Universidade do Minho
Escola de Engenharia

Ivo Miguel Menezes Silva

Self-healing Radio Maps of Wireless Networks for Indoor Positioning

Programa Doutoral em Telecomunicações MAP-tele
das Universidades do Minho, Aveiro e Porto

Universidade do Minho | universidade de aveiro | U.PORTO

Abril 2022

Ivo Miguel Menezes Silva
Self-healing Radio Maps of Wireless Networks for Indoor Positioning

UMinho | 2022

Universidade do Minho
Escola de Engenharia

Ivo Miguel Menezes Silva

Self-healing Radio Maps of Wireless Networks for Indoor Positioning

Tese de Doutoramento
Programa Doutoral em Telecomunicações MAP-tele
das Universidades do Minho, Aveiro e Porto

Universidade do Minho

universidade de aveiro

U.PORTO

Trabalho realizado sob orientação de
Professor Doutor Adriano Moreira
Professor Doutor Cristiano Pendão

Abril 2022

Direitos de Autor e Condições de Utilização do Trabalho por Terceiros

Acknowledgements

First and foremost, I would like to thank my supervisors, Prof. Adriano Moreira and Prof. Cristiano Pendão, for their guidance and the countless discussions over the course of this PhD. I am thankful to Adriano Moreira who encouraged me to enrol in this PhD and accompanied me throughout my academic journey. He has been one of my role models. I am also thankful to Cristiano Pendão, for our numerous conversations, for motivating and helping me persevere when I faced challenges, and most of all, for being a friend.

I would like to thank Joaquín Torres-Sospedra for his help and collaboration. Being passionate about the things he loves, including research work, has motivated me to be better. His insight was essential to improve this work. I hope we continue this fruitful collaboration for years to come.

I am especially grateful to Vera, for her love, friendship, and for being always there for me in good and bad times. I hope to see the world the way you do someday.

I would like to thank my parents, Valdemar and Violeta, for their unconditional love and support throughout my life, and my brother, André, for his friendship and words of encouragement. My parents taught me that being dedicated and consistent leads to greater success. This work is dedicated to them.

In these acknowledgements, I remember my friends Hugo Carvalho, Carlos Carvalho, and Ricardo Peixoto, that have been there for me when I needed.

My gratitude extends to the Algoritmi Research Center for hosting this PhD and to PIEP (Pólo de Inovação em Engenharia de Polímeros) for the provision of a testing space and for their support during that period.

Lastly, I acknowledge the MAP-Tele Doctoral Programme scientific committee and the FCT (Fundação para a Ciência e Tecnologia) for the PhD grant[1] that allowed to develop this work.

[1] PD/BD/137401/2018

Statement of Integrity

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, 29th April 2022

Full Name: Ivo Miguel Menezes Silva

Signature:

Resumo

A Indústria 4.0 está a impulsionar a mudança para novas formas de produção e otimização em tempo real nos espaços industriais que beneficiam das capacidades da *Internet of Things* (IoT) nomeadamente, a localização de veículos para monitorização e optimização de processos. Normalmente os espaços industriais possuem uma infraestrutura Wi-Fi que pode ser usada para localizar pessoas, bens ou veículos, sendo uma oportunidade para aumentar a produtividade. Os mapas de rádio são importantes para os sistemas de posicionamento baseados em Wi-Fi, porque representam o ambiente de rádio e são usados para estimar uma posição. Os mapas de rádio são constituídos por amostras Wi-Fi recolhidas em posições conhecidas e degradam-se ao longo do tempo devido a vários fatores, por exemplo, efeitos de propagação, adição/remoção de APs, entre outros. O processo de construção do mapa de rádio costuma ser exigente em termos de tempo e recursos humanos, constituindo um desafio considerável. Os veículos, que operam em ambientes industriais podem ser explorados para auxiliar na construção de mapas de rádio, desde que seja possível localizá-los e rastreá-los. O objetivo principal desta tese é desenvolver um sistema de posicionamento para veículos industriais com mapas de rádio auto-regenerativos (capaz de manter os mapas de rádio atualizados). Os veículos são localizados através da fusão sensorial de Wi-Fi com sensores de movimento, que permitem anotar novas amostras Wi-Fi para o mapa de rádio auto-regenerativo. São propostas duas abordagens de fusão sensorial, baseadas em *Loose Coupling* e *Tight Coupling*, para a localização dos veículos. A abordagem *Tight Coupling* inclui uma métrica de confiança para determinar quando é que as amostras de Wi-Fi devem ser anotadas. Deste modo, esta solução não requer calibração nem esforço humano para a construção e manutenção do mapa de rádio. Os resultados obtidos em experiências sugerem que esta solução tem potencial para a IoT e a Indústria 4.0, especialmente em serviços de localização, mas também na monitorização, suporte à navegação autónoma, e interconectividade.

Palavras chave: Fusão Sensorial, Indústria 4.0, Mapas de Rádio, Posicionamento Indoor, Veículos Industriais.

Abstract

Industry 4.0 is driving change for new forms of production and real-time optimization in factories, which benefit from the Industrial Internet of Things (IoT) capabilities to locate industrial vehicles for monitoring, improving safety, and operations. Most industrial environments have a Wi-Fi infrastructure that can be exploited to locate people, assets, or vehicles, providing an opportunity for enhancing productivity and interconnectivity. Radio maps are important for Wi-Fi-based Indoor Position Systems (IPSs) since they represent the radio environment and are used to estimate a position. Radio maps comprise a set of Wi-Fi samples collected at known positions, and degrade over time due to several aspects, e.g., propagation effects, addition/removal of Access Points (APs), among others, hence they should be periodically updated to maintain the IPS performance. The process to build and maintain radio maps is usually time-consuming and demanding in terms of human resources, thus being challenging to perform. Vehicles, commonly present in industrial environments, can be explored to help build and maintain radio maps, as long as it is possible to locate and track them. The main objective of this thesis is to develop an IPS for industrial vehicles with self-healing radio maps (capable of keeping radio maps up to date). Vehicles are tracked using sensor fusion of Wi-Fi with motion sensors, which allows to annotate new Wi-Fi samples to build the self-healing radio maps. Two sensor fusion approaches based on Loose Coupling and Tight Coupling are proposed to track vehicles. The Tight Coupling approach includes a reliability metric to determine when Wi-Fi samples should be annotated. As a result, this solution does not depend on any calibration or human effort to build and maintain the radio map. Results obtained in real-world experiments suggest that this solution has potential for IoT and Industry 4.0, especially in location services, but also in monitoring and analytics, supporting autonomous navigation, and interconnectivity between devices.

Keywords: Indoor Positioning, Industry 4.0, Industrial Vehicles, Radio Maps, Sensor Fusion.

Table of Contents

Direitos de Autor e Condições de Utilização do Trabalho por Terceiros ii

Acknowledgements iii

Statement of Integrity iv

Resumo v

Abstract vi

List of Tables xiii

List of Figures xv

List of Acronyms and Abbreviations xviii

1 Introduction and Motivation 1

1.1 Research Background and Motivation 1

1.2 Objectives 3

1.3 Research Challenges 4

1.4 Proposed Hypothesis 5

1.5 Approach 6

1.6 Contributions 7

1.7 Thesis Structure 10

2 State of the Art **11**

2.1 Radio Maps and Indoor Signal Propagation . . . 11

2.1.1 Characteristics of Signal Propagation . . . 12

2.1.2 Impairments of Radio Signals in Indoor Environments . . . 13

2.2 Main Concepts of Indoor Positioning . . . 15

2.2.1 Position Estimation Algorithms and Techniques . . . 16

Triangulation – Lateration and Angulation . . . 16

Fingerprinting or Scene Analysis . . . 19

Map Matching . . . 22

Dead Reckoning (DR) . . . 22

Sensor Fusion . . . 23

2.2.2 Positioning Specifications and Performance Metrics . . . 23

2.3 Indoor Positioning Systems for Indoor Vehicles . . . 25

2.4 Methods to Build and Maintain Radio Maps . . . 27

2.4.1 Collaborative Approaches . . . 28

2.4.2 SLAM Approaches . . . 29

2.4.3 Interpolation Methods . . . 30

2.4.4 Other Methods . . . 33

2.5 Summary . . . 35

3 Radio Map Degradation **37**

3.1 Approach . . . 38

3.1.1 Radio Map Definition . . . 38

3.1.2 Radio Map Degradation Ratio . . . 39

3.1.3 Wi-Fi Fingerprinting . . . 39

3.2 Long-term Wi-Fi Data Collection . . . 40

3.2.1 Experiments Setup . . . 40

3.2.2 Manual Site Survey . . . 41

3.2.3 Monitoring Devices Data . . . 42

3.3 Radio Environment Over Time . . . 43
3.4 Radio Map Degradation Over Time . . . 46
3.4.1 Manual Site Surveys Data . . . 46
3.4.2 Monitoring Devices Data . . . 48
3.5 Events Timeline . . . 49
3.6 AP beacon receiving probability . . . 51
3.7 Summary . . . 53

4 Indoor Positioning and Tracking of Industrial Vehicles 55
4.1 Introduction . . . 55
4.2 Sensors for Vehicle Tracking . . . 57
4.2.1 Heading (Orientation) . . . 58
4.2.2 Displacement (Travelled Distance) . . . 60
4.3 Particle Filter Sensor Fusion Approaches . . . 62
4.4 Particle Filter . . . 63
4.5 Loose Coupling Approach . . . 66
4.5.1 Initialization . . . 66
4.5.2 Motion Model . . . 67
4.5.3 Update Weights . . . 68
4.5.4 Resampling . . . 69
4.5.5 Wi-Fi Position Estimation . . . 71
4.5.6 Vehicle Pose Estimation . . . 72
4.5.7 Computational Complexity . . . 73
4.6 Tight Coupling Approach . . . 73
4.6.1 Initialization . . . 74
4.6.2 Motion Model . . . 75
4.6.3 Update Weights . . . 75
4.6.4 Resampling . . . 78
4.6.5 Vehicle Pose estimation . . . 79

4.6.6 Confidence and Dynamic Alpha . . . 80
4.6.7 Simulation . . . 81
Scenarios . . . 81
Synthetic Data . . . 82
Analysis of Results using Constant α . . . 83
Analysis of Weight Threshold w_{th} . . . 84
Simulation Results . . . 85
Positioning Error vs Confidence . . . 85
4.6.8 Computational Complexity . . . 86
4.7 Parameters . . . 87
4.8 Implementation and Visualization Interface . . . 89
4.9 Real-world Experiments . . . 90
4.9.1 Testing Scenario . . . 91
4.9.2 Vehicle Prototype . . . 92
4.9.3 Results . . . 93
4.10 Summary . . . 96

5 Self-healing Radio Maps using Industrial Vehicles 100

5.1 Introduction . . . 100
5.2 Architecture . . . 102
5.3 Scenarios and Radio Map Configurations . . . 103
5.4 Enhanced Particle Filter . . . 105
5.4.1 Initialization . . . 106
5.4.2 Update Weights . . . 106
5.4.3 Dynamic Alpha . . . 108
5.4.4 Other Methods . . . 109
5.5 Vehicle (Self-healing) Radio Map . . . 109
5.6 Real-world Experiments . . . 110
5.6.1 Setup and Mobile Unit Prototype . . . 111

5.6.2 Low-cost IMU . . . 113
5.6.3 Particle Filter Parameters . . . 114
5.6.4 Results . . . 114
Estimated trajectories and annotated Wi-Fi Samples . . . 116
Computational Complexity - Optimization of Annotated Wi-Fi samples . . . 121
Vehicle Radio Map vs Traditional Radio Map . . . 123
Comparison with Similar Systems . . . 124
5.7 Radio Map Construction and Maintenance - effort . . . 126
5.8 Summary . . . 127

6 Conclusions and Future Work **128**
6.1 Summary of the Main Contributions . . . 128
6.2 Conclusions . . . 129
6.3 Future Work . . . 131

Bibliography **134**

Appendices **150**

A Long-term Collection of Wi-Fi Samples using Monitoring Devices **151**
A.1 Hardware . . . 151
A.2 Software . . . 153
A.2.1 Monitoring Device Application . . . 153
A.2.2 Server Application . . . 155
Data Collection Module . . . 155
Device Monitoring Module . . . 156
A.3 Summary . . . 157

B Dioptra: An Open Access, Open Source, Synthetic Data Generator **158**
B.1 Introduction . . . 158
B.2 Architecture . . . 159

B.3 Graphical User Interface 161
B.4 Generating a Radio Map for Pedestrian Tracking 162

C Interpolation of Radio Maps **165**
C.1 Radio Map Interpolation Methods 166
C.1.1 Kriging Method 166
C.1.2 Inverse Distance Weighted 167
C.1.3 Radial Basis Functions 168
C.2 Radio Map Interpolation using Radial Basis Functions 169
C.2.1 Inputs 170
C.2.2 Algorithm 172
C.2.3 Real-world validation 173
C.3 Summary 176

List of Tables

1.1 List of publications classified by publication type and authorship. 9

2.1 Comparison of different systems for localization of vehicles in indoor environments. . . 27

4.1 Positioning results (in meters) of simulated trajectories for different α (top) and w_{th} (bottom) values. 84

4.2 Positioning results of Loop Trajectory (LT), Random Trajectory (RT) and all trajectories using a dynamic α (in meters). 85

4.3 Parameter definition of both particle filter approaches. 87

4.3 Continued from previous page. 88

4.3 Continued from previous page. 89

4.4 Overall positioning results (in meters) of experiments with industrial vehicle. 95

4.5 Overview of the proposed approaches in comparison to similar systems. 98

5.1 Particle filter parameters. 114

5.2 Positioning results for different scenarios and radio map configurations (in meters). . . . 115

5.3 Annotated Wi-Fi samples in trajectory 1 for the no Floor Plan (nFP) scenario. 117

5.4 Annotated Wi-Fi samples in trajectory 1 for the partial Floor Plan (pFP) scenario. 118

5.5 Annotated Wi-Fi samples in trajectory 1 for the complete Floor Plan (cFP) scenario. . . . 119

5.6 Positioning results of nFP+full Radio Map (fRM) scenario using simple random sampling to limit the number of Wi-Fi samples (errors in meters). 122

5.7 Plain Wi-Fi fingerprinting results (in meters) of each scenario in comparison to an initial Radio Map (iRM). 124

5.8 Comparison with similar systems for the construction of radio maps. 125

C.1 Positioning results for different r_0 values (in meters). 174

C.2 Positioning results for different merging intervals (in meters). 175

C.3 Positioning results for different k values (in meters). 175

List of Figures

1.1 Diagram of the contributions of this thesis (dashed boxes represent the main contributions). 7

2.1 Radio wave effects: (a) reflection, (b) refraction, (c) absorption, (d) diffraction and (e) scattering. 13

2.2 Examples of wireless-based indoor positioning technologies and their positioning techniques, according to cost, accuracy and scale (adapted from [50]). 17

2.3 Positioning based on lateration: (a) Time of Arrival (ToA)/Roundtrip Time of Flight (RToF) measurements; (b) Angle of Arrival (AoA) measurements (adapted from [50]). 18

2.4 Smartphone localization using Wi-Fi fingerprinting (demonstrative scenario). 21

3.1 School of Engineering at University of Minho, Departamento de Sistemas de Informação (DSI) is on the first floor of the right side of the building. 41

3.2 DSI building: monitoring devices (blue hexagons) and test points (salmon circles). . . . 42

3.3 Long-term data collection at DSI: (a) manual site survey; (b) Monitoring Device installed in one of the rooms. 43

3.4 APs detected by Monitoring Devices over time. 44

3.5 APs detected by monitoring devices over time. 45

3.6 Manual site survey metrics: (a) mean error of Wi-Fi fingerprinting; (b) Radio Map Degradation Ratio (RMDR). 47

3.7 Monitoring devices metrics: (a) mean error of Wi-Fi fingerprinting; (b) RMDR. 49

3.8 Significant events in DSI building over time. 50

3.9 APs beacon receiving probability vs. Received Signal Strength Indicator (RSSI) mean and Standard Deviation (STDEV) of 60 APs. 52

4.1 Inertial Measurement Unit (IMU) sensors: (a) Xsens MTi-300 (Industrial-grade); (b) Adafruit BNO055 (low-cost). 59
4.2 Measuring the displacement using an absolute encoder. 61
4.3 Loose Coupling and Tight Coupling sensor fusion approaches. 62
4.4 Particle filter main algorithm. 65
4.5 Resampling scenarios in Loose Coupling approach. 70
4.6 Normalized similarity between a Wi-Fi sample and radio map samples. 77
4.7 Assigning a similarity value to each particle. 78
4.8 Density plot of position error vs confidence. 86
4.9 Particle Filter (PF) visualization interface. 90
4.10 PIEP building at University of Minho: installed APs (upper right); reference tags marking Reference Points (RPs) (lower right). 91
4.11 Floor plan of testing scenario: reference points (gray squares) and the installed APs according to their model, network standard and height (z). 92
4.12 Industrial vehicle (tow truck) used in tests. 93
4.13 Hardware diagram of the positioning module of the vehicle prototype. 93
4.14 Comparison between dead reckoning, Loose Coupling and Tight Coupling estimated trajectories. 94
4.15 Error over time of each trajectory using the Loose Coupling (LC) and the Tight Coupling (TC). 95
4.16 CDF of the positioning error comparing the Loose Coupling and Tight Coupling approaches with Wi-Fi fingerprinting. 96

5.1 Architecture of the self-healing radio maps solution. 103
5.2 Considered floor plan scenarios: (a) complete Floor Plan (cFP); (b) partial Floor Plan (pFP); and, (c) no Floor Plan (nFP). 104
5.3 Estimated similarity of each particle. 108
5.4 Dynamic α based on distance between particle and nearest radio map Wi-Fi sample. . . 109

5.5 Setup at the PIEP building: mapping of the RPs (lower left); one of the installed APs (lower center); collection of Wi-Fi samples for the iRM (lower right). 111
5.6 Mobile unit prototype with installed sensors. 112
5.7 Hardware diagram of the positioning module of the mobile unit. 113
5.8 Confidence vs positioning error of all trajectories for the no Floor Plan (nFP) scenario: (a) vehicle Radio Map (vRM) configuration; (b) fRM configuration. 120
5.9 Cumulative Distribution Function (CDF) curves of all scenarios and radio map combinations. 121

A.1 Raspberry Pi 3 B+. 152
A.2 Architecture of the solution for the long-term collection of Wi-Fi samples. 153
A.3 Wi-Fi samples collection: Entity Relationship Diagram. 156

B.1 Architecture of Dioptra [17]. 160
B.2 Graphical user interface of Dioptra [17] (dark mode enabled). 161
B.3 Example of configuration file for radio map collection. 163

C.1 Wi-Fi samples merging. 170
C.2 Estimated RSSI values for a given AP using the Radial Basis Functions (RBF) technique. 174
C.3 Average positioning error at each testing location. 176

List of Acronyms and Abbreviations

AHRS Attitude and Heading Reference System

AoA Angle of Arrival

AP Access Point

BLE Bluetooth Low Energy

cFP complete Floor Plan

CDF Cumulative Distribution Function

CSS Chirp Spread Spectrum

DR Dead Reckoning

DSI Departamento de Sistemas de Informação

EKF Extended Kalman Filter

FP Fingerprint

fRM full Radio Map

GLONASS GLObal NAvigation Satellite System

GNSS Global Navigation Satellite System

GP-LVM Gaussian Process Latent Variable Models

GPS Global Positioning System

GUI Graphical User Interface

IDW Inverse Distance Weighted

IMLE Improved Maximum Likelihood Estimated

IMU Inertial Measurement Unit

INS Inertial Navigation System

IoT Internet of Things

IPIN Indoor Positioning and Indoor Navigation

IPS Indoor Position System

iRM initial Radio Map

KF Kalman Filter

k-NN k-Nearest Neighbour

LDPL Log-Distance Path Loss

LiDAR Light Detection And Ranging

LT Loop Trajectory

mm-Wave Millimeter Wave

MAC Media Access Control

nFP no Floor Plan

NLOS Non-line-of-sight

OSM OpenStreetMap

PDR Pedestrian Dead Reckoning

PF Particle Filter

pFP partial Floor Plan

PoA Phase of Arrival

RBF Radial Basis Functions

RF Radio Frequency

RFID Radio-Frequency Identification

RMDR Radio Map Degradation Ratio

RMSE Root Mean Square Error

RP Reference Point

RPi Raspberry Pi

RSS Received Signal Strength

RSSI Received Signal Strength Indicator

RT Random Trajectory

RTLS Real-time Location System

RToF Roundtrip Time of Flight

SLAM Simultaneous Localization and Mapping

SMP Smallest M-vertex Polygon

SOP Signals of OPportunity

STDEV Standard Deviation

SVM Support Vector Machine

TDoA Time Difference of Arrival

ToA Time of Arrival

UKF Unscented Kalman Filter

UWB Ultra-Wide Band

VDR Vehicle Dead Reckoning

VLC Visible Light Communications

vRM vehicle Radio Map

Wi-Fi Wireless Fidelity (IEEE 802.11 Wireless LAN)

WLAN Wireless Local Area Network

WSN Wireless Sensor Network

Chapter 1

Introduction and Motivation

This chapter comprises the introduction to this topic starting with the motivation and research background. It also presents the main objectives of this thesis, as well as the research challenges and the approach to overcome them. Finally, the contributions of this work are discussed.

1.1 Research Background and Motivation

Industry 4.0 represents a new stage in the organization and control of the industrial value chain. The technology development for Industry 4.0 has shown a need for controlling all parts of the industrial processes. Cyber-physical systems have embedded software and are connected to the internet, hence Industrial IoT combines the physical and digital worlds, providing integration of these systems in industrial processes. The positioning information of people, goods, vehicles or robots can be used to monitor its activity, enable new means of communication, and consequently improve the operations.

As part of Industry 4.0, factories rely on connected systems and devices in the factory environment to monitor and control the tasks and operations. To do so, mobile robots or human operated vehicles in these environments depend on a positioning system that is able to estimate their position accurately. Smart factories benefit from these capabilities, leveraging real-time optimization, automation, and autonomous decision-making processes.

Wireless Local Area Networks (WLANs) are widespread in most indoor environments, i.e. shopping

centres, airports, schools, hospitals and also in factory plant buildings and warehouses. Since these networks are already installed and cover most of the building's area, they can support indoor positioning without the need to install additional infrastructure. However, interference may cause problems when using WLAN due to the unplanned and uncontrolled deployment of Wi-Fi APs and hotspots, because these devices share the same transmission bands with other systems which usually have overlapping channels.

Because of the proliferation of WLAN, Wi-Fi-based positioning systems have been used since the 2000s [1] to locate and track users inside buildings. Since then, many relevant Wi-Fi-based positioning systems have been developed for real-time navigation [2, 3]. Wi-Fi fingerprinting, used in RADAR [1], is one of the most used techniques for indoor positioning based in WLAN networks. This technique requires the construction of a radio map, the collection of several fingerprints, in an offline or calibration phase, which represent the radio environment in the specific location where they were collected. A positioning engine uses an algorithm that compares an online sample with the previously collected fingerprints to provide a position estimate. Systems based on Wi-Fi fingerprinting also have an intrinsic limit on its accuracy due to the dynamic nature of radio signals and the radio map used to obtain position estimates, usually achieving up to $\approx$2 m of mean error, with a presence of large errors in individual estimates.

The initial calibration to build the radio map is a time-consuming process, which has to be repeated over time to maintain the radio map up to date, because the radio propagation environment suffers alterations due to interference, the addition or removal of APs or alterations in the physical layout of the indoor space. When the radio propagation environment changes, the existing radio map no longer represents the reality accurately. Therefore, the IPS requires an updated radio map to maintain the expected position accuracy. Currently, the best way to overcome this challenge is the re-calibration of the system through the manual collection of new radio samples.

Positioning systems for indoor vehicles are usually expensive and require the installation of infrastructure in the factory plant and expensive sensors in the vehicle, which slows the deployment of the system and increases its overall cost. Since WLANs are already available in these buildings, the existing WLANs infrastructure can be used to locate vehicles, allowing to reduce the cost of the system.

Although Wi-Fi is ubiquitous nowadays, it was designed for communication and not for indoor positioning, hence it is still limited in terms of positioning performance. The recent development of 5G includes support for new solutions for indoor positioning [4, 5], but they are still in their early stages and require

hardware that is still not commercially available. These solutions use signal propagation information and Global Navigation Satellite System (GNSS) fusion with 5G, respectively, to estimate the position. Both solutions show a promising future for 5G.

Despite the latest developments in indoor positioning for vehicles, there still is a lack of solutions that are easy to install, setup and configure, with the ability to provide accurate position estimates while maintaining its performance through time. In industrial environments, it is difficult to maintain radio maps updated due to the frequent layout changes and intense activity from human operators and/or vehicles. Since vehicles are constantly moving in these environments, it is an opportunity to locate and track them, thus allowing not only to improve the factory's productivity and safety but also to build and maintain the radio map automatically. This self-healing radio map may be used to locate any device as long as it is provided with a Wi-Fi interface.

1.2 Objectives

The primary goal of this research work is to develop an IPS with self-healing radio maps using indoor vehicles, i.e. tugger vehicles, mobile robots or stackers. In this thesis, a self-healing radio map is defined as being capable of overcoming the variations and changes in the radio environment, keeping itself updated over time. An IPS with self-healing radio maps should have the ability to overcome the problems that affect its functioning or negatively affect its performance. This system should take advantage of industrial vehicles in operation so that it is easy to install and deploy, reducing the effort it takes to set up, create the radio map, and keep it up-to-date. To achieve this, it is necessary to go beyond the current state-of-the-art solutions that require re-calibration and configuration over time in order to maintain the same performance of the positioning system.

Providing position estimates of indoor vehicles requires a high accuracy, a low maximum error, and the capability of tracking the vehicle, hence the positioning system should use multiple sensors. Combining Wi-Fi with information from other sensors allows to get enhanced position estimates for monitoring and tracking vehicles in indoor environments, as well as to support indoor navigation of autonomous mobile robots.

Given the current state-of-the-art in positioning systems for indoor vehicles which explore expensive

industrial-grade sensors, another goal of this thesis is to explore the use of vehicles as support for self-healing radio maps using an innovative solution which is low-cost, by exploring low-cost sensors as an alternative to industrial-grade sensors to optimize the cost of the IPS.

1.3 Research Challenges

This section describes some challenges and how they were addressed during the research work.

An IPS for an industrial environment has to provide a few guarantees. For example, the system should be available at all times, it should have low latency, support high device density, and maintain the same performance, in terms of accuracy, over time. The geographic scalability of the IPS to be developed is important because it defines the area that will be covered by the system. Device density is also an important aspect of an IPS because it defines the number of simultaneous devices that the system supports.

The radio map construction in an industrial scenario poses a challenge. As the number of floors and buildings supported by the system increases, the complexity of the radio map to be constructed also increases. To help understand the challenges associated with the creation of the radio map let us suppose that a factory plant has three floors, each with a usable area of 1000 m^2. Assuming a distance of 1 m between Reference Points (RPs), it represents 1000 RPs per floor. Each RP has to be mapped properly before the collection of Wi-Fi fingerprints (also called Wi-Fi samples), which means that 3000 RPs have to be properly mapped. According to [1], the data collection process requires the collection of 20 Wi-Fi samples in four different orientations, for each RP. If we consider that the data collection in one RP takes 2 minutes, it would take 100 hours just to create the radio map. Therefore, the manual creation of a radio map requires a lot of time to collect the data and to properly map the RPs using accurate measuring tools. There are alternative solutions that facilitate the radio map construction, for example, using Simultaneous Localization and Mapping (SLAM) [6, 7], or using interpolation in a radio map with fewer samples [8], however the radio map quality is usually lower.

Motion sensors are necessary to track vehicles in indoor environments. Considering the sensor fusion of data from potentially low-cost sensors for indoor positioning poses several challenges, i.e. noisy sensors, drift in the heading sensors, estimation of the absolute heading and the ability to measure the displacement. For instance, the displacement can be measured using encoder sensors attached to the wheel's axle, and

the heading can be measured with an Inertial Measurement Unit (IMU) sensor. IMUs have accelerometers, magnetometers and gyroscopes and can be used to provide the acceleration, heading and inclination of the device. These sensors are usually affected by the accumulated error and by magnetic field disturbances or other metal objects such as heavy machinery. The most reliable IMUs are expensive[2], but there are low-cost alternatives even though they exhibit larger errors.

Normally, IPSs that provide accurate position estimates use expensive hardware to achieve high accuracy. Since one of the purposes of this research work is to find a low-cost solution that provides accurate position estimates, it will be necessary to use advanced techniques such as a PF to achieve high accuracy using information from motion sensors combined with Wi-Fi measurements. The reliability of an IPS in industrial environments is important, not only because vehicles share the space with human operators but also because the objective is to explore industrial vehicles to build and maintain radio maps. The reliability of the IPS is related to the degree of confidence in the estimated position. Higher confidence in the estimated position means that it is more likely to have a low error associated.

The evaluation of the developed solution requires a mechanism to obtain the ground truth. As positioning errors become lower, i.e. as the precision and accuracy of the system increase, the system that provides the ground truth has to be better. Therefore, the ground truth system has to be much better than the system being evaluated. Selecting and setting up the system to provide accurate and reliable ground truth is also a challenge.

1.4 Proposed Hypothesis

In contrast with previous works that have explored expensive technologies or sensors for indoor tracking of industrial vehicles, the hypothesis of this work is that exploring sensor fusion techniques of low-cost sensors/technologies (i.e. motion sensors and Wi-Fi) is a cost efficient solution for industrial environments, which can also be explored as a means to construct and maintain the radio map. This hypothesis gives response, not only to the proposed objectives, but also to the challenges associated. Therefore, it should be tested and evaluated in real-world experiments to evaluate how it performs the demanding requirements in accuracy, coverage and reliability.

