$  is the predicted state vector.

For the measurement updates it was decided that only parameters relative to the previous epoch should be estimated to avoid jumps in the time series of the heading. The change in the state vector,  $\delta\mathbf{x}$ , contains the following elements:

$$\delta\mathbf{x} = \begin{pmatrix} \delta x \\ \delta y \\ \delta \dot{x} \\ \delta \dot{y} \\ \delta h \end{pmatrix}. \quad (6.20)$$

The observations, the pre-processed vehicle sensor data, are reduced by subtracting the predictions:

$$\mathbf{z}_{k, \text{reduced}} = \mathbf{z}_k - \mathbf{h}_k(\tilde{\mathbf{x}}_k) = \begin{pmatrix} v \\ \delta h \\ 0 \end{pmatrix} - \begin{pmatrix} \tilde{x} \cdot \cos(\tilde{h}) - \tilde{y} \cdot \sin(\tilde{h}) \\ 0 \\ \tilde{x} \cdot \sin(\tilde{h}) + \tilde{y} \cdot \cos(\tilde{h}) \end{pmatrix}. \quad (6.21)$$

Equation 6.21 includes three observations: the mean vehicle speed, the heading change, and the speed in the lateral direction of the vehicle, which is zero. To subtract the predicted speeds, they have to be brought to the same coordinate system as the measurements. The predicted parameters of the state vector are given in a local-level frame, whereas the vehicle speed measurements refer to the body frame of the vehicle. As the speeds are relative and the vehicle is expected to move parallel to the  $(xy)$ -plane of the local level system, only a rotation around the  $\mathbf{x}_3$ -axis by the heading has to be performed. The predicted heading change is always zero as the linear motion model does not assume changes in the direction of movement.

The linearized design matrix has the following form:

$$\mathbf{H}_k = \begin{pmatrix} 0 & 0 & \cos(\tilde{h}) & -\sin(\tilde{h}) & -\tilde{x} \cdot \sin(\tilde{h}) - \tilde{y} \cdot \cos(\tilde{h}) \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \sin(\tilde{h}) & \cos(\tilde{h}) & \tilde{x} \cdot \cos(\tilde{h}) - \tilde{y} \cdot \sin(\tilde{h}) \end{pmatrix}. \quad (6.22)$$

The uncertainties of the observations are assumed to be uncorrelated:

$$\mathbf{R}_k = \begin{pmatrix} \sigma_v^2 & 0 & 0 \\ 0 & \sigma_{\delta h}^2 & 0 \\ 0 & 0 & \sigma_{v_{lateral}}^2 \end{pmatrix}. \quad (6.23)$$

Now, all matrices are defined to calculate the Kalman gain according to Equation 6.1. The measurement update for EKF when only relative parameters of the state vector (as defined in Equation 6.20) are used can be simplified to:

$$\delta \hat{\mathbf{x}} = \mathbf{K}_k \cdot \mathbf{z}_{k, reduced}. \quad (6.24)$$

The covariance matrix  $\mathbf{P}_k$  is updated according to Equation 6.3.

The absolute parameters of the state vector are obtained by adding the estimated change  $\delta \hat{\mathbf{x}}$  to the predicted state vector:

$$\hat{\mathbf{x}} = \tilde{\mathbf{x}} + \delta \hat{\mathbf{x}}. \quad (6.25)$$

**Measurement update with fingerprinting.** The measurement update for fingerprinting is simpler, as the relationship between the observations and parameters of the state vector is linear. To allow the simultaneous updating of both fingerprinting and vehicle sensor data, the fingerprinting filter was also designed to update the relative parameters of the state vector (Equation 6.20).

The observation vector is therefore reduced as follows:

$$\mathbf{z}_{k, reduced} = \mathbf{z}_k - \mathbf{H}_k \tilde{\mathbf{x}}_k = \begin{pmatrix} x_{fp} \\ y_{fp} \end{pmatrix} - \begin{pmatrix} \tilde{x}_k \\ \tilde{y}_k \end{pmatrix}, \quad (6.26)$$

with the design matrix

$$\mathbf{H}_k = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \quad (6.27)$$

The observations are assumed to be uncorrelated:

$$\mathbf{R}_k = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}. \quad (6.28)$$

The Kalman gain is computed according to Equation 6.1. The measurement update is calculated analogously to Equation 6.24:

$$\delta \hat{\mathbf{x}} = \mathbf{K}_k \cdot \mathbf{z}_{k, reduced}. \quad (6.29)$$

As for the measurement update with vehicle sensor data, the covariance of the parameters is updated with Equation 6.3 and the state vector is updated with Equation 6.25.

### 6.3.3 Pseudo-heading update

Given that only relative parameters are estimated and the heading update is only computed from heading changes, the heading itself is never directly updated from absolute (fingerprinting) data. Correlations between the position and heading during the filter process were investigated, but they were of a very low magnitude ( $10^{-3}$ ). In order to overcome this problem, a pseudo-heading-update calculated from fingerprinting data can be implemented.

The heading update cannot be computed from the filtered position. Assume that a wrong start-heading was chosen and only vehicle sensor data are available. In this case, the filtered trajectory would be rotated with respect to the real trajectory by the difference between the chosen start-heading and the real heading. If a heading were computed from these positions, it would not yield any additional information and also point into the direction of the rotated, filtered trajectory.

Taking these thoughts into consideration, it becomes clear that the heading update must be calculated from positions obtained by fingerprinting. Since the positions obtained by fingerprinting have a high standard deviation, the heading updates should only be computed when the vehicle drives straight ahead for a longer period of time to minimize the influence of outliers. To ensure that the vehicle is driving straight ahead when the heading update is computed, a heading change threshold is set ( $1.5^\circ$ ). The yaw rates are multiplied by their respective time intervals and added up (from last to first) until the cumulative sum  $h_{change}$  exceeds the threshold:

$$h_{change} = \sum_i \dot{h}_i \cdot \Delta t_i. \quad (6.30)$$

When the threshold is exceeded, the time stamp  $t_{exceed}$  of the exceeding heading is stored. The pseudo- heading is then computed as the azimuth from the first fingerprinting position, where  $t_{fp} \geq t_{exceed}$ , to the current fingerprinting position:

$$h_{pseudo} = \tan^{-1} \left( \frac{\Delta y}{\Delta x} \right). \quad (6.31)$$

The pseudo-heading update is added to the fingerprinting measurement update in the Kalman filter. The reduced observations are computed according to the following equation:

$$\mathbf{z}_{k,reduced} = \begin{pmatrix} x_{fp} \\ y_{fp} \\ h_{pseudo} \end{pmatrix} - \begin{pmatrix} \tilde{x}_k \\ \tilde{y}_k \\ \tilde{h}_k \end{pmatrix}. \quad (6.32)$$

The design matrix is extended by a third row:

$$\mathbf{H}_k = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad (6.33)$$

The measurement update with the pseudo-heading is then computed analogously to the fingerprinting update.

## Part III

# Practical investigations and results

# Chapter 7

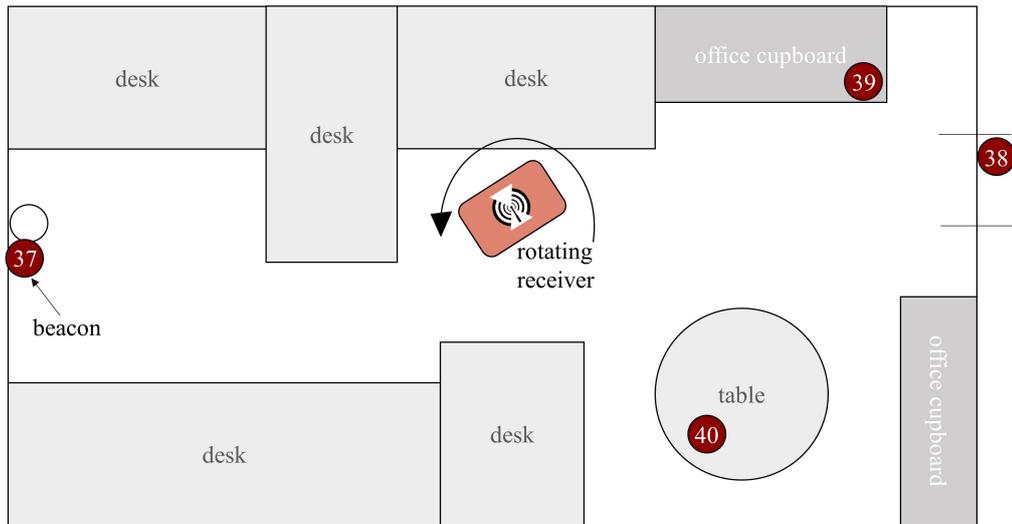
## Practical investigations

This chapter presents the practical investigations which were carried out for this thesis. It deals with considerations about different measurement setups, which lead to the final test measurements.

### 7.1 RSS distribution

For probabilistic fingerprinting, the likelihood  $P(\mathbf{r}|\mathbf{p})$  of measuring an RSS sample  $\mathbf{r}$  at the position of a reference point  $\mathbf{p}$  needs to be modelled. If the likelihood distribution is modelled using parametric approaches, assumptions concerning the underlying distribution have to be made. In radio fingerprinting, a normal distribution is most commonly assumed [31]. To find out whether this assumption is valid or not, test measurements were carried out in the scope of this thesis.

An office at the Institute of Geodesy at the Graz University of Technology (Steyrergasse 30, 2nd floor) served as the test bed for the measurements. The measurement setup is depicted in Figure 7.1. Four BLE beacons were deployed in the room: two with no obstacles in the line of sight (beacons 39 and 40), one behind the metal frame of a door (beacon 38) and one partly hidden behind a metal cable channel (beacon 37). The beaconing rate was set to 10 Hz. The receiver was placed on an office chair so that it could easily be rotated. Four test time series were measured, two with a static receiver, and two when the receiver was rotated. The measurements were recorded for 5 minutes. In each measurement series between 2100 and 2800 RSS samples were obtained per beacon.



**Figure 7.1:** Measurement setup for testing the RSS distributions.

The time series obtained were analysed with histograms and quantile-quantile plots (QQ-plots). Histograms show the relative frequency of how many of the samples fall into certain bins with a defined bin width. An approach to estimate the number of bins and the bin width was already presented in Chapter 3 (in Equation 3.9 and Equation 3.10). In a QQ-plot, quantiles of the sample are plotted against the quantiles of a theoretical distribution. If the plotted points lie on a straight line, the sample data follow the distribution against which they were plotted [45]. To test whether the RSS samples are normally distributed, they were plotted against a standard normal distribution.

An analysis of the time series obtained showed that when the receiver is not rotated while recording the RSS measurements, the data are not normally distributed. The time series for beacons 38, 39 and 40 are analysed in more detail. Figure 7.2 shows the histogram and QQ-plot for a static measurement of the RSS received from beacon 39. The curve of a normal distribution  $\mathcal{N}(\mu, \sigma^2)$  with mean  $\mu$  and variance  $\sigma^2$  estimated from the time series is shown in black together with the histogram. In the QQ-plot, the line is depicted on which the samples should lie if they were normally distributed. It can be seen that the RSS distribution is slightly left-skewed. Figure 7.3 shows the histogram and QQ-plot for the static measurement setup with RSS received from beacon 40. The sample data are strongly left-skewed and do not follow a normal distribution. The left-skewness of RSS samples was also observed in [31].

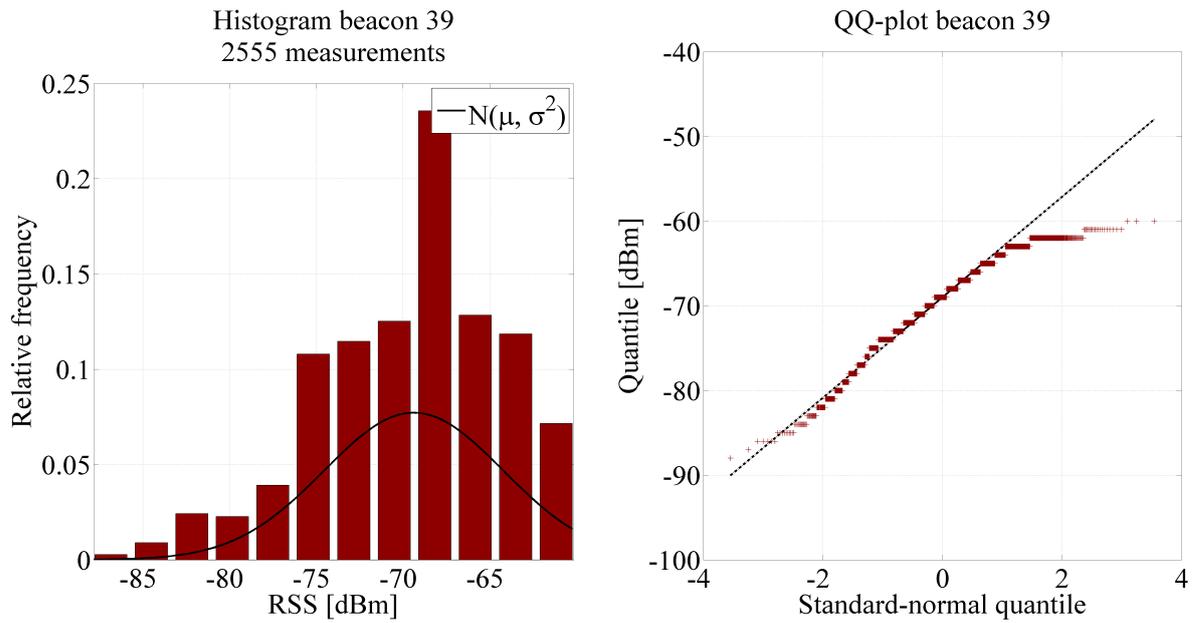


Figure 7.2: Histogram and QQ-plot for the static measurement setup for beacon 39.

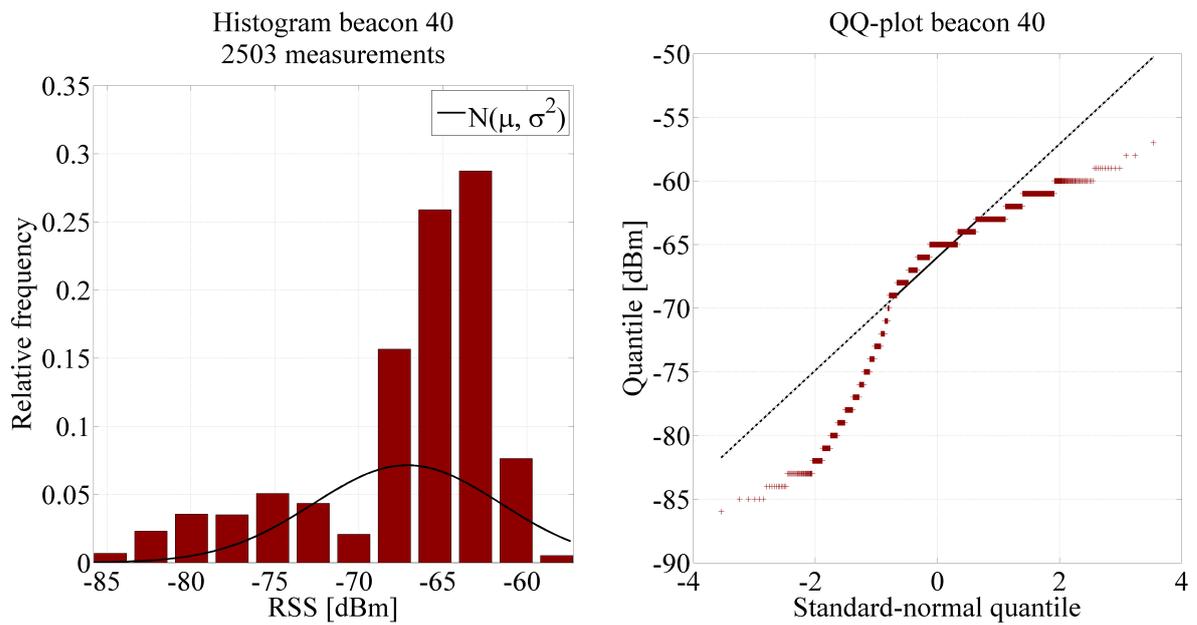
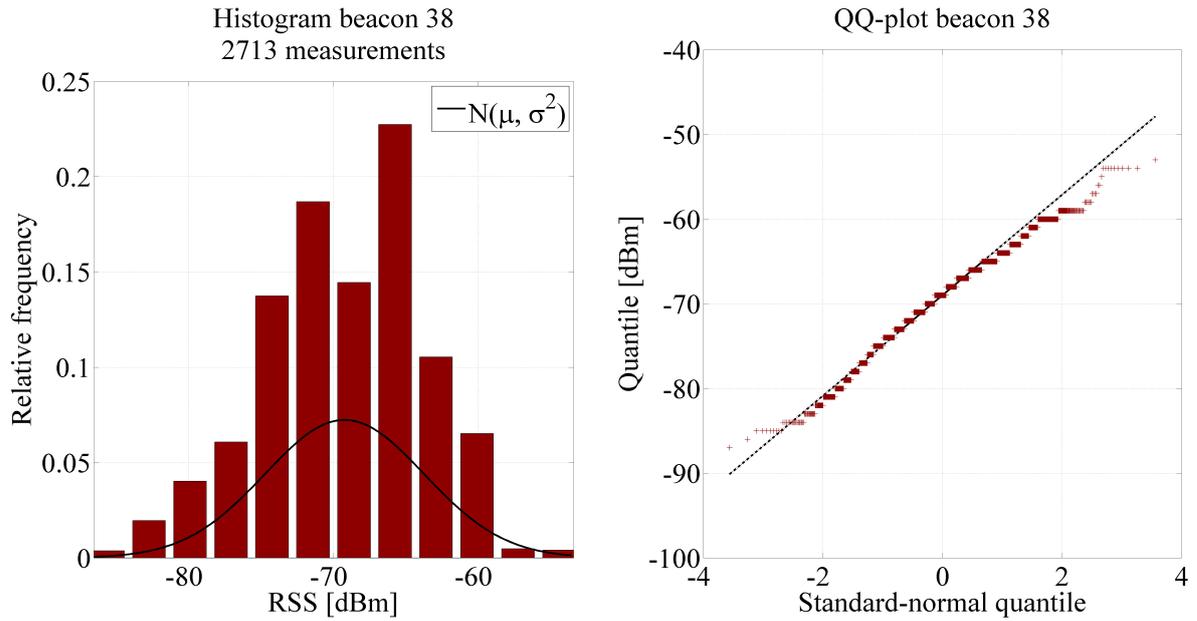
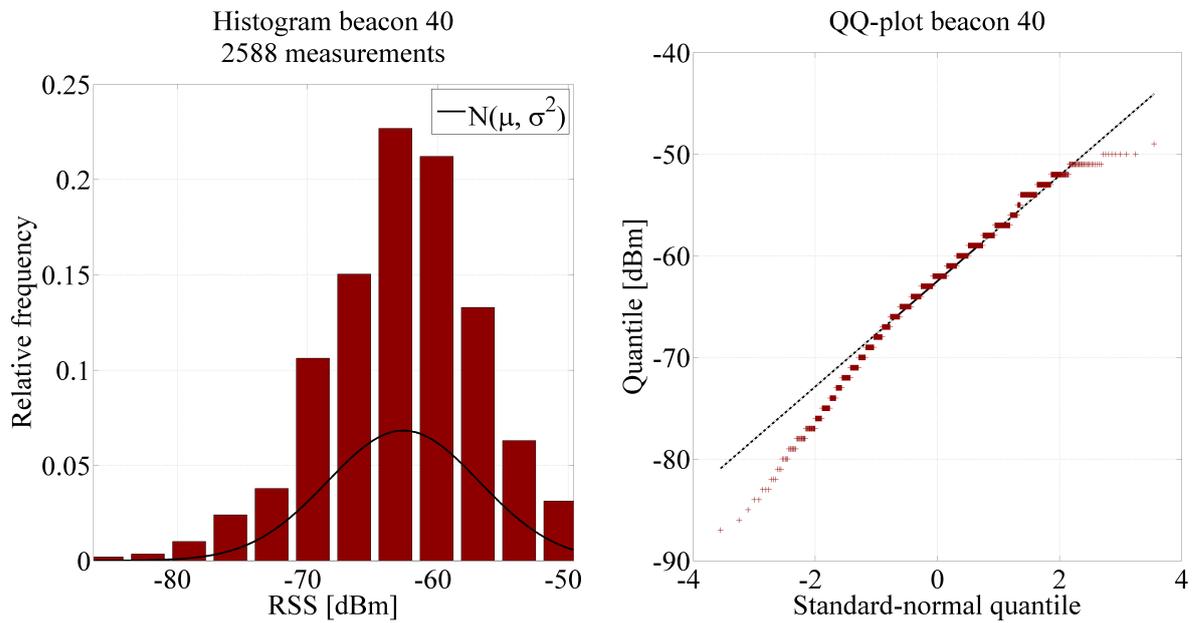


Figure 7.3: Histogram and QQ-plot for the static measurement setup for beacon 40.



**Figure 7.4:** Histogram and QQ-plot for the measurement setup with a rotating receiver for beacon 38.



**Figure 7.5:** Histogram and QQ-plot for the measurement setup with a rotating receiver for beacon 40.

Figures 7.4 and 7.5 show histograms and QQ-plots obtained from two RSS time series

when the receiver was rotated. The data seem to follow a normal distribution, however the time series for beacon 40 is also slightly left-skewed.

## 7.2 Test bed for probabilistic fingerprinting

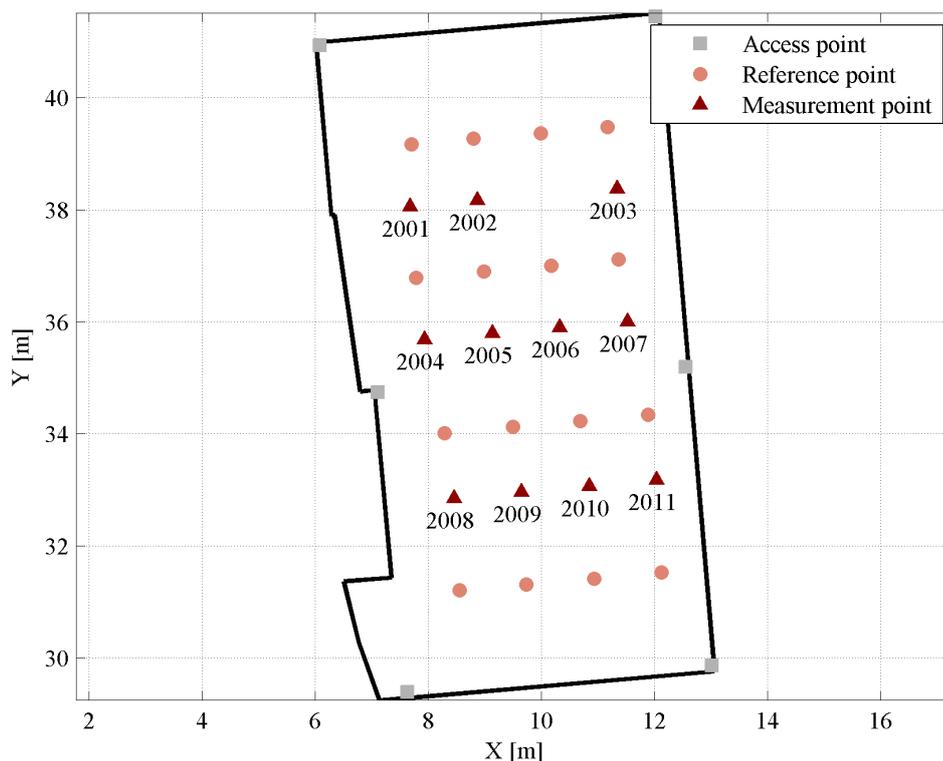
In a previous project ([54], [55], [56]), a radio map had already been recorded in a large parking garage (4800 m<sup>2</sup>) which also served as a test site for the investigations carried out for this thesis. Probabilistic fingerprinting was not yet taken into consideration during the previous project, thus measurements at the reference points were taken at a frequency of only 1 Hz. At each reference point, RSS were measured for 30 seconds. This leads to a maximum of 30 observations received from each AP at each reference point. Such a short time series is not ideal to compute the underlying stochastics of the data set and leads to a poor estimate of the mean and standard deviation if normal distribution is assumed.

In order to test probabilistic fingerprinting and compare it to its deterministic counterpart, a new radio map with a sufficiently large number of observations per reference point to each AP had to be measured. A test scenario was set up in a lecture room of the geodesy building at the Graz University of Technology (Steyrergasse 30) on the first floor. A picture of the lecture room is shown in Figure 7.6.



**Figure 7.6:** Lecture room *AE01* at Graz University of Technology.

In total, 6 BLE beacons were set up along the walls of the room and set to broadcast a signal at 10 Hz. At each reference point, a Lenovo IdeaPad 510S laptop (operating system Linux Mint 18) equipped with a LogiLink BT0015 antenna was set up. The programme *Vehicle Positioning Suite*, developed at the Institute of Geodesy, was used to record the RSS. Every 30 seconds, the antenna was turned by 90° so that after 2 minutes measurements had been taken in four directions. The procedure was repeated for all 16



**Figure 7.7:** Overview of reference points, access points and measurement points in lecture room *AE01*.

reference points. The coordinates of the reference points were obtained from a building plan and distometer measurements.

The deterministic radio map was constructed by averaging the RSS samples received from each of the AP at every RP. To obtain the likelihood densities for the probabilistic radio map, a parametric approach was chosen and the parameters of a normal distribution were estimated from the sample data.

In the online phase, the laptop was positioned at 11 measurement points. For 30 seconds, the laptop remained static at each measurement point and measured the RSS received from the surrounding access points. The *true* reference coordinates of the measurement points were obtained from distometer measurements. An overview of the reference, access and measurement points is given in Figure 7.7.

In post-processing, the RSS obtained at the measurement points were sampled down to 1 Hz by averaging. Both deterministic and probabilistic fingerprinting algorithms were used to estimate the position of the measurement points. A WKNN-approach was used

with  $K = 4$  reference points. For deterministic fingerprinting, the reciprocal value of the Euclidean distance (Equation 3.2) was used as the weight. In the case of probabilistic fingerprinting, logarithmic weighting was applied (see Equation 4.21). Finally, the Root Mean Square Error (RMSE) was computed as follows:

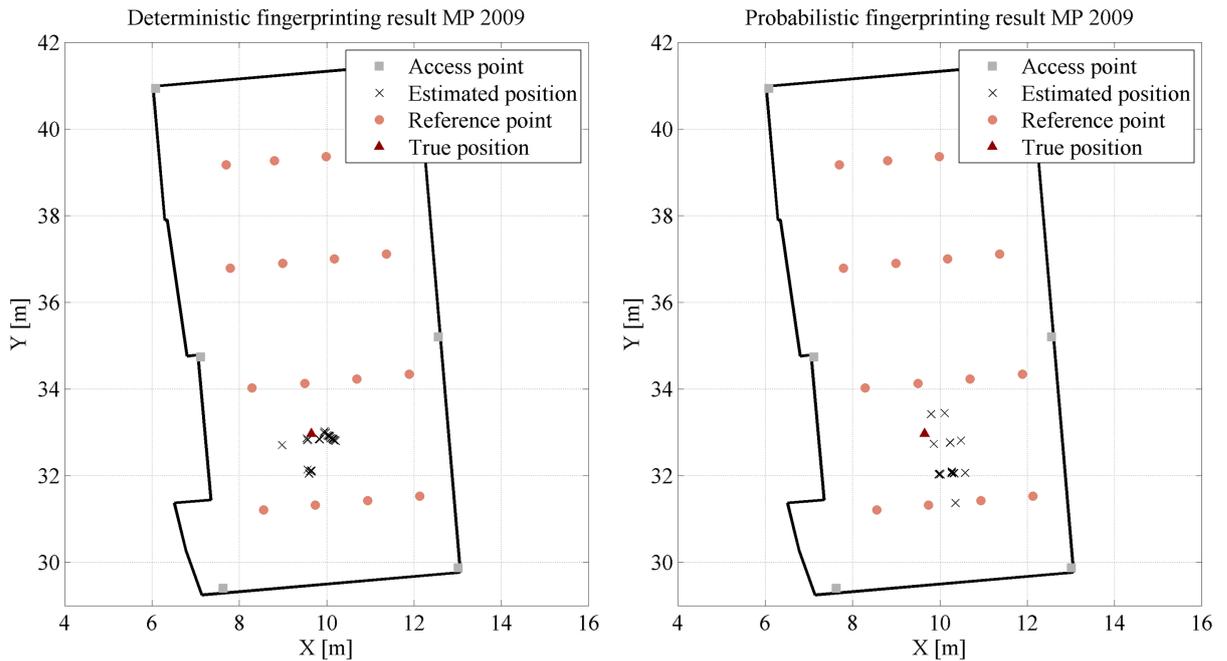
$$RMSE = \sqrt{\frac{\sum_{i=1}^n (x_{fp,i} - x_{ref})^2 + (y_{fp,i} - y_{ref})^2}{n}}, \quad (7.1)$$

where  $(x_{ref}, y_{ref})$  are the coordinates of the *true* (reference) position,  $(x_{fp}, y_{fp})$  the coordinates of the position computed from fingerprinting and  $n$  is the number of fingerprinting positions.

The RMSE results are depicted in Table 7.1. When relating the values to the placement of the measurement points in Figure 7.7, it can be seen that the points located in the edges of the setup (especially 2001, 2003 and 2011) have the highest RMSE. For almost all measurement points (except for points 2006 and 2011), probabilistic fingerprinting yields a higher RMSE than deterministic fingerprinting. Figure 7.8 shows a comparison of the results of deterministic and probabilistic fingerprinting for measurement point 2009. It can be seen that the estimated positions resulting from probabilistic fingerprinting are more scattered than those resulting from deterministic fingerprinting. A similar result was observable for the other measurement points as well. It can be concluded that the probabilistic fingerprinting algorithm is more sensitive to small changes in the observations, whereas the deterministic algorithm is more robust.

**Table 7.1:** RMSE of deterministic and probabilistic fingerprinting in lecture room *AE01*. A WKNN approach with  $K = 4$  reference points was used to obtain the results.

Measurement point	Deterministic fingerprinting RMSE [m]	Probabilistic fingerprinting RMSE [m]
2001	2.73	3.83
2002	2.92	3.62
2003	4.48	4.54
2004	1.14	1.83
2005	0.86	1.03
2006	1.98	1.52
2007	0.57	0.72
2008	1.57	2.03
2009	0.55	1.03
2010	3.71	3.80
2011	0.88	0.71



**Figure 7.8:** Comparison of deterministic and probabilistic fingerprinting for measurement point 2009.

### 7.3 First real-time tests

On 2 May 2017 the real-time filter developed to integrate vehicle sensor data and fingerprinting was first tested. The tests were carried out in a parking garage in Thondorf (Liebenauer Hauptstraße 316), which had already served for tests in a previous project [54]. A deterministic radio map for the garage had already been recorded in the scope of the previous project.

Figure 7.9 shows a picture of the test vehicle which was provided by Magna Steyr. The CAN interface was accessible as the car is a prototype. A sensor platform provided by the Institute of Geodesy was mounted on the roof of the vehicle. On the sensor platform, both the IMU for the reference trajectory and the BLE antenna for fingerprinting was mounted.

**Hardware used.** To collect the RSS measurements, a *LogiLink BT0015* BLE antenna was connected via a USB cable to the laptop, a *Lenovo IdeaPad 510S*. A *Peak-System PCAN-USB adapter* was needed to connect the laptop to the vehicle bus interface.



**Figure 7.9:** Test vehicle with the sensor platform mounted on the roof.

**Reference trajectory.** In order to investigate the achievable accuracy, a reference solution had to be set up. In a previous project [54], a tachymeter was used for tracking the vehicle. The disadvantage of this approach is that the tracking was only possible when there were no obstructions in the line of sight between the tachymeter and the vehicle, so only a small part of the actual trajectory was tracked.

To overcome these limitations, a new approach for the reference trajectory was developed using an IMU and a tachymeter. An IMU (iMar iNav-RQH) was mounted on the sensor platform. This highly precise IMU contains accelerometers and gyroscopes, whose sensor outputs can be used to calculate a trajectory with a strapdown algorithm. The strapdown algorithm requires the initial position and attitude as starting values. To obtain these initial values, a tachymeter was used. Figure 7.10 shows the IMU positioned on the platform and the tachymeter which was used for the measurements to obtain the absolute coordinates and attitude of the initial vehicle position.