[2]https://shop.xsens.com/shop/mti-100-series/mti-100-imu/mti-100-imu-2a8g4-dk

1.5 Approach

Sensor fusion techniques, such as the PF are suitable for accurate tracking of industrial vehicles equipped with several sensors, since they perform the sensor fusion of the Wi-Fi information with Dead Reckoning (DR). The combination of Wi-Fi fingerprinting and DR algorithms in a PF can be beneficial because combining both algorithms minimizes the main problems of each. The accuracy of the Wi-Fi-based algorithm could be improved once the Wi-Fi data is combined with data from the heading and displacement sensors, while the initial position of the DR algorithm could be obtained from the Wi-Fi position estimates. Additionally, Wi-Fi-based positioning could also be exploited as a means to correct the drift from the inertial sensors.

In the first phase of research, it is crucial to identify and enumerate the key factors that affect the radio signals and degrade the radio map, and to quantify how much they actually degrade over time. This process will be done through a literature review and also by conducting real-world experiments to quantify the degradation of radio maps.

The construction of the radio map is a bottleneck regarding the system's deployment because it takes some time to conclude and, without it, the system will not function properly. Therefore, the construction of the radio map has to be simplified in such a way that it can be done rapidly in large buildings with multiple floors. A way to address this challenge is to use fewer RPs for the radio map, which will be used as a starting point for a functional positioning system.

In order to keep the radio map fresh and up to date, the positioning system needs to continuously monitor the radio environment, allowing to overcome the variations in radio signals. The addition or removal of APs in the network are examples of situations that affect the radio environment, as well as the automatic configuration of the radio channels, leading to changes in the radio signals whenever the APs reconfigure automatically. Two possible solutions can be explored to address this challenge: one that uses several monitoring devices that listen to the radio environment, by collecting Wi-Fi samples; and another, that explores vehicles to automatically collect Wi-Fi samples during the utilization of the IPS to update the radio map. The first solution comprises the use of devices that are installed in known positions and constantly monitor the radio environment. The second solution takes advantage of the vehicles in operation. As they move through the indoor space, they can provide useful information for the self-healing radio maps because

the collected Wi-Fi samples can update the radio map over time. A Wi-Fi sample can be associated with the position estimate provided by the IPS. This strategy eliminates the need to periodically recalibrate the system, however, the position estimates require a degree of confidence in order to update the radio map with useful information. Finding the method to calculate confidence in a position estimate will be one of the research challenges in this matter.

The approach to validate the developed solutions will be based on a prototype that will be used in real-world experiments.

1.6 Contributions

The first contribution of this work is an analysis of the radio map degradation over time, considering datasets from a long-term data collection (over two years of Wi-Fi samples). Radio map degradation was evaluated with two metrics (mean positioning error and the Radio Map Degradation Ratio (RMDR)) which allowed to detect and understand the events that cause significant radio map degradation. This analysis can also contribute to the telecommunications field by identifying areas affected by heavy interference and other effects that may cause communication problems and may be subject to large positioning errors. This information can be used to improve WLAN deployment, by determining where APs should be deployed and the frequency channels that should be used to maximize coverage and capacity of the network. Overcoming the challenges associated with the radio map degradation in industrial environments, led to other contributions, illustrated in Figure 1.1 and described next.

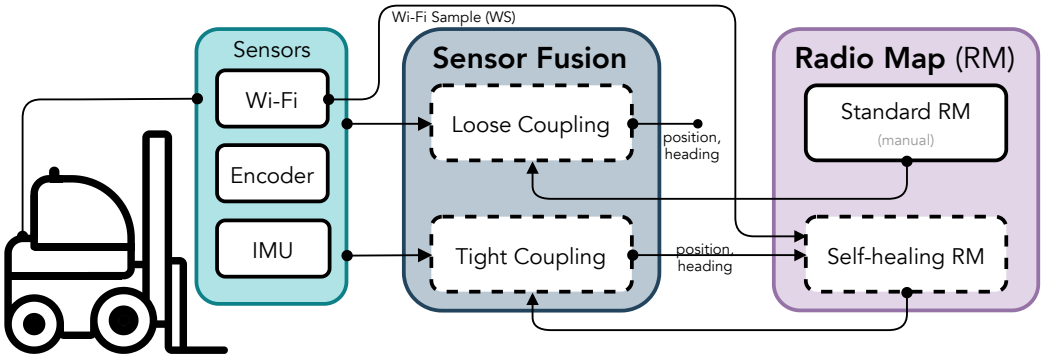


Figure 1.1: Diagram of the contributions of this thesis (dashed boxes represent the main contributions).

The second contribution of this work is the development of two distinct localization and tracking systems for industrial vehicles, based on Loose Coupling and Tight Coupling approaches. Each approach explores a different implementation of a PF to perform the sensor fusion of data collected from different sensors (Wi-Fi interface, IMU, and encoder). The Loose Coupling approach combines a Wi-Fi position estimate (obtained with Wi-Fi fingerprinting) with a DR position obtained from the IMU and encoder. Instead of obtaining a Wi-Fi position estimate, the Tight Coupling approach uses all available radio map information, by computing the set of similarities between a new Wi-Fi sample and the ones in the radio map. This information is then combined with data from the other sensors to obtain a position estimate. In this document, the terms Wi-Fi sample, fingerprint and Wi-Fi fingerprint are used interchangeably, when referring to the set of RSSI values of detected APs obtained from a scan of the radio environment using a Wi-Fi interface. These tracking systems for industrial vehicles contribute to the Industry 4.0 and IoT fields in multiple ways: integration of vehicles in factories processes; enhance the communication between vehicles and the factory's management services; improve productivity by tracking of materials across the supply chain; better management by monitoring vehicles in operation. The benefits of this research can also be applied as support for autonomous vehicles, thus allowing to increase automation.

The third contribution is a self-healing radio maps solution for industrial environments. A distinctive feature of this solution is that it explores industrial vehicles operating indoors and uses the above-mentioned Tight Coupling approach to locate and track industrial vehicles which allow to map the indoor environment collecting the radio map as they navigate through the space, therefore it reduces the mapping effort to build and maintain radio maps of industrial environments. The proposed self-healing radio maps solution for indoor positioning is considered to be low-cost in two senses, it is cheaper to deploy and install (as it explores already existing Wi-Fi infrastructure), and it uses inexpensive inertial sensors. It is an alternative to most solutions that require the installation of hardware to support the entire indoor area, as well as expensive sensors to track the vehicle's trajectory.

The scientific publications and datasets that were produced during this research work are presented in Table 1.1 classified by authorship and publication type.

Table 1.1: List of publications classified by publication type and authorship.

Type	Authorship	Title
Journal	Co-Author	F. Potorti, S. Park, A. Crivello, F. Palumbo, M. Girolami, P. Barsocchi, *et al.*, "The IPIN 2019 Indoor Localisation Competition—Description and Results," *IEEE Access*, vol. 8, pp. 206 674–206 718, 2020. DOI: 10.1109/ACCESS.2020.3037221
Journal	Co-Author	F. Potorti, J. Torres-Sospedra, D. Quezada-Gaibor, A. R. Jimenez, F. Seco, A. Perez-Navarro, *et al.*, "Off-line Evaluation of Indoor Positioning Systems in Different Scenarios: The Experiences from IPIN 2020 Competition," *IEEE Sens. J.*, pp. 1–1, 2021. DOI: 10.1109/JSEN.2021.3083149
Journal	First Author	I. Silva, C. Pendao, J. Torres-Sospedra, and A. Moreira, "TrackInFactory: A Tight Coupling Particle Filter for Industrial Vehicle Tracking in Indoor Environments," *IEEE Trans. Syst. Man, Cybern. Syst.*, pp. 1–12, 2021. DOI: 10.1109/TSMC.2021.3091987
Journal	First Author	I. Silva, C. Pendao, and A. Moreira, "Real-World Deployment of Low-Cost Indoor Positioning Systems for Industrial Applications," *IEEE Sens. J.*, vol. XX, no. XX, pp. 1–1, 2021. DOI: 10.1109/JSEN.2021.3103662
Conference	Co-Author	J. Torres-Sospedra, A. Moreira, G. M. Mendoza-Silva, M. Joao Nicolau, M. Matey-Sanz, I. Silva, J. Huerta, and C. Pendao, "Exploiting Different Combinations of Complementary Sensor's data for Fingerprint-based Indoor Positioning in Industrial Environments," in *2019 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Sep. 2019, pp. 1–8. DOI: 10.1109/IPIN.2019.8911758
Conference	First Author	I. Silva, A. Moreira, M. J. Nicolau, and C. Pendão, "Floor Plan-free Particle Filter for Indoor Positioning of Industrial Vehicles," in *ICL-GNSS Int. Conf. Localization GNSS*, Tampere, Finland, 2020
Conference	Co-Author	J. Torres-Sospedra, F. J. Aranda, F. J. Alvarez, D. Quezada-Gaibor, I. Silva, C. Pendao, and A. Moreira, "Ensembling Multiple Radio Maps with Dynamic Noise in Fingerprint-based Indoor Positioning," in *2021 IEEE 93rd Veh. Technol. Conf.*, Helsinki, Finland: IEEE, Apr. 2021, pp. 1–5. DOI: 10.1109/VTC2021-Spring51267.2021.9448947
Conference	First Author	I. Silva, C. Pendão, J. Torres-Sospedra, and A. Moreira, "Quantifying the Degradation of Radio Maps in Wi-Fi Fingerprinting," in *2021 Int. Conf. Indoor Position. Indoor Navig. IPIN 2021*, 2021
Conference	Co-Author	C. Pendão, I. Silva, A. Moreira, and J. Torres-sospedra, "Dioptra – A Data Generation Application for Indoor Positioning Systems," in *2021 Int. Conf. Indoor Position. Indoor Navig. IPIN 2021*, Lloret de Mar, 2021
Conference	Co-Author	J. Torres-Sospedra, I. Silva, L. Klus, D. Quezada-Gaibor, A. Crivello, P. Barsocchi, C. Pendão, E. S. Lohan, J. Nurmi, and A. Moreira, "Towards Ubiquitous Indoor Positioning : Comparing Systems across Heterogeneous Datasets," in *2021 Int. Conf. Indoor Position. Indoor Navig. IPIN 2021*, 2021
Dataset	Co-Author	A. Moreira, M. J. Nicolau, I. Silva, J. Torres-Sospedra, C. Pendão, and F. Meneses, "Wi-Fi Fingerprinting dataset with multiple simultaneous interfaces," version 1.0, Sep. 2019. DOI: 10.5281/zenodo.3342526
Dataset	Co-Author	A. Moreira, I. Silva, and J. Torres-Sospedra, "The DSI dataset for Wi-Fi fingerprinting using mobile devices," version 1.0, Apr. 2020. DOI: 10.5281/zenodo.3778646

1.7 Thesis Structure

This chapter (Chapter 1) presented the introduction to this thesis, including the primary research objectives as well as the approach and contributions. Chapter 2 introduces the state of the art, which includes an analysis of the propagation effects and impairments of radio signals, the concepts of indoor positioning, a literature review of IPSs for industrial vehicles, and solutions that dynamically adapt to radio environment changes. In Chapter 3, an empirical analysis of the radio map degradation in a long-term experiment is made. It includes the evaluation of the radio map degradation over time and how it affects positioning performance. In Chapter 4, the solution is presented for the indoor positioning and tracking of industrial vehicles exploring two approaches for sensor fusion, Loose Coupling and Tight Coupling. Chapter 5 describes the self-healing radio maps solution, which is based on an improved version of the positioning solution based on Tight Coupling. It explores the vehicles in operation to automatically build and maintain radio maps, overcoming radio map degradation. Finally, the conclusions and future work are discussed in Chapter 6.

Chapter 2

State of the Art

This chapter provides an introduction to the most relevant concepts regarding radio maps, signal propagation and indoor positioning, including the characteristics of signal propagation and impairments of radio signals in indoor environments. Then, the main positioning techniques and algorithms are presented, followed by the positioning specifications and performance metrics. Finally, an analysis of existing IPSs for indoor vehicles and several methods to build and maintain radio maps is presented.

2.1 Radio Maps and Indoor Signal Propagation

A radio map is defined by the set of radio signals observed in indoor environments. Many devices equipped with a radio interface (Wi-Fi or Bluetooth) are sources of these signals, e.g., smartphones, Access Points and laptops. Since these signals can be properly measured and retrieved by these types of devices, there is extensive research that explores radio signals for indoor positioning [1, 6, 21]. Radio maps are used in positioning based on Wi-Fi or Bluetooth, because a set of signal levels measured at a specific position is expected to have a unique set of values which differs from the signal levels measured at other positions in the building. However, radio maps are constantly changing due to the dynamic nature of radio signals, which are affected by propagation effects. Therefore, it is important to keep them updated. The following section describes how radio signals behave in indoor environments and how they are affected by different propagation effects.

2.1.1 Characteristics of Signal Propagation

In Radio Frequency (RF) propagation, path loss accounts for all elements that affect the signal propagation, i.e., free-space loss, atmospheric losses, fading loss due to multipath and other various effects based on the wave frequency and the environment [22]. If the main path of the signal is governed by free-space loss, it can be modelled by the free-space loss Friis equation:

$$L = G_T G_R \left(\frac{\lambda}{4\pi d}\right)^2 \tag{2.1.1}$$

where L is a gain, G_T and G_R represent the transmitting and receiving antenna gains, λ is the wavelength, and d represents the distance between the two antennas.

If the gains of the transmitting and receiving antennas are excluded from the the path loss expression, free-space loss can be defined as:

$$L_{FSL\ dB} = -20 \log\left(\frac{\lambda}{4\pi d}\right) \tag{2.1.2}$$

When there is a radio transmission between two devices, the receiver sensitivity is an important factor that defines the threshold for acceptable operation. The received signal must have an minimum signal-to-noise ratio (SNR) or bit energy to noise power spectral density (E_b/N_0), that enables the receiver to recover the received information. The noise component of the signal is set by the receiver front-end. It is also known as thermal noise, because of the thermal agitation of electrons at the front end of the receiver. Thermal noise is modelled through a White Gaussian distribution with a constant power spectral density, meaning that the power density will be the same over all frequencies of interest ($N_0/2$). Therefore, thermal noise is considered to be "white" and additive because it is Gaussian and independent of the signal involved. The following equation from [22] defines the thermal noise power of a receiver:

$$N = \kappa T_0 B + \kappa T_e B \qquad [W] \tag{2.1.3}$$

The Boltzmann's constant is represented by $\kappa = 1.38 \times 10^{-23}$ J/K, $T_0 = 290$ K is the standard

noise temperature, T_e is the equivalent noise temperature of the receiver, and B is the noise-equivalent bandwidth (Hz). Since the thermal noise is defined by a Gaussian distribution and is generated at the receiver, one can assume that the noise is constant, therefore the SNR will be mainly defined by the variations in the signal levels. When the signal strength is below the sensitivity of the receiver, the transmitted information cannot be retrieved properly.

2.1.2 Impairments of Radio Signals in Indoor Environments

The propagation of radio signals can be compared to the propagation of light waves. Therefore, these waves can be affected by the phenomena of reflection, refraction, diffraction, absorption and scattering, as depicted in Figure 2.1. The materials that go into the walls, floors and doors of the building have a deep influence on the propagation of radio signals.

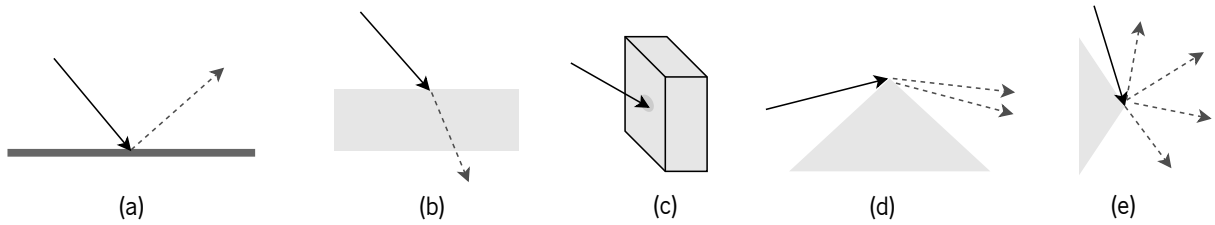


Figure 2.1: Radio wave effects: (a) reflection, (b) refraction, (c) absorption, (d) diffraction and (e) scattering.

Radio signals can bounce off certain surfaces, like water or metal. These surfaces that reflect the radio signals can be found in many indoor scenarios, such as laboratories, offices, factory buildings or warehouses. A signal is refracted when it enters a medium with a different propagation speed. The wave is bent when the signal enters the new medium. For instance, water and glass are two media that cause the refraction of radio waves. The phenomenon called absorption represents the amount of power the waves lose depending on their frequency and on the material that absorbs the wave [23]. The diffraction effect occurs when the wave goes around corners. It is an apparent bending of radio waves when hitting an object [23]. Scattering is similar to diffraction, however, it is more unpredictable. It happens when the signal hits an object and scatters in all directions, which can have a significant impact on the signal power. Whenever an electromagnetic wave goes through some material, it may be weakened or dampened.

The radio waves often reach the receiver by more than one path, due to reflection, diffraction and scattering caused by the building's structures. This phenomenon is known as multipath [24]. A distorted version of the original signal is obtained at the receiver, composed of various signal components that follow different indirect paths and (possibly) a direct path component. Each signal component arrives at the receiver with a different delay and a different signal power that is characterized by the attenuation of the channel.

The Wi-Fi signals inside buildings may be affected by one or more of the above-mentioned effects at the same time. Radio signals from APs suffer from the channel attenuation that causes the signal to lose power as it propagates through the propagation medium. In addition, Wi-Fi radio signals in indoor environments may vary significantly over time due to various factors:

- **Interference**: Any device that generates electromagnetic waves can cause interference in signals to wireless APs inside buildings, as long as the transmitting frequency is close to the one of the APs in the network. Most WLAN environments use the 2.4 GHz band (i.e., IEEE 802.11b/g/n) for communication, which has several overlapping channels which interfere with each other. Intra-channel interference and inter-channel interference are common in scenarios where the Wi-Fi infrastructure is mostly composed of 2.4 GHz APs.
- **Addition/removal of APs**: When the network configuration changes due to the addition or removal of APs, it affects the propagation of signals. Newly added APs can cause interference when they share the same transmission channel as other ones. In many situations, this causes APs to reconfigure to transmit in a different frequency band, thus leading to other changes in the radio environment. In addition, it is known that APs frequently have a dynamic power control to adjust to network conditions [25]. Mobile APs (hotspots) also contribute to an increase in the interference between Wi-Fi enabled devices because they transmit in the same frequency bands as standard APs. Moreover, they are a challenge for fingerprinting-based solutions since they can be easily moved, and switched on and off frequently. Therefore, mobile APs are not useful for fingerprinting-based positioning, because it usually relies on RSSI values from fixed APs.
- **People moving**: It is known that the mobility of people and the human body have an influence on the propagation of signals from APs, inside buildings. Since humans are mostly made of water, the radio waves are absorbed by the body because it has an absorption similar to that of water [23].

- **Layout modifications**: Changes in the layout of indoor buildings also affect the propagation of radio waves, hence it causes the radio map to become outdated. Particularly in dynamic environments such as factories, the periodic change in the indoor layout leads to alterations in the propagation of radio signals from APs, because they will find different paths until they reach the receiver device.

2.2 Main Concepts of Indoor Positioning

Positioning systems have numerous applications in indoor and outdoor environments. In outdoor environments, GNSS uses satellites to provide geographic positioning. There are several GNSS systems such as Global Positioning System (GPS)[3], Galileo[4], GLObal NAvigation Satellite System (GLONASS)[5] or BeiDou[6], that use satellite constellations to provide the geolocation of the receiver. For outdoor scenarios, GNSS is used for tracking devices, locating firefighters, assisting in navigation, helping in emergencies, and in many other applications. GNSS systems are reliable in these applications, however, they are not reliable in indoor environments because GNSS signals cannot be properly retrieved inside buildings due to attenuation and severe multipath.

For indoor positioning, alternative systems have been developed to provide coverage inside buildings where GNSS systems are not reliable. In indoor environments, positioning systems use other technologies such as Wi-Fi [1, 26, 27, 28], Radio-Frequency Identification (RFID) [29, 30], Ultra-Wide Band (UWB) [31, 32], Bluetooth [33, 34], ultrasound [35], or combinations of multiple technologies [30, 36, 37, 38, 39]. These systems are mostly used to track people or objects inside buildings, but there is a recent trend in IPSs for indoor navigation of autonomous mobile robots [3, 40, 41, 42, 43].

The research over the past decade has shown several applications for indoor positioning, i.e., mobile robots navigation and location-based services. Since the research community is always looking for new technologies to use, Visible Light Communications (VLC) has been used in [44, 45] as an alternative technology for indoor positioning. Although these systems achieve very high accuracy (mean error below 5 cm), in order to function properly, they require a line-of-sight signal and are limited to small deployments

[3] https://www.gps.gov
[4] http://www.esa.int/esaNA/galileo.html
[5] https://www.glonass-iac.ru/en/
[6] http://en.beidou.gov.cn/SYSTEMS/System/

inside an indoor room.

More recently, with the research and development of 5G wireless networks, several experiments for indoor positioning using Millimeter Wave (mm-Wave) technology were developed [4, 46, 47, 48, 49]. mm-Wave shows a promising future because this technology has a wide radio spectrum and high bandwidth available, that can be used in many applications. Simulation constitutes most of the research work in this area because it requires hardware that is not readily available and deployed in indoor buildings nowadays. Hence, it is expected that most applications using mm-Wave will be deployed in the coming years.

Outdoor localization systems usually provide position estimates based on geographic coordinates, however, in indoor environments the location information should consider that buildings can have multiple floors, each floor can have many rooms and areas, and finally, the position can be defined as a set of coordinates that represent the absolute position inside the building.

2.2.1 Position Estimation Algorithms and Techniques

For IPSs, accuracy is a very important aspect, which mainly depends on the technology of the IPS. Figure 2.2 shows a graph that plots the cost against the accuracy of several technologies for indoor positioning. It also shows the typical coverage scale of an IPS based on each technology. The most accurate IPSs are the most expensive, e.g., the ones based on UWB or mm-Wave, and they are better suited for small scale positioning systems because of the required infrastructure. Conversely, Wi-Fi and Bluetooth-based positioning systems are low-cost but have lower accuracy.

Different position estimation algorithms may be used depending on the technology. Most technologies rely on accurate measurements of signal characteristics, such as the time, angle or phase of arrival, or even based on the signal strength. The following sub-sections describe several position estimation techniques used in indoor environments.

Triangulation – Lateration and Angulation

Triangulation uses geometric properties of triangles to estimate the target location. This technique includes two variants: lateration and angulation. Lateration takes measurements on the distance between the target

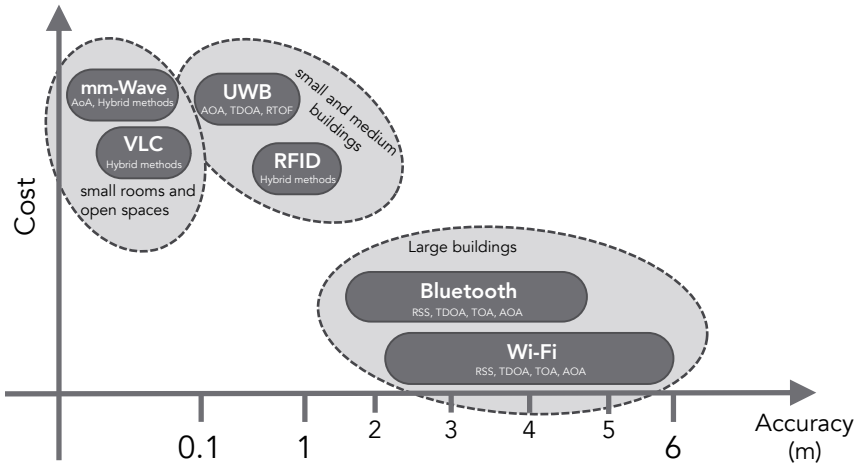


Figure 2.2: Examples of wireless-based indoor positioning technologies and their positioning techniques, according to cost, accuracy and scale (adapted from [50]).

object and the RPs to estimate the position of an object. Angulation estimates the location of the object from measured angles associated with the RPs. In turn, lateration may use the following techniques:

- **Time of Arrival (ToA)**

  This radio-based technique is based on the assumption that the distance from the mobile terminal to the measuring unit is directly proportional to signal propagation time. ToA measures the propagation time between a transmitter and a receiver. As shown in Figure 2.3 (a), ToA measurements must be from at least three RPs, in order to enable 2D positioning. For ToA-based systems, one-way propagation time is measured to calculate the distance between the measuring unit and the signal transmitter. However, direct ToA faces two main problems. First, all devices must be time-synchronized precisely. Second, a labelled timestamp in the transmitting signal is necessary in order to allow the measuring unit to determine the distance that the signal has travelled. This technique can be used by various transmission techniques such as Direct-Sequence Spread-Spectrum [51] or UWB [52, 53, 54].

  Systems based on UWB take advantage of the radio signals that have high temporal resolution and provide accurate ToA measurements in multipath environments. The UWB system presented in [54] was developed with simple components and achieved very promising results with location error below 0.3 m inside a laboratory. The main disadvantage of UWB based systems is the implementation

costs associated with the need to install multiple sensors to cover the entire indoor area.

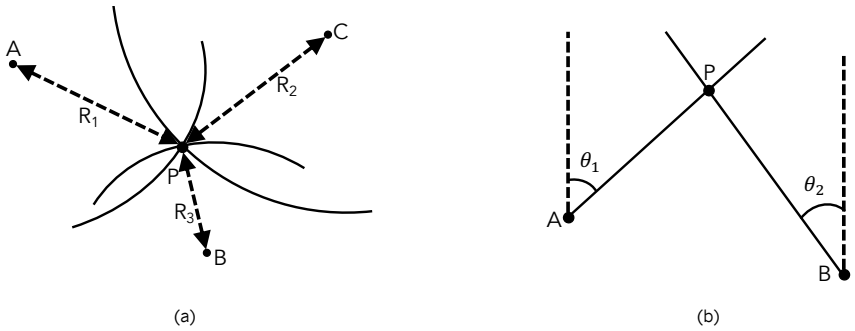


Figure 2.3: Positioning based on lateration: (a) ToA/RToF measurements; (b) AoA measurements (adapted from [50]).

- **Roundtrip Time of Flight (RToF)**

  This algorithm measures the time-of-flight of a signal travelling from the transmitter to the measuring unit and back. RToF requires better clock synchronization in comparison with ToA and its measuring unit is considered as a radar.

  The UWB-based system presented in [53] uses RToF combined with a bandwidth of more than 1 GHz to deal with multipath environments and provide high accuracy. These features are ideal for the demanding requirements of indoor location. Another interesting solution, proposed in [55], uses UWB with RToF to locate mobile robots in indoor environments with positioning errors of 20 cm for a coverage area of 120 m^2.

- **Time Difference of Arrival (TDoA)**

  The main idea of this technique is to measure the time difference of the arrival of one or more signals to a device in order to estimate its relative position.

  Cricket [56] and Active Bat [57] systems use ultrasonic pulse TDoA to enable high precision 3D

positioning in indoor environments. The accuracy provided by these systems is 9 cm and 2 cm for Active Bat and Cricket, respectively. The accuracy is very high, but Cricket only works with a range of 10 m and Active Bat 50 m, which is low in many scenarios.

- **RSS-based (or Signal Attenuation-based)**

  The Received Signal Strength (RSS) technique can be used in two ways: trilateration and fingerprinting. Regarding the trilateration, this method uses the attenuation that affects the radio signal as the measuring unit to apply a trilateration algorithm, resulting in the location's estimation. Fingerprinting is described in detail later in this section.

- **Phase of Arrival (PoA) or Received Signal Phase Method**

  In this method, the range is estimated from the carrier phase (or phase difference). The phase difference is calculated by comparing the received signal with a pure sinusoidal signal with zero phase offset.

Angulation techniques (Angle of Arrival (AoA) Estimation) estimate location by intersecting multiple pairs of angle direction lines, each formed by a circular radius from a base station or a beacon station to the mobile target. AoA uses at least two known RPs and two measured angles in order to derive the 2D location of the target. Figure 2.3 (b) shows how the position is obtained from two angles.

Fingerprinting or Scene Analysis

Fingerprinting comprises algorithms that rely on the collection of features from a scene (fingerprints) and then estimate the location of an object based on online measurements that are matched with fingerprints previously collected at known positions. RSS-based location is commonly used in scene analysis. This technique is based on the idea that each place has a unique set of radio signals that allow to identify it to the detriment of other places. A fingerprint is known as an association of a unique measurement to a place.

Location fingerprinting includes two phases, the offline or calibration phase and the online or real-time phase. In the calibration phase, a radio map is built by collecting fingerprints in multiple RPs spread through the indoor space. It is ideal to collect several fingerprints in each RP, due to the dynamic nature of radio signals. It improves the system's robustness because it uses more relevant data to estimate the position.

The level of detail in the offline phase is associated with the accuracy of the systems that use fingerprinting. A higher detail in the calibration phase provides higher accuracy and better results. As mentioned before, building a radio map is a very arduous task that becomes even more complicated to perform, especially for large buildings, since the collection of data requires a larger number of RPs. Keeping the radio map updated is important because radio signals are constantly varying due to propagation effects, alterations in the indoor layout of the building, and APs can be moved to another place.

In the online phase, the positioning algorithm estimates the location by comparing previously collected data in the calibration phase with the obtained signal strength values from the device. Fingerprinting based techniques use algorithms based on pattern recognition, e.g., probabilistic methods, neural networks, k-Nearest Neighbour (k-NN), Support Vector Machine (SVM), and Smallest M-vertex Polygon (SMP).