The tachymeter (a *Leica total station* provided by the working group on *Remote Sensing and Photogrammetry* at the Institute of Geodesy) was positioned on a tripod in the parking garage. The distances, horizontal and zenith angles to three points with known coordinates (which had already been placed in the parking garage in the previous project) were measured so that a 3D resection could be performed. Before every test round, a prism was placed at a marked point on top of the IMU and a second prism was placed on the sensor platform further at the front, in the direction of the  $x_1$ -axis of the vehicle. The total station then measured the distance from and angles to these two prisms so that



**Figure 7.10:** An IMU and tachymeter measurements are needed to obtain the reference trajectory.

the coordinates of their positions could later be obtained in the post-processing stage.

**Time synchronisation.** In order to compare the reference trajectory from IMU and tachymeter measurements to the trajectory obtained from the integration of BLE fingerprinting and vehicle sensor data, the time stamps of the trajectories need to be synchronised. To achieve this synchronisation, the IMU and the laptop logging the RSS from the surrounding BLE AP as well as vehicle sensor data were connected to GNSS receivers. The IMU was connected to a *Javad SIGMA TriPad* and the *Lenovo IdeaPad 510S* to a *u-blox* receiver. Antennas connected to the receivers were held out of the windows of the parking garage so that they were able to receive signals from GNSS satellites. The offset between Global Positioning System (GPS)-time and system time of the laptop was logged. The same procedure was repeated after the test rounds. In post-processing, the time stamps of the reference and the computed integrated trajectories were brought to GPS time.

**Position of the BLE antenna.** Two different scenarios for the position of the BLE antenna were tested (see Figure 7.11). First, the BLE antenna was mounted on the central front pillar of the sensor platform. For the second rounds of testing it was placed inside the vehicle to find out whether this change had an impact on the accuracy of the positioning result.

### 7.3.1 Comparison to the reference trajectory

To compute the reference trajectory, the tachymeter data were processed using the software *Geosi VERM* (version 17) by *IDC EDV GmbH*. First, a 3D resection was



**Figure 7.11:** The BLE antenna was first mounted on the platform (left) and then placed inside the vehicle (right).

performed to obtain the position and orientation of the standpoint. Then the coordinates of the prisms were calculated with the first principal task. For the first principal task, the coordinates of the standpoint, the oriented direction to the unknown point, and the distance from the standpoint to the unknown point are used to calculate the coordinates of the unknown point. All coordinates were obtained in a Gauß-Krüger (GK) system.

Using the coordinates of the prism placed in the front part of the sensor platform and the coordinate of the position of the IMU, the initial heading was calculated with Equation 6.31. The heading and position of the IMU were used as initial values for the strapdown algorithm. The data obtained from the IMU were processed using the *Inertial Explorer®* by *NovAtel*.

*Inertial Explorer®* requires the input data to be given in the World Geodetic System 1984 (WGS84). Thus, the coordinates obtained through the tachymeter measurements had to be transformed from GK to WGS84. The reference trajectory calculated with *Inertial Explorer®* then had to be transformed to GK-coordinates. For information on coordinate transformations the reader is referred to [20].

In the next step, the lever arm between IMU and BLE antenna had to be considered. The coordinates of the reference trajectory refer to the centre of the IMU, whereas the integrated trajectory refers to the position of the BLE antenna. The reference trajectory was brought to the same position as the integrated trajectory by shifting it by  $r = 1.73$  m in the direction of the heading  $h$ :

$$y_{BLE} = y_{IMU} + r \sin(h) \quad (7.2)$$

$$x_{BLE} = x_{IMU} + r \cos(h). \quad (7.3)$$

Now, the RMSE of the trajectory computed from fingerprinting and the integration of vehicle sensor data with respect to the reference trajectory can be computed as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (x_{meas,i} - x_{ref,i})^2 + (y_{meas,i} - y_{ref,i})^2}{n}}, \quad (7.4)$$

where  $n$  is the number of epochs of the trajectories,  $(x_{meas}, y_{meas})$  are the coordinates of the computed trajectory and  $(x_{ref}, y_{ref})$  are the coordinates of the reference trajectory.

### 7.3.2 Results

The preliminary real-time tests showed that it is possible to integrate BLE fingerprinting and vehicle sensor data with the filter described in Section 6.3. When the achieved accuracy was assessed in post-processing, the accuracy of the fingerprinting solution was worse than expected. An overview of the RSME obtained from a deterministic, unfiltered fingerprinting-only solution is given in Table 7.2. A WKNN approach with  $K = 4$  and the inverse Euclidean norm as weights was used. The table shows the results obtained when the full radio map was used as well as the results for sparse fingerprinting, with 33 % and 50 % of the original fingerprints remaining respectively.

**Table 7.2:** RMSE of the unfiltered, deterministic fingerprinting solution for the test rounds on 2 May 2017.

Full radio map		Sparse radio map	
<i>Antenna above vehicle</i>		<i>33% of original data</i>	<i>50% of original data</i>
RMSE [m] round 1	6.81	6.11	6.48
RMSE [m] round 2	5.81	5.47	5.80
<i>Antenna inside vehicle</i>		<i>33% of original data</i>	<i>50% of original data</i>
RMSE [m] round 1	4.67	4.49	4.80

It can be seen that the RMSE is lower when the antenna is placed inside the vehicle and that the sparse radio map leads to a higher accuracy than the full radio map. These somewhat illogical results and the poor absolute accuracy lead to the conclusion that the radio map is outdated and needs to be measured anew.

## 7.4 Recording a new radio map for the Thondorf parking garage

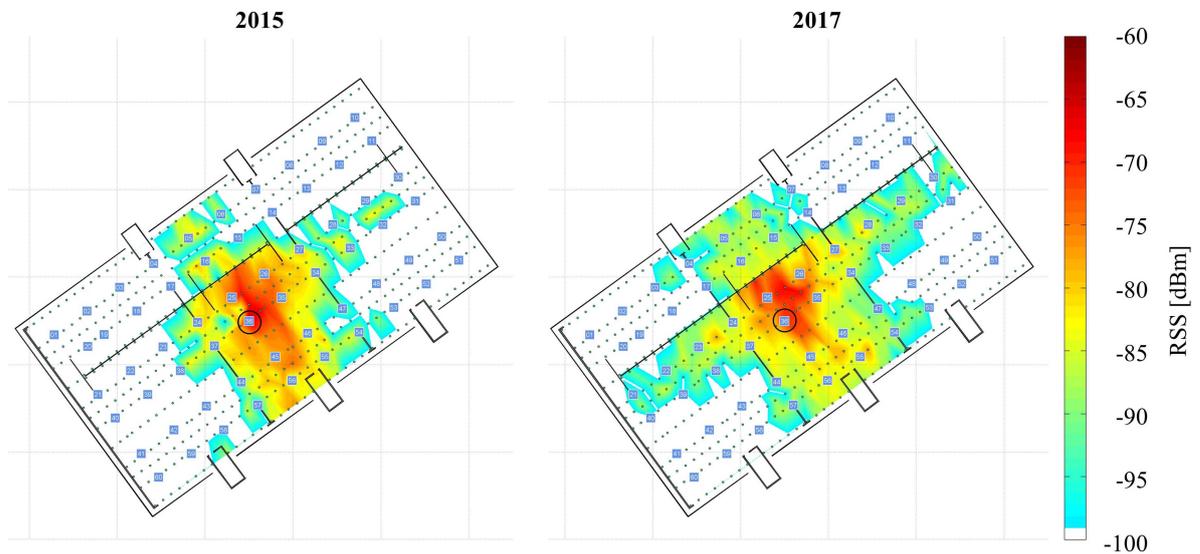
On 23 May 2017, the radio map for the 5th floor of the Thondorf parking garage was measured anew. In total, RSS measurements were recorded at 568 reference points. 60 BLE beacons were placed on the steel girders at the ceiling of the parking garage. Their beaconing rate was set to 10 Hz.

To record the radio map within one day, two laptops were used simultaneously. When recording radio maps, it is important that the measurements are taken at a height at which the BLE antenna is later located during the online-phase [43]. Both laptops were therefore placed on small tables. To avoid obstructions in the signal paths, the *LogiLink BT0015* BLE antennas were placed on top of the laptops. Furthermore, the observers knelt down while the RSS measurements were being recorded to avoid attenuating the BLE signals sent from the BLE beacons located at the ceiling. At each RP, RSS were recorded for 30 seconds. Figure 7.12 shows two pictures of the measurement setup at each reference point.

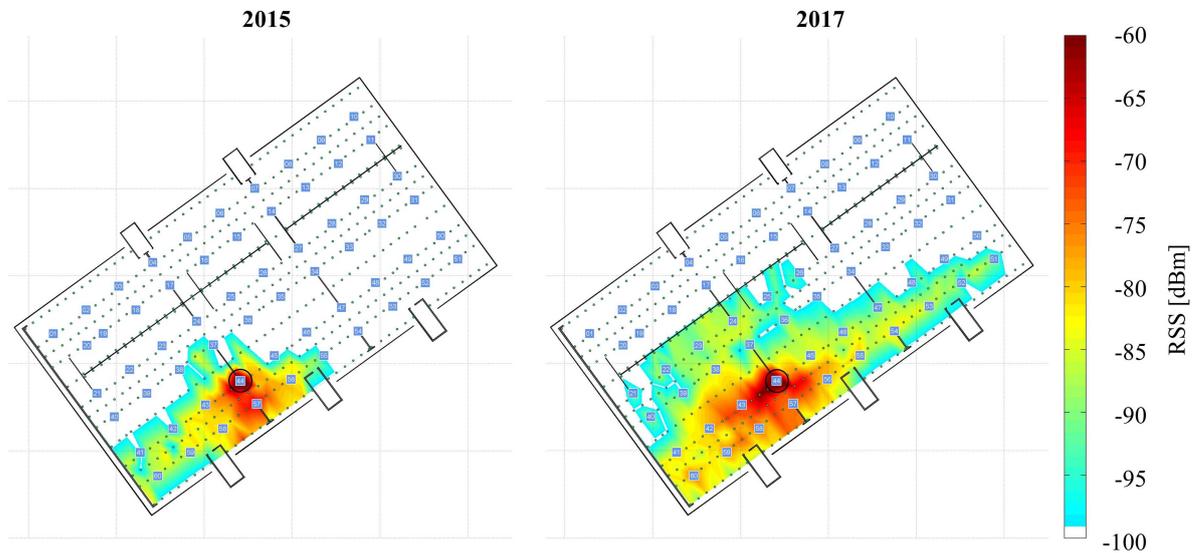


**Figure 7.12:** To avoid obstructing the signal path, the observers knelt down while the RSS were being recorded.

**Results.** Figures 7.13 and 7.14 show a comparison of the RSS received from AP 36 and AP 44, for the radio maps recorded in 2015 and in 2017. The plots were obtained through a linear interpolation between the RP in *Matlab*.



**Figure 7.13:** Comparison of the RSS received from AP 36, for the radio maps recorded in 2015 (left) and 2017 (right).



**Figure 7.14:** Comparison of the RSS received from AP 44, for the radio maps recorded in 2015 (left) and 2017 (right).

When comparing the RSS patterns from 2015 and 2017, it can be seen that the BLE

signals recorded in 2017 seem to have a greater range than those recorded in 2015. This can be attributed to the changes of the objects located in the parking garage. In 2015, when the first radio map was recorded, hardly any vehicles were parked on the 5th floor of the parking garage. In 2017, almost all parking spaces were occupied. As metal is a strong reflector for 2.4 GHz radio signals [17], the vehicles cause multipath effects. Due to multiple reflections, the BLE signals are carried further when more vehicles are present.

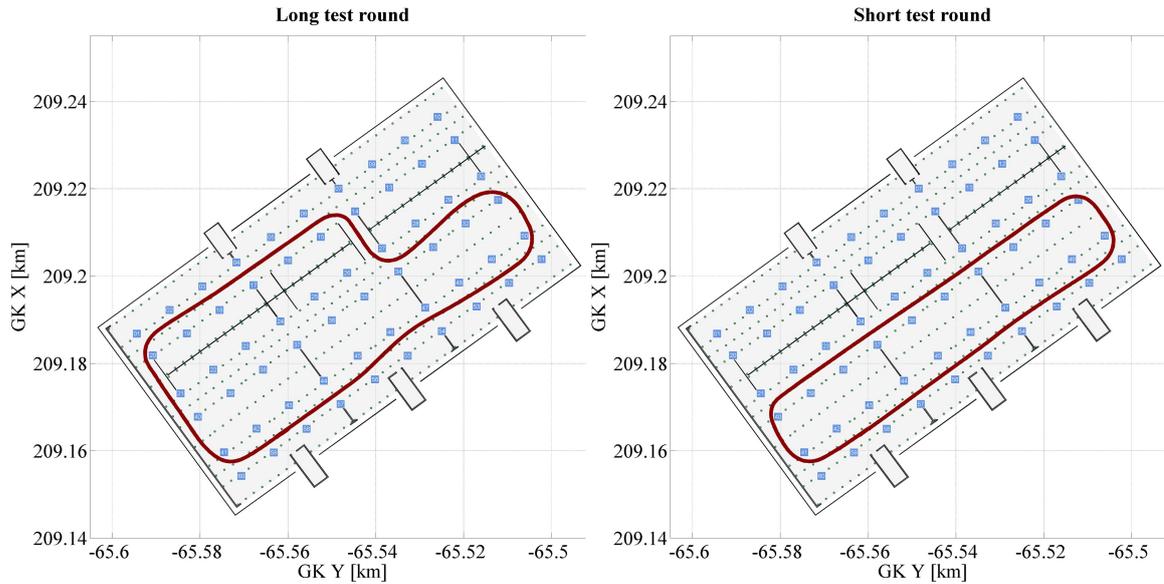
The new radio map was then used to compute trajectories from the test rounds on 2 May 2017. The results are shown in Table 7.3. When comparing it to the results obtained from the outdated radio map in Table 7.2, it can be seen that the accuracies improved with the use of the new radio map. All RMSE are below 5 m. When comparing these results to a study by Faragher and Harle (2015), who also investigated fingerprinting with BLE beacons, these results seem plausible. When Faragher and Harle (2015) used one BLE beacon per 100 m<sup>2</sup>, they achieved a positioning accuracy of < 5.5 m. The Thondorf parking garage has an area of approximately 4800 m<sup>2</sup>. With 60 beacons deployed, the beacon density is one beacon per 80 m<sup>2</sup>. With this beacon density, an accuracy of < 5 m was achieved.

**Table 7.3:** RMSE of the unfiltered, deterministic fingerprinting solutions for the test rounds on 2 May 2017 using the newly recorded radio map.

Full radio map		Sparse radio map	
<i>Antenna above vehicle</i>		<i>33% of original data</i>	<i>50% of original data</i>
RMSE [m] round 1	4.40	4.05	4.56
RMSE [m] round 2	4.25	4.45	4.58
<i>Antenna inside vehicle</i>		<i>33% of original data</i>	<i>50% of original data</i>
RMSE [m] round 1	4.24	4.61	4.64

## 7.5 Final test measurements

On 19 June 2017, the final test measurements took place in cooperation with colleagues from the Signal Processing and Speech Communication (SPSC) Laboratory of Graz University of Technology, who tested Ultra-Wideband (UWB) for the indoor localization of the test vehicle.



**Figure 7.15:** Two types of test rounds were used: a longer one (left) and a shorter one (right).

The test measurements were carried out analogously to the preliminary test measurements, which were already described in Section 7.3. Again, the prototype vehicle provided by Magna Steyr was used, and the IMU and tachymeter measurements were recorded for the reference trajectory. In addition to the IMU and BLE antenna, an UWB receiver was mounted on the sensor platform. In total, 13 test rounds were recorded, in 8 of which the BLE antenna was located on the sensor platform and in 5 of which the antenna was inside the vehicle, as depicted in Figure 7.11. The test rounds differed in length and are illustrated in Figure 7.15. The short test round only went through the south-eastern part of the parking garage, as the UWB solution by the SPSC Laboratory only provided coverage for this area.

Already before the final test measurements, battery problems with some BLE beacons occurred. When examining the BLE beacons again after the test rounds, it was found that some had stopped sending signals. To obtain representative measurements when all 60 beacons broadcast BLE signals, it was decided to repeat the test measurements.

60 new batteries were ordered to replace the old batteries in the BLE beacons. On 22 June 2017, the final test measurements were repeated. As the team of the SPSC Laboratory did not have to repeat their measurements, only long test rounds (see left side of Figure 7.15) were recorded. In total, 10 test rounds were measured, 5 with the antenna mounted on the sensor platform on top of the vehicle, and 5 with the antenna placed inside the vehicle. Before and after the test rounds, it was investigated whether

all beacons broadcast their advertisements. No battery problems occurred on 22 June 2017.

The recorded final test measurements serve as the basis for the investigations of fingerprinting optimization carried out in post-processing. The results will be presented in Chapter 8.

# Chapter 8

## Results

This chapter presents the results of the investigations of fingerprinting optimization. The radio map recorded in May 2017 (see Section 7.4) in the parking garage in Thondorf and the test rounds driven in June 2017 (see Section 7.5) served as the basis for these investigations.

Section 8.1 will describe the algorithmic optimization, Section 8.2 the economic optimization and Section 8.3 the integration of vehicle sensor data. To conclude this chapter and the thesis, an outlook for further research will be given in Section 8.4.

### 8.1 Algorithmic optimization

This section investigates the algorithmic optimization carried out in the scope of this thesis. The main focus is placed on finding the ideal weights for deterministic fingerprinting and on testing probabilistic fingerprinting.

#### 8.1.1 Deterministic fingerprinting

For deterministic fingerprinting, the radio map was obtained by averaging the RSS time series received from each AP at every RP. A WKNN approach with the  $K$  most similar RP was used to estimate the positions in the online phase.  $K$  was set to four.

Various norms were tested to find the  $K$  most similar RP and then used as weights in the WKNN algorithm. Of the norms presented in Section 4.1.2, the Euclidean, Manhattan, Sørensen, Neyman, Neyman<sub>2</sub> and  $\chi^2$  norms were compared.

Tables 8.1 and 8.2 show the RMSE of the unfiltered trajectories obtained from deterministic WKNN fingerprinting on 19 and 22 June respectively. When comparing the results for the measurement setups with different BLE antenna locations (above and inside the vehicle), it can be seen that all RMSE are smaller than or equal to 5.05 m when the antenna is located above the vehicle and smaller than or equal to 6.03 m when the BLE antenna is located inside the vehicle.

The different norms used have hardly any impact on the achievable positioning accuracy. The difference in minimum and maximum RMSE for each round is a few decimetres only, with the maximum range being 75 cm (round 7 on 22 June 2017, where the Sørensen norm yielded an RMSE of 5.46 m and Neyman<sub>2</sub> an RMSE of 4.71 m).

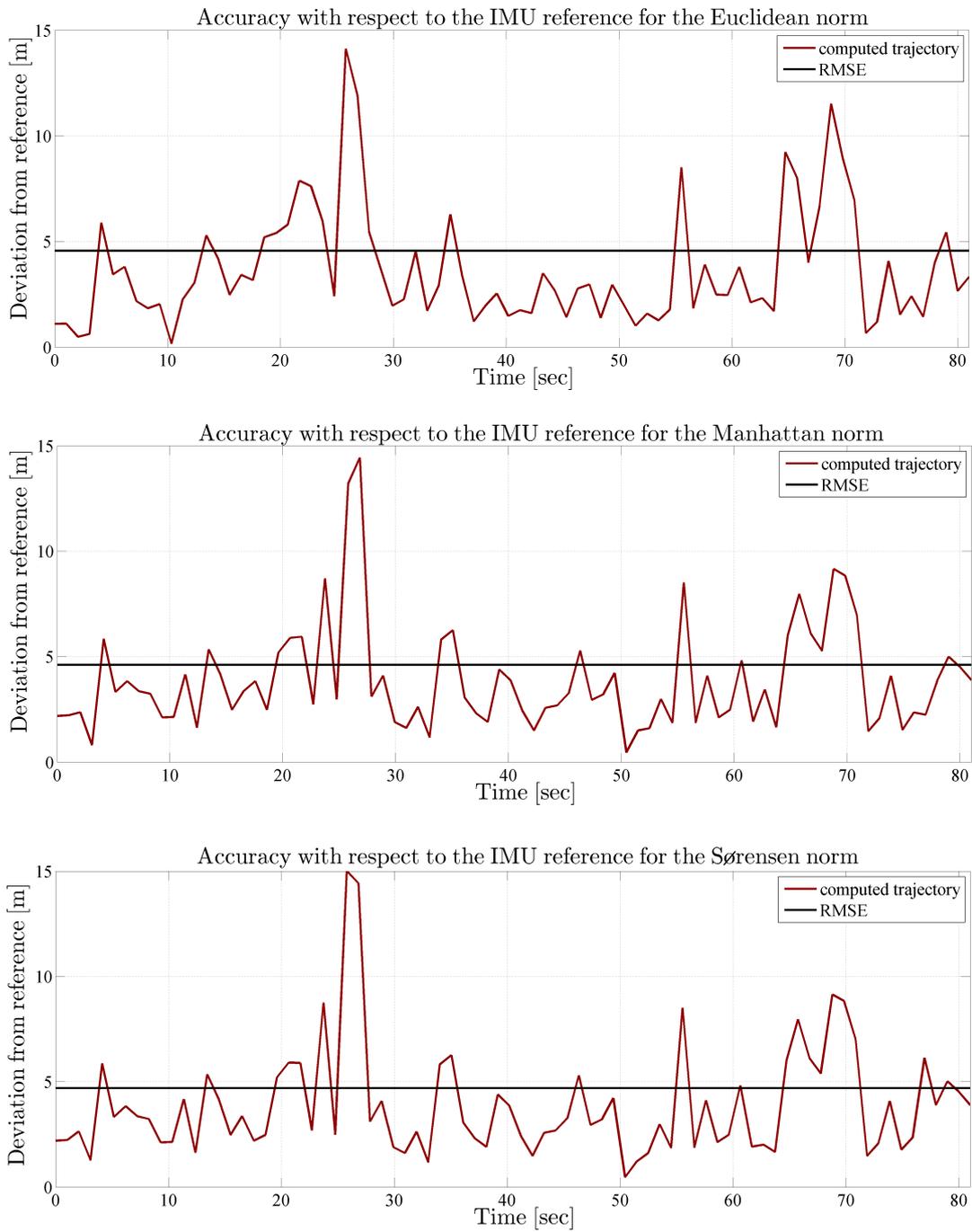
**Table 8.1:** Deterministic fingerprinting results for the measurements on 19 June 2017.

RMSE [m]						
Norm used	Euclidean	Manhattan	Sørensen	Neyman	Neyman <sub>2</sub>	$\chi^2$
<i>Antenna above vehicle</i>						
Round 1 (long)	4.12	4.03	4.08	4.05	4.10	4.11
Round 2 (long)	3.74	3.66	3.74	3.80	3.71	3.76
Round 3 (long)	3.88	3.92	3.94	4.02	3.87	3.91
Round 4 (long)	4.08	4.48	4.49	4.19	4.04	4.05
Round 5 (short)	4.96	4.70	4.70	4.98	4.95	4.99
Round 6 (short)	4.14	3.99	4.00	4.27	4.15	4.20
Round 7 (short)	4.60	4.82	4.93	4.61	4.58	4.63
Round 8 (short)	4.88	4.68	4.75	4.80	4.86	4.80
<i>Antenna inside vehicle</i>						
Round 9 (long)	5.48	5.65	5.71	5.53	5.48	5.49
Round 10 (long)	5.54	5.37	5.44	5.66	5.61	5.62
Round 11 (long)	4.60	4.57	4.61	4.93	4.65	4.76
Round 12 (long)	4.64	4.49	4.52	4.70	4.63	4.71
Round 13 (long)	4.46	4.77	4.86	4.83	4.54	4.73

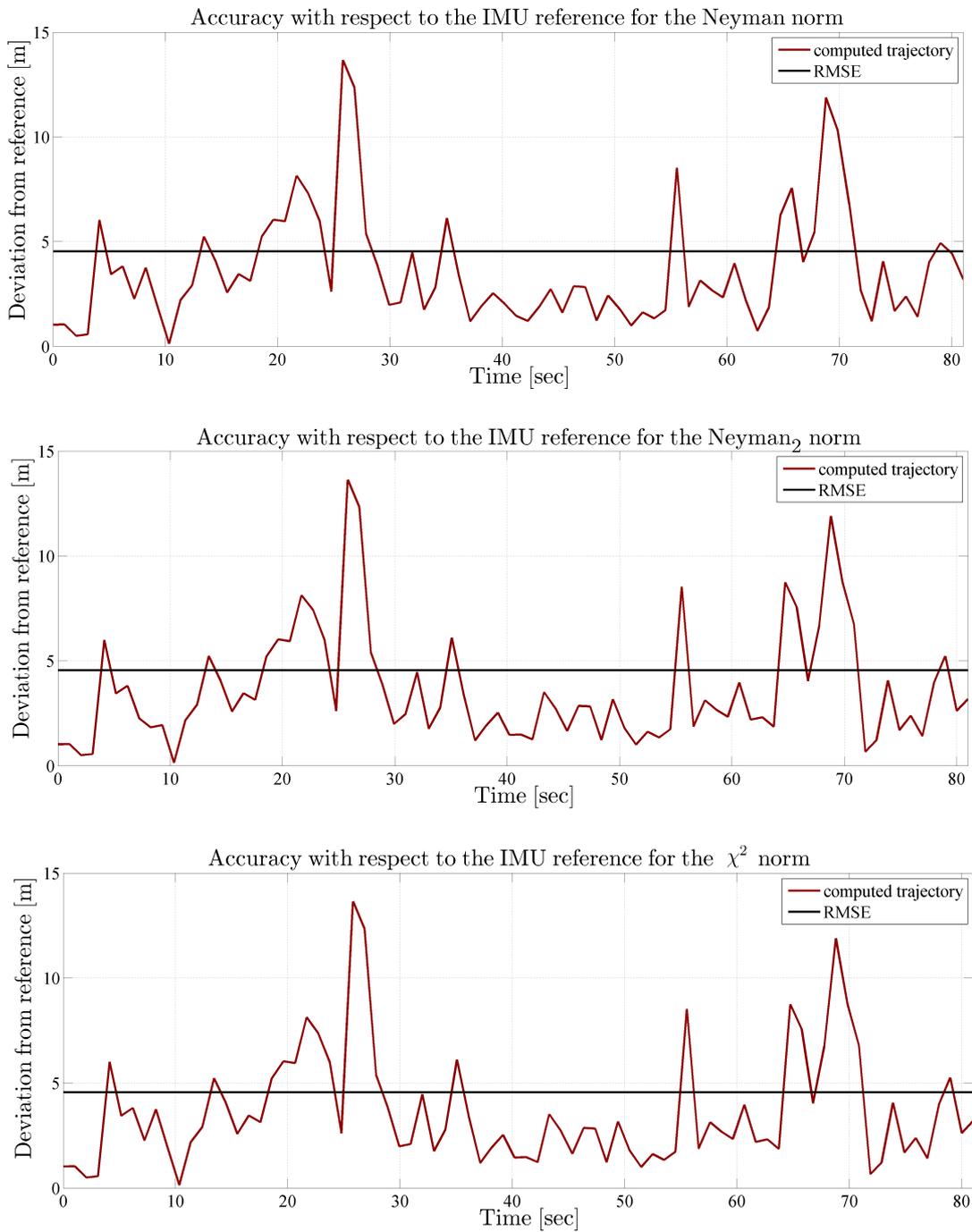
**Table 8.2:** Deterministic fingerprinting results for the measurements on 22 June 2017.

RMSE [m]						
Norm used	Euclidean	Manhattan	Sørensen	Neyman	Neyman <sub>2</sub>	$\chi^2$
<i>Antenna above vehicle</i>						
Round 1 (long)	4.56	4.60	4.70	4.53	4.54	4.56
Round 2 (long)	4.22	4.37	4.33	4.32	4.19	4.25
Round 3 (long)	4.60	4.54	4.66	4.61	4.53	4.54
Round 4 (long)	5.05	4.86	4.92	5.05	5.03	5.04
Round 5 (long)	4.45	4.40	4.38	4.43	4.39	4.41
<i>Antenna inside vehicle</i>						
Round 6 (long)	5.49	5.08	5.16	5.61	5.49	5.54
Round 7 (long)	4.74	4.92	5.46	4.89	4.71	4.77
Round 8 (long)	4.90	4.80	4.98	5.33	4.96	5.11
Round 9 (long)	6.03	6.05	6.19	5.99	6.03	6.05
Round 10 (long)	5.59	5.68	5.70	5.60	5.48	5.56

Since the RMSE provides only one value per trajectory, one might argue that hardly any changes are visible when different norms are used as weights because the RMSE averages the results of the individual positioning solutions. However, the individual positioning solutions were also investigated. Figures 8.1 and 8.2 show the time series of the deviations of the unfiltered fingerprinting solutions from the IMU reference trajectory for test round 1 on 22 June 2017. Here the term *deviation* refers to the distance between the coordinates of the reference trajectory and the coordinates of the computed trajectory obtained through fingerprinting. All time series show a similar course, meaning that the different norms used yield a similar positioning accuracy for the individual deterministic fingerprinting solutions. When investigating the maxima of the time series, it can be seen that the biggest outlier occurs after approximately 26 seconds and has a magnitude of 15 metres. The Euclidean norm as well as Neyman, Neyman<sub>2</sub> and  $\chi^2$  show a second outlier of over 10 metres at epoch  $t = 69$  seconds. For the Manhattan and Sørensen norm, the deviation from the reference trajectory at epoch  $t = 69$  is approximately 9 metres. Both Manhattan and Sørensen belong to the  $L_1$  family of norms, which seems to be more robust against outliers. Norms based on the  $L_2$  or Euclidean norm, such as the Euclidean itself or the Neyman, Neyman<sub>2</sub> and  $\chi^2$  norms, which belong to the squared  $L_2$  family, are more sensitive to outliers.



**Figure 8.1:** Accuracy of the unfiltered fingerprinting solution using the Euclidean, Manhattan and Sørensen norms for round 1 on 22 June 2017.



**Figure 8.2:** Accuracy of the unfiltered fingerprinting solution using the Neyman, Neyman<sub>2</sub>, and  $\chi^2$  norms for round 1 on 22 June 2017.

### 8.1.2 Probabilistic fingerprinting

For probabilistic fingerprinting, a parametric approach was used to estimate the likelihood density  $P(\mathbf{r}|\mathbf{p}_i)$  at every reference point. As the examinations of the RSS distribution carried out in Section 7.1 showed that the shape of the RSS likelihood distribution is similar to the shape of a Gaussian bell curve if the receiver is rotated while the radio map is recorded, the parameters of a normal distribution (mean  $\mu$  and variance  $\sigma^2$ ) were estimated from the measured RSS samples.

In the online phase, a WKNN maximum likelihood estimator taking the four most probable reference points was used. Both linear (see Equation 4.20) and logarithmic weights (see Equation 4.21) were tested.

**Table 8.3:** Deterministic and probabilistic fingerprinting results for the test rounds on 19 June 2017.

RMSE [m]			
Weight used	Deterministic Euclidean norm	Probabilistic linear weights	Probabilistic logarithmic weights
<i>Antenna above vehicle</i>			
Round 1 (long)	4.12	5.83	5.87
Round 2 (long)	3.74	5.42	4.31
Round 3 (long)	3.88	5.87	5.56
Round 4 (long)	4.08	6.37	5.89
Round 5 (short)	4.96	6.71	6.34
Round 6 (short)	4.14	6.21	5.66
Round 7 (short)	4.60	6.16	6.05
Round 8 (short)	4.88	6.16	5.70
<i>Antenna inside vehicle</i>			
Round 9 (long)	5.48	6.11	5.95
Round 10 (long)	5.54	5.75	5.44
Round 11 (long)	4.60	5.08	5.12
Round 12 (long)	4.64	5.30	5.24
Round 13 (long)	4.46	5.49	5.38

Table 8.3 shows the RMSE of the trajectories obtained from weighted probabilistic fingerprinting. To compare the results to deterministic fingerprinting, the RMSE of

**Table 8.4:** Deterministic and probabilistic fingerprinting results for the test rounds on 22 June 2017.

Weight used	RMSE [m]		
	Deterministic Euclidean norm	Probabilistic linear weights	Probabilistic logarithmic weights
<i>Antenna above vehicle</i>			
Round 1 (long)	4.56	6.20	6.22
Round 2 (long)	4.22	5.82	5.53
Round 3 (long)	4.60	5.83	5.51
Round 4 (long)	5.05	6.72	6.54
Round 5 (long)	4.45	6.45	6.53
<i>Antenna inside vehicle</i>			
Round 6 (long)	5.49	7.33	7.09
Round 7 (long)	4.74	6.39	6.07
Round 8 (long)	4.90	6.13	5.51
Round 9 (long)	6.03	6.18	5.87
Round 10 (long)	5.59	6.43	6.61

WKNN deterministic fingerprinting with the Euclidean norm is shown in the same table. Table 8.4 is structured in the same way and shows the RMSE for the trajectories measured on 22 June 2017. When comparing the probabilistic weights, it can be seen that the logarithmic weights yield a lower RMSE for almost all test rounds (except for rounds 1 and 11 on 19 June and rounds 1, 5 and 10 on 22 June 2017, where the linear weights yield slightly better results). When the probabilistic results are compared to their deterministic counterpart, it can be seen that deterministic fingerprinting yields a higher accuracy for almost all test rounds (except for round 10 on 19 June and round 9 on 22 June). The RMSE of the deterministic trajectories are all smaller than or equal to 6.03 m, whereas the RMSE of the probabilistic trajectories are up to 7.33 m. A study which compared the performance of deterministic and probabilistic fingerprinting as well as neural network approaches [35] also found that deterministic WKNN fingerprinting yields the best results in terms of accuracy and robustness.