Figure 2.4 displays an indoor scenario with a Wi-Fi fingerprinting positioning system. During the offline phase, several Wi-Fi samples were collected at each RP, represented by the white-filled circles. In the online phase, a mobile device is used to obtain online Wi-Fi samples. In the figure, the collected Wi-Fi sample is composed of the RSSI values of each detected AP in that position. The online Wi-Fi sample represented by the red-filled circle can be compared with the ones previously collected during the offline phase. Then, a position estimate is obtained using one of the positioning algorithms described above. Assuming the k-NN algorithm is used, the position estimate is the centroid between the k radio map Wi-Fi samples that are more similar to the online Wi-Fi sample collected at the red-filled circle.

Probabilistic methods take advantage of statistical models to provide the position estimates. These methods calculate the likelihood that the mobile node is in a specific location. The decision rule is made using Bayes' formula.

Neural networks take advantage of the calibration phase to use the collected RSS measurements as the inputs for the training purpose. Weights are obtained after the training of the neural network. A position estimate is obtained following the multiplication of the input vector with RSS measurements by the trained input. The output can be a vector of two or three elements, which represents the 2D or 3D of the estimated position. Neural networks have been successfully used to improve indoor localization in [26, 58, 59].

The k-NN algorithm uses the measured RSS values in the online phase to compute the distance to the closest k matches of known locations from the calibration phase. An estimated location is obtained by averaging k location candidates. Optionally, the distances in the signal space can be used as weights to

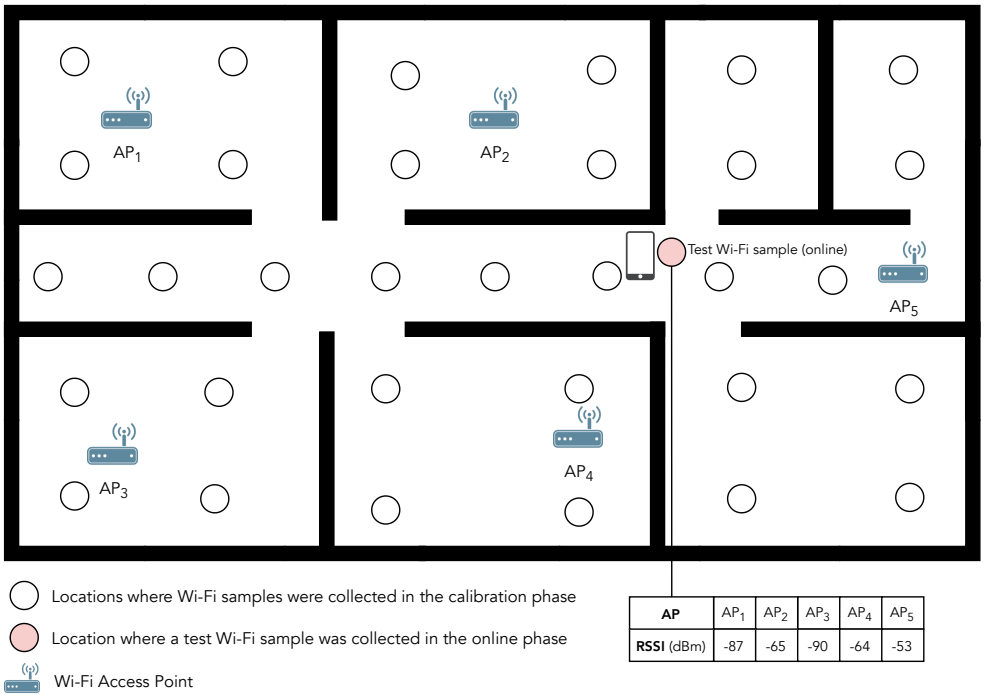


Figure 2.4: Smartphone localization using Wi-Fi fingerprinting (demonstrative scenario).

obtain the position estimate. The k parameter specifies the number of neighbors to which the distances are calculated, and the performance of this algorithm depends on finding an optimal k value for each case. RADAR [1] is one of the most well-known IPSs that use the k-NN algorithm.

SVM is a tool for statistical analysis and machine learning that has been used in many applications for data classification and regression. In [60], an SVM is used to classify the location of a wireless client from its signal strength measurements.

In the SMP algorithm, the online RSS measurement is used to search the location database for M candidate locations from each access point. M-vertex polygons are formed by choosing at least one candidate location from each access point. The location estimate results from the average of the coordinates of vertices of the smallest polygon. SMP has been used in MultiLoc [37, 61].

Most systems based on scene analysis use Wi-Fi or Bluetooth Low Energy (BLE) as the radio technologies. Currently, most buildings have Wi-Fi coverage, therefore the information from the APs' radio signals can be used to collect the radio map, which can be used to estimate the position of a Wi-Fi-enabled device. RADAR [1] and Redpin [36] are examples of Wi-Fi fingerprinting systems capable of estimating the indoor

location. RADAR has an accuracy between 2 and 3 m and Redpin can estimate the correct room location 9 out of 10 times. Since these systems take advantage of existing Wi-Fi infrastructures, they are cheap to implement because they are mostly based on software modules and do not require the installation of additional hardware.

Map Matching

The map matching technique associates the device's position with a digital map. The main goal of this algorithm is to identify the segment where the object moves and estimate the location of the object in that segment. Map matching algorithms are based on the pattern recognition theory by comparing the estimated location of the device with the digital map. Commonly, this technique is used in car navigation to combine position estimates provided by the GNSS system with information from road maps to obtain better position estimates (lower errors). In indoor environments, map matching can improve navigation of pedestrians [62, 63].

Dead Reckoning (DR)

The main goal of DR is to determine the position, starting from a known point (previously determined position) and estimating the position using the speed (measured or estimated), time and course (movement direction). The positioning accuracy of dead reckoning is affected by cumulative errors that appear over time due to errors in sensor readings. Inertial Navigation Systems (INSs) are one of the most used sensors for DR. They provide continuous position, velocity, and orientation estimates, which are accurate for a short term but are subject to cumulative errors due to the noise of the sensor [62]. Filtering techniques can be used to reduce the effect of measurement noise.

A system that integrates a Wi-Fi-based positioning system with an INS for indoor mobile positioning is presented in [2]. It takes the advantages of both systems to provide better accuracy. Positioning based in Wi-Fi provides an accuracy of 3 m, but combining it with an INS allows to achieve a mean error of about 1.53 m. Fusing the INS information with a Wi-Fi fingerprinting system has three advantages namely, the improvement in system performance regarding positioning, the ability of obtaining the device's initial position, and the ability of tracking the device even when Wi-Fi is not available (DR navigation).

Sensor Fusion

Sensor fusion comprises the combination of sensory data or data that is derived from sensory data, which results in information that is in some sense better than would be possible when using these sources separately [64]. It relies on many techniques and algorithms to produce estimates. For example, the Kalman Filter (KF) is one of the most well-known algorithms that uses a series of measurements over time, containing statistical noise and other inaccuracies to produce estimates of unknown variables. Particle Filters (PFs) are also used in sensor fusion, they are used in non-linear and non-Gaussian problems that the KF cannot solve. PFs are based on Monte Carlo algorithms and comprise the online sampling of particles that have an associated weight at each time point. Particles closer to the position of the device are the ones with higher weights. Conversely, particles farther from the position of the device are the ones with lower weights. The particles' weight is updated every time the sampling is performed and the particles with lower weights are replaced with new ones with higher weights, therefore, they converge to a specific area.

Sensor fusion is used for mobile robot and orientation estimation in [65, 66], and for the localization and tracking of mobile users exploring Pedestrian Dead Reckoning (PDR) techniques in [67, 68, 69].

2.2.2 Positioning Specifications and Performance Metrics

Depending on the application, an IPS may provide different types of locations: physical, symbolic, absolute or relative [70]. A system that provides the physical location is the GNSS system with the latitude and longitude coordinates, therefore a physical location is expressed in coordinates that identify a point in a 2D/3D referential [50]. A symbolic location is represented by abstract ideas of where something is [70], e.g.: in the office, in Building 1, or in a bus approaching Lisbon. A system that provides physical location can be extended to provide a symbolic location. The symbolic location is very useful in many scenarios where it is important to locate people or objects inside buildings in particular rooms or areas. However, symbolic locations are not useful in some cases, for example, when it is necessary to know the exact position inside a room. An absolute location shares the same reference grid for all located objects [70], i.e., GPS shares the same coordinate system (latitude, longitude) for all located objects. In a relative location system, each object can have its own reference frame. Usually, the relative location information

is based on the proximity to known RPs or base stations [50].

An IPS is characterized by a set of metrics that define how the system performs [50]: accuracy, scalability, cost, precision, complexity, and robustness are among the most relevant. First, the accuracy or location error is probably the most important requirement of positioning systems. It is usually defined by the mean Euclidean distance between the real position and the corresponding estimated position. Second, the scalability can be achieved in two domains, geography and density. Geography scale means that as the space dimension increases, the system is scalable if it supports larger areas. Density scale means that the system is able to support a higher number of units located per geographic area/space per time period. Scalability can also be measured in terms of support for 2D or 3D spaces, as some systems support only one and others are able to support both 2D and 3D spaces. Third, the cost of a positioning system depends on a number of factors: the installation, the coverage area of the system, the operation and maintenance. The selected technology of the positioning system also affects its overall cost, however, Wi-Fi-based positioning systems usually take advantage of the existing infrastructure having lower cost. Fourth, precision provides a measurement regarding the consistency of the positioning system, because it allows to observe the variation in performance in many trials with the CDF. Using the CDF, it is possible to compare two different systems that have the same accuracy. Fifth, the complexity of a positioning system can be attributed to hardware, software and operation factors. If a positioning system is too complex and not fit for online operation, it will not be feasible for operation in the real-world. Finally, the robustness of a system is associated with the ability to keep the system working when under severe conditions. For example, considering a Wi-Fi-based positioning system, it is robust if it can function when there is interference between radio signals, or when there is a lack of radio signals from APs.

In more demanding circumstances, such as the indoor positioning of industrial vehicles, the system should have high accuracy (mean error below 1 m) with low maximum error, and it should consider additional metrics, namely, reliability, update rate, delay, convergence time, and uptime [71]. A system is considered to be reliable if the provided position estimates can be trusted. Therefore, an important feature is a system being able of assessing whether the position estimates have a low or high error associated. However, most IPSs are not able to obtain this type of information. The update rate is the rate at which the positioning system generates new position estimates, it is crucial when it is necessary to track a vehicle's movement with precision. The delay characterizes the time it takes to generate a position estimate. Once

the system is running, the delay should be as low as possible. Convergence time defines the time it takes since the system is initialized until it starts producing accurate position estimates, it should be as low as possible. Uptime is the activity time of the system, which should be the same as the time during which vehicles are in operation. When devising a positioning system for industrial vehicles, all these aspects have to be considered to optimize the cost/accuracy relationship without sacrificing any of the other requirements.

Currently, there is a lack of standards for evaluation of IPSs that enable a fair comparison between them. Usually, each IPS is tested under specific conditions and the reported results are difficult to compare with other solutions because they were evaluated in different scenarios, with different limitations. Another challenge regarding benchmarking of IPSs is the method used to measure the positioning error. Although the ISO/IEC 18305:2016 [72] standard provides an evaluation method for IPSs by defining the magnitude of the error as the 2D (or 3D) Euclidean distance between the estimated and true positions, many researchers use other methods to evaluate their systems. For example, in [73], a method is proposed for distance error measurement in pedestrian indoor localization, that accounts for floor transitions. The Indoor Positioning and Indoor Navigation (IPIN) competition [9] has been contributing to the benchmarking of IPSs in on-line and off-line tracks. Since all competing systems are evaluated under the same conditions, the results can be fairly compared against each other. To measure the positioning performance, errors are usually measured in 2D, with an added penalty when the floor (approximate z coordinate) is wrongly estimated.

2.3 Indoor Positioning Systems for Indoor Vehicles

There is a need for locating vehicles in industrial environments, in some scenarios to manage the operations and in other scenarios to support autonomous navigation. Vehicles in indoor environments can be manually operated (by a human) or be autonomous mobile robots. Most of the research in this area comprises IPSs for autonomous mobile robots because they are mostly used to support indoor navigation, however, they can also be used to monitor and track the position of indoor vehicles. The main principles of IPSs for autonomous mobile robots can be used for manned or even hybrid vehicles (vehicles that can be both autonomously and manually operated), therefore their analysis is relevant for the purpose of this research work. Different technologies may be used for the localization of industrial vehicles depending on the application requirements, being the cost, accuracy and reliability the most important factors when choosing

the technology. The localization of industrial vehicles indoors may resort to camera vision, ultrasounds, magnetic, Light Detection And Ranging (LiDAR), UWB, RFID, Wi-Fi or Bluetooth. These technologies are often combined with motion sensors using sensor fusion to improve positioning performance. Additional details about real-time localization systems in industrial settings can be found in [74]. In the following, an analysis of several IPSs for industrial vehicles based on different technologies/sensors and localization techniques is presented.

UWB systems are used for vehicle localization in [75, 76]. In [75], UWB is combined with GPS to provide indoor/outdoor positioning of industrial vehicles. Sensor fusion is performed with a PF, achieving an average error of 0.16 m in simulations. A drawback of this solution is that it depends on line-of-sight propagation and it is expensive due to the use of UWB.

In [77], a forklift is located using several algorithms to combine laser range scanners (LiDAR) with odometry information. The approach in this paper combines AMCL (Adaptive Monte Carlo Localization), scan matching and Discrete Fourier Transform-based pose estimation refinement into one algorithm stack for high-precision localization in industrial indoor environments.

Wu et al. [43] proposed an IPS for mobile robot localization that does not depend on additional infrastructure because positioning relies on magnetic field and LiDAR data. The initial position is obtained from magnetic fingerprinting while a finer position is obtained by laser scan matching. The empirical experiments performed in an area of 57x19 m revealed a Root Mean Square Error (RMSE) of 1.43 m, considering that the initial pose is known.

RFID has been explored in [30, 66] for mobile robot localization. In [66], passive tags placed on walls of the space are detected by a robot equipped with RFID antennas. The robot's pose is obtained from sensory fusion of RFID with odometry, using a similarity sensor model. Experiments have revealed a mean error of 0.14 m in an area with around 230 m^2. Both solutions are complex to scale for larger buildings due to the installation of RFID tags and since they can also suffer from interference from Wi-Fi or Bluetooth.

Although UWB, LiDAR and RFID technologies can meet the demanding requirements of indoor tracking of industrial vehicles, they are expensive. Alternatively, Wi-Fi is low-cost and ubiquitous since it is present in most buildings, including factories, thus it is an ideal technology for deployments that take advantage of already existing infrastructure. Wi-Fi has been explored for localization in industrial environments to improve productivity in [78, 79], and explored for vehicle tracking in [3, 80]. The IPS for vehicles based on Wi-Fi

fingerprinting [80] uses a raw data smoothing technique to feed a classifier based on neural networks, which can deal with noisy Wi-Fi signal strength measurements. This solution achieved a mean error of 2.25 m in performed tests. When combined with other sensors using sensor fusion, Wi-Fi can be used to provide coverage of a building to provide accurate tracking of industrial vehicles. Table 2.1 summarizes the above-mentioned IPSs according to the technologies/sensors, the positioning techniques, their relative cost and mean error.

Table 2.1: Comparison of different systems for localization of vehicles in indoor environments.

IPS	**Technology / Sensor(s)**	**Positioning Technique**	**Cost**	**Mean Error (m)**
[75]	UWB +GPS	Particle Filter	High	0.16
[77]	LiDAR + odometer	AMCL + DFT + scan matching	High	0.013
[43]	LiDAR + magnetometer	Magnetic Fingerprinting + laser scan matching	High	1.43
[66]	RFID + odometer	Similarity model sensor fusion	Medium	0.14
[80]	Wi-Fi	Wi-Fi Fingerprinting	Low	2.25

AMCL – Adaptive Monte Carlo Localization; **DFT** – Discrete Fourier Transform.

2.4 Methods to Build and Maintain Radio Maps

In order to understand how a self-healing radio maps solution can be developed using industrial vehicles, first it is necessary to study how radio maps are built and maintained. Building a radio map for Wi-Fi fingerprinting is a laborious and time-consuming task because it requires the mapping of RPs and the collection of Wi-Fi samples at each RP. The complexity of this task increases proportionally to the size of the building and the grid size that defines the distance between neighbor RPs. For instance, if the grid size is denser, more RPs are surveyed so there will be a higher cost and the accuracy is expected to increase, hence there is usually a trade-off between cost and accuracy when deciding the grid size of the radio map. Once finished, the radio map represents a snapshot of the building's radio environment at that time. With

time, the radio map becomes outdated due to many reasons, such as propagation effects, changes in the indoor layout of the building, moving furniture, or the addition or removal of APs. Therefore, the radio map needs to be periodically updated, leading to additional costs in terms of time and human resources. The standard practice in Wi-Fi fingerprinting is to manually perform these tasks, however, it is not practical. To avoid manually building or maintenance of the radio map, the following approaches have been explored to simplify this process:

- collaborative approaches - users actively or passively contribute with samples to the radio map, hence it depends on the users' movement and, in some cases, on their feedback in order to be effective;
- SLAM - the system locates itself whilst collecting Signals of OPportunity (SOP), including Wi-Fi samples, to map the radio environment;
- interpolation - using Wi-Fi monitors that collect Wi-Fi samples in a few known locations, then using an interpolation technique to estimate APs' signal levels in all RPs.

2.4.1 Collaborative Approaches

The idea of collaborative radio maps where users independently contribute with Wi-Fi samples to build a joint radio map was explored in [21, 81, 82, 83]. There are different approaches to collaborative radio maps: (i) users explicitly and manually report their position to the system while collecting Wi-Fi samples; (ii) users collect data passively (e.g., Wi-Fi and motion data), which can be used to estimate PDR trajectories and, subsequently, annotate the collected Wi-Fi samples; (iii) the system uses an opportunistic sensing approach and, from time to time, prompts the user to report whether they are in the estimated position. Since these solutions depend on user participation, they raise some challenges, for instance, users may choose not to share their information or indicate a wrong position and they may not visit all areas of the building (leaving those areas unmapped).

Kim *et al.* [81] proposed an autonomous solution for the construction of radio maps as well as the localization of users. Inertial data is automatically collected by users on their daily routine, without needing to survey the entire building. The system progressively builds the radio map as more users contribute with data. The position of users is obtained by merging PDR with Wi-Fi fingerprinting. The average positioning

error obtained in experiments is 6.9 m for a building with 60x66 m.

Both works in [21, 82] use Wi-Fi fingerprinting systems that take user feedback to improve radio maps. These systems start with a baseline Wi-Fi fingerprinting solution which already has an existing radio map, but users contribute with new Wi-Fi samples collected automatically by a smartphone. These systems also use users' feedback to build the radio map. Taking the users' input has its shortcomings, for instance, the system can be hindered by malicious actors that feed the system with bad data. The systems in [21, 82] were designed for pedestrians, hence they depend on the motion model of a pedestrian, which cannot be applied to industrial vehicles. In the proposed approach in this thesis, a proper motion model is considered for the vehicles, which are tracked with motion sensors (an encoder and an IMU) for improved accuracy.

2.4.2 SLAM Approaches

Some works explore mobile robots to perform SLAM of its surroundings [42, 84, 85, 86]. SLAM systems are usually equipped with sensors to map the building, i.e., sonar, LiDAR or camera to detect indoor features such as walls, obstacles, or other elements. With the ability to locate themselves while collecting SOP, they may be used for creating the radio map of the building, however, they are expensive due to the sensors they use and usually require complex configuration. In [42] a mobile robot equipped with sonars, a wheel encoder and a depth camera builds a radio map by locating itself with a two-stage process. Initially, it performs obstacle-avoidance-based navigation and odometry-based correction for bearing angles. Then, it uses depth-camera images (RGB-D) for SLAM. Experiments with Wi-Fi fingerprinting showed a mean error of 5.2 m, in a corridor 54 m long. In [85], a robot equipped with several sensors (LiDAR, magnetometer, light sensor, smartphone (Wi-Fi)) generates an indoor grid map while simultaneously collecting signals from Wi-Fi interfaces and other sensors, allowing to build and keep the radio map up to date. The LiDAR allows to map the space using an Improved Maximum Likelihood Estimated (IMLE) scan-matching algorithm, while collecting data from other sensors, including Wi-Fi interfaces. Since Wi-Fi samples are obtained while the robot is moving, signal strength values of Wi-Fi samples collected within a 1 m radius of a RP are averaged.

The SLAM approach described in [86] explores smartphones to collaboratively localize and construct the radio map of the building. The system merges tracks from different users, performs loop closure detection (when users pass by a previously visited location), and optimizes a graph to generate an accurate

radio map. Two PDR methods have been employed, one based on a step counter (using the smartphone's IMU) and another based on Project Tango (smartphone with additional sensors for high accuracy tracking). The advantages of this approach are that it does not require any knowledge of the map and locations of the APs. Experimental results in an area of 130x70 m have shown a positioning accuracy of 0.6 m and 4.76 m for Tango-based PDR and step counter-based PDR, respectively.

2.4.3 Interpolation Methods

Other alternatives eliminate the need to build and maintain the radio map through interpolation (estimation of signal strength values), for example, using RBF, Kriging, Gaussian Process Latent Variable Models (GP-LVM), Inverse Distance Weighted (IDW), Voronoi tesselation, Neural Networks, Model Tree, or Log-Distance Path Loss (LDPL). RBF are a class of artificial neural networks that were explored in [87, 88, 89] to interpolate RSS values of a radio map from observations at known RPs. Generally, RBF networks have an input layer, a hidden layer with radial basis functions and an output layer. RSS values can be estimated using RBF without knowing the position of APs and the indoor layout such as the walls and building materials, hence it can be useful especially in situations where the building's floor plan is not available and there are many APs.

The Kriging method, used in [90, 91, 92], interpolates the RSS values at a given position by creating a spatial structure that can be represented by a mathematical function to provide continuous values. The Kriging method assumes that the RSS values from neighboring locations are more correlated than the RSS values from locations that are farther away, hence it considers that a spatial random variable is spatially dependent. The semivariogram between the RP and the unobserved points to be estimated is necessary to estimate the RSS values. With only a few calibration points, the Kriging method can create a complete radio map of the building. This method shows good interpolation results because, theoretically, the estimation errors show minimum variance.

A different approach to build radio maps is explored in [6], where a motion dynamics model is combined with GP-LVM to reconstruct the radio map from a set of RSS values. Gaussian processes are used to determine the model of each AP present in the building, allowing to interpolate the radio map of the building. A user collects data to be used as training data for the GP-LVM algorithm, which then builds the

radio map of the environment. Performed experiments using the estimated radio map achieved a mean error of 3.97 m.

The IDW method, explored in [93], follows a deterministic model to interpolate unknown points. RSS values can be estimated for unobserved locations by calculating a weighted average of the RSS values available at the known locations. IDW uses spatial autocorrelation that gives more emphasis to values from points that are closer, while values from points farther away have less effect. Since the IDW method only considers the distance when interpolating RSS values, it will not consider the indoor layout of the building, i.e., the walls, the doors, building materials, etc. Therefore, it might have errors in the interpolated RSS values. The Kriging method provides a more robust model in relation to the IDW method [94], but is more complex to implement. Conversely, the IDW method is simpler to implement but has larger interpolation errors.

An interpolation method based on Voronoi tessellation was introduced in [95]. It refines the propagation model for each cell of the target area tessellated by a high-order Voronoi diagram. Therefore, it accounts for the signal fading caused by walls and obstacles more accurately. This method comprises two phases. In the first, the positions of APs are estimated using the LDPL model, then in the second phase, signal fading parameters of the APs are estimated for each Voronoi cell. Obtained results showed that this interpolation method outperformed RBF [87] and IDW [96]. Unfortunately, this method still depends on the manual collection of RSS values at known positions.

The solution proposed in [97] explores neural networks to adapt to changes when they are detected in the radio environment. It is not a fully autonomous solution because it depends on a static radio map that is initially created by collecting Wi-Fi samples in many RPs. Then, Wi-Fi samples are manually collected at a few calibration points (much less than the number of RPs). This paper suggests the hypothesis that there is a relationship between the RSS values detected in the RPs and the calibration points, therefore it uses a neural network to solve this problem. First, the neural network is trained with the static radio map because it contains the RSS values at all RPs. Then, it uses the RSS values obtained at each calibration point to estimate the RSS values for all RPs, hence updating the radio map. Experimental results in a WLAN indoor environment showed that this solution improves the positioning accuracy within 2 m from 68% to 75%. Even though this solution can adapt to the RSS variations from the detected APs, it has a drawback which is the need to manually collect Wi-Fi samples at the calibration points.

Model trees are explored in [98]. It starts with an already created radio map and uses RF receivers at known positions to estimate the updated radio map for all RPs. This method uses an already created radio map to learn the functional relationship in the RSS values between a mobile device and the RSS values obtained by the RF receivers. A model tree is used to estimate the RSS values at the mobile device based on the RSS values collected by the RF receivers. Consequently, it allows adapting to the dynamic changes in the radio environment due to the continuous operation of the RF receivers.

Various studies [1, 99, 100] focus on LDPL-based models to estimate a Wi-Fi radio map. In free space, where there is a direct line-of-sight between the transmitter and receiver, the LDPL model is defined as:

$$PL = PL_0 + 10\eta \log_{10}\left(\frac{d}{d_0}\right) + \chi \qquad [\text{dB}] \tag{2.4.1}$$

where PL represents the total path loss measured in dB, PL_0 is the path loss at reference distance d_0, d is the distance between transmitter and receiver, η is the path loss exponent and χ is a zero-mean Gaussian distributed random variable (in dB).

As the radio signal propagates in indoor environments, it suffers power losses that are not linear with distance due to the presence of people, obstacles and other elements that affect the propagation of Wi-Fi signals, consequently resulting in an attenuated signal. The attenuation of a signal is a factor of distance and also of the obstacles, still or moving, that can influence the propagation of these signals causing the absorption, reflection, diffraction and scattering. Since most times there is not a line-of-sight between the transmitter and receiver, the models used to interpolate radio signals need to account for the effects that impair the Wi-Fi signals. In order to overcome these, some systems described in the literature [1, 99, 100] depend on the floor plan of the building as well as the position of APs in order to take into consideration the attenuation that the signal suffers between the AP's position and the receiver.

In RADAR [1], the wall attenuation factor model considers the attenuation effects from walls along the direct path between the transmitter (AP) and receiver on the same floor. RADAR achieves an average positioning error of 4.3 m using an interpolated radio map by knowing the position of three APs and the floor plan of the building floor.

Other systems such as the probability model described in [60] and the multi-wall-and-floor model in [101], depend on site-specific models that account site-specific parameters such as building materials,

geometrics and thickness. These models may be accurate, but they require extensive measurements and detailed material characteristics and geometry properties in order to accurately represent the building.

A deeper analysis of some interpolation methods is made in Appendix C, which also includes the mathematical models of these methods and an implementation of the RBF method in a real-world experiment.

2.4.4 Other Methods

A combination of SLAM and propagation models is used in [102] to automatically build the radio map of an industrial space. Initially, a 3D force-directed graph is built from Wi-Fi samples collected by fixed Wi-Fi monitors. Then, the graph estimates APs' positions which, then allow to model the radio environment using a mobile unit equipped with Wi-Fi and motion sensors. As a result, the radio map is automatically built without calibration or knowledge about the space. Since this method estimates APs' positions, it is prone to errors which can affect the positioning accuracy of the radio map. In addition, it requires the installation of Wi-Fi monitors, which can be challenging due to the number of beacons and the locations where they have to be installed.

Pallas [103] relies on Wi-Fi monitors and crowdsourcing to automatically create the radio map and keep it up to date. Through the knowledge of the indoor floor plan and the location of the Wi-Fi monitors, Pallas statistically maps the collected RSS traces to specific indoor pathways. Unlike other systems based on fingerprinting, Pallas relies on passive Wi-Fi samples, which are signal strength measurements from deployed Wi-Fi monitors when the signal-emitting devices are at different locations. Consequently, it depends on the identification of passive Wi-Fi landmarks within the collected RSS traces in order to map them into Wi-Fi samples, which are used for passive localization. Since RSS traces are collected by user's smartphones as they move through the indoor area, the system needs several active users in order to have most of the indoor environment properly mapped with passive Wi-Fi samples. The Pallas system was evaluated in an office floor with 1710 m^2 and in an open canteen environment with 1450 m^2, achieving an average localization error of 2.6 m and 3.5 m, respectively.

The EZ system [104], does not require calibration of the indoor environment and does not need any knowledge regarding the APs location and physical layout of the building. However, it requires that the devices inside the building occasionally report GPS coordinates that might have been obtained when the

device was near an entrance or a window. The EZ system uses an algorithm that locates its users by computing the distances between the devices and the APs using the LDPL propagation model. A Wi-Fi fingerprint can be referenced to a location when the device has a GPS signal. All devices send Wi-Fi fingerprints to the server, some are obtained in known locations, where GPS is available, and others are obtained in unknown locations, where GPS is not available. Once the server gathers the fingerprints from the devices, it has to solve a set of LDPL equations in order to obtain the locations of the APs. A large number of APs leads to an increase in the number of equations, and therefore, it takes more time to solve them. To update the model of the indoor radio environment, this algorithm needs to be run again. It can take a few minutes to several hours depending on the size of the problem. In the two experiments that were carried, the obtained median error is 2 m compared with 2-3 m from RADAR [1], however, RADAR required a time-consuming calibration process. The dependence on the GPS signal is a disadvantage of the EZ system, especially because there are many buildings where the GPS signals are not available.

In [105], Atia proposed a dynamic online-calibrated indoor positioning system based on WLANs. The system automatically creates and updates the radio map without the need to perform offline surveys. A wireless monitor device is placed beside each installed AP to send information to a server, every second. Each wireless monitor sends the AP's location as well as the Media Access Control (MAC) address and RSSI of all detected APs (power profiles). The APs' power profiles information is gathered in the server and processed using log-distance models and Bayesian modelling with Gaussian Regression. The processed information constitutes the created radio map, which comprises the estimated power profiles of all APs. The radio map remains updated because the wireless monitors are constantly sending updated information regarding the radio environment. When it receives a Wi-Fi sample from the mobile device, the server estimates the location of the mobile device using the k-NN algorithm. To test this system, two different experiments were performed in buildings with 375 m^2 and 1000 m^2, respectively. Four APs were used in both experiments and the results show that the estimated radio map provides a 2-3 m accuracy which is very similar to the accuracy found in other fingerprinting-based systems that use an manually created radio map. This system has several advantages, namely, it achieves the same accuracy as other Wi-Fi-based IPSs, it does not need an offline survey to create the radio map, and it does not need to know the floor plan of the building or have any previous knowledge of the building. However, this solution also presents some disadvantages, specifically, it requires high computing power to estimate the radio map and if the

solution is not properly integrated into the APs' firmware, it requires the installation of a wireless monitor for each AP.

2.5 Summary

In this chapter, the most important concepts of indoor positioning were reviewed, including signal propagation, impairments of radio signals indoors, and the techniques to estimate the position, with a particular emphasis in Wi-Fi-based positioning systems. Some specifications and performance metrics were described and a distinction was made between positioning accuracy and positioning precision.

An analysis of existing IPSs for indoor vehicles was made. The main finding of this review has shown that sensory fusion techniques are often adopted to achieve improved accuracy and that more expensive technologies such as UWB or LiDAR lead to better performance. Alternative technologies, such as Wi-Fi, take advantage of the existing WLAN infrastructure, thus being lower cost and simpler to deploy. As a disadvantage, it has lower accuracy which can be overcome using additional sensors installed in the vehicles. The analysed systems do not report the performance degradation over time, i.e., whether the system is reliable after being deployed over several months, or longer.

A review of the methods to build and maintain radio maps was also made. It focused on solutions that simplify the building/maintenance of radio maps, using different approaches, e.g., collaborative methods that depend on users' participation, SLAM-based methods that locate and map the space using sensors such as LiDARs, and interpolation methods that require the collection of data in a few points and interpolate signal levels for the entire space. The main disadvantage of interpolation-based methods is that they provide approximations of signal strength levels, that rarely account for signal effects like reflection, refraction and multipath. Conversely, in SLAM and collaborative-based methods, Wi-Fi samples are collected on-site, hence they are more reliable and represent the radio environment more accurately. Collaborative methods take advantage of existing users, with no additional hardware, which is necessary for SLAM and interpolation-based methods. However, users can feed the system with malicious feedback to hinder the system. Interpolation methods usually start from a simple radio map with just a few Wi-Fi samples or a set of Wi-Fi monitors that autonomously collect data and implement an interpolation technique to estimate RSS values. On the one hand, the simplest interpolation methods are less accurate since they

use less information. On the other hand, complex interpolation methods are more accurate but depend on information regarding the AP's positions, the floor plan, and in some cases, the building materials. Obtaining this information is not possible most times.

Despite having numerous ways of building and updating radio maps, there is a lack of studies that investigate how radio maps degrade over time, and how it affects the positioning performance. The following chapter focuses on a long-term evaluation of the radio map degradation in a building. This study comes as a preliminary analysis of the events that cause radio map degradation, which is an important characterization of the problem before introducing the self-healing radio map solution.

Chapter 3

Radio Map Degradation

One of the most common assumptions regarding IPSs based on Wi-Fi fingerprinting is that the radio map becomes outdated and has to be updated to maintain the positioning performance [106, 107]. The problem with the radio map is that it represents a snapshot of the radio environment at the time when Wi-Fi samples were collected, and it becomes outdated after some time and needs to be maintained. As stated in the literature [1, 22, 108, 109], some factors associated with radio map degradation are: the propagation of radio signals indoors (reflection, scattering, multipath, etc.); modifications in the indoor layout (moving furniture or other objects); the removal/displacement/addition of APs; and the presence/absence of people and their Wi-Fi enabled devices, including Wi-Fi hotspots that often appear, change position, and disappear from the building.

Radio map maintenance can be accomplished by performing periodic manual site surveys, or by exploring methods, as described in Section 2.4. Despite those ways of updating the radio map (manually or automatically), selecting the appropriate timing for the update is still an open issue. Detecting when a radio map has actually degraded significantly –requiring to be updated in order to provide reliable indoor positioning– can be considered a major challenge.

This chapter describes an empirical study based on real-world experiments, to evaluate how and why radio maps degrade. Several site surveys were conducted and monitoring devices were deployed to analyze the radio environment of one building over 2+ years. This allowed to identify significant changes/events that caused the degradation of radio maps.

3.1 Approach

To conduct this study, two types of long-term Wi-Fi fingerprinting datasets were collected in the same building. The first is based on a comprehensive manual site survey and the second relies on several fixed Monitoring Devices, also known as anchors, that periodically collect Wi-Fi samples. By having several versions of the building's radio map, it is possible to conduct an empirical analysis of the radio map degradation. Analysing the Wi-Fi infrastructure allows to detect changes regarding the addition/removal of APs and observe how these events may affect the Wi-Fi fingerprinting performance using the collected radio maps. This study also included an analysis of the AP beacon receiving probability, i.e., the number of times an AP is detected over the total number of scans, and determine why some APs are detected more or less often.

To quantify the radio map degradation, two metrics are considered: the positioning error and the Radio Map Degradation Ratio (RMDR). The RMDR metric compares two radio maps and measures how different they are. Obtained results show that positioning performance is significantly better when radio maps are collected on the same day as test data, and although radio map degradation tends to increase over time, it only leads to large positioning errors when significant changes occur in Wi-Fi infrastructure, making previous radio maps outdated.

This work starts by following the same assumption as the research community, i.e., radio maps tend to degrade over time, with the purpose of quantifying the radio map degradation and finding whether the degradation of radio maps is related to events that have caused significant changes in the radio environment. This section focuses on the definition of the radio map and the metrics for measuring radio map degradation.

3.1.1 Radio Map Definition

The radio map is defined as $rm = \{(\rho_1, ws_1), ..., (\rho_m, ws_m)\}$, which is the set of Wi-Fi samples, each associated to the location of the RP where it was collected, represented by ρ. Each Wi-Fi sample, defined as $ws = \{RSSI_1, ..., RSSI_n\}$ represents the set of RSSI values of each AP detected by the Wi-Fi interface.

3.1.2 Radio Map Degradation Ratio

The RMDR metric measures how much a radio map has changed since it was initially collected by comparing it to a more recent one. It calculates the difference in measured RSSI values at each position, assuming that two versions of the radio map share the same set of RPs. Essentially, the RMDR defines the amount of RSSI variation (degradation) per RP per AP. In case there are multiple Wi-Fi samples per RP, they are averaged into one sample with averaged RSSI values of all APs. This allows to reduce noise from consecutive Wi-Fi samples. The RMDR between two radio maps a and b (b older than a) is defined as:

$$RMDR = \frac{\varpi_{a,b}}{N_{RP} \times N_{AP}} \qquad [\text{dBm}] \tag{3.1.1}$$

where N_{RP} represents the number of RPs, N_{AP} represents the number of APs detected in both radio maps, and $\varpi_{a,b}$ represents the dissimilarity between pairs of Wi-Fi samples from radio maps a and b collected at the same RP, defined as:

$$\varpi_{a,b} = \sum_{i=1}^{N_{RP}} \sum_{j=1}^{N_{AP}} \left| RSSI_{ij}^{a} - RSSI_{ij}^{b} \right| \tag{3.1.2}$$

where $RSSI_{ij}$ represents RSSI of AP_j in RP i for radio maps a and b. In order to account for missing APs, whenever an AP is not detected in a Wi-Fi sample its RSSI is assumed to be -120 dBm, which allows to penalize the missing APs. The RMDR is higher when the differences between radio maps are also higher. Conversely, it is lower when both radio maps are similar, hence it is zero when computed for the same radio map, meaning that no degradation occurred.

3.1.3 Wi-Fi Fingerprinting

A complementary metric to evaluate the degradation of radio maps is the positioning error (which is the ultimate goal). Wi-Fi fingerprinting has been used since the beginning of the century when RADAR was introduced [1]. For the estimation algorithm, it was adopted the k-NN method, where an operational Wi-Fi sample is compared against the radio map Wi-Fi samples using a distance function. In this case the

Manhattan distance was used, defined as:

$$d_M(ws^*, ws) = \sum_{i=1}^{N_{AP}} \left| RSSI^i_{ws^*} - RSSI^i_{ws} \right| \tag{3.1.3}$$

where N_{AP} defines the number of detected APs, ws^* and ws represent the operational and radio map Wi-Fi samples, respectively. If an AP is missing in either Wi-Fi sample, a default RSSI of -120 dBm is assumed. After computing the Manhattan distance between the operational Wi-Fi sample and all the radio map Wi-Fi samples, the estimated position is computed using k-NN, by obtaining the centroid of the k most similar radio map Wi-Fi samples:

$$\rho_{wifi} = \frac{\sum_{i=1}^{k} \rho_i(x, y)}{k} \tag{3.1.4}$$

where ρ_i represents the position of a ws.

3.2 Long-term Wi-Fi Data Collection

In order to evaluate the degradation of radio maps, two procedures were adopted to collect Wi-Fi samples over time. The first is based on a site survey, where Wi-Fi samples were manually collected in the building at a set of RPs. The second is based on Monitoring Devices, also known as anchors, which were installed in known locations continuously collecting Wi-Fi samples. The manual site survey method has the advantage of containing Wi-Fi samples from more reference locations, thus being a more dense radio map. The method based on Monitoring Devices has the advantage of being able to autonomously and continuously collect Wi-Fi samples without human intervention for a long time.

3.2.1 Experiments Setup

Experiments were conducted at Departamento de Sistemas de Informação (DSI) located in the University of Minho's Azurém Campus (Figure 3.1). The building measures 106 $\times$ 19 m and has several office rooms, laboratories, and classrooms. It is used daily by a couple of hundred people to attend classes and to conduct research activities. Figure 3.2 depicts the floor plan of the building, the test points (RPs where Wi-

Fi samples were collected in the site survey), and the Monitoring Devices (used for continuous collection of Wi-Fi samples). The DSI building is equipped with multiple Wi-Fi APs to ensure that all users of the building can access the internet efficiently. These APs are easily identified in Wi-Fi scans due to their SSID being the same on the entire campus, however many other networks are also found in Wi-Fi scans. These networks are usually associated with printers, mobile APs, or other APs installed in laboratories and office rooms.

Figure 3.1: School of Engineering at University of Minho, DSI is on the first floor of the right side of the building.

3.2.2 Manual Site Survey

On a period over 2 years, 12 datasets were obtained using a Raspberry Pi 3 B+. Site surveys in which datasets were obtained are represented in blue in Figure 3.8. Each dataset is composed of 20 Wi-Fi samples at each testing point (see Figure 3.2) and can be seen as a different version of the building's radio map. Figure 3.3 (a) depicts one of the manual site surveys at DSI using a manually pushed trolley to collect the data. Unfortunately, due to the pandemic, it was not feasible to perform the manual site surveys during most of 2020.

Although it is a time-consuming and laborious task, the datasets resulting from this work can be directly compared to evaluate the radio map degradation. Since these datasets contain samples from a wider set

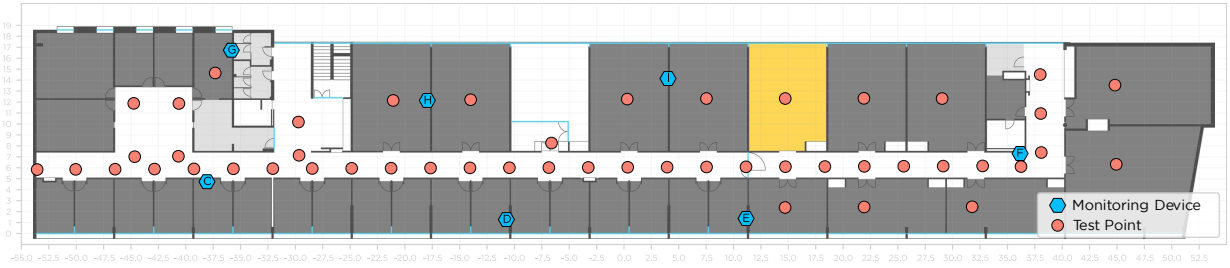


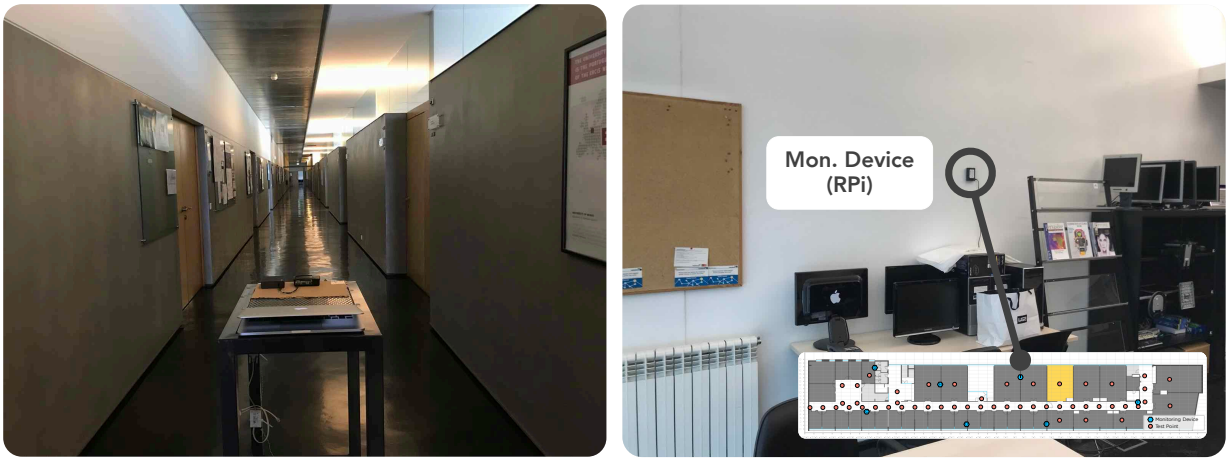
Figure 3.2: DSI building: monitoring devices (blue hexagons) and test points (salmon circles).

of test points (a total of 49), they better represent the building's radio environment than the datasets obtained with Monitoring Devices (only 7 in total). In order to enrich these datasets, in addition to the manually surveyed points, Wi-Fi samples from Monitoring Devices collected around the same time were also considered. In summary, 12 datasets were collected at the following dates: 2019-02-19; 2019-03-25; 2019-06-11; 2019-07-03; 2019-07-29; 2019-09-11; 2019-10-09; 2019-11-13; 2019-12-11; 2020-01-15; 2020-02-19; 2021-04-23. In each dataset contains 1120 Wi-Fi samples collected across 56 distinct positions.

3.2.3 Monitoring Devices Data

Although the installation of infrastructure to collect data tends to be expensive, the Raspberry Pi 3B+ was chosen to implement the Monitoring Device. The main advantages of this device are that it is low-cost, it already includes 2.4 and 5 GHz Wi-Fi interfaces (802.11b/g/n/ac), it is simple to configure and deploy, and it is based on Linux which facilitates the development of software modules. A Java program was developed to collect a new Wi-Fi sample every 60 seconds. In order to avoid getting outdated signal strength values, the program performs two scans and considers the last one. This program also sends Wi-Fi samples to a centralized server that gathers data from all Monitoring Devices. It also saves data locally into an SQLite database.

After testing the data collection program, seven Monitoring Devices were deployed in the DSI building (Figure 3.3 (b) shows one of the deployed Monitoring Devices) on 2019-02-19 and started collecting data on that day. Since the Monitoring Devices are continuously collecting Wi-Fi samples, one can easily obtain the radio map of the building based on Monitoring Device data, at any time period. Although it has a low number of RPs (only 7) for the size of the building considered, this radio map can be used with Wi-



(a) (b)

Figure 3.3: Long-term data collection at DSI: (a) manual site survey; (b) Monitoring Device installed in one of the rooms.

Fi fingerprinting as a metric to evaluate if the positioning performance degrades when the radio map is older. Monitoring Device data was used to create radio maps in two ways: to complement the manual site surveys, and to generate radio maps integrating Monitoring Device data only. Blue and yellow marks in Figure 3.8 represent dates when these data were obtained. More than 7 million Wi-Fi samples were collected from all Monitoring Devices since they started collecting data.

Appendix A documents the solution that was implemented to enable the long-term data collection. It explains how data from Monitoring Devices is gathered and how it monitors whether Monitoring Devices are collecting data or not.

3.3 Radio Environment Over Time

The continuous monitoring provided by Monitoring Devices allows to observe variations in the radio environment over time. Figure 3.4 shows the number of APs detected in the building according to their frequency band (2.4 or 5 GHz).

The Wi-Fi radio spectrum is crowded with many APs in both bands, but with a larger number of APs in the 2.4 GHz band. A total of 4865 APs were detected over the data collection period. This high number of APs detected in the building means that there is interference due to the overlapping of neighboring

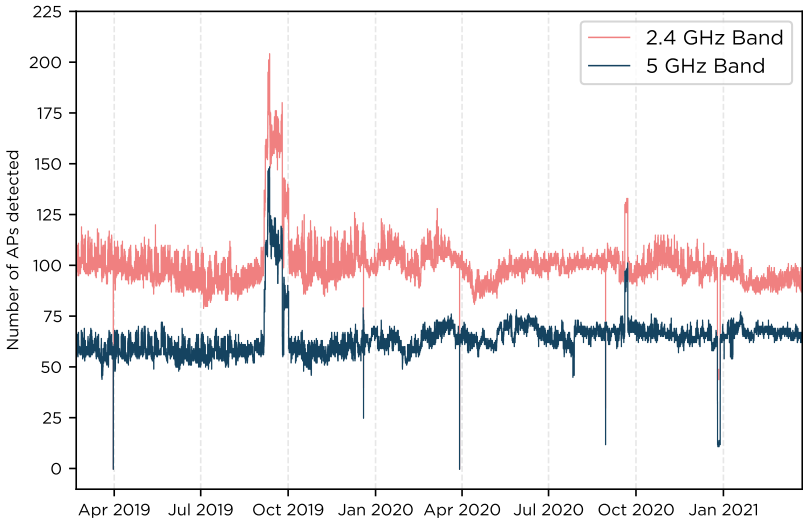


Figure 3.4: APs detected by Monitoring Devices over time.

channels, especially in the 2.4 GHz band.

Figure 3.5 shows all visible APs, since the beginning of this work, in 2019. This plot aggregates data from all Monitoring Devices deployed in the building, hence, it shows all visible APs over the analyzed period. APs are sorted according to the first time of detection. Since some Monitoring Devices are deployed near windows, they can detect APs from neighbor buildings, which are also included in Figure 3.5. Each line, represented in a different color, depicts the period during which an AP was detected. A total of 417 unique APs were detected during this period, considering that each unique MAC address counts as one AP. A total of 4 448 APs that were rarely observed (in less than 1% of Wi-Fi samples) were ignored and are not displayed in the plot.

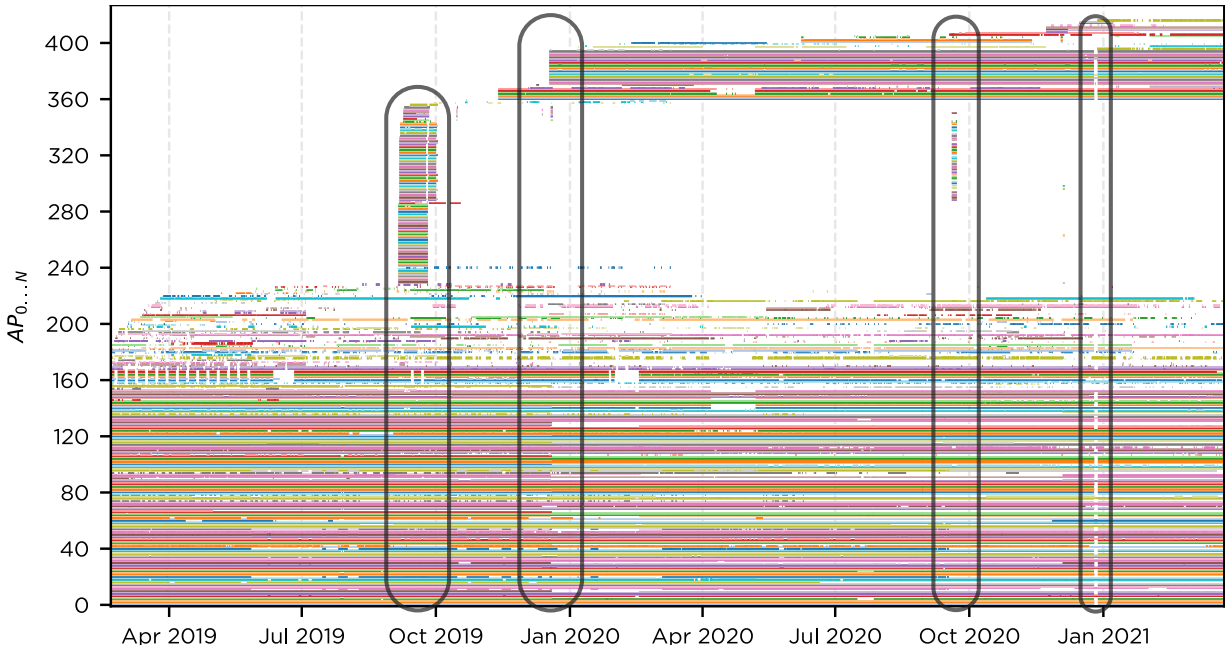


Figure 3.5: APs detected by monitoring devices over time.

Figure 3.5 serves as a tool for analysing changes in the radio environment. It shows that there are some APs that were detected over the entire observation period. Some APs have intermittent behavior (around AP_{160}), being switched on and off several times. There is also the case where APs only show during a short period (in Oct. 2019 and Sept. 2020). Some APs appear at a specific time (Dec. 2019) and remain visible until the end of the data collection. The variations observed in Figures 3.4 and 3.5, demonstrate that the radio environment is highly dynamic, including many short-term variations and several significant changes in the Wi-Fi infrastructure over time. A description of the more significant events (highlighted in grey in Figure 3.5) is presented next.

Sept. 2019 – A set of new APs are detected over one month, then disappear and are not observed again. A deeper analysis through the Wi-Fi samples from that period showed that many new APs, associated with the University's WLAN infrastructure, were detected during this period.

Dec. 2019 – Many APs stopped being detected and several new ones started being detected around the same time. Due to the significant amount of APs that changed in this period, this is a dramatic change in the radio environment.

Sept. 2020 – It is a similar behavior to the one observed in Sept. 2019, but the newly added APs

were detected during a shorter period.

Dec. 2020 – Immediately before Jan. 2021, almost all APs stopped being detected during a couple of days (vertical white space around the time indicating that the vast majority of APs was switched off). This suggests that there was a power outage that caused APs to be down during this period. APs return to normal operation after this period. See Section 3.5 for further details.

Other events – Many other variations and periods in which APs are switched on and off can be found in Figure 3.5, demonstrating that the radio environment is highly dynamic not only in long-term but also in short-term. These short-term variations are probably associated with the building's activity where people use their APs and hotspots, which are easily moved, switched on and off, hence they are not as constant as the APs from the university's WLAN infrastructure.

3.4 Radio Map Degradation Over Time

Whether or not the highly dynamic radio environment has an impact on IPSs is analyzed in this section, where the results regarding the radio map degradation over time, using the mean positioning error and the RMDR metrics, are presented.

3.4.1 Manual Site Surveys Data

Each dataset was used as a radio map and all subsequent datasets (the ones obtained after the radio map) were used as test datasets. This allows the evaluation of the radio map over time since the day it was initially collected. For example, the 2019-12-11, 2020-01-15, 2020-02-19, 2021-04-23 datasets were used as test data when considering the 2019-12-11 dataset as radio map. In the case where the radio map and test datasets are the same (when the radio map is tested on the same day it was created), 5 Wi-Fi samples were selected as test data and the remaining 15 as radio map data, for each testing point. During most of 2020, it was not possible to perform the manual site survey due to COVID.

Figure 3.6 (a) depicts the mean error achieved for each version of the radio map. Each line in the plot represents the times when a radio map was evaluated, over time. As expected, better results are obtained when the radio map is more recent. When the radio map is obtained on the same day as the

test dataset, the results are excellent. The plot shows that the degradation is not incremental over time and it is not guaranteed that time itself is the reason for an increase in the radio map degradation. There are particular events that cause significant degradation of results, for example, the radio map from Sept. 2019 performs poorly in all subsequent tests, and all radio maps created in 2019 had significantly worse positioning results in 2020.

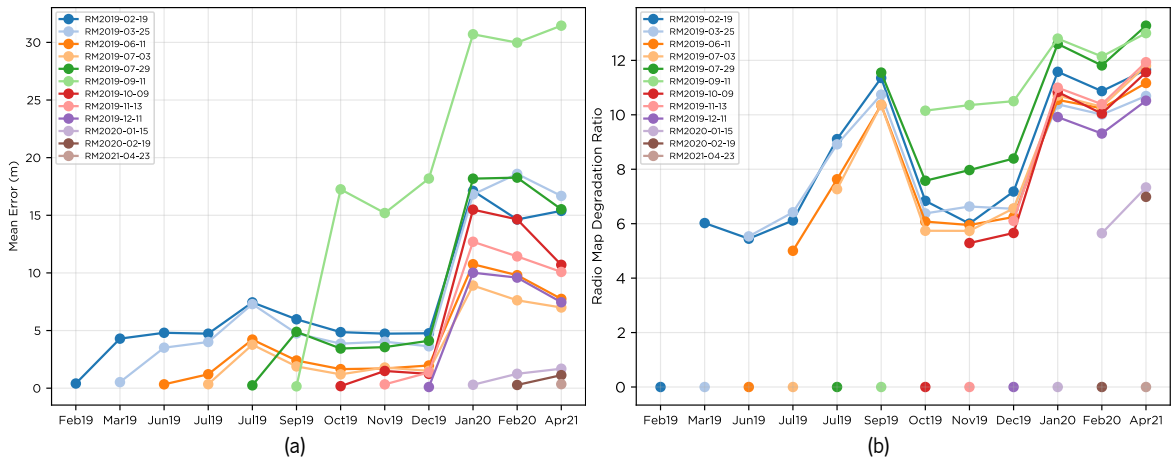


Figure 3.6: Manual site survey metrics: (a) mean error of Wi-Fi fingerprinting; (b) RMDR.

The positioning results obtained with radio maps and testing data collected on the same day are much better than results obtained in more realistic conditions where testing data is collected days or months after the radio map was created. Many results reported in the literature might suffer from this problem, where the results being reported are obtained with radio maps and test data collected in the same day, thus not representing the real performance of the IPSs.

Results show that Sept. 2019 was not a good period to build the radio map because, as described in Section 3.3, during this period many new APs were added and removed just about a month later. Therefore, collecting a radio map before an infrastructure change renders the radio map unusable after the change. All tests with the Sept. 2019 radio map had worse performance than older radio maps. This shows that when the radio map contains a set of APs that are removed, it affects the positioning results based on Wi-Fi fingerprinting since the implemented version assigns a default RSSI to APs that are not detected.

A clear degradation of results in all radio maps collected before Jan. 2020 is observed in 2020. As shown in Figure 3.5, in Dec. 2019, there were many APs that were disconnected, and new ones were

added, causing significant changes in the radio environment. Therefore, all radio maps that were created before Jan. 2020 became outdated, as shown by the worse positioning results in 2020.

Despite a very long time between the last two datasets (2020-02-19 and 2021-04-23), the radio maps that were obtained in 2020 did not have significantly worse results than the 2021 dataset. This is justified by the reduced activity in the building and almost no changes in the Wi-Fi infrastructure during 2020.

The RMDR was evaluated for each radio map version by comparing it with itself and the subsequent ones. Figure 3.6 (b) shows the RMDR of each radio map over time. When compared with itself, the RMDR is zero, as shown in the plot. The RMDR achieved for all radio maps follows a similar behavior as the one observed in Figure 3.6 (a), except for the values observed in 2019-07-29 and 2019-09-11. During this period, the RMDR increased, but it did not translate into significantly worse positioning results. The RMDR rose substantially in radio maps after Dec. 2019, following the same behavior observed in Figure 3.6 (a). This shows that changes that occurred in the radio environment after Dec. 2019 differ from the ones observed in Jul. and Sept. 2019 (where RMDR was higher, but the mean error remained almost the same). Our analysis of the building's Wi-Fi infrastructure (see Section 3.5) shows that these changes are related to distinct events.

In sum, these results demonstrate that better results are achieved when the radio map and test dataset are collected on the same day. When a radio map is tested after some time, there is a degradation in the positioning performance. Radio map degradation is directly related to changes in the Wi-Fi infrastructure, especially when several APs are removed from the building, and not directly correlated simply with the age of the radio map.

3.4.2 Monitoring Devices Data

The same approach as the one in Section 3.4.1 was adopted to evaluate radio map degradation using Wi-Fi fingerprinting and the RMDR metric. As a result of being continuously collecting Wi-Fi samples, it allowed to obtain more datasets in comparison to the manual site surveys. In this experiment, the Monitoring Devices datasets were from the same dates as the manual site survey experiments, to validate whether the results are similar. Monitoring Devices datasets include data from months during which the university was closed or partially opened, due to COVID.