## 8.2 Economic optimization

For sparse fingerprinting, either the number of AP or the number of RP can be reduced. The radio map in the Thondorf parking garage consists of 60 BLE AP and 568 RP. When the RP are thinned out, every  $n^{\text{th}}$  RP can be taken out automatically. When the number of AP is reduced, special attention has to be paid to the coverage and the geometry of the AP. All investigations carried out for the economic optimization were done with deterministic fingerprinting as it yielded more accurate and robust results (see Section 8.2).

Section 8.2.1 presents the results of the fingerprinting positioning solution with a reduced number of reference points. Furthermore it covers the achievable accuracy when simple path loss models are used to patch the sparse radio map. Section 8.2.2 shows the results for different scenarios with a reduced number of access points.

### 8.2.1 Reducing the number of reference points

When the number of reference points is reduced, the positioning accuracy inevitably decreases. To investigate how strongly the positioning accuracy deteriorates, the number of reference points in the radio map was reduced by 50% and 67%, so that one half or one third of the RP remained. Again, the RMSE was computed for each of the trajectories measured on 19 and 22 June 2017. To show the difference between the sparse and the full radio maps, the difference  $\Delta_{full}$  is introduced:

$$\Delta_{full} = RMSE_{sparse} - RMSE_{full}. \quad (8.1)$$

If  $\Delta_{full}$  is positive, the RMSE obtained with the sparse radio map is higher than the one obtained from the full radio map. If  $\Delta_{full}$  is negative, the sparse radio map yields a better solution than the full radio map.

Tables 8.5 and 8.6 show the RMSE of the trajectories computed from the sparse radio maps with 50% and 33% of the original reference points remaining. The BLE measurements were taken on 19 June 2017 for Table 8.5 and 22 June 2017 for Table 8.6, respectively. To compare the results to the full radio map, the RMSE obtained from the computations with the full radio map as well as the differences  $\Delta_{full}$  are also shown in the same tables.

When the number of fingerprints is reduced by 50%, the maximum RMSE increase is 70 cm for the trajectories obtained on 19 June and 73 cm for the trajectories obtained

on 22 June 2017. The average RMSE increase is only 10 cm for 19 June and 18 cm for 22 June, as the sparse radio map yields lower RMSE for some trajectories. When the radio map is reduced further so that only 33% of the original fingerprints remain, the maximum RMSE increase is 1.20 m for 19 June and 1.25 m for 22 June 2017, respectively. The average RMSE increases are 24 cm for the trajectories of 19 June and 15 cm for the trajectories obtained on 22 June, as again the sparse radio map yields better results for some trajectories. Even though  $\Delta_{full}$  is sometimes negative, meaning that the positioning accuracy achieved with the sparse radio map is better for the trajectory investigated, the positioning accuracy deteriorates more often than it improves when fingerprints are taken out of the radio map. It can therefore be concluded that the Thondorf radio map does not consist of too many reference points.

**Table 8.5:** RMSE obtained from the sparse radio maps (50% and 30% of the original fingerprints remaining) for the measurements on 19 June 2017.

RMSE [m]					
	Full radio map	Sparse radio map (50%)	$\Delta_{full}$	Sparse radio map (33%)	$\Delta_{full}$
<i>Antenna above vehicle</i>					
Round 1 (long)	4.12	4.24	0.12	4.54	0.42
Round 2 (long)	3.74	4.26	0.52	4.17	0.43
Round 3 (long)	3.88	3.85	-0.03	4.07	0.19
Round 4 (long)	4.08	4.78	0.70	5.25	1.17
Round 5 (short)	4.96	4.75	-0.21	4.68	-0.28
Round 6 (short)	4.14	4.20	0.06	4.12	-0.02
Round 7 (short)	4.60	4.28	-0.32	4.68	0.08
Round 8 (short)	4.88	4.90	0.02	4.72	-0.16
<i>Antenna inside vehicle</i>					
Round 9 (long)	5.48	5.76	0.28	5.17	-0.31
Round 10 (long)	5.54	5.29	-0.25	5.46	-0.08
Round 11 (long)	4.60	4.71	0.11	4.75	0.15
Round 12 (long)	4.64	4.91	0.27	4.93	0.29
Round 13 (long)	4.46	4.46	0.00	5.66	1.20

**Radio map interpolation.** When the number of reference points is reduced, the radio map can be patched or densified through interpolation with path loss models. A simple attenuation model (see Equation 5.1) was used to interpolate the sparse radio maps for

**Table 8.6:** RMSE obtained from the sparse radio maps (50% and 30% of the original fingerprints remaining) for the measurements on 22 June 2017.

RMSE [m]					
	Full radio map	Sparse radio map (50%)	$\Delta_{full}$	Sparse radio map (33%)	$\Delta_{full}$
<i>Antenna above vehicle</i>					
Round 1 (long)	4.56	4.60	0.04	4.76	0.20
Round 2 (long)	4.22	4.71	0.49	4.58	0.36
Round 3 (long)	4.60	4.95	0.35	4.79	0.19
Round 4 (long)	5.05	5.24	0.19	4.95	-0.10
Round 5 (long)	4.45	4.74	0.29	4.50	0.05
<i>Antenna inside vehicle</i>					
Round 6 (long)	5.49	5.36	-0.13	5.02	-0.47
Round 7 (long)	4.74	5.47	0.73	5.99	1.25
Round 8 (long)	4.90	5.16	0.26	5.33	0.43
Round 9 (long)	6.03	6.08	0.05	5.96	-0.07
Round 10 (long)	5.59	5.11	-0.48	5.28	-0.31

the Thondorf parking garage. Both of the sparse radio maps, with 50% and 30% of the original access points remaining, were interpolated so that the original number of reference points was restored. The attenuation exponent was determined empirically and found to be  $n = 2.75$ . The closest neighbouring original reference point was taken to compute the reference distance  $d_0$  and to obtain the power at the reference distance  $P(d_0)$  for every interpolated RP.

Table 8.7 shows the RMSE for the trajectories measured on 19 June 2017 and computed from the interpolated radio maps with 50% and 30% of the original RP remaining. Again, the RMSE for the full radio map as well as the differences between the RMSE of the interpolated maps and the RMSE of the full radio map are shown for comparison. Table 8.8 shows the same results for the trajectories measured on 22 June 2017.

When 50% of the RP are interpolated, the maximum RMSE increase of the trajectories is 54 cm for 19 June and 1.39 m for 22 June 2017. The average RMSE increases are 29 cm and 66 cm for 19 and 22 June respectively, when 50% of the original RP are kept and the rest is interpolated. When only 33% of the original RP are kept and the rest is interpolated, the maximum RMSE increase rises to 1.46 m for the trajectories obtained

on 19 June and 1.59 m for the trajectories of 22 June. The average RMSE increase when two thirds of the total reference points of the radio map are interpolated is 87 cm for both measurement days.

When the radio map is interpolated with a simple path loss model, the RMSE increases are higher than they are in the case when the RP are just taken out and not interpolated. The weighting with the sparse, truly measured RP yields better results than WKNN fingerprinting with interpolated RP. This can be attributed to the fact that the simple path loss model assumes an ideal propagation medium in which the waves attenuate equally in all directions. In reality, the BLE signals are attenuated, reflected or refracted when they hit obstacles in the parking garage. As the BLE beacons are placed on steel girders, the closest RP to the interpolation point might lie on the other side of the steel girder, where the RSS which serves as the reference power might be much stronger or much weaker than it is for the point which is interpolated.

**Table 8.7:** RMSE obtained from the interpolated radio maps (with 50% and 33% of the original reference points) for 19 June 2017.

RMSE [m]					
	Full ra- dio map	Interpolated ra- dio map (50%)	$\Delta_{full}$	Interpolated ra- dio map (33%)	$\Delta_{full}$
<i>Antenna above vehicle</i>					
Round 1 (long)	4.12	4.37	0.25	5.58	1.46
Round 2 (long)	3.74	4.06	0.32	4.90	1.16
Round 3 (long)	3.88	4.35	0.47	5.07	1.19
Round 4 (long)	4.08	4.20	0.12	5.28	1.20
Round 5 (short)	4.96	5.07	0.11	6.05	1.09
Round 6 (short)	4.14	4.55	0.41	5.17	1.03
Round 7 (short)	4.60	4.79	0.19	5.39	0.79
Round 8 (short)	4.88	5.36	0.48	5.33	0.45
<i>Antenna inside vehicle</i>					
Round 9 (long)	5.48	6.02	0.54	5.87	0.39
Round 10 (long)	5.54	5.58	0.04	5.98	0.44
Round 11 (long)	4.60	4.98	0.38	4.85	0.25
Round 12 (long)	4.64	4.85	0.21	5.43	0.79
Round 13 (long)	4.46	4.74	0.28	5.55	1.09

**Table 8.8:** RMSE obtained from the interpolated radio maps (with 50% and 33% of the original reference points) for 22 June 2017.

RMSE [m]					
	Full ra- dio map	Interpolated ra- dio map (50%)	$\Delta_{full}$	Interpolated ra- dio map (33%)	$\Delta_{full}$
<i>Antenna above vehicle</i>					
Round 1 (long)	4.56	5.16	0.60	5.57	1.01
Round 2 (long)	4.22	5.02	0.80	5.81	1.59
Round 3 (long)	4.60	5.99	1.39	5.95	1.35
Round 4 (long)	5.05	5.94	0.89	5.94	0.89
Round 5 (long)	4.45	5.54	1.09	5.62	1.17
<i>Antenna inside vehicle</i>					
Round 6 (long)	5.49	5.61	0.12	5.79	0.30
Round 7 (long)	4.74	5.33	0.59	5.80	1.06
Round 8 (long)	4.90	5.38	0.48	5.43	0.53
Round 9 (long)	6.03	6.70	0.67	6.73	0.70
Round 10 (long)	5.59	5.59	0.00	5.73	0.14

**Removing the reference points in areas where the vehicle should not drive.**

Another approach which was chosen to test sparse fingerprinting in parking garages was to keep reference points only on the roadway and not in the parking spaces. In a certain way, removing RP in areas where the vehicle is not allowed to drive is similar to adding additional information to the radio map: if only a simple nearest neighbour fingerprinting algorithm was used, the position estimate must lie on the roadway. (Note that as soon as a  $K$  nearest neighbour algorithm is used, the position estimate can also lie between two roadways.)

Tables 8.9 and 8.10 show the RMSE for the fingerprinting trajectories measured on 19 and 22 June 2017 and computed from the radio map with reference points located on the roadway only. A WKNN deterministic fingerprinting algorithm was used with the inverse Euclidean distance as weight. The RMSE obtained from the computations with the full radio map as well as the difference  $\Delta_{full}$  between the sparse radio map and the full radio map are shown for comparison. For all rounds where the antenna was mounted on the sensor platform (above the vehicle), the sparse radio map with fingerprints on the roadway always yielded better results (i.e. a lower RMSE) than the full radio map.

When the antenna was placed inside the vehicle, the positioning accuracy sometimes deteriorated when the sparse radio map was used.

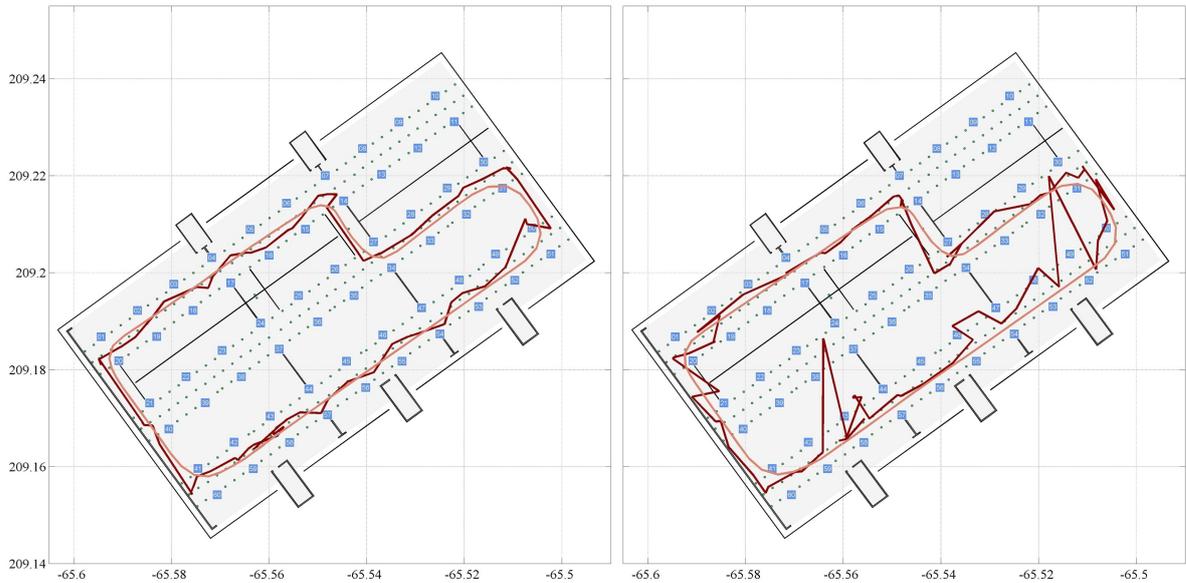
**Table 8.9:** RMSE obtained from the sparse radio map with fingerprints only in areas where the vehicle should drive. The trajectories were obtained during the final test measurements on 19 June 2017.

RMSE [m]			
	Full radio map	Sparse radio map (road only)	$\Delta_{full}$
<i>Antenna above vehicle</i>			
Round 1 (long)	4.12	3.88	-0.24
Round 2 (long)	3.74	3.28	-0.46
Round 3 (long)	3.88	3.06	-0.82
Round 4 (long)	4.08	3.94	-0.14
Round 5 (short)	4.96	3.87	-1.09
Round 6 (short)	4.14	3.63	-0.51
Round 7 (short)	4.60	4.11	-0.49
Round 8 (short)	4.88	3.80	-1.08
<i>Antenna inside vehicle</i>			
Round 9 (long)	5.48	5.38	-0.10
Round 10 (long)	5.54	5.67	0.13
Round 11 (long)	4.60	4.55	-0.05
Round 12 (long)	4.64	5.03	0.39
Round 13 (long)	4.46	4.60	0.14

**Table 8.10:** RMSE obtained from the sparse radio map with fingerprints only in areas where the vehicle should drive. The trajectories were obtained during the final test measurements on 22 June 2017.