Figure 3.7 (a) depicts the Wi-Fi fingerprinting mean error over time, for each radio map version. These results are in accordance with the ones of Figure 3.6 (a). Again, the Sept. 2019 radio map is only valid during the period when it was collected. It has a large positioning error in all other tests. The event from Dec. 2019 significantly affected the results of radio maps that were built before that. In 2020, new radio maps created after Dec. 2019 have good performance results, demonstrating that no significant changes in the radio environment have occurred during the year. In this period, the mean error is low because only 7 test points (Monitoring Devices positions) were considered, and the building's activity was substantially reduced due to COVID restrictions. Regarding the RMDR, Figure 3.7 (b) shows that radio maps from Monitoring Devices also register high RMDR values in Sept. 2019 and Jan. 2020, having a similar behavior to the results achieved with the site survey radio maps (Figure 3.6 (b)).

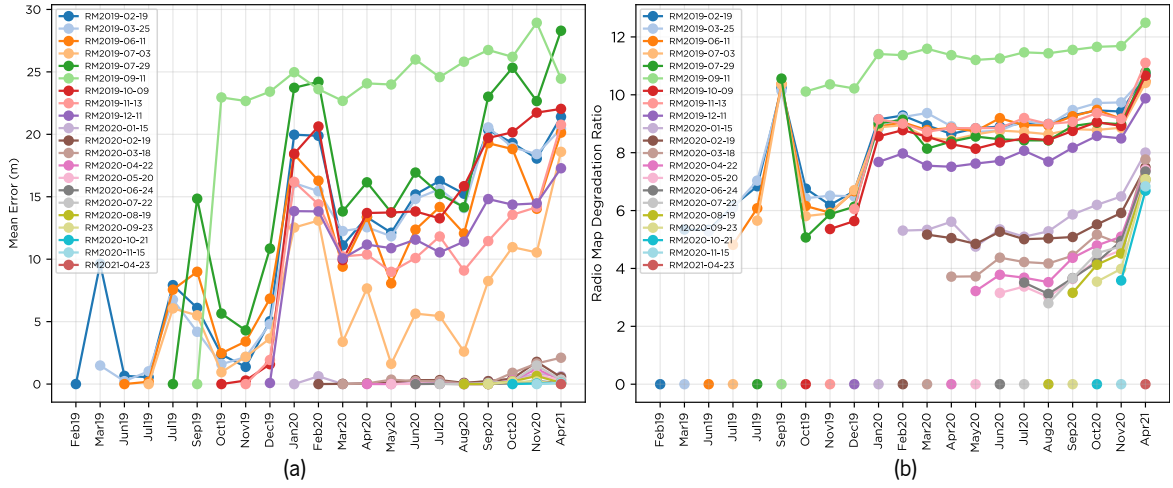


Figure 3.7: Monitoring devices metrics: (a) mean error of Wi-Fi fingerprinting; (b) RMDR.

Although the results are similar to the ones obtained with site surveys, the great advantage of using the Monitoring Devices is that data are autonomously and continuously collected without human intervention which allows to obtain the radio map of the building from any period.

3.5 Events Timeline

In order to establish a cause/effect relationship between the changes in the radio environment (identified in Section 3.3) and the obtained results (in Section 3.4), Figure 3.8 shows the major events that occurred

during the data collection period.

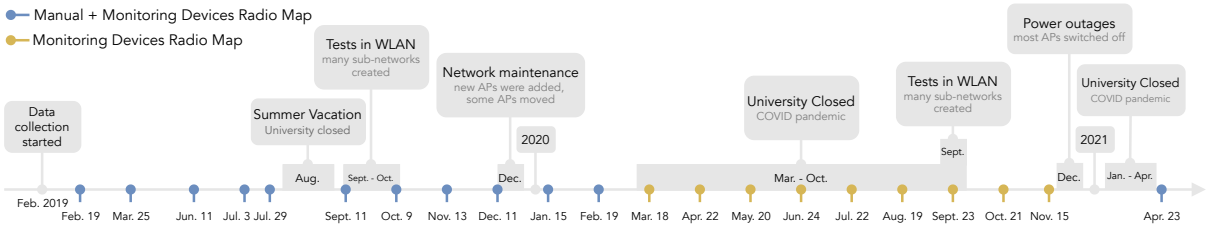


Figure 3.8: Significant events in DSI building over time.

This timeline includes times when there were interventions in the building's Wi-Fi infrastructure, periods during which the university was closed due to COVID, and times (blue and yellow marks) when datasets were obtained. Blue and yellow marks represent times when datasets were collected. Yellow marks represent dates when only Monitoring Device datasets were collected. The last time a manual site survey was conducted was on 2021-04-23, a few days after the university re-opened after being closed due to COVID restrictions. No datasets from Monitoring Devices were obtained between Nov. 2020 and Apr. 2021, because one device stopped working, and was only fixed in Apr. 2021. The main interventions in the building's Wi-Fi infrastructure, which affected the results (see Section 3.4), are described next:

- In Sept. 2019, with the beginning of the school year, the technicians performed several tests in the network, during which they used the existing APs to transmit several new SSIDs which were not previously detected. A similar test was also conducted in Sept. 2020[7].
- The building's WLAN suffered a maintenance intervention in Dec. 2019. New APs with a larger user and traffic capacity were installed in classrooms (right side in Figure 3.2), and some of the replaced APs were moved into the office areas to increase coverage (left side in Figure 3.2).
- In Dec. 2020, a power outage caused problems in the building's power supply and a significant number of APs were switched off for a couple of days. This caused the network to be down, and since it occurred around Christmas time, the technicians were only able to solve this problem after a couple of days. Although the building suffered power outages, the circuit that powers Monitoring Devices seems to be independent of the one that powers the APs, since there are a few APs that are still visible during this period (Figure 3.5), hence Monitoring Devices were still operating and some APs

[7] Information about operations in the Wi-Fi infrastructure was obtained from the University services.

were also working. It is probable that the APs detected during this period are either rogue APs (not from the university's dedicated WLAN), or APs from neighbor buildings, which can be detected by Monitoring Devices. After this issue was solved, the APs returned to their previous state and were detected by Monitoring Devices.

3.6 AP beacon receiving probability

The Wi-Fi samples collected by Monitoring Devices allowed to conduct an analysis of the APs beacon receiving probability. APs are not always detected in Wi-Fi samples due to being far from the receiver device, or due to characteristics of signal propagation. When an AP is missing in a Wi-Fi sample, it is ignored or assigned a default value. Usually, this value serves as a penalization in the localization algorithm for the AP not detected in the Wi-Fi sample.

Researchers tried to characterize the APs RSSI distribution [28, 98, 110] by showing histograms of AP's signal levels over a time window. Although there are some conflicting conclusions about which model better fits RSSI distribution, most authors agree that RSSI histograms resemble a Gaussian distribution in most cases. Other distributions are also considered in the literature due to the diversity of RSSI histograms. Despite analyzing the signal distribution in RSS space, these works miss an important aspect which is the ratio of the times the AP is detected over the total number of Wi-Fi samples analyzed. This aspect is important, especially in solutions that perform AP selection, either to improve positioning performance or to reduce computational effort [111].

In order to characterize APs according to the ratio of times they are detected in Wi-Fi samples, the AP beacon receiving probability was defined as:

$$\Gamma = \frac{n}{N_{ws}} \tag{3.6.1}$$

where n represents the number of times the AP was detected over the total of Wi-Fi samples considered N_{ws}.

To analyze AP beacon receiving probability, over 3 million Wi-Fi samples collected from all Monitoring Devices between 2019-04-01 and 2020-04-01 were considered. Each Monitoring Device detects unique APs, and

for each Monitoring Device, one can obtain the beacon receiving probability of each detected AP in the considered set of Wi-Fi samples. In order to understand whether the APs beacon receiving probability is affected by signal strength, it was also obtained the mean and STDEV for each AP. Each AP detected by Monitoring Devices in the considered period is represented by a point in Figure 3.9 representing its beacon receiving probability, mean RSSI, and RSSI STDEV defined by the color of each point. Only the APs from the university's Wi-Fi infrastructure were considered, in order to ignore mobile APs.

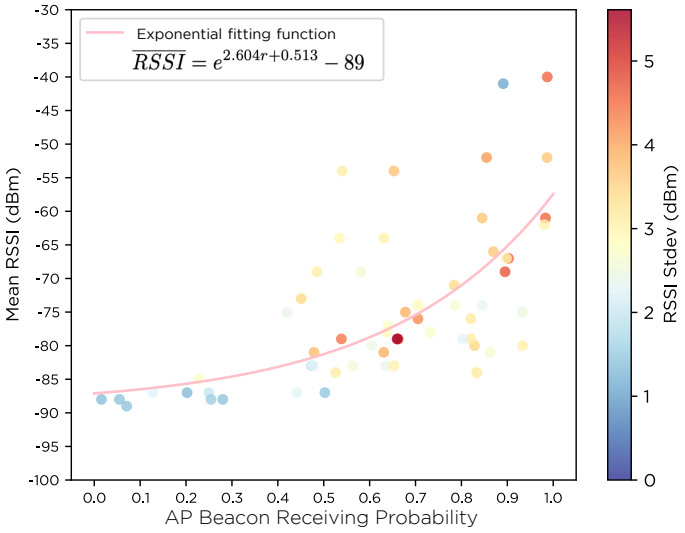


Figure 3.9: APs beacon receiving probability vs. RSSI mean and STDEV of 60 APs.

The APs that have higher beacon receiving probability ($\Gamma > 0.8$) register higher mean RSSI and STDEV. These signals are more variable than the ones for APs with low beacon receiving probability. APs with a lower mean RSSI have a lower beacon receiving probability, this is expected since these signals are closer to the receiver sensitivity, and if there is a variation in the signal propagation, the AP is not detected. An exponential fitting function was applied to model the beacon receiving probability based on the RSSI.

$$\overline{RSSI} = e^{2.604\Gamma+0.513} - 89 \tag{3.6.2}$$

Regarding the RSSI STDEV of APs, a higher concentration of APs is observed whose RSSI STDEV is close to zero when the mean RSSI is low. This is probably related to the number of Wi-Fi samples in which

an AP is detected, being that, when the AP has lower signal strength, it is detected fewer times (low beacon receiving probability), which consequently leads to lower RSSI STDEV. There is no clear tendency regarding the APs whose RSSI STDEV is high, because these APs appear in areas with low and high beacon receiving probability.

This study is important to support works that model the behavior of APs. Many models, such as the LDPL model, ignore the probability of an AP not being detected in a Wi-Fi sample. This empirical study shows that, in order to accurately model the behavior of radio signals at the receiver, propagation models should account for the AP's beacon receiving probability according to the signal strength, being that APs whose signal is stronger are detected more often.

3.7 Summary

This chapter presented an empirical study on the radio maps degradation at an office/classroom building. Long-term Wi-Fi data was collected by performing several manual site surveys and deploying multiple Monitoring Devices that autonomously collected Wi-Fi data.

The long-term radio map degradation was assessed with the positioning error and the proposed RMDR metric. Achieved results showed that, although there is a tendency to increase over time, the radio map degradation is mostly caused by dramatic changes in the Wi-Fi infrastructure. The positioning performance is significantly affected when many APs are removed from the building, making all radio maps created before this change, outdated. Therefore, the radio map should be updated after significant changes occur in the Wi-Fi infrastructure. The RMDR measures how different two radio maps are by comparing existing APs and their signal levels for each RP. The RMDR can detect radio map degradation, but it is not an indicator of worse positioning performance. When just a few APs are added to the Wi-Fi infrastructure, it does not affect the positioning performance, but still leads to an increase of the radio map degradation defined by the RMDR. Also, the positioning results obtained with training and testing data collected on the same day might be misleading by providing unrealistically good performance.

Regarding the APs beacon receiving probability, this research showed that there is a correlation between signal strength and AP beacon receiving probability, i.e., APs whose signal is stronger are detected more often than APs whose signal is weaker. This analysis can contribute to the definition of propagation models,

e.g., based on LDPL, that account for AP beacon receiving probability in the simulated data.

Now that it is clear that radio map degradation can cause significant changes in the positioning performance, it is important to keep the radio map updated. Although the study reported in this chapter was not conducted in an industrial environment, the same behavior in the radio map degradation is also expected when there are Wi-Fi infrastructure changes. Furthermore, industrial environments are more dynamic than academic buildings, since they often suffer interior layout changes to accommodate new production lines or storage areas. This shows that in industrial environments radio maps should be continuously updated.

To achieve that, two methods may be explored: (i) using Monitoring Devices that continuously collect Wi-Fi samples with an interpolation technique to estimate RSSI values in the building; (ii) taking advantage of industrial vehicles to collect Wi-Fi samples as they move through the space.

The Monitoring Device-based approach has two disadvantages. First, additional infrastructure is necessary (installation of Monitoring Devices), which require a dense setup to achieve a higher quality radio map. Second, most interpolation models used to estimate signal strength values are simple and do not consider important aspects that affect signal propagation indoors, for instance, the geometry of the space and its building materials (concrete, glass, wood). There are more complex models that use the floor plan information and building materials, however, designing an accurate model of the space and all indoor elements is a complex task and, on many occasions, it is not possible because the information is not available. Furthermore, many indoor environments are subject to changes in the layout, which also affect signal propagation indoors, thus it is not ideal to use complex models to interpolate signal strength values in dynamic environments. Both challenges can be overcome with the approach that explores industrial vehicles to annotate Wi-Fi samples as they operate. This strategy allows to build and keep the radio map up to date with real Wi-Fi samples of signal strength values obtained in-situ. Before being able to annotate and add Wi-Fi samples to the radio map while operating, it is necessary to locate and track the vehicles accurately. That part will be introduced in the following chapter.

Chapter 4

Indoor Positioning and Tracking of Industrial Vehicles

This chapter presents the solution that was adopted for indoor vehicle tracking, which is divided into two distinct sensor fusion methods, Loose Coupling and Tight Coupling. The chapter starts with an introduction explaining different methods and technologies that can be used for indoor vehicle tracking. Then the sensor fusion solution is explained, with dedicated sections for each method. Finally, these methods are compared against each other in real-world experiments.

4.1 Introduction

Vehicles have an important role in the distribution of materials in industrial environments, thus current and future generation factories (Industry 4.0) can benefit from tracking these vehicles to increase productivity, improve logistics processes and enable device intercommunication. Besides, materials can be tracked along the supply chain by continuously monitoring the position of the vehicles carrying them, improving internal and external logistics. With the rapid development of research in this area [75, 77], there are already autonomous vehicles as well as hybrid vehicles (can be human-operated and also operate autonomously) in factories, which can benefit from positioning and tracking.

The positioning of industrial vehicles can have two distinct applications, namely, continuous tracking and monitoring of the vehicle's position as it operates (main focus of this work), and as a support for navigation of autonomous vehicles [112, 113]. Both applications have similar requirements, including coverage of the entire operating area (factory plant), sub-meter accuracy, low maximum error, availability and high reliability. The reliability of a positioning system is especially important for the localization of vehicles, as it defines if the provided estimates can be trusted. A positioning system with low reliability can provide very accurate estimates as well as estimates with large error, without providing any feedback to whether or not those estimates are accurate. Conversely, a reliable positioning system is able to determine if a position estimate has a high/low error associated, i.e., if the position estimate can be trusted.

Wi-Fi-based positioning systems have been explored [1, 114] to enable the indoor positioning in indoor environments, where GPS signals cannot be properly retrieved due to the attenuation caused by the building. Wi-Fi is present in most buildings hence, this technology can easily offer coverage of a factory plant, but usually provides accuracy with a mean error between 2 and 4 m and presence of sporadic large errors higher than 10 m [114]. Even though Wi-Fi may provide an absolute position, it is not effective for tracking vehicles, due to jumps between consecutive position estimates. Alternatively, vehicles can be tracked using Dead Reckoning which depends on sensors such as IMUs and wheel encoders. DR depends on the knowledge about the initial pose (position and heading), which must be estimated using other methods, and it suffers from cumulative errors that affect IMU sensors, causing drift in the estimated trajectory. Sensor fusion of Wi-Fi with motion sensors allows to benefit from both approaches while minimising their drawbacks.

Traditionally, sensor fusion for positioning can be performed using Loose Coupling or Tight Coupling approaches. In Loose Coupling, sensor fusion is performed at the position level, i.e., positions obtained from distinct estimators are combined into a new estimate. In Tight Coupling, sensor fusion is performed at the measurement level to generate a unique position estimate. Different Bayesian techniques have been explored in the literature about sensor fusion in indoor positioning, namely, the Kalman Filter (KF) [67], the Extended Kalman Filter (EKF) [115], and the Particle Filter (PF) [75, 116, 117]. The KF models the uncertainty to be Gaussian distributed, hence it is suitable for problems where the noise in data tends to be Gaussian and uni-modal. The EKF, a variation of the KF for non-linear problems, is used in GPS and navigation as well as in the positioning of mobile robots [115]. Similarly to the EKF, the Unscented

Kalman Filter (UKF) is used in non-linear problems but has improved performance. The PF is a subset of Monte Carlo methods suitable for non-linear and non-Gaussian problems because it does not assume linearity and Gaussian nature of noise in data. PFs are suitable tools for indoor positioning and tracking since they comprise a set of particles that simultaneously represent many possible positions, and support different sensor types, motion dynamics and noise distributions. They are simple to implement and can take advantage of the building's floor plan to improve position estimates.

Based on the requirements of indoor vehicle tracking, and its unique characteristics, the PF was chosen as the technique to perform sensor fusion of motion sensors and Wi-Fi data. Even though the PF is computationally expensive, it is possible to adjust the number of particles, adapting it to the available computing resources. In this chapter, two sensor fusion approaches (based on Loose Coupling and Tight Coupling) for the indoor tracking of vehicles are proposed.

4.2 Sensors for Vehicle Tracking

Accurate tracking of industrial vehicles can be achieved using specific sensors to measure certain aspects of the vehicle's movement, for instance, the travelled distance and the heading (the orientation in which the vehicle is headed). This information can be explored in DR techniques to infer the trajectory of the vehicle. Although these techniques are reliable in the short term, they are subject to cumulative errors, thus requiring a solution that allows to mitigate these errors. This section includes an analysis of some of these sensors. Typically, IMUs are used for measuring the heading, and the rotary encoders are used for measuring the travelled distance, but there are also alternatives for the rotary encoders.

Indoor positioning for pedestrians based on smartphones has several challenges because the relative position of the device is not known and there is a need to support many models and brands of smartphones that use different sensors that provide different types of data. It is not the case when the purpose of the IPS is to locate vehicles in indoor environments. In these scenarios, the sensors installed on the vehicles can be selected according to the desired specifications and can be installed in a known position on the vehicle.

4.2.1 Heading (Orientation)

IMUs are self-contained systems with tri-axis magnetometers, accelerometers and gyroscopes that measure the magnetic field (μT), acceleration (m·s^{-2}) and angular velocity (rad/s), providing information regarding the relative orientation of the device. An IMU with on-board processing that provides attitude and heading information is called an Attitude and Heading Reference System (AHRS). The representation of the device's orientation may be represented in Euler angles (roll, pitch and yaw) or quaternion. In this context, the vehicle's heading is the angle associated with the yaw axis.

Depending on the IMU, the heading may be absolute or relative. Some IMUs provide absolute heading estimates, which means that the heading is referenced to the true north. Others only provide relative heading estimates using the starting point as the first reference and all heading estimates are relative to that initial orientation.

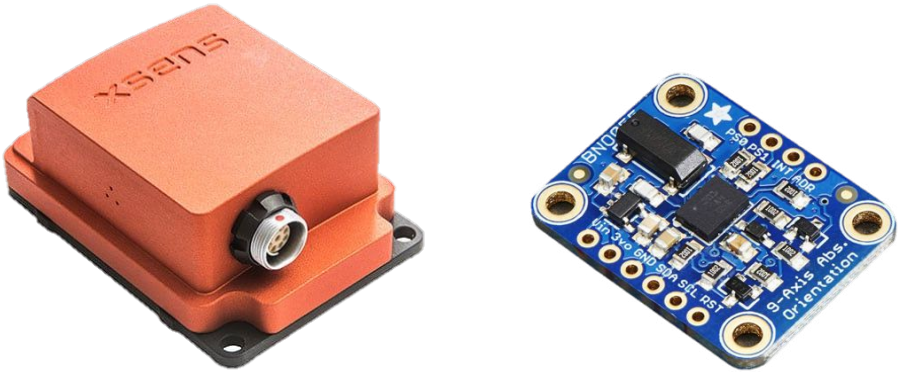
Magnetic disturbances of ferromagnetic elements affect the magnetometer's ability to measure the earth's magnetic field. The presence of large metal objects, metal furniture and other elements are the source of errors in the estimated orientation [118].

IMU sensors suffer from integration drift, small errors in the measurement of angular and linear velocity are integrated into larger errors that accumulate over time since the initial position. Accelerometers and gyroscopes have sensor bias, it is the variation that occurs in the average signal output when there is no movement. Sensor bias varies with time, hence it contributes to drift in the heading. Drift may be reduced by using another type of navigation system to reset the initial orientation/position. Most IMU's specification sheet includes information regarding the drift, namely gyro bias stability, roll/pitch and yaw in degrees per hour.

Orientation filters are responsible for estimating the orientation angles, by merging the accelerometer and magnetometer data with angular velocity from the gyroscope. Examples of the orientation filter are the complementary filter [119], the KF [120] and the general-purpose Bayesian filters, the Mahony [121] filter or the Madgwick [122] filter. A comparison between the Mahony and Madgwick filters is made in [123], revealing that the Madgwick filter provides better orientation estimates. The main advantages of the Madgwick filter are that it has a low error, is computationally inexpensive and has better performance than the Kalman-based algorithms [122]. Since it has a low computational load and is able to operate at low

sampling rates, the Madgwick filter is a good solution for real-time applications that have limited processing power or require high sampling rates.

Industrial-grade IMUs are very expensive because their reliability is highly dependent on the quality of the integrated sensing elements as well as the signal processing pipeline. For instance, the Xsens MTi-300 IMU[8], depicted in Figure 4.1 (a), has a gyro bias stability of 10°/h and costs 2900 €. It can be used to track vehicles or mobile robots, to assist in autonomous navigation or to provide accurate inertial data from wind turbines deployed in remote areas.



(a) (b)

Figure 4.1: IMU sensors: (a) Xsens MTi-300 (Industrial-grade); (b) Adafruit BNO055 (low-cost).

Source (a) https://www.digikey.be/product-detail/en/xsens-technologies-bv/MTI-300-2A8G4/1551-MTI-300-2A8G4-ND/1066747

Source (b) https://www.amazon.com/Adafruit-Absolute-Orientation-Fusion-Breakout/dp/B017PEIGIG

Alternative, low-cost IMUs (e.g., the Adafruit BNO055 shown in Figure 4.1 (b)), are commonly used in educational robot projects because they can be easily integrated with the Arduino or Raspberry Pi platforms. Lower accuracy and higher drift are their main disadvantages in comparison to industrial IMUs. In Section 4.4, a PF is proposed as the solution to help cope with the heading drift and noise from the IMU sensor.

In this work, AHRS IMUs are used because they integrate sensor fusion algorithms to convert raw data from the magnetometer, gyroscope and accelerometer sensors into the orientation of the device. AHRS sensors take advantage of reference vectors (i.e., gravity and the earth's magnetic field) to compensate for the gyroscopes integration drift, resulting in more reliable orientation data (lower drift).

[8]https://www.xsens.com/mti-300

4.2.2 Displacement (Travelled Distance)

In order to properly estimate a position using DR, the heading and the displacement are necessary. The algorithm begins with an initial position and it estimates the following ones every time new heading and displacement samples are obtained. The displacement defines the distance that was travelled since the previous position. In the following, several solutions for measuring the displacement are presented.

The displacement can be measured using rotary encoders attached to one of the vehicle's wheels. Rotary encoders can be absolute or incremental. Absolute encoders allow to read the absolute angle at which the encoder is positioned at, whenever a new sample is obtained. Incremental encoders generate a series of pulses during movement which can be converted into a velocity since the previous sample. When the absolute encoder is powered, it always produces a known position. In contrast, the incremental encoder always starts from zero and acts as a counter for each pulse generated in operation. Usually, encoders are attached to the motor or the wheel's axle, however there are no standard mounts compatible with different types of indoor vehicles. This demands a custom made mount that is expensive and takes time to be designed and manufactured. Thus, it increases the cost of the deployment in the vehicle. A disadvantage of using wheel attached encoders is that they may suffer from wheel drag while the vehicle turns, but it can be compensated with a PF.

Accelerometers are low-cost sensors that can be used to measure displacement by obtaining the speed through the integration of the acceleration in function of time. Since the acceleration is integrated twice, it has high sensitivity to noise and the drift will be high. Even with filtering or calibration, very high errors will be noticed, as shown in [124]. Since accelerometers are present in IMUs one can use them to obtain the heading and displacement of the device.

Optical flow may be used for obtaining odometry information with a camera to extract features in order to estimate the travelled distance. Optical flow is the displacement of pixel values in an image sequence induced by the movement of a camera or the scene observed by it. In [125], optical flow is applied to estimate the velocity of micro aerial vehicles, whilst in [126], optical flow is used as an odometer for mobile robots. Different algorithms may be applied to estimate the travelled distance with optical flow, namely, the Lucas-Kanade algorithm, the Horn-Schunck algorithm and the block matching algorithm. Optical flow depends on optical sensors, such as cameras to capture the frames in which the optical flow algorithms

are applied, therefore the pixel count in a frame has a direct impact on the computational power required for onboard optical flow computation.

In this work, the absolute encoder was selected as the sensor for measuring the travelled distance because it is reliable for measuring the displacement, and it is simple to read and convert the measured angles into a distance value. Figure 4.2 depicts how the angles measured by the absolute encoder can be converted into the distance.

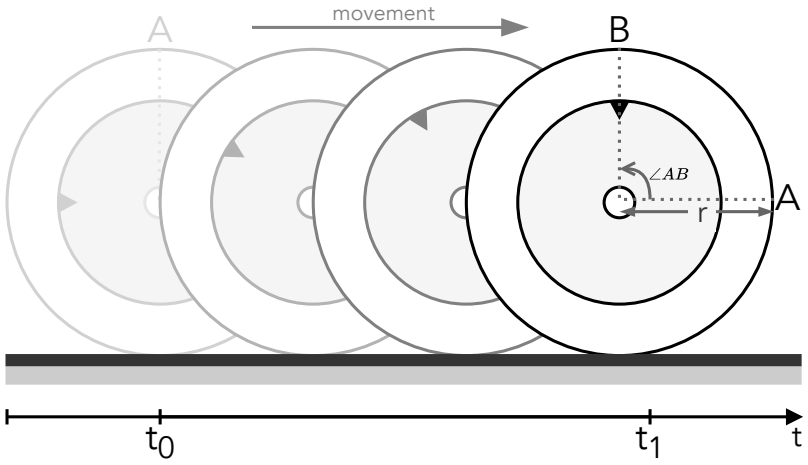


Figure 4.2: Measuring the displacement using an absolute encoder.

First, it is necessary to observe the angle at which the encoder is positioned in two instants (t_0 and t_1). Second, the angle difference between those two observations is measured. Third, based on the measured angle and the wheel radius, one can convert the angle into the displacement, as follows:

$$d = \frac{\angle AB}{360} \times 2\pi r \tag{4.2.1}$$

where d represents the measured displacement in the time interval between t_0 and t_1, $\angle AB$ represents the angle (in degrees) measured in the same time interval, and r represents the wheel radius.

4.3 Particle Filter Sensor Fusion Approaches

Figure 4.3 shows a diagram of the sensor fusion approaches explored for indoor vehicle tracking. The PF is the main algorithm that combines data originating from the Wi-Fi, the encoder and the IMU sensors. DR information is obtained from the movement model of the vehicles, which is based on the motion sensors (encoder and IMU). The information regarding the absolute position of the vehicle is inferred from the Wi-Fi information, considering two distinct approaches, the Loose Coupling and the Tight Coupling approach. Overall, the main difference between these approaches is that the Loose Coupling approach combines a Wi-Fi position estimate with the DR information, while the Tight Coupling approach combines the Wi-Fi similarities with the DR information to estimate the vehicle's pose.

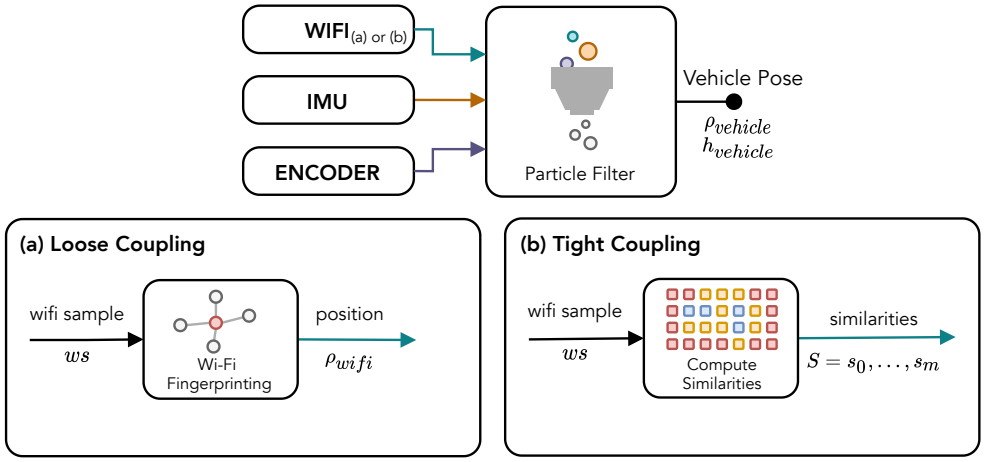


Figure 4.3: Loose Coupling and Tight Coupling sensor fusion approaches.

The Loose Coupling approach (Figure 4.3 (a)), combines a Wi-Fi position estimate with data from other sensors in the fusion algorithm. For instance, a Wi-Fi fingerprinting method based on k-NN first computes the similarity between the Wi-Fi sample (sensor measurement) and the reference samples from the radio map. Then, a position estimate is obtained based on the most similar Wi-Fi samples and it is combined with data from other sensors to produce a more accurate position estimate. Large errors are usual in Wi-Fi-based positioning due to different signal propagation effects, hence this approach has the additional challenge of trying to eliminate the large positioning errors.

The Tight Coupling approach (Figure 4.3 (b)), computes the set of similarities between a new Wi-Fi sample and the radio map, then the set of similarities is combined with data from the motion sensors

to obtain a PF position estimate. In contrast with the Loose Coupling approach, which only considers a small set of radio map samples, this approach considers the entire radio map, therefore it benefits from all available information. Since Tight Coupling does not rely on a Wi-Fi position estimate, it is less influenced by the large errors in Wi-Fi position estimates.

Previous works already explored these techniques for tracking pedestrians, e.g., in [38, 127] (Loose Coupling approach) and in [128] (Tight Coupling approach). Although these approaches are not new, in this work they are applied to industrial vehicle tracking, which explores a different motion model than pedestrians and depends on dedicated sensors to measure the displacement and heading.

4.4 Particle Filter

PFs are based on the idea that a set of particles represents the belief of the vehicle's pose and comprise three iterative steps named predict, update, and resampling. In the predict step, particles states are estimated based on the previous particle state and a motion model, considering the noise models of each sensor. In the update step, particles' weights are updated according to sensor measurements. In the resampling step, particles with lower weights are replaced by new ones that are based on the particles with higher weights. Prior to running these steps, it is necessary to initialize the PF, by creating the particles.

Different assumptions may be considered regarding the initial position and heading of the vehicle, depending on the requirements of the IPS. While some IMUs provide the absolute heading by providing true North-reference yaw, there are others that provide a relative yaw that resets every time the sensor is powered, thus making it challenge to determine the initial heading of the vehicle. Similarly, there are some cases where the initial position of the vehicles is available (e.g., vehicles always start operation from a known location), and other cases where the initial position of the vehicle is unknown because vehicles may start operation from any location of the building.

In this chapter, the following assumptions are made regarding the initial pose of the vehicle: initial position and heading are unknown. Several mechanisms are included in the proposed implementations to overcome this challenge, e.g., Wi-Fi fingerprinting is used to determine an initial position. Even though the initial position may have a potentially large error, it is still better than spreading particles uniformly in the space. To cope with the initial heading being unknown, each particle is assigned with a random heading

offset, a specific attribute unique for each particle, to guarantee that the distribution of particles' headings represents the true heading of the vehicle. As the vehicle starts moving, particles start converging into a position where it is more likely for the vehicle to be at. Different weights are assigned to particles, being that particles that are closer to the vehicle's position have higher weights, conversely, particles further from the vehicle's position have lower weights. Particles further from the true position of the vehicle will be eliminated, and the ones that better represent the vehicle's position and heading will be copied in the resampling method, allowing to estimate the position and heading of the vehicle after a few iterations.

Figure 4.4 represents the main algorithm of the proposed PF, which is the same for the Loose Coupling and Tight Coupling implementations. It starts with the particles' initialization, then waits for samples from sensors (displacement, heading or Wi-Fi). Particles move according to heading and displacement measurements, considering noise. Upon receiving a new Wi-Fi sample, the particles' weights are updated and then the particles are subject to resampling. The vehicle's position and heading are updated after this process. Since the Loose Coupling and Tight Coupling differ in some methods, Figure 4.4 also includes the respective algorithms and equations used in each step. For instance, the update weights method is different for the Loose Coupling (LC) and Tight Coupling (TC), but the motion model is the same for both implementations, hence the same equation is used to update particles' positions and headings.

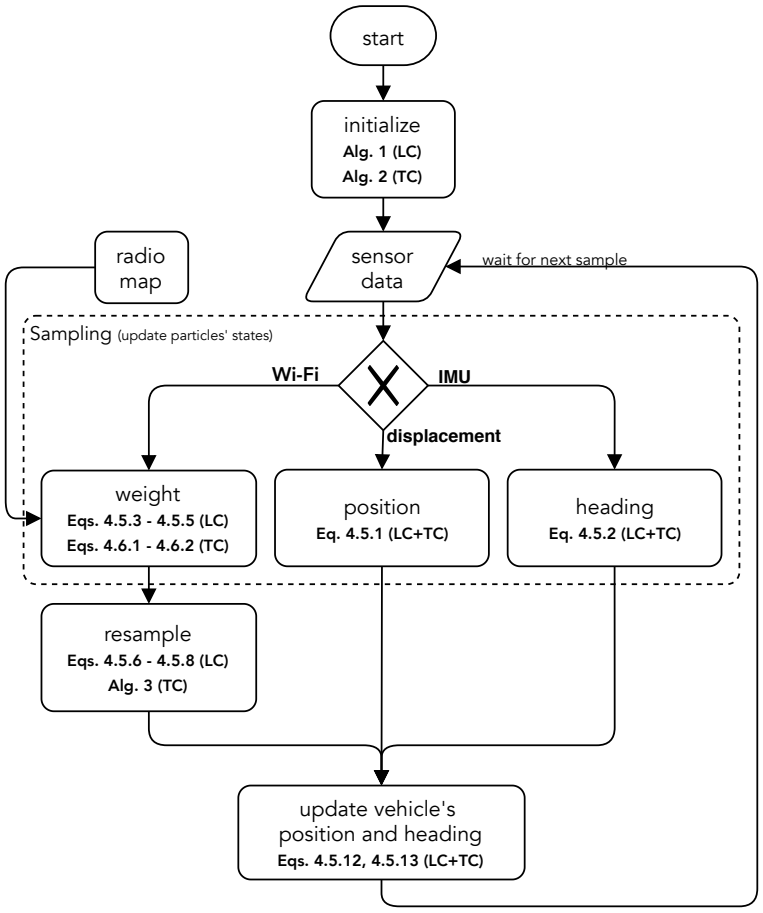


Figure 4.4: Particle filter main algorithm.

The PF prediction and update steps are performed simultaneously when new sensor samples are obtained. The prediction is based on the propagation of particles according to heading and displacement samples, which consider noise modelled after the IMU and encoder, respectively. Therefore, whenever a new heading or displacement sample is collected, the sampling process includes the predict (added noise) and the update (new observations) in the same step (Section 4.5.2). The update step also includes the process to update the particles' weights, which is based on the particle's current state and the latest Wi-Fi sample.

A particle is defined as $p = (w, x, y, z, h, ho)$, where w represents the weight of the particle, x, y, and z represent the position coordinates of the particle, h represents the heading and ho represents the heading offset, necessary to estimate the heading of the vehicle (see Section 4.5.1).

4.5 Loose Coupling Approach

The most straightforward method to combine Wi-Fi information with motion sensors is to explore Wi-Fi fingerprinting to determine an absolute position and combine it with DR information to track the vehicles. This approach follows a loosely coupled configuration where a Wi-Fi position estimate is combined with information from other sensors to determine an improved position provided by the sensor fusion algorithm. As already mentioned, the PF was selected to perform the sensor fusion of sensor data.

In this context, the Wi-Fi position estimate (obtained from Wi-Fi fingerprinting) is necessary to update the particles' weights, assigning higher weights to particles closer to the Wi-Fi position estimate. This information is combined with the motion sensors by updating the particles' positions and headings based on the sensor measurements. The estimated trajectory is corrected as the vehicle moves by eliminating the particles with lower weights in the resampling step. Wi-Fi information has a crucial role in the sensor fusion algorithm as it allows to determine which particles will ultimately be removed based on their weight in the resampling step.

4.5.1 Initialization

Usually, particles are uniformly distributed throughout the navigable space when they are initialized. This approach leads to large errors in initial iterations, as particles take some time before converging into a small area. Two distinct approaches are explored in particles initialization, both aim for initialising particles near the locations where it is more likely for the vehicle to be near. As the initial heading of the vehicle is unknown, each particle has an initial heading offset (ho) randomly set. Once particles start moving, the ones with the correct heading offset will remain alive as they are the ones with higher weights. Hence, after a few iterations the particles converge into the correct heading.

In this approach, particles are uniformly distributed around reference points that are closer to the centroid of the first Wi-Fi position estimates. RPs are selected based on a radius distance between the centroid and each reference point. Algorithm 1 describes how particles are initialized.

Algorithm 1 Initialize particles - Loose Coupling approach.

Input

$WiFi_n$ - number of Wi-Fi position estimates considered in initialization

$\mathcal{N}$ - total number of particles

Output

P - set of particles

procedure Initialize Particles Loose Coupling
  c=centroid of first $WiFi_n$ Wi-Fi position estimates
  RP=list of reference points within a r_{ini} radius of c
  $np = \mathcal{N}/\#(RP)$
  for rp in RP **do**
    for $i = 1$ until np **do**
      $w = 1/\mathcal{N}$
      $(x, y, z) = (rp.x + \text{rand}(0, r), \quad rp.y + \text{rand}(0, r), \quad 1.0)$
      $h = 0$
      $ho = \text{rand}(-\pi, \pi)$
      $p = (w, x, y, z, h, ho)$
      $P = P \cup \{p\}$

4.5.2 Motion Model

Continuous tracking of the vehicle's position is achieved by updating particles' positions and headings using the data provided by the encoder and IMU sensors, which measure the displacement and heading (orientation), respectively.

For each new displacement sample, particles positions are updated as follows[9]:

$$\begin{aligned} x_t &= x_{t-1} + (d_t + n_d) \times \sin(h_t) \\ y_t &= y_{t-1} + (d_t + n_d) \times \cos(h_t) \end{aligned} \tag{4.5.1}$$

where x_{t-1}, y_{t-1} represent the previous position of the particle, d_t represents the displacement (in meters) with respect to the previous reading, h_t represents the heading, and n_d=N(0, σ_d) is a zero-mean Gaussian distributed random variable added to the displacement measurement.

When a new heading sample sample is received from this sensor, the particles' headings are updated

[9]for simplicity, the vehicle is assumed to move in a plane, therefore with constant z.

as follows:

$$h_t = \theta_t + n_\theta + ho \tag{4.5.2}$$

where θ_t represents the heading (yaw) measurement from the sensor at instant t, n_θ=N(0, σ_θ) is a zero-mean Gaussian distributed random variable, and ho represents the heading offset (necessary to estimate the initial heading, as described in Section 4.5.1) of the particle.

4.5.3 Update Weights

The current weight of a particle is updated based on the Euclidean distance between the particle's position and the latest Wi-Fi position estimate. The larger the distance between the particle and the latest Wi-Fi position estimate, the lower its weight will be. The final weight of each particle also depends on its previous value:

$$w_t = w_{t-1} \times (1 - \alpha) + \alpha \times (1 - d_n) \tag{4.5.3}$$

where w_{t-1} represents the previous particle weight, d_n is the normalized distance between the particle and the latest Wi-Fi position estimate, and α is a value between 0 and 1 which represents how much the distance contributes to the particle weight. Conversely, $(1 - \alpha)$ represents how much the previous weight contributes to the particle's weight. The normalized distance, d_n, is given by:

$$d_n = \frac{d - \mathsf{min}(D)}{\mathsf{max}(D) - \mathsf{min}(D)} \tag{4.5.4}$$

where d represents the Euclidean distance between the particle and the latest Wi-Fi position estimate, and D represents the set of distances between all particles and the latest Wi-Fi position estimate.

Warm-up

Once particles are initialized, the PF starts estimating the vehicle's position. During the first iterations, the confidence on the vehicle's estimated position is low because the initial heading is unknown and because the particles take some time to converge into a concentrated cluster. Therefore, during the first iterations,

the importance of Wi-Fi position estimates should be higher when updating the particles' weights. After this period, Wi-Fi position estimates can be given a lower importance in updating the particles' weights. This can be done by adjusting the value of alpha in Equation 4.5.3 over time. It adopts an exponential decay with a decay constant of $1/35$, starting with alpha equal to one and reducing its value during the warm-up period up to a minimum value (α_{min}), as follows:

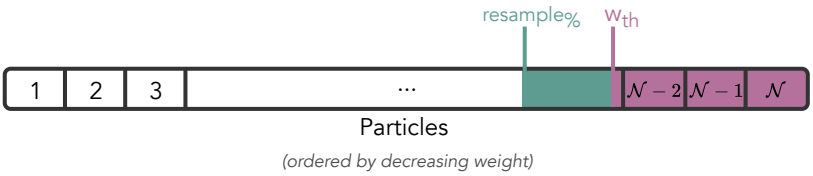
$$\alpha(t) = \begin{cases} (1 - \alpha_{min}) * (e^{-t/35}) + \alpha_{min} & , t < t_{warm-up} \\ \alpha_{min} & , t \geq t_{warm-up} \end{cases} \tag{4.5.5}$$

where t denotes the time since particles were created, α_{min} denotes the value of α after warming-up, and $t_{warm-up}$ defines the warm-up time.

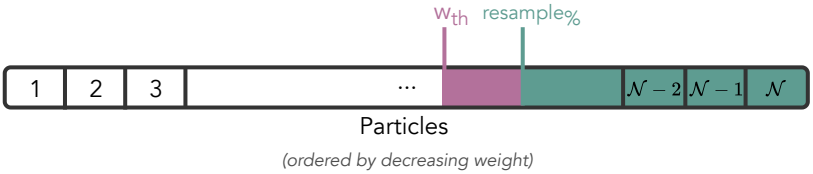
The Loose Coupling approach does not depend on the floor plan of the building to perform collision detection. Whenever a particle is farther than r meters of any RP within the indoor space, its weight is set to zero, because it means that the particle has moved into an area where the vehicle cannot navigate through. The r parameter depends on the distance between adjacent RPs.

4.5.4 Resampling

Two parameters are used to define how particles are removed, the resample percentage and the weight threshold. The first ($resample_{\%}$) defines a percentage of particles with lower weights that are removed. The second (w_{th}) defines a threshold where particles with weights lower than it are removed. In the resampling process, two distinct cases (depicted in Figure 4.5) may occur depending on the $resample_{\%}$, the w_{th} and the particles' weights. Figure 4.5 (a) shows the case in which the percentage of samples removed from the w_{th} criterion is lower than $resample_{\%}$, while Figure 4.5 (b) illustrates the opposite case.



(a) Percentage of samples removed based on the w_{th} criteria is lower than $resample_{\%}$.



(b) Percentage of samples removed based on the w_{th} criteria is higher than $resample_{\%}$.

Figure 4.5: Resampling scenarios in Loose Coupling approach.

In both scenarios, the particles removed due to the $resample_{\%}$ (represented in green in Figure 4.5) are replaced by new particles that are created near the latest Wi-Fi position estimate, following a similar process as the one described in Section 4.5.1. The weight of new particles near the Wi-Fi position estimate is defined by:

$$w = \begin{cases} w_{min} & , d_{vw} > d_{max} \\ f(d_{vw}) & , d_{vw} \leq d_{max} \end{cases} \tag{4.5.6}$$

where d_{vw} represents the distance between the vehicle's latest position estimate ($\rho_{vehicle}$) and the Wi-Fi position estimate (ρ_{wifi}), d_{max} defines the maximum allowed distance between $\rho_{vehicle}$ and ρ_{wifi}, and,

$$f(d_{vw}) = \frac{d_{max} - d_{vw}}{d_{max}} \times (w_{max} - w_{min}) + w_{min} \tag{4.5.7}$$

where the distance d_{vw} is converted into a weight value between w_{min} and w_{max}, the lower the distance the higher the weight value. This gives more emphasis to ρ_{wifi} estimates when they are closer to $\rho_{vehicle}$ and have a lower impact when ρ_{wifi} is further away from $\rho_{vehicle}$, which occurs when ρ_{wifi} is an outlier.

IMU sensors are susceptible to magnetic perturbations which affect magnetometer readings, consequently affecting the estimated heading, which usually leads to drift in the heading. Drift can be minimized

by assigning different heading offsets to resampled particles because there are particles that follow the real trajectory whilst others will move away from the trajectory or move into non-navigable areas. Particles that follow the real trajectory are the ones that hold the most likely heading offset, therefore they will end up having higher weights. To minimize drift in the heading and improve heading estimation, resampled particles created around ρ_{wifi} have a new heading offset, defined as:

$$ho = ho' + n_{ho} \tag{4.5.8}$$

where ho' is the heading offset of a randomly selected particle (from the subset of particles with weight higher than w_{th}) and n_{ho}=N(0, σ_{ho}) represents a zero-mean Gaussian distributed random angle.

When the $resample_{\%}$ of particles is lower than the number of removed particles due to the w_{th} (Figure 4.5 (b)), it is necessary to create new particles (represented in purple in Figure 4.5 (b)) to maintain the total number of particles constant. These new particles are created from the sub-set of particles that have weights higher than w_{th}, which are randomly selected and copied.

4.5.5 Wi-Fi Position Estimation

A new technique was implemented for improving positioning performance with Wi-Fi fingerprinting [28]. With this approach, Wi-Fi samples from multiple Wi-Fi interfaces are merged into one Wi-Fi sample which is then compared with the samples from the radio map. Since RSS values from distinct interfaces are not correlated (see [28]), they can be averaged into a unique sample, which results in lower mean and maximum errors.

Let us assume that I interfaces are used and that all of them collect a sample at the same time. A Wi-Fi sample ws_t^i is defined as the list of RSSI values measured by the interface i, from the R detected APs, in time instant t:

$$ws_t^i = \left(RSSI_1^i, ..., RSSI_R^i\right) \tag{4.5.9}$$

A combined sample can be obtained by computing the average of the RSSI values from the interfaces:

$$ws_t = \left(\widehat{RSSI}_1, ..., \widehat{RSSI}_R\right) \tag{4.5.10}$$

where,

$$\widehat{RSSI}_j = \frac{\left(RSSI_j^1 + ... + RSSI_j^I\right)}{I} \tag{4.5.11}$$

The calibration phase comprises the collection of several combined Wi-Fi samples at known RPs spread through the navigable space. In the real-time phase, the similarity of a Wi-Fi sample is computed for all samples of the radio map. The Manhattan distance was chosen as the dissimilarity function and the k-NN algorithm is applied to obtain the estimated position, that is defined as ρ_{wifi}. This is already explained in Section 3.1.3.

4.5.6 Vehicle Pose Estimation

The vehicle's pose (position and heading), is defined by the weighted average of the particles' positions:

$$\rho_{vehicle}(x, y, z) = \frac{\sum_{i=1}^{\mathcal{N}} p_i(x, y, z) \times p_i.w}{\sum_{i=1}^{\mathcal{N}} p_i.w} \tag{4.5.12}$$

and the weighted average of the particle's headings:

$$h_{vehicle} = \tan^{-1}\left(\frac{\sum_{i=1}^{\mathcal{N}} \sin(p_i.h) \times p_i.w}{\sum_{i=1}^{\mathcal{N}} \cos(p_i.h) \times p_i.w}\right) \tag{4.5.13}$$

Since the state of the PF is updated whenever a new sample from either sensor is obtained, the vehicle's pose can also be updated. In order to reduce computing time in the PF, this method is called whenever a new heading sample is received. The IMU sensor has a sample rate of 20 Hz, therefore, the vehicle's pose is updated every 50 ms, which is adequate for tracking vehicles.

4.5.7 Computational Complexity

PFs are known for having demanding computational requirements due to the high sample rate of sensors, which demands that the main operation is executed fast enough so that there is no delay between position estimates provided by the PF. The computational burden of PFs depends mostly on the number of particles, as most operations affect all particles and their states over time. When new heading or displacement measurements are obtained, the sampling algorithm has a low complexity of $O(\mathcal{N})$, with $\mathcal{N}$ representing the number of particles.

In the process of updating the particles' weights, the Euclidean distance between each particle and the latest Wi-Fi position estimate must be computed, having a complexity of $O(\mathcal{N})$. The same complexity is also present in the resampling method since the algorithm depends only on the particles and their weights.

When a new Wi-Fi sample is received, a Wi-Fi position estimate is obtained. In this process, the latest Wi-Fi sample is compared against the samples from the radio map, then the k-NN algorithm is used to estimate a position. This process has an asymptotic complexity of $O(N_{ws} \times N_{AP})$, where N_{ws} represents the number of Wi-Fi samples in the radio map, and N_{AP} represents the number of APs detected in the latest Wi-Fi sample.

The system handles part of the complexity problem by fixing the number of particles being used, and the achieved performance reported is for that specific number of particles. Moreover, the number of particles used can be the same for larger spaces, not increasing the computational cost of the system. This leaves the radio map as the only input that can increase the computational cost of the system. For larger spaces, the number of samples in the radio map may increase, however, this problem can be minimized by applying a clustering or filtering technique [129] which allows the reduction of radio map search in large areas without compromising the accuracy. As reported in [129] there are techniques that effectively reduced the radio map size while improving the accuracy in some cases.

4.6 Tight Coupling Approach

The main objective of the Tight Coupling approach was to improve the Loose Coupling approach in several ways. First, by using all available Wi-Fi information to update particle's weights. Instead of using the

Wi-Fi position when updating particles weights like is done in the Loose Coupling approach, it explores all available information with Wi-Fi similarity, better estimating the weights of particles. Second, it takes advantage of the building's floor plan to remove particles that hit walls or obstacles, to correct the trajectory. Third, it dynamically adapts to conditions, allowing to determine the influence that Wi-Fi data has in the process to update particles' weights. This is essential during the initial period during which particles are more spread throughout the space, and the Wi-Fi information allows to favor the particles that are in areas with higher similarity. In the following, the PF processes of the Tight Coupling approach are described.

4.6.1 Initialization

Before creating particles, an absolute position is necessary to determine where to place them. Wi-Fi fingerprinting allows to obtain an absolute position, however, there is a certain probability to have a high error in the initial position. To minimize the initial error, APs signal strengths of the first few Wi-Fi samples are averaged into a merged Wi-Fi sample (only RSSI values of detected APs are considered), to increase the probability of placing the particles closer to the real position [28]. The similarity between the averaged sample and the radio map samples is obtained, resulting in a list of samples ordered by similarity. Particles are created around the positions (RPs) where the most similar radio map samples were collected. Algorithm 2 describes how particles are created around RPs where the merged Wi-Fi sample is more similar.

Algorithm 2 Initialize particles - Tight Coupling approach.

Input

rm - radio map, set of Wi-Fi samples

$\overline{ws}$ - Wi-Fi sample of averaged RSSIs values from the first M Wi-Fi samples

J - number of RPs where particles will be created around

$\mathcal{N}$ - total number of particles $\{\mathcal{N} \in \mathbb{Z} \mid \mathcal{N} > J\}$

Output

P - set of all particles

procedure Initialize Particles Tight Coupling
    $P = \{\}$
    $\mathcal{D} = \mathsf{d_M}(rm, \overline{ws})$ *//compute Manhattan dist. (dissimilarity) between the rm and the $\overline{ws}$*
    $RP =$ get the set of J ref. points that are more similar with $\overline{ws}$ based on D
    $np = \mathcal{N}/\#(RP)$ *//number of particles to be created around each RP*
    for $rp \in RP$ **do**
        $s_n =$ normalized similarity between ws and rp, defined in Eq. 4.6.1
        for $i = 1 \in \{1 \ldots np\}$ **do**
            $w = s_n$
            $(x, y, z) = (rp.x + \mathsf{rand}(0, r), \quad rp.y + \mathsf{rand}(0, r), \quad 1)$
            $h = 0$
            $ho = \mathsf{rand}(-\pi, \pi)$
            $p = (w, x, y, z, h, ho)$
            $P = P \cup \{p\}$

4.6.2 Motion Model

In the Tight Coupling approach, the same motion model as the one described in Section 4.5.2 was adopted.

4.6.3 Update Weights

Wi-Fi ensures absolute positioning coverage of the entire operational area, therefore it is used to update particles weights as it allows to assign higher weights to particles that are closer to the most probable positions, and lower weights to particles that are further away from the most probable positions.

In this context, we refer to the radio map (rm) as the set of samples that were collected at a site survey, and we refer to Wi-Fi sample (ws) as the set of RSSI values detected by a Wi-Fi interface, as previously defined in Section 3.1.

In Wi-Fi fingerprinting, a position estimate is usually obtained by computing the centroid of the most similar radio map samples. While this estimated position can be used to update the weights of the particles (Loose Coupling), this approach does not use all the available information (set of all the similarities between the new Wi-Fi sample and all the radio map samples). Statistically, radio map samples that are more similar to the Wi-Fi sample are the ones closer to the true position of the Wi-Fi sample, as depicted in Figure 4.6. At the top part of the figure, there are some RPs whose similarity is better than in the center of the operational area. This is good for a Tight Coupling approach that considers entire set of similarities instead of the estimated position (black triangle) in Loose Coupling, which is displaced with respect to the real location of the Wi-Fi sample.

In the Tight Coupling approach, the set of all similarities is used when updating particles weights. Two Wi-Fi samples are compared using a distance (dissimilarity) or similarity function, e.g., Manhattan, Euclidean or Minkowski distances, or the cosine similarity. The Manhattan distance was chosen to compare the Wi-Fi samples since it showed good performance in Wi-Fi fingerprinting [130, 131, 132]. The expression that defines the Manhattan distance (d_M) is defined in Equation 3.1.3. If an AP is missing either in the radio map sample or the Wi-Fi sample, a default RSSI value of -90 dBm is used because the Wi-Fi receiver sensitivity is $\approx$-90 dBm. Once this is performed for all samples of the radio map, the Manhattan distance is normalized and converted into the Manhattan normalized similarity, as defined as:

$$s_n = 1 - \frac{d_M - \min(\mathcal{D})}{\max(\mathcal{D}) - \min(\mathcal{D})} \tag{4.6.1}$$

where $\mathcal{D}$ represents the set of the Manhattan distances between the Wi-Fi sample and the radio map samples. This expression converts the Manhattan distance into a similarity value, because it normalizes and inverts the Manhattan distance, so that it represents the normalized similarity, s_n, where $s_n \in [0, 1]$.

Figure 4.6 shows an example of the normalized similarity between a Wi-Fi sample and the radio map samples. Areas with higher similarity are depicted at the top and bottom of the figure. If the traditional Loose Coupling approach was used, the centroid could fall in an area with low similarity (black triangle), which is displaced with respect to the real location of the Wi-Fi sample. This issue does not exist in a Tight Coupling approach because it considers all available information and it does not depend on the estimated position.

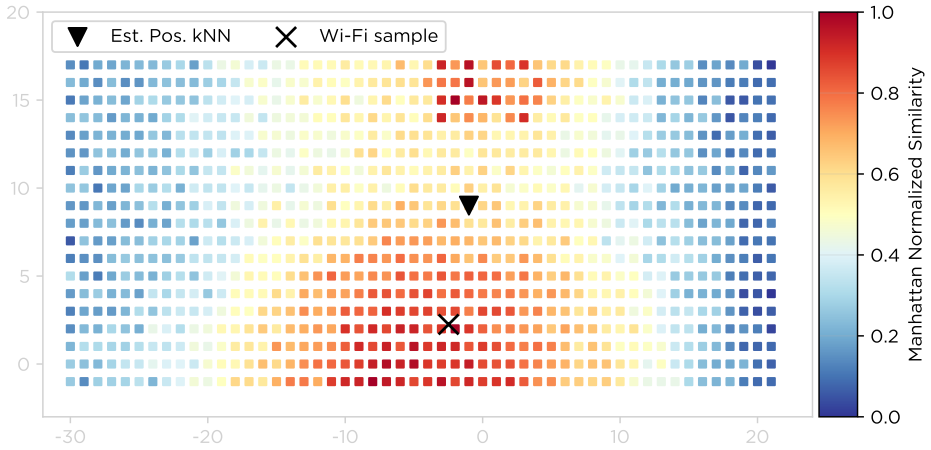


Figure 4.6: Normalized similarity between a Wi-Fi sample and radio map samples.

The particles' weights have an important role in the PF as they define which particles are removed in the resampling process, and they are used when estimating the vehicle's position and heading. The weight of a particle is defined as:

$$w_t = w_{t-1} \times (1 - \alpha) + s_n \times \alpha \tag{4.6.2}$$

where w_{t-1} represents the previous particle weight, s_n represents the normalized similarity, and α is a factor used to define how much the latest available Wi-Fi information affects the particles' weights.

The normalized similarity (s_n) component of the particle's weight is obtained from the nearest RP as shown in Figure 4.7. As the particle's weight depends on its closest RP, a dense radio map, with a low distance between neighboring RPs (e.g., 1x1 m grid), will improve the system performance because it ensures that the particle's closest RP is always at a distance lower than the grid size. A sparse radio map, with a larger distance between RPs, can also be adopted as long as its density is increased using LDPL models [1], neural networks [97], Voronoi tesselation [95], or other techniques.

The floor plan information is used in the Tight Coupling approach to perform collision detection by checking whether the line segment between the previous and the current particle's position has crossed a wall or obstacle, using floor plan data. When it happens, the particle's weight is set to zero.