RMSE [m]			
	Full radio map	Sparse radio map (road only)	$\Delta_{full}$
<i>Antenna above vehicle</i>			
Round 1 (long)	4.56	4.03	-0.53
Round 2 (long)	4.22	3.71	-0.51
Round 3 (long)	4.60	4.39	-0.21
Round 4 (long)	5.05	4.07	-0.98
Round 5 (long)	4.45	4.15	-0.30
<i>Antenna inside vehicle</i>			
Round 6 (long)	5.49	5.69	0.20
Round 7 (long)	4.74	4.80	0.06
Round 8 (long)	4.90	4.81	-0.09
Round 9 (long)	6.03	5.32	-0.71
Round 10 (long)	5.59	5.47	-0.12

Figure 8.3 shows two unfiltered trajectories obtained from sparse fingerprinting where the radio map contained reference points which were only located in areas where the vehicle is allowed to drive. The fingerprinting solution is shown in dark red, the reference trajectory in light red. The left side of Figure 8.3 shows the trajectory obtained from measurement round 1 on 22 June 2017, where the BLE antenna was placed on the sensor platform on the roof of the vehicle. This trajectory is very smooth and follows the reference closely. The right side of Figure 8.3 shows the trajectory of the 6th measurement round recorded on the same day, when the antenna was placed inside the vehicle. With the antenna placed inside, the fingerprinting trajectory is noisier and shows two large outliers. The outliers occur when the position is estimated to be on a roadway where the vehicle is allowed to drive, but it is not the roadway where the vehicle is truly located. Outliers of this magnitude ( $>20$  m) only occurred when sparse radio maps were used.



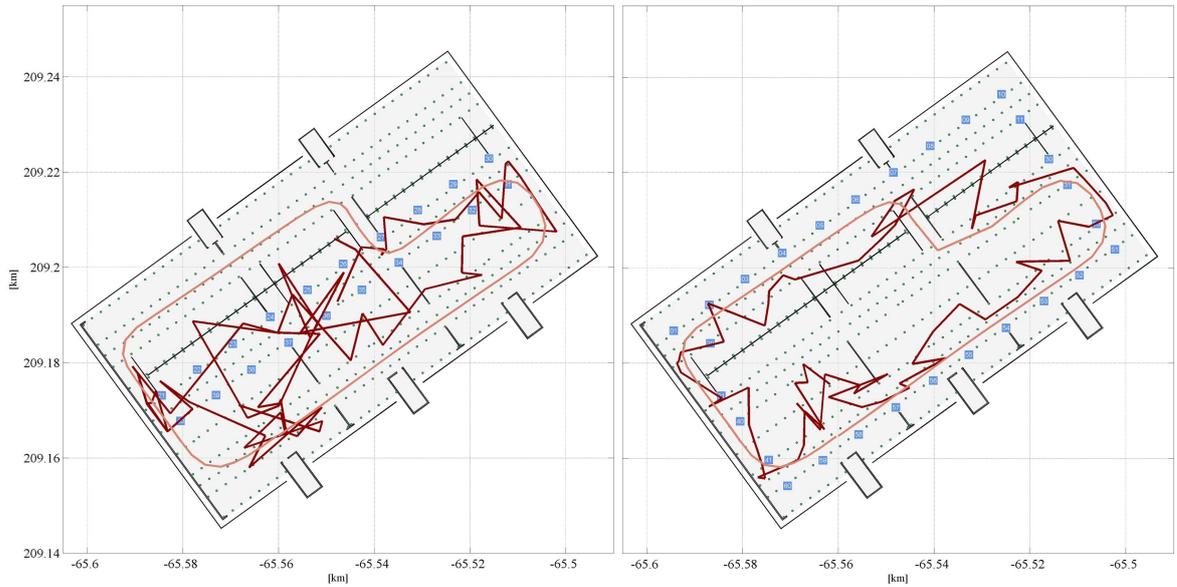
**Figure 8.3:** Trajectories obtained from sparse fingerprinting with reference points only in areas where the vehicle is allowed to drive. The fingerprinting trajectory is shown in dark red, along with the reference trajectory in light red. While recording the BLE measurements on 22 June 2017, the antenna was first placed on top of the vehicle (left plot, round 1) and then inside the vehicle (right plot, round 6).

### 8.2.2 Reducing the number of access points

The number of AP was first reduced automatically by taking out every 4<sup>th</sup>, 5<sup>th</sup> and 6<sup>th</sup> AP from the radio map. When every 6<sup>th</sup> AP was taken out, the RMSE of the trajectories deteriorated by about 50 cm, when every 4<sup>th</sup> and 5<sup>th</sup> AP were taken out, the RMSE deteriorated by approximately 1 m. However, as soon as larger numbers of AP are taken out, the RMSE is not representative any more, as large outliers occur close to the regions where the AP were taken out.

Different scenarios were tested with AP located only in the corners, only in the centre or only in the corners and around the ramps of the parking garage. Huge outliers of up to 60 m occurred and, especially for test rounds where the BLE antenna was located inside the vehicle, no suitable positioning estimates were obtained.

The left side of Figure 8.4 shows an example of a fingerprinting trajectory which was obtained when the antenna was located inside the vehicle and only AP in the middle row



**Figure 8.4:** Trajectories obtained from sparse fingerprinting when only AP in the middle row (left side) or only AP in the outer row (right side) of the parking garage were kept. The result shown was computed for test round 9 on 22 June 2017, where the antenna was located inside the vehicle. The fingerprinting trajectories are shown in dark red, the references in light red.

of the parking garage were used. The estimated positioning solutions are far away from the actual trajectory and cannot be used in a positioning system for a parking garage.

The right side of Figure 8.4 shows the same trajectory obtained when AP are only deployed close to the surrounding edges of the parking garage. The computed positions are closer to the reference trajectory than the ones obtained from the radio map if the only AP present are in the middle row. This solution might be better because the position estimate tends to be close to the position of the AP, and the AP in the outer circle (right side of Figure 8.4) are closer to the reference trajectory than the AP in the inner circle (left side of Figure 8.4).

When a combination of AP in the outer circle and one row of AP in the middle of the parking garage are used, the RMSE of the trajectories only increases by approximately 1.5 m in comparison to the trajectory of the full radio map. However, large outliers with a magnitude of more than 20 m offset from the reference still occur, more frequently when the antenna is placed inside the vehicle than when it is placed outside.

In conclusion it can be said that the number of AP should not be reduced, not even if the estimated positions from fingerprinting are only used as initial values for a filter

which then only continues with relative positioning updates through a DR algorithm. As soon as the number of AP is reduced, outliers of the positioning estimate occur, with a magnitude of over 20 m away from the true position. When a filter relying only on DR uses a starting position 20 m away from the true starting position, the trajectory computed will have this offset from the true trajectory.

### 8.3 Integration of vehicle sensor data

The real-time integration of BLE fingerprinting and vehicle sensor data was tested in the course of the final test measurements on 19 and 22 June 2017. Statements about the achievable accuracy can only be made when the integrated trajectories are compared to the reference trajectory in post-processing, which will be described in this section. First, the settings for the Kalman filter will be discussed, then the final results will be presented.

**Filter settings.** For the fingerprinting solutions, a deterministic WKNN algorithm with the four most similar RP was used. The weights were computed from the inverse Manhattan norm, as  $L_1$  norms proved to be more stable than  $L_2$  norms (see Section 8.1.1).

The initial position was calculated from the RSS of the first fingerprinting epoch. The start heading was entered manually. A new fingerprinting update was computed every second, while the updates with vehicle sensor data were carried out at 10 Hz.

The initial covariance matrix was modelled as follows:

$$\mathbf{P}_0 = \begin{pmatrix} \sigma_x^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\dot{x}}^2 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\dot{y}}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_h^2 \end{pmatrix} = \begin{pmatrix} 1^2 & 0 & 0 & 0 & 0 \\ 0 & 1^2 & 0 & 0 & 0 \\ 0 & 0 & 1^2 & 0 & 0 \\ 0 & 0 & 0 & 1^2 & 0 \\ 0 & 0 & 0 & 0 & \left(\frac{10\pi}{180}\right)^2 \end{pmatrix}, \quad (8.2)$$

where  $(\sigma_x, \sigma_y)$  are given in metres,  $(\sigma_{\dot{x}}, \sigma_{\dot{y}})$  in  $m/s$  and  $\sigma_h$  in radians. To model the system noise, the covariance of the parameters neglected by the motion model is needed.

It was designed as:

$$\mathbf{R}_n = \begin{pmatrix} \sigma_{\dot{x}}^2 & 0 & 0 \\ 0 & \sigma_{\dot{y}}^2 & 0 \\ 0 & 0 & \sigma_h^2 \end{pmatrix} = \begin{pmatrix} 2^2 & 0 & 0 \\ 0 & 2^2 & 0 \\ 0 & 0 & \left(\frac{90\pi}{180}\right)^2 \end{pmatrix}, \quad (8.3)$$

with  $(\sigma_{\dot{x}}, \sigma_{\dot{y}})$  in  $m/s^2$  and  $\sigma_h$  in radians.

For the standard deviations of the observations, the standard deviations of the velocities (both forward and right) obtained from vehicle sensor data were set to be  $\sigma_v = 0.1m/s$ . The standard deviation of the heading change was set to  $\sigma_h = 0.1^\circ$ . The accuracy of the fingerprinting updates was modelled with  $\sigma_{x,y} = 0.3m$ , which is a much lower standard deviation than the actual standard deviation of fingerprinting. This high weight was given to the fingerprinting observations due to the fact that if the initial position or heading are incorrect, the integrated trajectory will not be corrected and drawn closer to the true trajectory when the relative observations have a much lower standard deviation and therefore a much higher weight.

**Final results.** Table 8.11 and Table 8.12 show the RMSE of the integrated, filtered trajectories in comparison to the deterministic, fingerprinting-only trajectories for the test rounds recorded on 19 and 22 June 2017, respectively. It can be seen that when vehicle sensor data are added as additional information, the RMSE are lowered for all trajectories. The improvement, which is calculated as

$$Improvement = RMSE_{fingerprinting} - RMSE_{integrated\ solution}, \quad (8.4)$$

is also shown in the same tables. The average improvement for all test rounds recorded on both days is 2.24 m.

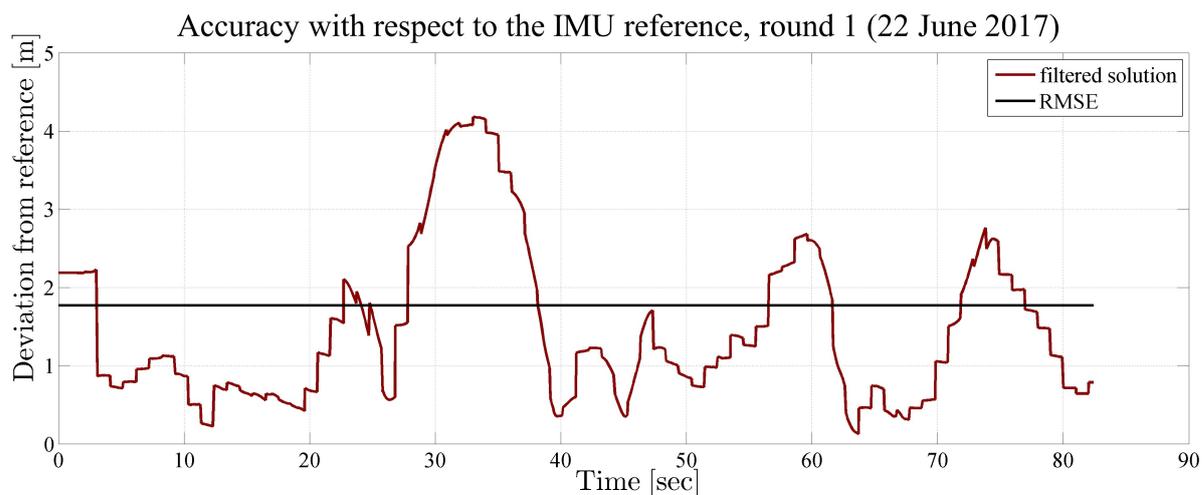
When the antenna is placed above the vehicle, slightly better results can be achieved than when it is placed inside the vehicle. The maximum RMSE when the antenna is placed outside is 2.82 m for the tests on 19 June and 2.76 m for the test rounds recorded on 22 June 2017. When the antenna is placed inside the vehicle, the maximum RMSE for the test rounds on 19 June is 3.49 m and 2.91 m for 22 June 2017.

**Table 8.11:** RMSE of the integrated trajectories in comparison to the fingerprinting-only trajectories for the test rounds recorded on 19 June 2017.

RMSE [m]			
	Deterministic Manhattan norm	Integrated solution with vehicle sensor data	Improve- ment
<i>Antenna above vehicle</i>			
Round 1 (long)	4.03	2.11	1.92
Round 2 (long)	3.66	1.71	1.95
Round 3 (long)	3.92	2.13	1.79
Round 4 (long)	4.48	2.82	1.66
Round 5 (short)	4.70	2.66	2.04
Round 6 (short)	3.99	2.06	1.93
Round 7 (short)	4.82	2.64	2.18
Round 8 (short)	4.68	2.36	2.32
<i>Antenna inside vehicle</i>			
Round 9 (long)	5.65	3.49	2.16
Round 10 (long)	5.37	2.48	2.89
Round 11 (long)	4.57	2.80	1.77
Round 12 (long)	4.49	2.32	2.17
Round 13 (long)	4.77	3.42	1.35

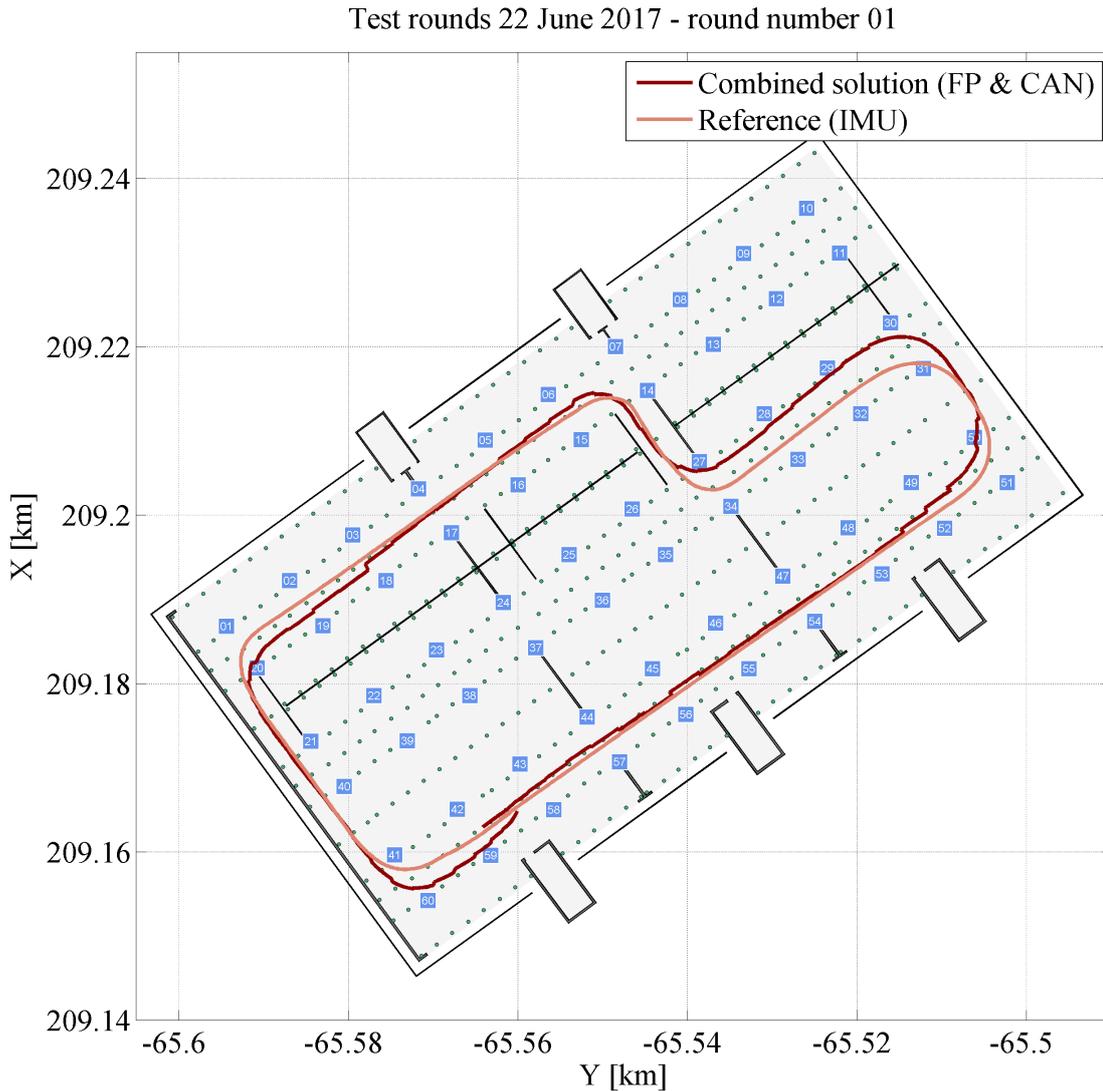
**Table 8.12:** RMSE of the integrated trajectories in comparison to the fingerprinting-only trajectories for the test rounds recorded on 22 June 2017.