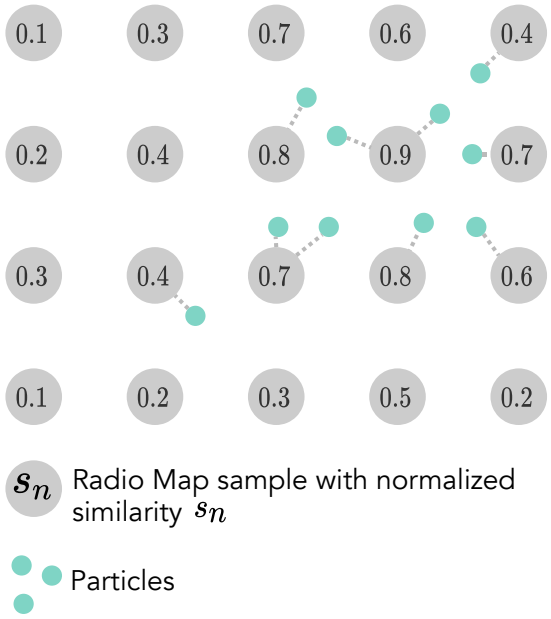


Figure 4.7: Assigning a similarity value to each particle.

4.6.4 Resampling

In the Tight Coupling approach, particles with weights lower than a given threshold are removed and new particles are created from the sub-set of particles with higher weights. In order to improve heading estimation and cope with drift in the heading, which is cumulative over time, new particles have a new heading offset based on the higher weight particle that was selected to be copied. This process is described in Algorithm 3. Once the particles with lower weights are removed (line 2, Alg. 3), the algorithm enters a loop (line 3, Alg. 3) to create new particles until a total of $\mathcal{N}$ particles exists in P'. The multinomial resampling method [133] was adopted in the resampling step. In this resampling method, particles are randomly selected according to their weight, using the weight as the probability of being selected (line 5, Alg. 3). This allows particles with higher weight to be selected with higher frequency than the ones with lower weights. Since the particle's previous weights are considered when updating the particle's weight, the PF does not follow a standard implementation in resampling where weights are normalized to sum up to one, and all particles have the same weight of $1/\mathcal{N}$. Instead, newly created particles that are copies of particles with higher weights inherit the original particle's weight.

In the initialization process, a random heading offset is assigned to each particle and, as the PF starts converging, particles that were initialized with the correct heading offset will have a higher probability of being selected and copied in this process, which will allow all particles to converge into similar heading offset values. It is known that IMU sensors suffer from cumulative drift due to sensor bias, hence, the heading offset cannot be constant. In the resampling process, new particles are created with a new heading offset to compensate for drift in the heading (line 6, Alg. 3).

Algorithm 3 Resampling based on the multinomial method.

Input

w_{th} - weight threshold, particles with weight lower than it are removed

n_{ho} - zero-mean Gaussian distributed random variable, to minimize drift and improve heading estimation

Output

P' - set of resampled particles

1: **procedure** Resample Particles
2: $P' =$ list of particles with weights higher than w_{th}
3: **while** #$(P') < \mathcal{N}$ **do** (multinomial resampling)
4: *//create copies of particles from* P'
5: $p =$ particle selected from P' using its weight as probability
6: $ho = p.ho + n_{ho}$ *//define particle's new heading offset by adding noise to particle's original heading offset*
7: $p' = (p.w, p.x, p.y, p.z, p.h, ho)$
8: $P' = P' \cup \{p'\}$ *//add copied particle to the set of resampled particles*

4.6.5 Vehicle Pose estimation

In this approach (Tight Coupling), the weighted average of all particles positions and headings represents the vehicle's estimated pose, as defined in Section 4.5.6.

4.6.6 Confidence and Dynamic Alpha

A distinctive feature of the Tight Coupling approach is that it dynamically adapts the influence that Wi-Fi has over the latest particle's weight. Larger positioning errors may occur during the first iterations after initialization because the initial position and heading are unknown. During this period, Wi-Fi should have a higher influence on particles' weights because it will result in removing particles in areas where it is less likely for the vehicle to be at. As particles are dispersed throughout the space during the first iterations of the PF, higher weights will be attributed to particles in areas where it is more likely for the vehicle to be, and lower weights to particles that are in unlikely positions. As Wi-Fi defines how particles' weights are updated, it also has an important role to minimize drift in the heading. The α parameter, introduced in Section 4.6.3, defines how much the particle's weight is influenced by the Wi-Fi similarity.

Based on this behavior of the PF, a confidence measure is proposed that has two purposes. First, it can be used as a reliability metric, indicating whether the estimated position is more likely to have a larger error associated. Second, it can be used to dynamically adjust the value of α. When the confidence is lower (meaning that a higher positioning error might exist), a larger value is assigned to α, thus it gives a higher influence to Wi-Fi.

The dispersion of particles ϑ (in meters), defined as the average distance between the particles' positions and the PF estimated position, is given by:

$$\vartheta = \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} \mathsf{d_E}(\rho_{vehicle}, p_i) \times w_i \tag{4.6.3}$$

where $\mathsf{d_E}$ defines the Euclidean distance between the ith particle position p_i and the PF estimated position $\rho_{vehicle}$, and w_i represents the particle's weight.

The dispersion is used to define a metric of confidence, defined as:

$$C(\vartheta) = \begin{cases} -\frac{1}{r_{max}}\vartheta + 1, & 0 \leq \vartheta < r_{max} \\ 0, & \vartheta \geq r_{max} \end{cases} \tag{4.6.4}$$

where r_{max} defines the maximum allowed dispersion (in meters) between particles. It was defined $r_{max} = 4\ m$, since, even for tracking, more than 4 m of radius is a large area and therefore it can be associated to significant position errors.

The confidence was modelled after an empirical analysis of the particles' dispersion in simulations, as described in Section 4.6.7. Upon an analysis, it was found that when the cloud of particles is smaller (lower dispersion), there is usually higher confidence associated, and when the dispersion of particles is larger (e.g., during first iterations) the positioning error tends to be higher.

The confidence measure has to be a reliable indicator of whether there is a large error in position estimates, hence several experiments with synthetic data were conducted to define the appropriate function that defines the confidence. Section 4.6.7 shows the correlation between the confidence and the positioning error, demonstrating that the confidence function is reliable.

Even though α could be a direct translation of the confidence value, i.e., $\alpha = 1 - C$, it would greatly impact the PF estimated position as Wi-Fi would have too much influence in particles' weights resulting in large errors. Tests performed with simulated data have shown large positioning errors may occur when $\alpha > 0.6$ (see Section 4.6.7). Taking this into consideration, the following expression was used to make sure that when confidence is low, the Wi-Fi has less influence:

$$\alpha = 0.6 - 0.6 \times C \tag{4.6.5}$$

4.6.7 Simulation

Simulation experiments were performed to tune the PF parameters and define the appropriate function for α. Initially, a simulated environment was created then, synthetic data of displacement, heading and Wi-Fi measurements were generated. Finally, the proposed Tight Coupling approach was tested and its parameters were tuned to achieve the best results.

Scenarios

Two simulation scenarios were considered, one representing an empty building (50 $\times$ 20 m), without obstacles, and another representing a building with the same dimensions but with space restrictions (the vehicle cannot navigate in some areas). Six trajectories were considered in these experiments: 3 random

trajectories and 3 loop trajectories. The purpose of random trajectories is to show that the PF is capable of converging, even without any space restrictions, and the purpose of loop trajectories is to show how the PF performs in a realistic scenario, with obstacles and space restrictions. Each random trajectory is defined by a random movement with a travelled distance of approx. 500 m. Each loop trajectory is defined by a cyclic movement, where the vehicle performs several laps in a pre-defined trajectory and has an average travelled distance of about 1750 m.

Synthetic Data

The simplest method to test positioning techniques and algorithms is through simulation, using synthetic data. The main advantage of this process is that it avoids the need to perform real-world experiments, and allows to validate the positioning solution under different conditions without additional effort.

As the PF depends on data from Wi-Fi and motion sensors, it is necessary to emulate these sensors by generating synthetic data (heading, displacement and Wi-Fi) synchronized in time. To achieve that, it is assumed that the vehicle follows a trajectory from point A to point B at a constant speed of 1 m/s, then it stops for 1 s at B, before proceeding to the next point. A trajectory is composed of many points. Based on the output of sensors to be used, sampling rates of heading, displacement and Wi-Fi were defined as 20 Hz, 50 Hz and 1/2 Hz, respectively. Wi-Fi scanning in the 2.4 GHz frequency range usually takes more than 1.5 s to collect the signal strength values of APs, hence it was decided that Wi-Fi samples were generated every 2 s.

A displacement sample is defined as:

$$ds = l + n_{ds} \tag{4.6.6}$$

where l represents the true traveled distance and n_{ds} represents a zero-mean Gaussian distributed random variable that was defined as N(0, 0.004 m).

Each heading sample is generated as:

$$hs = \beta + n_{hs} + \phi \tag{4.6.7}$$

where β represents the true absolute heading, n_{hs} represents a zero-mean Gaussian distributed random variable, and ϕ represents the cumulative drift defined in degrees per hour, meaning that drift increases over time, hence after one hour drift has a value of ϕ degrees. It was defined n_{hs}=N(0, $10°$) and ϕ=20°/h.

A simple model based on LDPL was used to calculate signal strength values of APs. A signal strength value is defined as:

$$RSSI = RSSI_0 - 10 \times \eta \times log_{10}(\xi) + n_{rssi} \tag{4.6.8}$$

where $RSSI_0$ represents the RSSI value at 1 m distance of the AP, η represents the path loss exponent, ξ represents the distance between the AP and the point where the signal strength is estimated, and n_{rssi} is a zero-mean Gaussian distributed random variable. It was defined $RSSI_0 = -40$ dBm, n_{rssi}=N(0, 4 dBm), and $\eta = 2$.

Both radio map samples as well as Wi-Fi samples of trajectories were created using this model. The radio map of the simulation scenario comprises 20 Wi-Fi samples at each RP for a 1x1 m grid of RPs.

In the definition of displacement and heading noise components (n_{hs} and n_{ds}) and drift in the heading (ϕ) it was assumed that the sensors produced more noise than normal (industrial-grade) sensors to test the proposed solution under challenging conditions. The Wi-Fi noise component was defined considering that Wi-Fi signals have high variability in indoor environments due to multipath and other propagation effects.

The work developed in this simulation tool has contributed to the Dioptra tool which is a synthetic data generator that can be used for fair comparison between IPSs. The Dioptra tool is described in Appendix B.

Analysis of Results using Constant α

Before modelling the confidence function, previously described in Section 4.6.6, it is necessary to study how α affects results, using a constant α in all trajectories.

The PF was run three times for each trajectory, using different random seeds. Table 4.1 (top part) shows the overall positioning error that includes position estimates of all runs of loop and random trajectories. The used error metric is the Euclidean distance between the estimated and true positions.

A clear deterioration of results is observed for $\alpha \geq 0.6$. This suggests that in the long run, the latest Wi-Fi sample should have a low influence over the particles' weights, therefore it was defined the value of α (Equation 4.6.5) varying between 0 and 0.6 as a function of the confidence. When the confidence is

Table 4.1: Positioning results (in meters) of simulated trajectories for different α (top) and w_{th} (bottom) values.

α	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
Mean	**0.70**	0.71	0.82	1.47	1.63	3.81	4.36	6.51	4.07	4.63
Median	**0.49**	0.50	0.56	0.64	0.69	0.80	0.94	3.19	1.12	1.24
P_{75th}	**0.86**	0.90	1.07	1.29	1.44	2.70	4.04	5.36	3.27	4.06
P_{99th}	3.69	**3.28**	3.54	19.45	18.24	31.24	29.75	46.94	43.83	41.26
Max	8.46	6.93	**6.51**	27.91	36.78	38.10	35.30	52.19	52.43	50.08
w_{th}	**0.1**	**0.2**	**0.3**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**0.9**	**1.0**
Mean	0.88	0.87	0.87	0.81	0.77	0.73	**0.71**	12.30	16.59	N.A.
Median	0.49	0.49	0.49	0.49	**0.48**	0.49	0.50	11.61	16.01	
P_{75th}	0.94	0.92	0.93	0.92	**0.90**	0.91	**0.90**	18.02	23.14	
P_{99th}	8.28	8.03	7.99	6.81	5.65	4.49	**3.28**	31.96	44.86	
Max	13.11	10.95	11.07	10.79	10.29	8.58	**6.93**	34.93	47.22	

high, α is low, and vice versa. This approach is important in the warm-up period during which Wi-Fi has more importance while the particles are still converging into a position and heading that is more likely to be the true vehicle's pose.

Analysis of Weight Threshold w_{th}

The PF performance was also analyzed for different values of the weight threshold (w_{th}) used in the resampling process. In this process, if the weight threshold is too low, particles will move freely in random trajectories or will move freely until they hit walls or obstacles in loop trajectories. If it is too high, too many particles are removed in the resampling process.

Based on the previous results, the same experiment was conducted for loop and random trajectories using $\alpha = 0.2$ and tested different values of w_{th} to understand how this parameter affects results. Table 4.1 (bottom part) shows the overall positioning error that includes position estimates of loop and random trajectories, obtained for different w_{th} values. When $w_{th} = 0.7$ it achieves the best mean, 75th percentile, 99th percentile and maximum error, therefore it was the selected value for w_{th} in the performed experiments, including the ones of Section 4.6.7. As expected, the PF performance degrades when w_{th} is too high, i.e., when $w_{th} \geq 0.8$ the PF removes an excessive amount of particles in resampling, hence the PF is not able to converge.

Simulation Results

The following results consider α as a function of C, as described in Section 4.6.6. Each trajectory was run three times, as previously. Positioning error statistics include all position estimates since $t = 0$, including the initial warm-up period. Table 4.2 shows the positioning results of loop trajectories, random trajectories, and all trajectories. In comparison with the constant α results, the major improvement is observed in the overall maximum error from 6.93 m (for $\alpha = 0.2$) to 5.95 m, besides, all other performance metrics are also improved. Also, better results are achieved in loop trajectories, in comparison to random trajectories, because many particles are removed when they hit walls or obstacles. This improves positioning and tracking because particles remain in areas where vehicles navigate. Conversely, the positioning error is larger in random trajectories because particles are more dispersed, and they are removed in the resampling process or when they hit the limits of the open-space building.

Table 4.2: Positioning results of Loop Trajectory (LT), Random Trajectory (RT) and all trajectories using a dynamic α (in meters).

	LT	**RT**	**All**
Mean	0.39	1.07	0.66
Median	0.34	0.86	0.48
P_{75th}	0.50	1.35	0.80
P_{99th}	1.51	3.93	3.15
Max	3.55	5.95	5.95

Positioning Error vs Confidence

Figure 4.8 depicts a density plot of the positioning error versus confidence, considering all position estimates of all simulations according to Section 4.6.7. It shows that there is a relationship between the confidence (given by the dispersion of particles) and the positioning error, where the vast majority of position estimates with high confidence ($>$0.8) have positioning errors lower than 1 m. There is a strong negative correlation between the positioning error and confidence, which is statistically significant, $\mathcal{R}_{Pearson} = -0.70$, $p_{value} = 0.0$. Note that a logarithmic scale was used in the color scale of Figure 4.8 to better represent these results.

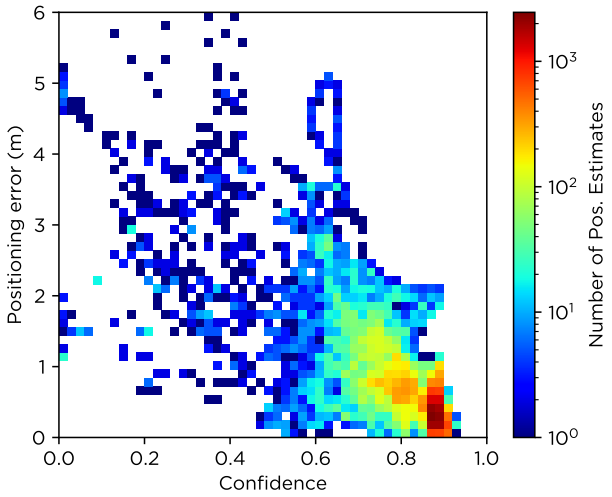


Figure 4.8: Density plot of position error vs confidence.

4.6.8 Computational Complexity

As already stated in Section 4.5.7, the computational burden of the PF depends mostly on the number of particles, because most operations affect the particles' states. The sampling algorithm that updates particles' positions and headings has a low complexity of $O(\mathcal{N})$, where $\mathcal{N}$ represents the number of particles.

In the process of updating the particles' weights, the number of radio map Wi-Fi samples (N_{ws}), as well as, the number of detected APs in the latest Wi-Fi sample (N_{AP}), lead to an asymptotic complexity of $O(N_{ws} \times N_{AP})$. A similar complexity is achieved in the process to find the closest RP to each particle, where a comparison is made between the positions of N_{ws} Wi-Fi samples and $\mathcal{N}$ particles, thus achieving an asymptotic complexity of $O(N_{ws} \times \mathcal{N})$. The resampling algorithm, which implements the multinomial resampling approach [133], has a complexity of $O(\mathcal{N} \log_2(\mathcal{N}))$, lower than the complexity in updating particles weights, but also presents asymptotic complexity.

Similarly to the Loose Coupling approach, the computational complexity of the Tight Coupling algorithm depends mostly on the number of radio map Wi-Fi samples and the number of particles in the PF. The number of particles can be adjusted according to the available computing resources, and the number of Wi-Fi samples can also be optimized to reduce computational effort (e.g., multiple Wi-Fi samples collected in the same RP may be averaged into one).

4.7 Parameters

Preliminary experiments were conducted to tune the parameters of the PFs. These experiments consisted in collecting test data in a few trajectories and running the PFs using these data as input. Since there are many possible combinations of parameters, the methodology used to optimize the set of parameters focused on selecting a few possible combinations of the parameters, running the PFs of both approaches, and choosing the combination of parameters that leads to better positioning results. The Tight Coupling approach also benefited from the simulations that allowed to test different scenarios and optimize the confidence function based on the simulation results.

Table 4.3 shows the list of parameters used in each approach. They share some parameters (e.g., the number of particles $\mathcal{N}$ or the weight threshold w_{th}, among others), but several parameters are specific to each approach. The values defined in this table are the ones used in real-world experiments with the Loose Coupling and Tight Coupling approaches. The Tight Coupling approach has the advantage of having less parameters than the Loose Coupling approach, therefore, being easier to configure.

Table 4.3: Parameter definition of both particle filter approaches.

	Description	**Loose Coupling**	**Tight Coupling**
Initialization			
$\mathcal{N}$	number of particles	3000	3000
$WiFi_n$	number of Wi-Fi samples used to obtain centroid	3	-
r	radius for removing particles, depends on the distance between RPs	$\sqrt{2}$ m	$\sqrt{2}$ m
r_{ini}	radius for selecting nearby RPs in initialization	4 m	-
k	k-NN parameter for Wi-Fi fingerprinting	5	-

Table 4.3: Continued from previous page.

	Description	Loose Coupling	Tight Coupling
J	number of RPs where particles will be created around	-	6
M	number of first Wi-Fi samples to be averaged	-	3
Disp. and Heading			
n_d	displacement zero-mean Gaussian distributed random variable	N(0, 0.004 m)	N(0, 0.01 m)
n_θ	heading zero-mean Gaussian distributed random variable	N(0, $2.8°$)	N(0, $1°$)
Update Weights			
α_{min}	α takes this value after $t_{warm-up}$	0.05	-
$t_{warm-up}$	warm-up time	180 s	-
Resample			
w_{th}	particles with weights lower than this threshold it are removed in resampling	0.2	0.7
n_{ho}	heading offset zero-mean Gaussian distributed random variable	N(0, $8°$)	N(0, $2°$)
$resample_{\%}$	percentage of particles with lower weights that are removed	10%	-
d_{max}	maximum allowed distance between $\rho_{vehicle}$ and ρ_{wifi}	4 m	-

Table 4.3: Continued from previous page.

	Description	Loose Coupling	Tight Coupling
w_{min}	used to convert a distance into a weight	0.05	-
w_{max}	used to convert a distance into a weight	0.5	-

4.8 Implementation and Visualization Interface

The implementation of both sensor fusion approaches was done using the Java programming language, therefore the programs can be easily extended or integrated into other Java programs. Since this programming language is available in most operating systems, it can run in any machine that supports it, including the Raspberry Pi, which is a low-cost computer that can run the PF in real-time. The operating principle of the application is simple, it parses the radio map and the building's floor plan, then it runs iteratively while there is new sensor data. A visualization interface, depicted in Figure 4.9, is integrated into the application to better understand how the PF works and to easily detect whether a certain configuration (set of parameters) works as expected. The interface displays the following data: the particles with different colors according to their weight; the floor plan, including the non-navigable areas; the estimated and the ground truth trajectories; the locations with Wi-Fi samples from the radio map (pink dots); the estimated heading in graphical and numerical formats; the weighted average of the particles; and, the latest positioning error given by the Euclidean distance between the ground truth and the estimated position.

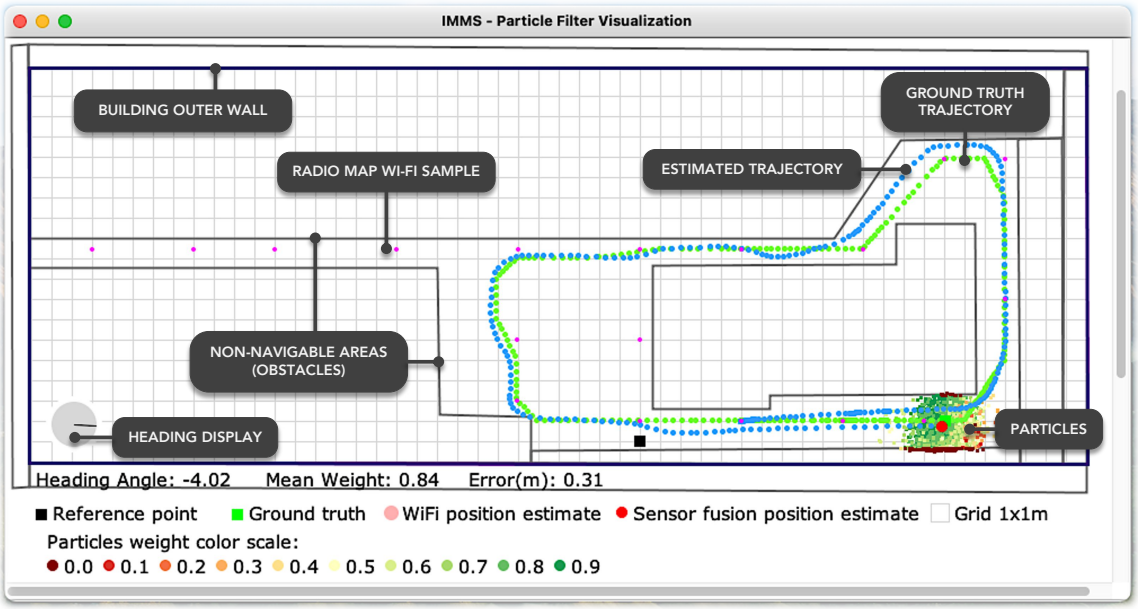


Figure 4.9: PF visualization interface.

The interface also shows several elements in a legend, including a color scale for the weights of particles, and a few markers that identify different locations, namely: reference point refers to point $(0, 0)$ in the building; ground truth represents the current ground truth position; the sensor fusion estimated position represents the current PF position estimate; and, the grid which represents the size of each square in the grid. The grid is also an additional element that allows to easily interpret the distances inside the building.

4.9 Real-world Experiments

An industrial vehicle was used to perform experiments in a factory-like building to evaluate both the Loose Coupling and Tight Coupling approaches. Preliminary tests included the collection of data from sensors installed in the industrial vehicle and the construction of the radio map at the testing scenario.

The real-world experiments described in this thesis were performed in two stages. First with the calibration (radio map) and test data collection, and then with the offline processing of the data, which allows to perform several experiments with the same data, and to assess the performance of each approach using

different parameters.

4.9.1 Testing Scenario

Experiments were conducted at the PIEP building (at the University of Minho) which is dedicated to research on plastic polymers, hence it is similar to a factory plant with large open spaces, large machinery and metal objects (injection molds), as shown in Figure 4.10. The building measures approximately 50 $\times$ 20 m and is more than 8 m high.

Figure 4.10: PIEP building at University of Minho: installed APs (upper right); reference tags marking RPs (lower right).

Figure 4.11 shows RPs where radio map samples were collected (grey squares), the positions and specifications of the APs present in the building (colored squares and diamonds), and the navigable areas (places where the vehicle can operate). The distance between adjacent RPs is 1 m in most cases. In order to build the radio map, 20 Wi-Fi samples were collected at each RP for a total of 178 RPs. It took about 180 minutes to build the radio map considering 40 s to collect the Wi-Fi samples at each RP and 30 s to move to the vehicle to a new RP. A set of tags affixed to the floor marking the positions of RPs allows to obtain ground truth data using a video camera pointed towards the floor where it recorded the time when the vehicle has passed each tag.

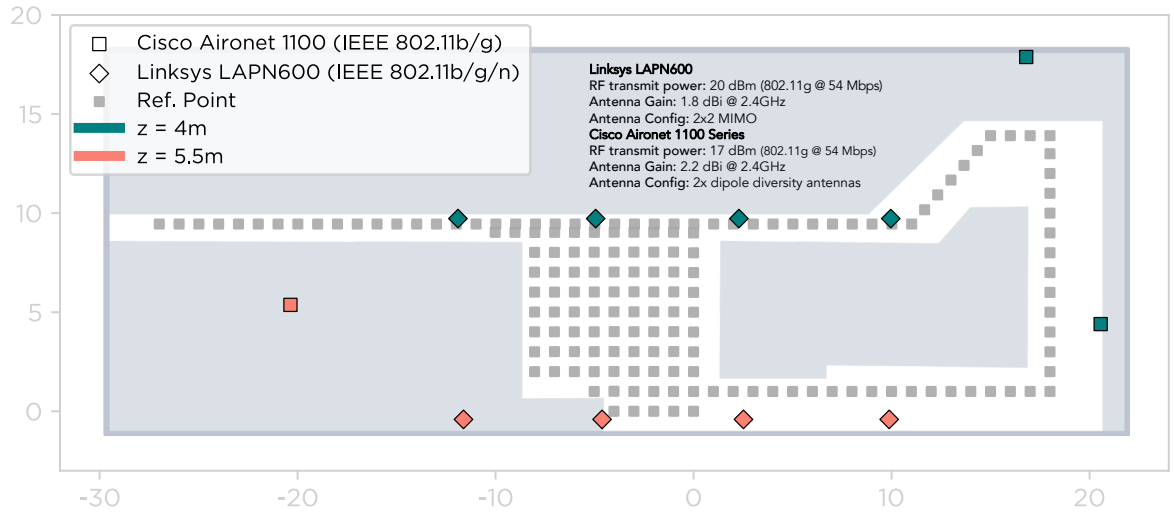


Figure 4.11: Floor plan of testing scenario: reference points (gray squares) and the installed APs according to their model, network standard and height (z).

4.9.2 Vehicle Prototype

A positioning module was installed in an industrial tow truck (Toyota BT Movit TSE300), used to transport materials in factories. The integrated positioning module, shown in Figure 4.12, is composed of a Raspberry Pi Model 3B+ as well as the following sensors: four external Wi-Fi interfaces (Edimax EW7811-Un); an Xsens MTi-300 AHRS IMU to measure the heading; and, an absolute encoder (US Digital A2) to measure the displacement. The hardware diagram, represented in Figure 4.13, represents how sensors are connected to the Raspberry Pi, the communication interface of each sensor and the power supply specifications necessary to power the Raspberry Pi and the encoder sensor. The encoder sensor is attached to the back of the vehicle with a custom mount.

An advantage of the positioning module is that it can be easily installed in other types of vehicles because it operates independently of the vehicle, however, it still depends on coupling the encoder to the vehicle.

Figure 4.12: Industrial vehicle (tow truck) used in tests.

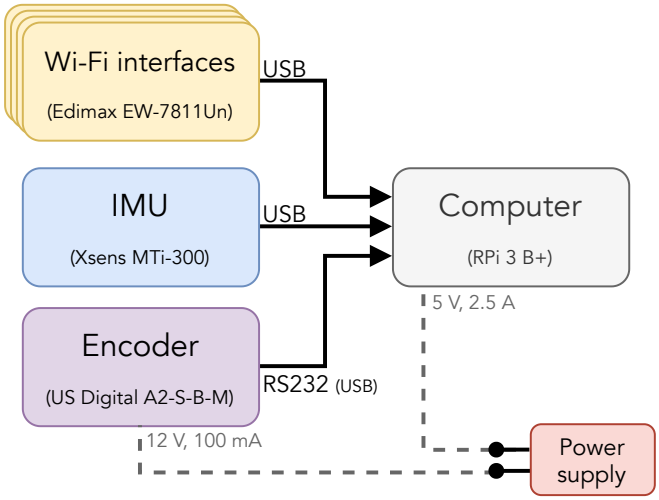


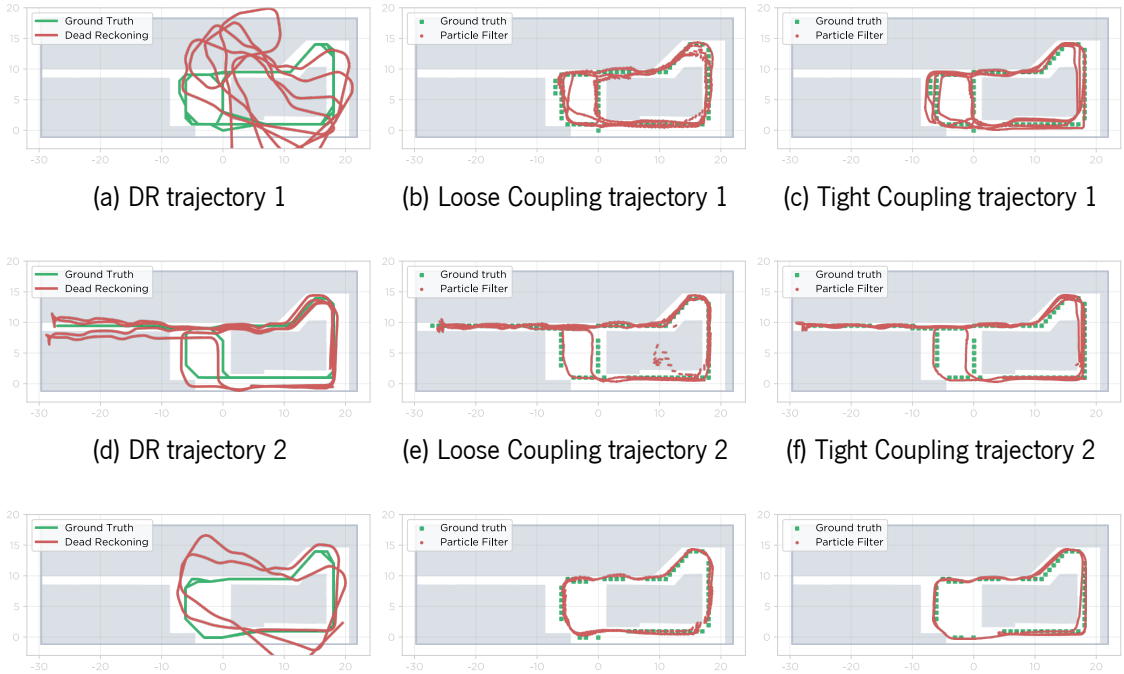
Figure 4.13: Hardware diagram of the positioning module of the vehicle prototype.