RMSE [m]			
	Deterministic Manhattan norm	Integrated solution with vehicle sensor data	Improve- ment
<i>Antenna above vehicle</i>			
Round 1 (long)	4.60	1.77	2.83
Round 2 (long)	4.37	2.05	2.32
Round 3 (long)	4.54	2.70	1.84
Round 4 (long)	4.86	2.76	2.10
Round 5 (long)	4.40	2.36	2.04
<i>Antenna inside vehicle</i>			
Round 6 (long)	5.08	2.45	2.63
Round 7 (long)	4.92	2.57	2.35
Round 8 (long)	4.80	2.05	2.75
Round 9 (long)	6.05	2.35	3.70
Round 10 (long)	5.68	2.91	2.77



**Figure 8.5:** Accuracy of the integrated solution with respect to the IMU reference for test round 1, recorded on 22 June 2017.

Figure 8.5 shows the deviations of the integrated, filtered solution from the reference trajectory as a time series. It can be seen that the maximum deviation from the true position is slightly higher than 4 m (at epoch  $t = 33$  seconds), which corresponds to the accuracy of fingerprinting. The absolute accuracy of a system integrating absolute and relative positioning data cannot be higher than the accuracy of the absolute positioning system, however the relative accuracy can be improved. When investigating the time series more closely, small jumps or steps can be seen every second. These jumps result from the absolute fingerprinting updates.

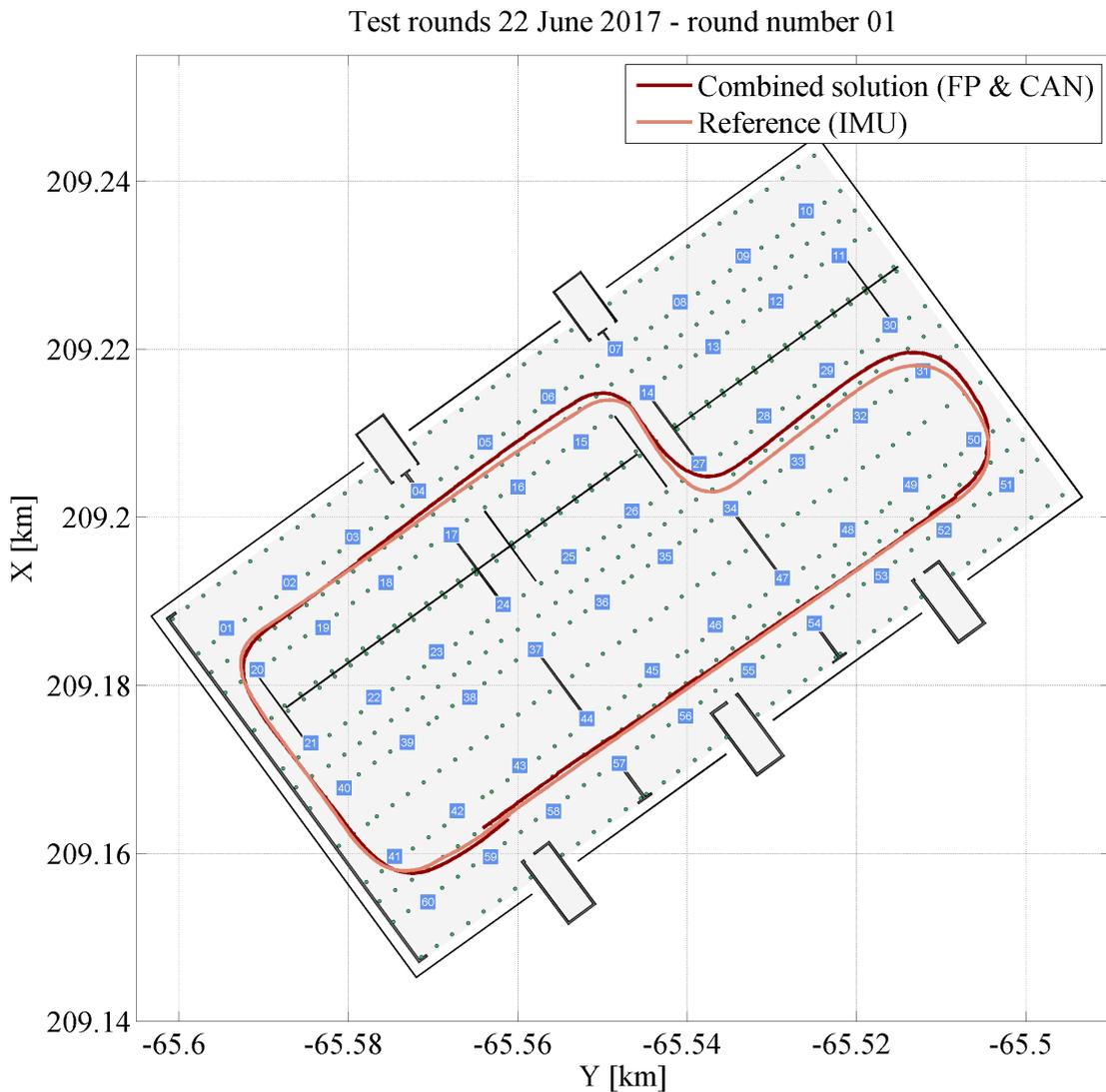


**Figure 8.6:** Integrated solution (dark red) in comparison to the IMU reference (light red) for test round 1, recorded on 22 June 2017. The fingerprinting accuracy was modelled as  $\sigma_{x,y} = 0.3$  m.

Figure 8.6 shows the combined, integrated solution of fingerprinting and vehicle sensor data for test round 1 recorded on 22 June 2017 in dark red. The IMU reference is shown in light red. It can be seen that the integrated trajectory closely follows the reference trajectory. Only in the north-eastern part does it slightly deviate from the reference. This deviation can also be seen in the time series in Figure 8.5, between seconds 30 and 40. Furthermore, small jumps can be seen in the integrated solution, which occur whenever a fingerprinting update is processed. These jumps can be reduced when a lower weight, i.e.

a higher standard deviation, is given to the fingerprinting observations.

Figure 8.7 shows the integrated trajectory for test round 1 on 22 June 2017 when a standard deviation of  $\sigma_{x,y} = 3$  m was given to the fingerprinting updates. When comparing it to Figure 8.6, which shows the exact same trajectory but with a standard deviation of  $\sigma_{x,y} = 0.3$  m for the coordinates obtained through fingerprinting, it can be seen that the trajectory with a fingerprinting standard deviation of 3 m is smoother and follows the reference trajectory more closely.



**Figure 8.7:** Integrated solution (dark red) in comparison to the IMU reference (light red) for test round 1, recorded on 22 June 2017. The fingerprinting accuracy was modelled as  $\sigma_{x,y} = 3$  m.

However, with a high standard deviation, the fingerprinting updates have hardly any influence on the course of the trajectory. In the case of Figure 8.7, the integrated trajectory is close to the true trajectory because the initial position (obtained from the first fingerprinting epoch) is close to the true position, a correct initial heading was used and the vehicle sensor data do not drift. If the initial position were 5 m away from the true position, or if the start heading was chosen wrongly, it would take a long time before the integrated trajectory came closer to the true trajectory, when only a low weight is given to the fingerprinting updates. Moreover, if the vehicle sensor data start to drift, this drift can be reduced better if a stronger weight is given to the absolute fingerprinting updates. To conclude, a standard deviation of  $\sigma_{x,y} = 0.3$  m was chosen as an ideal weight for the filter, even if, visually, higher standard deviations might provide a more appealing result in some cases. The filter is more stable when a higher weight is given to fingerprinting.

## 8.4 Conclusion and outlook

To summarize the results of this thesis, it can be said that deterministic WKNN fingerprinting yields more accurate results than probabilistic fingerprinting with uniform prior probabilities. Using different norms as a measure of RSS similarity and as weights in WKNN fingerprinting has hardly any impact on the achievable accuracy of the result. However, norms belonging to the  $L_1$  family proved to be more stable, i.e. more robust against outliers, than distance metrics derived from the  $L_2$  norm.

As far as sparse fingerprinting is concerned, the accuracy decreases when the number of fingerprints is reduced. When the radio map is interpolated with a simple path loss model, the accuracy of WKNN fingerprinting deteriorates even further, as this model does not describe the indoor propagation characteristics of radio waves accurately. The number of AP in the Thondorf parking garage should not be reduced, as this would severely degrade the achievable accuracy of the absolute positioning result. When fingerprints in areas where the vehicle is not allowed to drive are taken out of the radio map, the achievable accuracy of the positioning result can even be improved, but the BLE antenna should be placed outside the vehicle in the online positioning phase.

When it comes to the real-time integration of BLE fingerprinting and vehicle sensor data, the filter algorithm developed has proved to work. When fingerprinting is supported by adding vehicle sensor data, the positioning solution can be improved. As the initial filter

position is computed from the first fingerprinting epoch and the initial heading might be erroneous, a high weight should be given to the fingerprinting updates so that the integrated trajectory will approach the true trajectory faster in case of erroneous initial values.

As an outlook for further research, the filter developed in this study could be combined with GNSS measurements and also used for outdoor applications. In GNSS-denied areas such as tunnels, a continuous positioning solution for vehicles would be possible through the use of vehicle sensor data. As the filter is decentralized, BLE fingerprinting and GNSS positioning can easily be interchanged, so that the continuous position determination of the vehicle should be possible when it leaves the road and enters a parking garage.

For further improvements of the indoor positioning solution, an approach to determining the initial heading has to be found. Moreover the combination of probabilistic fingerprinting and particle filtering could be investigated, as particle filtering computes prior probabilities, which can be used in probabilistic fingerprinting.

Lastly, the radio maps in parking garages could be updated automatically. When vehicles are static, i.e. parked, and the position of the parking space is known, BLE data could be logged and used to update the radio map. Furthermore the radio map could be updated in combination with DR from vehicle sensor data, as described in Section 5.2.

# Bibliography

- [1] Apple Inc. (2014). Getting Started with iBeacon: Version 1.0.
- [2] Bae, H., Oh, J., Lee, K., and Oh, J.-H. (2016). Low-cost indoor positioning system using BLE (bluetooth low energy) based sensor fusion with constrained extended kalman filter. In *IEEE-ROBIO 2016*, pages 939–945. IEEE, Piscataway, NJ.
- [3] Bahl, P. and Padmanabhan, V. N. (2000). RADAR: An In-Building RF-based User Location and Tracking System. In IEEE, editor, *IEEE Infocom 2000*, pages 775–784. IEEE, Piscataway.
- [4] Bluetooth Special Interest Group (2010). Specification of the Bluetooth System, Version 4.0.
- [5] Cha, S.-H. (2007). Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions. In Srivastava, H. M., editor, *International Journal of Mathematical Models and Methods in Applied Sciences*, volume 4, pages 300–307.
- [6] Chai, X. and Yang, Q. (2007). Reducing the Calibration Effort for Probabilistic Indoor Location Estimation. In Institute of Electrical and Electronics Engineers and IEEE Computer Society, editors, *IEEE transactions on mobile computing*, volume 6, no. 6, pages 649–662. IEEE Computer Society, Los Alamitos CA.
- [7] Chen, J., Gary Chan, S.-H., He, J., and Soung-Chang, L. (2003). Mixed-Mode WLAN: The Integration of Ad Hoc Mode with Wireless LAN Infrastructure. In Chandran, C. Y. and Hoit, M. I., editors, *GLOBECOM '03*, pages 231–235. IEEE, Piscataway, NJ.
- [8] Chintalapudi, K., Padmanabha Iyer, A., and Padmanabhan, V. N. (2010). Indoor localization without the pain. In Vaidya, N., editor, *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, pages 173–184. ACM, Chicago, Illinois, USA.

- [9] Deasy, T. P. and Scanlon, W. G. (2007). Simulation or Measurement: The Effect of Radio Map Creation on Indoor WLAN-Based Localisation Accuracy. In *Wireless personal communications*, volume 42, pages 563–573. Springer, Dordrecht.
- [10] Del Corte-Valiente, A., Gómez-Pulido, J. M., and Gutiérrez-Blanco, O. (2009). Efficient Techniques and Algorithms for Improving Indoor Localization Precision on WLAN Networks Applications. In *International Journal of Communications, Network and System Sciences International Journal of Communications, Network and System Sciences*, volume 7, pages 645–651. Scientific Research.
- [11] Dorn, M. (2014). *Sensorintegration zum Zweck der RPAS-Navigation*. Master’s thesis, Graz University of Technology, Graz.
- [12] Dutzler, R., Ebner, M., and Brandner, R. (2013). Indoor Navigation by WLAN Location Fingerprinting: Reducing Trainings-Efforts with Interpolated Radio Maps. In Narzt, W., editor, *UBICOMM 2013*. INRIA.
- [13] Einicke, G. A. and White, L. B. (1999). Robust extended Kalman filtering. In Engineers, I. o. E., Electronics, IEEE Acoustics, S., Group., S. P., IEEE Acoustics, S., Society., S. P., and Society, I. S. P., editors, *IEEE transactions on signal processing*, volume 47 (9) of 47, pages 2596–2599. Institute of Electrical and Electronics Engineers, New York.
- [14] Faragher, R. and Harle, R. (2014). An Analysis of the Accuracy of Bluetooth Low Energy for Indoor Positioning Applications. In *Proceedings of the 27th International Technical Meeting of the Satallite Division of the Institute of Navigation (ION GNSS+2014)*, pages 201–210. Tampa, Florida.
- [15] Faragher, R. and Harle, R. (2015). Location Fingerprinting With Bluetooth Low Energy Beacons. In *IEEE Journal on Selected Areas In Communication, Vol. 33, No. 11*.
- [16] Feng, Y., Minghua, J., Jing, L., Xiao, Q., Ming, H., Tao, P., and Xinrong, H. (2014). An Improved Indoor Localization of WiFi Based on Support Vector Machines. In *International journal of future generation communication and networking*, volume 7, no.5, pages 191–206. SERSC, South Korea.
- [17] Fielitz, H. and ter Haseborg, J. L. (2014). On the EMC of the propagation of 2.4 GHz ISM signals in environments with complex metallic structures – a cost-

- effective measurement setup . In *2014 International Symposium on Electromagnetic Compatibility (EMC Europe 2014)*, pages 1043–1048. IEEE, Piscataway, NJ.
- [18] Grewal, M. S., Weill, L. R., and Andrews, A. P. (2007). *Global positioning systems, inertial navigation and integration*. Wiley, Hoboken, NJ, [elektronische ressource], 2. ed. edition.
- [19] Heydon, R. (2013). *Bluetooth Low Energy: The Developer’s Handbook*. Prentice Hall, Upper Saddle River, N.J.
- [20] Hofmann-Wellenhof, B., Legat, K., and Wieser, M. (2003). *Navigation: Principles of Positioning and Guidance*. Springer Vienna, Vienna.
- [21] Honkavirta, V., Perälä, T., Ali-Löytty, S., and Piché, R. (2009). A Comparative Survey of WLAN Location Fingerprinting Methods. In *6th Workshop on Positioning, Navigation and Communication, 2009*, volume 6 of *Hannoversche Beiträge zur Nachrichtentechnik 0*, pages 243–251. IEEE, Piscataway, NJ.
- [22] Hotta, S., Hada, Y., and Yaginuma, Y. (2012). A robust room-level localization method based on transition probability for indoor environments. In Rizos, C., Dempster, A. G., Li, B., and Gallagher, T., editors, *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8. Sydney.
- [23] Ji, M., Kim, J., Jeon, J., and Cho, Y. (2015). Analysis of positioning accuracy corresponding to the number of BLE beacons in indoor positioning system. In *2015 17th International Conference on Advanced Communication Technology (ICACT)*, pages 92–95. IEEE, Piscataway, NJ.
- [24] Ji, Y., Biaz, S., Pandey, S., and Agrawal, P. (2006). ARIADNE: A Dynamic Indoor Signal Map Construction and Localization System. In *Proceedings of the 4th international conference on Mobile systems, applications and services*, pages 151–162. ACM, New York, NY.
- [25] Kaemarungsi, K. and Krishnamurthy, P. (2004). Properties of Indoor Received Signal Strength for WLAN Location Fingerprinting. In Chlamtac, I., editor, *Proceedings of MobiQuitous 2004*. IEEE Service Center, Piscataway, NJ.
- [26] Khalajmehrabadi, A., Gatsis, N., and Akopian, D. (2017). Structured Group Sparsity: A Novel Indoor WLAN Localization, Outlier Detection and Radio Map Interpolation Scheme. In Vehicular Technology Society, Institute of Electrical and Electronics Engineers, and IEEE Vehicular Technology Group, editors, *IEEE transactions*

- on vehicular technology*, volume 66, No. 7, pages 6498–6510. Institute of Electrical and Electronics Engineers Inc, New York.
- [27] Kim, J. and Lee, I. (2015). 802.11 WLAN: history and new enabling MIMO techniques for next generation standards. *IEEE Communications Magazine*, 2015(53).
- [28] Koch, K.-R. (2000). *Einführung in die Bayes-Statistik*. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [29] Küpper, A. (2005). *Location-Based Services: Fundamentals and Operation*. John Wiley and Sons, Chichester.
- [30] Kushki, A., Plataniotis, K. N., and Venetsanopoulos, A. N. (2007). Kernel-based positioning in wireless local area networks. In Institute of Electrical and Electronics Engineers and IEEE Computer Society, editors, *IEEE transactions on mobile computing*, volume 6, pages 689–705. IEEE Computer Society, Los Alamitos CA.
- [31] Kushki, A., Venetsanopoulos, A. N., and Plataniotis, K. N. (2012). *WLAN positioning systems: Principles and applications in location-based services*. Cambridge University Press, Cambridge.
- [32] Laoudias, C., Eliades, D. G., Kemppe, P., Panayiotou, C. G., and Polycarpou, M. M. (2009). Indoor localization using neural networks with location fingerprints. In Alippi, C., Polycarpou, M., Panayiotou, C., and Ellinas, G., editors, *Artificial neural networks - ICANN 2009*, volume 5769 of *Lecture notes in computer science*, pages 954–963. Springer, Berlin.
- [33] Li, H., Wang, Y.-h., Sun, Q.-m., and Liu, J.-n. (2015). Indoor localization for sparse wireless networks with heterogeneous information. In *2015 International Conference on Information Networking (ICOIN)*, pages 200–204. IEEE, Piscataway, NJ.
- [34] Li, R., Liu, C., and Luo, F. (2008). A design for automotive CAN bus monitoring system. In *IEEE Vehicle Power and Propulsion Conference, 2008*. IEEE, Piscataway, NJ.
- [35] Lin, T.-N. and Lin, P.-C. (2005). Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks. In *International Conference on Wireless Networks, Communications and Mobile Computing, 2005*, pages 1569–1574. IEEE Operations Center, Piscataway, NJ.

- [36] Lohan, E., Talvitie, J., Silva, P., Nurminen, H., Ali-Löytty, S., and Piché, R. (2015). Received Signal Strength models for WLAN and BLE-based indoor positioning in multi-floor buildings. *International Conference on Localization and GNSS (ICL-GNSS)*, 2015.
- [37] Lott, M. and Forkel, I. (2001). A multi-wall-and-floor model for indoor radio propagation. In *IEEE VTS 53rd Vehicular Technology Conference, spring, 2001*, pages 464–468. IEEE, Piscataway, NJ.
- [38] Ma, R., Guo, Q., Hu, C., and Xue, J. (2015). An Improved WiFi Indoor Positioning Algorithm by Weighted Fusion. In Khoshelham, K. and Zlatanova, S., editors, *Sensors for Indoor Mapping and Navigation*, volume 9 of 15, pages 21824–21843. Molecular Diversity Preservation International (MDPI), Basel Switzerland.
- [39] Majeed, K., Sorour, S., Al-Naffouri, T. Y., and Valaee, S. (2015). Indoor Localization and Radio Map Estimation Using Unsupervised Manifold Alignment with Geometry Perturbation. In Institute of Electrical and Electronics Engineers and IEEE Computer Society, editors, *IEEE transactions on mobile computing*, volume 15, no. 11, pages 2794–2808. IEEE Computer Society, Los Alamitos CA.
- [40] McCann, S. and Ashley, A. (17.07.2017). Official IEEE 802.11 Working Group Project Timelines.
- [41] Mirowski, P., Milioris, D., Whiting, P., and Ho, T. K. (2014). Probabilistic radio-frequency fingerprinting and localization on the run. *Bell Labs Technical Journal*, 2014(18(4)):111–133.
- [42] Negus, K. and Petrick, A. (2008). History of Wireless Local Area Networks (WLANs) in the Unlicensed Bands. *George Mason University Law School Conference, Information Economy Project*, 2008.
- [43] Niedermayr, S. (2015). *Positionsbestimmung durch Kombination geometrie- und merkmalsbasierter Verfahren unter Einbeziehung der Qualität*. Dissertation, Vienna University of Technology, Vienna.
- [44] Presi, T. P. (2013). Design and development of PIC microcontroller based vehicle monitoring system using controller area network (CAN) protocol. In *International Conference on Information Communication and Embedded Systems (ICICES), 2013*, pages 1–7. IEEE, Piscataway, NJ.

- [45] Rayner, J. C. W., Best, D. J., and Thas, O. (2009). *Smooth tests of goodness of fit using R*. Wiley series in probability and statistics. John Wiley & Sons (Asia), Singapore and Hoboken, NJ, 2nd ed. edition.
- [46] Roos, T., Myllymäki, P., Tirri, H., Misikangas, P., and Sievänen, J. (2002). A probabilistic approach to WLAN user location estimation. *International Journal of Wireless Information Networks*, (9):155–164.
- [47] Sachs, L. (2004). *Angewandte Statistik: Anwendung statistischer Methoden*. Springer, Berlin and Heidelberg, 11 edition.
- [48] Schnabel, P. (2004). *Netzwerktechnik-Fibel: Grundlagen Netzwerktechnik; Übertragungstechnik; TCP/IP; Anwendungen und Dienste; Netzwerk-Sicherheit*. Schnabel, Ludwigsburg, 1 edition.
- [49] Scott, D. W. (1992). *Multivariate density estimation: Theory, practice, and visualization*. A Wiley-Interscience publication. Wiley, New York, NY.
- [50] Sorour, S., Lostanlen, Y., Valaee, S., and Majeed, K. (2015). Joint Indoor Localization and Radio Map Construction with Limited Deployment Load. In Institute of Electrical and Electronics Engineers and IEEE Computer Society, editors, *IEEE transactions on mobile computing*, volume 14, no. 5, pages 1031–1043. IEEE Computer Society, Los Alamitos CA.
- [51] Taheri, A., Singh, A., and Agu, E. (2004). Location Fingerprinting on Infrastructure 802.11 Wireless Local Area Networks (WLANs) using Locus. *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks (LCN'04)*, 2004.
- [52] Torres-Sospedra, J., Montoliu, R., Trilles, S., Belmonte, Ó., and Huerta, J. (2015). Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems. In Lin, B., editor, *Expert systems with applications*, volume 42, pages 9263–9278. Pergamon.
- [53] Welch, G. F. (2014). Kalman filter. In Ikeuchi, K., editor, *Computer vision*, Springer reference, pages 435–437. Springer, New York, NY.
- [54] Wilfinger, R. (2015). *Absolute positioning of vehicles in indoor environments using Wireless LAN and Bluetooth LE*. Master’s thesis, Graz University of Technology, Graz.
- [55] Wilfinger, R., Moder, T., and Wieser, M. (2016a). *Indoor Position Determination Using Location Fingerprinting and Vehicle Sensor Data: NETI Final Report*.

- [56] Wilfinger, R., Moder, T., Wieser, M., and Grosswindhager, B. (2016b). Indoor Position Determination Using Location Fingerprinting and Vehicle Sensor Data. In *2016 European Navigation Conference (ENC)*. IEEE, Piscataway, NJ.
- [57] Wilk, P., Karciarz, J., and Swiatek, J. (2015). Indoor radio map maintenance by automatic annotation of crowdsourced Wi-Fi fingerprints. In *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, Piscataway, NJ.
- [58] Wu, C., Yang, Z., Xiao, C., Yang, C., Liu, Y., and Liu, M. (2015). Static power of mobile devices: self-updating radio maps for wireless indoor localization. In *IEEE Conference on Computer Communications (INFOCOM), 2015*, pages 2497–2505. IEEE, Piscataway, NJ and Piscataway, NJ.
- [59] Yin, J., Yang, Q., and Ni, L. (2005). Adaptive temporal radio maps for indoor location estimation. In *Third IEEE International Conference on Pervasive Computing and Communications, 2005*. IEEE Computer Society, Los Alamitos, Calif.
- [60] Youssef, M. and Agrawala, A. (2005). The Horus WLAN location determination system. In *MobiSys '05 Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 205–218. ACM, NY.
- [61] Youssef, M., Agrawala, A., and Shankar, A. U. (2003). WLAN Location Determination via Clustering and Probability Distributions. In *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications*. IEEE, Los Alamitos, Calif.
- [62] Zhao, X., Xiao, Z., Markham, A., Trigoni N., and Ren, Y. (2014). Does BTLE measure up against WiFi? A comparison of indoor location performance. *Proceedings of 20th European Wireless Conference*, 2014:263–268.

# List of Figures

2.1	System architecture of WLAN in infrastructure mode. . . . .	6
2.2	BLE spectrum and three WLAN channels in the 2.4 GHz band . . . . .	8
2.3	BLE beacon by <i>Accent Systems</i> . . . . .	10
3.1	Schematic view of a deterministic radio map. At every reference point (dark red), the RSS received from the surrounding AP are stored as a single deterministic value. . . . .	14
3.2	Schematic view of a probabilistic radio map. At every reference point (dark red), a probability distribution which describes the likelihood that a certain RSS value will be received from a specific AP, is stored. . . . .	18
3.3	Histogram obtained from 2721 RSS measurements received from a BLE access point. . . . .	18
4.1	Influence of bandwidth $\sigma$ on the kernel density estimate. The training points lie at -65 dBm, -70 dBm and -73 dBm, a Gaussian kernel was used to obtain the density estimate. . . . .	28
4.2	Natural logarithm. Domain $\{x x \in \mathbb{R}_{>0}\}$ , range $\{y y \in \mathbb{R}\}$ . . . . .	29
5.1	If the distances between three receivers and three AP are known, the relative geometry of the constellation is rigid. . . . .	36
6.1	Schematic process of Kalman filtering based on Dorn (2014). . . . .	40
6.2	Uncoupled integration of GNSS and IMU. . . . .	44
6.3	Loosely coupled integration of GNSS and IMU. . . . .	45
6.4	Tightly coupled integration of GNSS and IMU. . . . .	45
6.5	Body system of a vehicle . . . . .	47
6.6	Decentralized filter architecture combining BLE fingerprinting and DR from vehicle sensor data. . . . .	48
6.7	Pre-processing stage of a filter combining RSS and vehicle sensor data. . . . .	49
6.8	Stages of the master filter. . . . .	52

LIST OF FIGURES

---

7.1	Measurement setup for testing the RSS distributions. . . . .	59
7.2	Histogram and QQ-plot for the static measurement setup for beacon 39. . . . .	60
7.3	Histogram and QQ-plot for the static measurement setup for beacon 40. . . . .	60
7.4	Histogram and QQ-plot for the measurement setup with a rotating receiver for beacon 38. . . . .	61
7.5	Histogram and QQ-plot for the measurement setup with a rotating receiver for beacon 40. . . . .	61
7.6	Lecture room <i>AE01</i> at Graz University of Technology. . . . .	62
7.7	Overview of reference points, access points and measurement points in lecture room <i>AE01</i> . . . . .	63
7.8	Comparison of deterministic and probabilistic fingerprinting for measurement point 2009. . . . .	65
7.9	Test vehicle with the sensor platform mounted on the roof. . . . .	66
7.10	An IMU and tachymeter measurements are needed to obtain the reference trajectory. . . . .	67
7.11	The BLE antenna was first mounted on the platform (left) and then placed inside the vehicle (right). . . . .	68
7.12	To avoid obstructing the signal path, the observers knelt down while the RSS were being recorded. . . . .	70
7.13	Comparison of the RSS received from AP 36, for the radio maps recorded in 2015 (left) and 2017 (right). . . . .	71
7.14	Comparison of the RSS received from AP 44, for the radio maps recorded in 2015 (left) and 2017 (right). . . . .	71
7.15	Two types of test rounds were used: a longer one (left) and a shorter one (right). . . . .	73
8.1	Accuracy of the unfiltered fingerprinting solution using the Euclidean, Manhattan and Sørensen norms for round 1 on 22 June 2017. . . . .	78
8.2	Accuracy of the unfiltered fingerprinting solution using the Neyman, Neyman <sub>2</sub> , and $\chi^2$ norms for round 1 on 22 June 2017. . . . .	79
8.3	Trajectories obtained from sparse fingerprinting with reference points only in areas where the vehicle is allowed to drive. The fingerprinting trajectory is shown in dark red, along with the reference trajectory in light red. While recording the BLE measurements on 22 June 2017, the antenna was first placed on top of the vehicle (left plot, round 1) and then inside the vehicle (right plot, round 6). . . . .	89

LIST OF FIGURES

---

8.4	Trajectories obtained from sparse fingerprinting when only AP in the middle row (left side) or only AP in the outer row (right side) of the parking garage were kept. The result shown was computed for test round 9 on 22 June 2017, where the antenna was located inside the vehicle. The fingerprinting trajectories are shown in dark red, the references in light red.	90
8.5	Accuracy of the integrated solution with respect to the IMU reference for test round 1, recorded on 22 June 2017. . . . .	94
8.6	Integrated solution (dark red) in comparison to the IMU reference (light red) for test round 1, recorded on 22 June 2017. The fingerprinting accuracy was modelled as $\sigma_{x,y} = 0.3$ m. . . . .	95
8.7	Integrated solution (dark red) in comparison to the IMU reference (light red) for test round 1, recorded on 22 June 2017. The fingerprinting accuracy was modelled as $\sigma_{x,y} = 3$ m. . . . .	96

# List of Tables

2.1	Release year, spectrum and channel width of the most important WLAN protocols according to [40] and [48] . . . . .	6
2.2	Development of the data rates of the Bluetooth standard according to Heydon (2013) . . . . .	8
3.1	Requirements that, according to Niedermayr (2015), attributes used for fingerprinting should fulfil. . . . .	13
7.1	RMSE of deterministic and probabilistic fingerprinting in lecture room <i>AE01</i> . A WKNN approach with $K = 4$ reference points was used to obtain the results. . . . .	64
7.2	RMSE of the unfiltered, deterministic fingerprinting solution for the test rounds on 2 May 2017. . . . .	69
7.3	RMSE of the unfiltered, deterministic fingerprinting solutions for the test rounds on 2 May 2017 using the newly recorded radio map. . . . .	72
8.1	Deterministic fingerprinting results for the measurements on 19 June 2017.	76
8.2	Deterministic fingerprinting results for the measurements on 22 June 2017.	77
8.3	Deterministic and probabilistic fingerprinting results for the test rounds on 19 June 2017. . . . .	80
8.4	Deterministic and probabilistic fingerprinting results for the test rounds on 22 June 2017. . . . .	81
8.5	RMSE obtained from the sparse radio maps (50% and 30% of the original fingerprints remaining) for the measurements on 19 June 2017. . . . .	83
8.6	RMSE obtained from the sparse radio maps (50% and 30% of the original fingerprints remaining) for the measurements on 22 June 2017. . . . .	84
8.7	RMSE obtained from the interpolated radio maps (with 50% and 33% of the original reference points) for 19 June 2017. . . . .	85

LIST OF TABLES

---

8.8	RMSE obtained from the interpolated radio maps (with 50% and 33% of the original reference points) for 22 June 2017. . . . .	86
8.9	RMSE obtained from the sparse radio map with fingerprints only in areas where the vehicle should drive. The trajectories were obtained during the final test measurements on 19 June 2017. . . . .	87
8.10	RMSE obtained from the sparse radio map with fingerprints only in areas where the vehicle should drive. The trajectories were obtained during the final test measurements on 22 June 2017. . . . .	88
8.11	RMSE of the integrated trajectories in comparison to the fingerprinting-only trajectories for the test rounds recorded on 19 June 2017. . . . .	93
8.12	RMSE of the integrated trajectories in comparison to the fingerprinting-only trajectories for the test rounds recorded on 22 June 2017. . . . .	93