4.9.3 Results

Three test trajectories were performed where the vehicle moves through the indoor space while a video camera records the tags as the vehicle passes by them, allowing to collect ground truth information. Two trajectories have a duration of 15 minutes and the third 7 minutes, covering distances of 305, 303 and 168 m, respectively. Positioning error reported in this section is obtained from the Euclidean distance between the ground truth position and the estimated position. The positioning error was measured at the RPs positions having a total of 225, 255, and 126 measurements for trajectories 1, 2, and 3, respectively.

This section presents a comparison between results achieved with the Loose Coupling and Tight Coupling approaches.

Figure 4.14 displays the DR and estimated trajectories using Loose Coupling and Tight Coupling approaches. The DR trajectories consider an initial position and heading, which is updated based on heading and displacement samples. The initial pose is unknown in the Loose Coupling and Tight Coupling trajectories. Drift in the heading is present in all DR trajectories, but the Loose Coupling and Tight Coupling approaches can compensate for the drift, as shown by the estimated trajectories. This is possible due to the techniques used to remove particles that have moved into non-navigable areas as well as the resampling methods, which update the heading offset of particles. When comparing the estimated Loose Coupling and Tight Coupling trajectories, the overall results are similar, however in trajectory 2, there is a large error in the Loose Coupling trajectory.



(a) DR trajectory 1 (b) Loose Coupling trajectory 1 (c) Tight Coupling trajectory 1

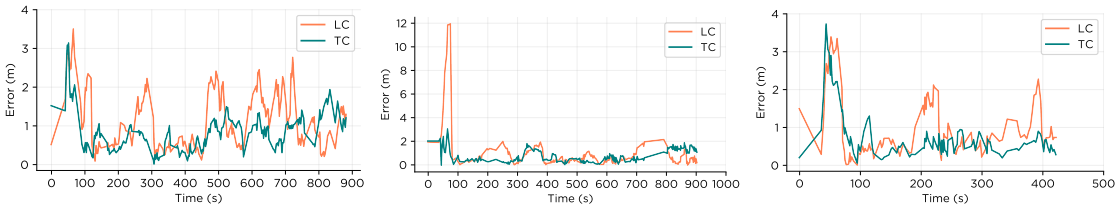
(d) DR trajectory 2 (e) Loose Coupling trajectory 2 (f) Tight Coupling trajectory 2

(g) DR trajectory 3 (h) Loose Coupling trajectory 3 (i) Tight Coupling trajectory 3

Figure 4.14: Comparison between dead reckoning, Loose Coupling and Tight Coupling estimated trajectories.

Figure 4.15 shows how the positioning error varies over time for both approaches. Errors are higher during the initial phase (warm-up period), while the PF is converging, which is the expected behavior. It

was arbitrated that the warm-up period comprises the position estimates in the interval $t \in [0\ s, 100\ s]$. The Tight Coupling approach converges faster, has a lower maximum error and a better mean error after the warm-up period. This is achieved due to the process of updating the particles' weights and resampling, which rapidly removes particles from improbable areas. The floor plan information also helps in reducing error, as it allows to remove particles that hit walls or obstacles.



(a) Trajectory 1 (b) Trajectory 2 (c) Trajectory 3

Figure 4.15: Error over time of each trajectory using the Loose Coupling (LC) and the Tight Coupling (TC).

Results of all trajectories are aggregated and presented in Table 4.4. It shows a comparison between the Loose Coupling, the Tight Coupling and pure Wi-Fi fingerprinting (k-NN with k=5). The Tight Coupling approach has an improvement of over 65% in the maximum error (from 12.26 m to 4.00 m) and over 20% in the third quartile (from 1.40 m to 1.08 m), in comparison to the Loose Coupling approach. After the warm-up period, the Tight Coupling approach also has improved results, especially in the maximum error.

Table 4.4: Overall positioning results (in meters) of experiments with industrial vehicle.

	Wi-Fi FP	**Loose coupling approach**		**Tight coupling approach**	
	Overall	**Overall**	**AW**	**Overall**	**AW**
Mean	2.15	1.03	0.85	**0.81**	**0.71**
Median	1.85	0.68	0.66	0.67	0.64
P_{75th}	2.97	1.40	1.23	**1.08**	**1.00**
P_{95th}	5.39	2.46	2.00	1.72	1.48
P_{99th}	7.48	6.87	2.33	**3.01**	**1.70**
Max	12.24	12.26	2.88	**4.00**	**2.06**

FP – Fingerprinting; **AW** – After warm-up.

Figure 4.16 shows the CDF curves of the Tight Coupling, Loose Coupling, and Wi-Fi fingerprinting for the experiments with the industrial vehicle. Both sensor fusion approaches (Tight Coupling and Loose Coupling) benefit from the motion sensors for improved tracking of the vehicle. This justifies why these approaches

have improved performance over Wi-Fi fingerprinting. The CDF curves of the Tight Coupling and Loose Coupling approaches are very similar for probabilities below 0.6, but after that point, there is a clear improvement of the Tight Coupling over the Loose Coupling one. The curves after the warm-up are the ones that better represent the positioning performance in a realistic scenario because it is expected to be running in vehicles for long periods, in which the larger errors in the beginning (while the PF is converging) are less relevant. In positioning applications with demanding requirements such as high accuracy and low maximum error, the performance in the last quartile is very important. The Tight Coupling approach is better suited for indoor vehicle tracking because it has a lower maximum error and an improved positioning performance over the Loose Coupling approach. The Tight Coupling approach achieved 1.0 m and 1.70 m in the 75th and 99th percentiles, respectively, showing that it can locate industrial vehicles accurately with a low maximum error.

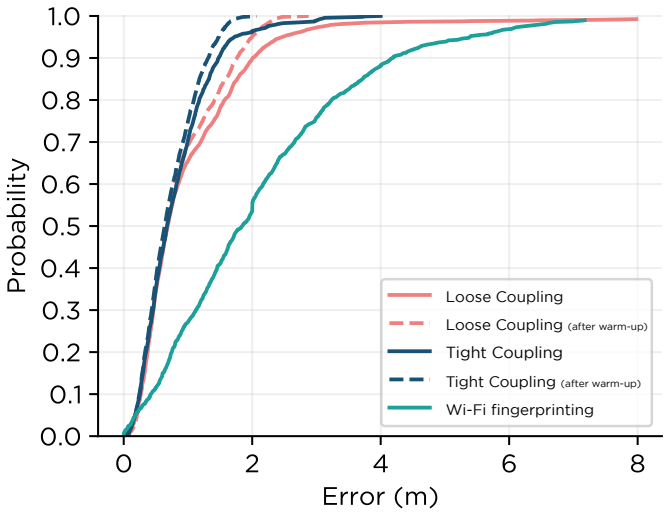


Figure 4.16: CDF of the positioning error comparing the Loose Coupling and Tight Coupling approaches with Wi-Fi fingerprinting.

4.10 Summary

Two sensor fusion approaches for indoor vehicle tracking using a PF were proposed in this chapter. The Tight Coupling approach performs the sensor fusion at the measurement level, whilst the Loose Coupling approach performs the sensory fusion at the position level because it considers a Wi-Fi position estimate

in its sensor fusion algorithm. In the Loose Coupling approach, a Wi-Fi position estimate is combined with data from motion sensors in the fusion algorithm. In the Tight Coupling approach, a Wi-Fi sample is merged with data from motion sensors, using the Wi-Fi similarity to update the particles' weights. Instead of using just one Wi-Fi interface to obtain Wi-Fi samples, a common practice in previous works, in this work multiple Wi-Fi interfaces are used to obtain better RSSI measurements and reduce noise [28]. Another distinctive aspect of these sensor fusion implementations is that the Loose Coupling approach works without requiring a floor plan, and the Tight Coupling approach supports floor plans and has a measure to tell whether the estimated position is reliable.

Due to the distinct features of each implementation, a detailed description of the PF processes (particles initialization, update weights and resampling) was presented for both implementations. Simulations were conducted to tune some parameters of the Tight Coupling implementation, and to define the appropriate confidence function that is used to dynamically adjust the α parameter when there might exist a larger error in position estimates. The confidence metric has an important role in vehicle tracking, as it provides information regarding the reliability of the system. When confidence is low, there is a strong possibility that the estimated position is wrong, and has a large error associated. Conversely, when it is high, the estimated position is very likely to be near the true position of the vehicle. Chronologically, the Loose Coupling approach was implemented first, and the Tight Coupling approach was implemented later aiming to overcome some limitations of the Loose Coupling approach. In addition to the differences in the way Wi-Fi information is integrated into the sensor fusion algorithm, the Tight Coupling implementation also included other features, such as the integration of floor plans in the PF, a new resampling method, and a confidence metric.

Real-world experiments were conducted in a factory-like building, where an industrial tow truck with a positioning module collected sensor data in three test trajectories. The positioning results of both approaches were compared against each other and also with Wi-Fi fingerprinting. Adding the motion sensors enables the tracking of the vehicles with much higher accuracy, as shown by the results achieved with both sensor fusion approaches in comparison to the Wi-Fi fingerprinting results. These sensor fusion approaches overcome some of the main limitations of Wi-Fi fingerprinting since they drastically reduce the errors in the last percentile and allow continuous tracking of the vehicle with a high sample frequency. Overall, the Tight Coupling approach has better positioning performance than the Loose Coupling approach mainly

because it uses all available information in the sensor fusion method, it dynamically adjusts to conditions based on the confidence metric, and it takes advantage of the floor plan of the building to correct the estimated trajectory. These improvements over the Loose Coupling approach translate into better performance, especially in the maximum error.

Table 4.5 shows an overview of the proposed approaches and several positioning systems for the localization of industrial vehicles, mobile robots, and pedestrians. This overview presents each system according to its technology, localization technique, and reported accuracy (mean error). An overall mean error of 0.81 m was achieved with the Tight Coupling approach, which performs better than the Loose Coupling approach (1.03 m mean error) using the same sensors. Regarding the other positioning systems, the results of both approaches are in line with systems with similar technologies based on PFs. The main contributions of this work are published in [14] and [11], detailing the Loose Coupling and Tight Coupling approaches, respectively.

Table 4.5: Overview of the proposed approaches in comparison to similar systems.

Solution	**Technology**	**Localization Technique**	**Mean Error (m)**
[75]	UWB + GPS	PF	0.16
TC approach	Wi-Fi + MS	PF w/ TC	0.81
LC approach	Wi-Fi + MS	PF w/ W-FP	1.03
[3]	Wi-Fi + odometry + LiDAR	PF	1.20
[128]	Wi-Fi + PDR	PF	1.20
[134]	Wi-Fi + IMU + FM transmitters	PF w/ W-FP	2.11
[80]	Wi-Fi	W-FP	2.25

PF – Particle Filter; **MS** – Motion Sensors (IMU and encoder); **W-FP** – Wi-Fi fingerprinting; **LC** – Loose Coupling; **TC** – Tight Coupling.

The participation in the IPIN 2019 Competition (Track 3) [9] with the UMINHO team has contributed to getting an understanding of sensor fusion techniques based on Wi-Fi fingerprinting and PDR. The adopted

approach used Wi-Fi fingerprinting combined with PDR and FastGraph [135] to estimate the trajectory of a pedestrian. Although IPIN's Track 3 is based on offline data collected from smartphone's sensors, it revealed challenges related to indoor vehicle tracking, for instance, the movement model of vehicles differs from pedestrians, the types of sensors that can be used in vehicles and how different the sensor fusion algorithms may depend on these aspects. The UMINHO team was 7th out of 12 teams, thus showing that the proposed system would need to be improved to achieve a better classification.

After the participation in the 2019 IPIN Competition [9], in 2020, the UMINHO team participated with a new approach inspired by the Loose Coupling implementation described in this chapter. The proposed system [10] used a PF to perform sensor fusion of Wi-Fi fingerprinting and PDR data which integrated the building's floor plan to correct the trajectory. The UMINHO's team classification improved from 7th in 2019 [9], to 5th in 2020 [10].

With the ability to accurately track industrial vehicles, they can be explored for building/maintaining radio maps. When the position of the vehicle is reliable, it can be associated with the Wi-Fi samples that are being collected while vehicles navigate through the industrial environment. In the next chapter, a self-healing radio maps solution based on the Tight Coupling approach is described.

Chapter 5

Self-healing Radio Maps using Industrial Vehicles

The previous chapter described two different approaches for the localization of industrial vehicles based on a PF. The deployment of an IPS like this in the real world raises many challenges, such as the installation of infrastructure, the calibration process or modelling of the building's floor plan. For Wi-Fi-based IPSs, deployments often require a laborious and time-consuming site survey to build and maintain a radio map.

In industrial environments, vehicles have an important role in the day-to-day tasks, mostly in the transportation of raw materials and finished goods. As mentioned before, locating these vehicles has numerous advantages, e.g., improving safety and monitoring, increasing productivity, or integration of the vehicles' position in the information systems. In addition, by localizing these vehicles as they operate, it is possible to collect Signals of OPportunity (SOP), for instance, Wi-Fi samples that can be added to the radio map, thus keeping it updated.

5.1 Introduction

Research papers about Indoor Position Systems (IPSs) usually present a general description of the solution and technologies but lack an analysis of the requirements and the deployment effort to set up the system,

e.g., the time and cost to set up, whether it requires the installation of infrastructure, or if a calibration process is necessary. Moreover, most systems are only evaluated in the laboratory or very small spaces. In industrial environments there are other challenges regarding the deployment of an IPS, for instance, there are spaces that often change the indoor layout due to reconfiguration of production lines, and industrial spaces whose floor plan is not available or is not detailed enough.

Building a radio map for Wi-Fi fingerprinting is a laborious and time-consuming task because it requires the mapping of RPs and the collection of Wi-Fi samples at each RP. The complexity of this task increases proportionally to the size of the building and the grid size that defines the distance between neighbor RPs. For instance, if the grid size is denser, more RPs are surveyed so there will be a higher cost and the accuracy is expected to increase. Hence, there is usually a trade-off between cost and accuracy when deciding the grid size of the radio map. In industrial environments, this task is even more challenging because it disrupts the operation of vehicles and human operators, thus it should be avoided at the expense of financial loss. Once finished, the radio map represents a snapshot of the building's radio environment at that particular time. As previously discussed, with time, the radio map becomes outdated due to many reasons, such as propagation effects, changes in the indoor layout of the building, moving furniture, or the addition or removal of APs. Therefore, the radio map needs to be periodically updated, leading to additional costs in terms of time and human resources. Although there are several methods to build radio maps, as described in Section 2.4, they usually require expensive hardware (e.g. LiDAR or radar in SLAM-based solutions), depending on user feedback in collaborative solutions, or requiring dedicated hardware to collect Wi-Fi samples autonomously in the case of interpolation methods.

Usually, the requirements and deployment configurations of IPSs are static and strict, for instance: deployment of infrastructure, e.g., UWB which is expensive and in some scenarios impossible to deploy; calibration (or site survey), necessary in fingerprinting-based methods (e.g. Wi-Fi or Bluetooth) that depend on a radio map that needs to be created and periodically updated; floor plan information, which is frequently used in IPSs especially the ones based on PFs, however, no floor plan or only a partial floor plan (low detail) may be available in some scenarios. Moreover, in many environments, floor plans frequently change due to alterations in the indoor layout, making previous floor plan models outdated. Since most IPSs have strict deployment configurations, they do not adapt well and do not work properly if the requirements cannot be ensured. To address the above-mentioned challenges, this chapter describes a new method for industrial

positioning with the following features:

- It does not require the installation of infrastructure since it takes advantage of the existing Wi-Fi infrastructure.
- It can operate with a complete floor plan, a partial floor plan or with no floor plan at all.
- When it is possible to perform a site survey, it can operate with a sparse radio map with just a few reference points, referred to as initial Radio Map (iRM). It also works without the iRM when it is not possible to perform a site-survey, but in this case it requires a partial or complete floor plan.
- Accurate vehicle tracking with a low-cost positioning module, composed of low-cost IMUs, an encoder and Wi-Fi interfaces.
- Mapping of the radio environment, vehicles operating act in a SLAM-based approach, as they are tracked by the IPS, they can annotate new Wi-Fi samples to build and maintain the radio map.
- Deployment effort is greatly reduced since it only requires information regarding the floor plan or the iRM to work.
- Maintenance is unnecessary since the radio map is automatically updated by vehicles operating, which annotate new samples and add them to the radio map. The set of Wi-Fi samples that are autonomously annotated by vehicles in operation are referred to as vehicle Radio Map (vRM).

The vRMs are self-healing because they grow and are updated dynamically, allow to build and maintain the building's radio map, and adapt to variations in the radio environment. In industrial environments, the vRMs can be used with Wi-Fi fingerprinting to locate human operators, industrial vehicles, or other assets, allowing to improve logistics, monitoring of operations, and safety of operators.

5.2 Architecture

Figure 5.1 shows the architecture of the self-healing radio maps solution, which comprises one or multiple industrial vehicles equipped with low-cost sensors. The localization and tracking of industrial vehicles, equipped with low-cost sensors, is achieved with an enhanced version of the Tight Coupling approach introduced in Chapter 4, which allows to automatically annotate and add new samples to the vRM, without human intervention.

Each vehicle includes an enhanced version of the Tight Coupling approach introduced in Chapter 4,

used to perform sensor fusion of data from sensors producing position estimates and corresponding confidence values which allows to automatically annotate and add new samples to the vRM, without human intervention. The radio map is used by the PF in the particles' weighting phase. The radio map is composed of the vRM and the iRM (optional). The vRM contains Wi-Fi samples annotated by vehicles and the iRM contains (previously collected) Wi-Fi samples that serve as a starting point for the system (traditional radio map). The confidence metric, already introduced in Section 4.6.6, is based on the dispersion of particles and determines whether the latest position estimate is reliable or not. When the confidence is high, it allows to build and maintain the vRM by assigning a position estimate to the Wi-Fi samples collected by vehicles.

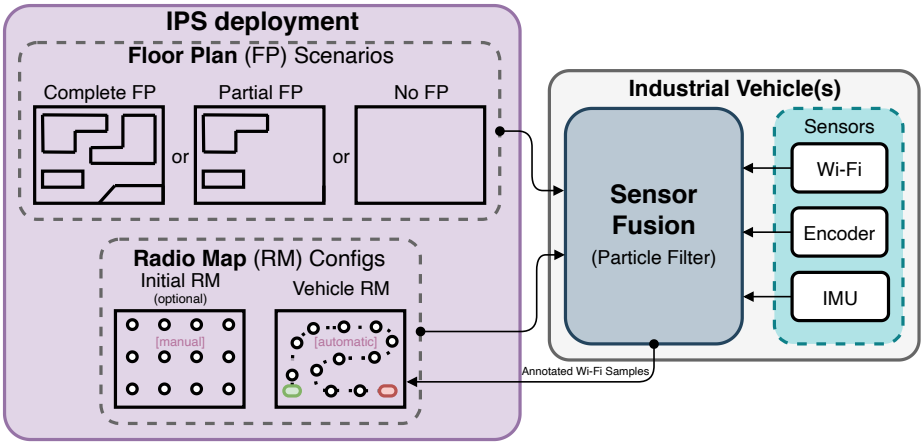


Figure 5.1: Architecture of the self-healing radio maps solution.

5.3 Scenarios and Radio Map Configurations

The navigable areas where industrial vehicles operate are important, as they can be represented in a floor plan to assist in indoor tracking and localization based on a PF. It allows improving the estimated trajectory by removing the particles that hit walls or obstacles or move into non-navigable areas.

The characteristics of industrial environments can vary significantly, depending on the business sector, e.g., it can be a simple storage warehouse or a complex space with several production lines and storage areas. Some industrial spaces have detailed floor plans available. There are other spaces where a large

part is an open area and only a partial floor plan is available, and there are also cases where the floor plan is not available. In order to account for these possibilities, three possible scenarios are considered (Figure 5.2):

- **complete Floor Plan (cFP)**: complete floor plan of the space is available;
- **partial Floor Plan (pFP)**: partial floor plan of the space is available (some areas are mapped in the floor plan, but there is a large open space);
- **no Floor Plan (nFP)**: no floor plan available, large open area.

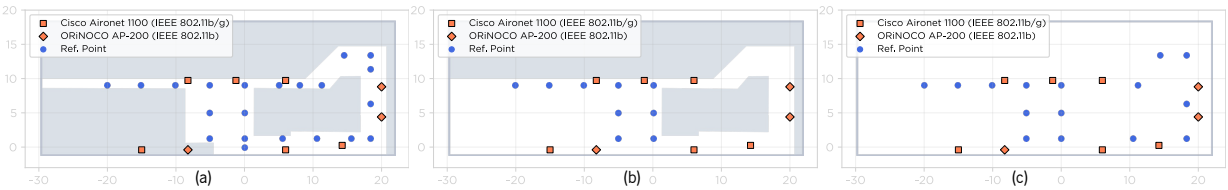


Figure 5.2: Considered floor plan scenarios: (a) complete Floor Plan (cFP); (b) partial Floor Plan (pFP); and, (c) no Floor Plan (nFP).

As previously explained, a radio map is necessary when using a PF to perform sensor fusion of Wi-Fi with motion sensors because the radio map serves as an absolute position reference. This information is used when updating the particles' weights to correct the estimated trajectory. Without the radio map, the PF works as a pure DR solution using only the motion sensors, and if available, the floor plan to assist in correcting the estimated trajectory.

In some situations, it might not be possible to build an iRM because the plant operates 24/7. In these cases, the Vehicle Dead Reckoning (VDR) trajectory containing drift is corrected by detecting collisions of particles in walls or obstacles, defined in the floor plan. In order to consider the existence of an iRM or its absence, the proposed solution may operate using the following radio map configurations:

- **Vehicle Dead Reckoning (VDR)**: without iRM and vRM, pure dead reckoning;
- **vehicle Radio Map (vRM)**: without iRM, but with vRM;
- **full Radio Map (fRM)**: full radio map composed of the iRM and the vRM.

Since building the iRM is time-consuming and difficult to accomplish in large scenarios, the iRM used in experiments is a sparse radio map (with a few samples) whose RPs are represented in blue in Figure 5.2.

These scenarios and radio map configurations represent the majority of IPS deployment constraints

in industrial environments. The idea is to evaluate the proposed system under different conditions to determine the minimum requirements that allow accurate localization of industrial vehicles to enable the construction and maintenance of radio maps.

5.4 Enhanced Particle Filter

The enhanced PF that will be introduced in this section is an improved version of the Tight Coupling implementation (described in Section 4.6). Several aspects differentiate this version from the previous one, namely: (i) it works in different scenarios, with and without the floor plan of the building; (ii) it uses a new method to update particles' weights using weighted k-NN to estimate Wi-Fi samples similarity at the position of each particle, and dynamically adjusting parameters; (iii) it works with or without an iRM.

The enhanced PF works as follows. First, particles are created around the vehicle's initial position. Second, as new sensor readings are received, particles' states are updated considering noise in the sensors. Third, particle's weights are updated based on Wi-Fi samples similarities, by comparing a new received Wi-Fi sample to the ones in the radio map (note that the radio map is dynamic and evolves with time because it includes samples of the vRM). Fourth, over time, some particles disperse due to the noise added to the sensors' readings, and when their weights become low, they are removed in the resampling process. In scenarios that consider a floor plan, particles that hit walls or obstacles are assigned a zero weight. In the resampling step, the particles with lower weights are removed and the particles with higher weights are duplicated to keep the number of particles constant. The continuous updating of particles' weights and the resampling process allow the estimated position to remain near the real position, minimising the cumulative drift in the heading. A reliability metric, named confidence, allows to determine when to add annotated Wi-Fi samples to the radio map.

As previously defined, the PF consists of a set of $\mathcal{N}$ particles defined as $p = (w, x, y, z, h, ho)$, where w represents the weight of the particle, x, y, and z represent the position coordinates of the particle, h represents the heading, and ho represents the heading offset used to improve heading estimates. In the following, the main differences between the previous PF based on Tight Coupling and this one are presented.

5.4.1 Initialization

In industrial environments, vehicles are usually parked in known positions (e.g. charging stations or dedicated parking spots) and start their operation from a known pose (position ρ_{ini} and heading h_{ini}). This information is used to initialize particles around the vehicle's position. As an advantage, the large maximum errors usually found during the first iterations of the PF are eliminated, and new Wi-Fi samples can be added to the vRM from the start.

Particles are uniformly distributed within a radius r of a given initial position ρ_{ini}, have an initial heading defined as h_{ini}, and an initial heading offset defined as $ho =$U$(0,n_{ho})$. As a result, each particle has a unique position and heading offset.

5.4.2 Update Weights

As previously explained, the purpose of the method to update particles' weights is to assign higher weights to particles that are closer to the real position and lower weights are assigned to particles that are probably further away from the real position. The Wi-Fi information available on the radio map is essential to determine where the vehicle is located. Although it does not provide an accurate position estimate, comparing a Wi-Fi sample with the ones in the radio map allows to identify the areas where the vehicle is more likely to be.

In this version of the PF, it is necessary to refine the method to update the particles' weights. Contrarily to the original Tight Coupling implementation, where a dense radio map was considered, in this implementation, there is a high probability that the radio map is sparse or empty upon the PF initialization.

This poses a challenge because in the Tight Coupling implementation the particles' weights were obtained based on the similarity of the Wi-Fi sample nearest to the particle's position and there was always a radio map Wi-Fi sample close to the particle's position. To overcome this issue, it is proposed the addition of the weighted k-NN algorithm to estimate the normalized similarity at the position of each particle, based on the radio map Wi-Fi samples nearer the particle's position. Instead of considering only the Wi-Fi sample that is closer to the particle, this approach considers several Wi-Fi samples closer to the particle. The distances between the particle and the positions of each radio map Wi-Fi sample are converted into weights in the algorithm so that Wi-Fi samples that are closer to the particle's position have more influence on the

estimated similarity. As a result, the estimated normalized similarity is more reliable. As a consequence, this approach is more complex and requires more computing power. In the following, a brief description of this method is made.

In the first step of this process, the normalized similarity between the Wi-Fi sample and the radio map Wi-Fi samples is computed, as previously described in Section 4.6.3. Then, the normalized similarity at the position of each particle is obtained using the weighted k-NN algorithm, as described in Algorithm 4. First, the Euclidean distances between the particle's position and the radio map Wi-Fi samples are calculated (line 2). Second, these distances are sorted in ascending order (line 3). Third, the distances are converted into weights using the Inverse Distance Weighted (IDW) method (lines 5 and 6), which assigns higher weights to shorter distances and lower weights to larger distances. Fourth, the estimated similarity is obtained using the weighted average of the k_{uw} Wi-Fi samples closer to the particle position (lines 7 and 8).

Algorithm 4 Estimate particle normalized similarity.

Input

k_{uw} - parameter of the weighted k-NN algorithm
p - particle
WS_ρ - set of Wi-Fi samples positions
S_n - set of Wi-Fi samples normalized similarities

Output

$\hat{s_n}$ - estimated similarity at particle's position

1: **procedure** Estimate Particle normalized Similarity
2: $D = \mathsf{d_E}(p, WS_\rho)$ //obtain set of Euclidean distances between the particle's position and each Wi-Fi sample
3: $D = \mathsf{sort}(D)$
4: $\mathcal{W} = \{\}$ //initialize set of weights
5: **for** $d \in D$ **do**
6: $\mathcal{W} = \mathcal{W} \cup \left\{\frac{1}{d^2}\right\}$ //convert distance into weight using IDW and add it to the set of weights
7: $\hat{s_n} = \frac{\sum_{i=1}^{k_{uw}} \mathcal{W}^i \times S_n^i}{\sum_{i=1}^{k_{uw}} \mathcal{W}^i}$ //compute the normalized similarity using the weighted k-NN algorithm
8: **return** $\hat{s_n}$

Figure 5.3 shows an example of the estimated similarities at the positions of particles using this approach. The 5 particles refer to the state of the PF at a given moment. The 3 radio map samples show the normalized similarity after the collection of a new Wi-Fi sample. The estimated similarity of each particle using the algorithm above is shown at the position of each particle, as expected, each Wi-Fi sample of the

radio map contributes to the estimated similarity, being that the nearest Wi-Fi samples have more influence over the estimated value. The k_{uw} parameter defines the number of nearest Wi-Fi samples considered in the algorithm. Once the particle's normalized similarity is obtained, its weight is updated considering the same expression used in the Tight Coupling approach, defined by Equation 4.6.2.

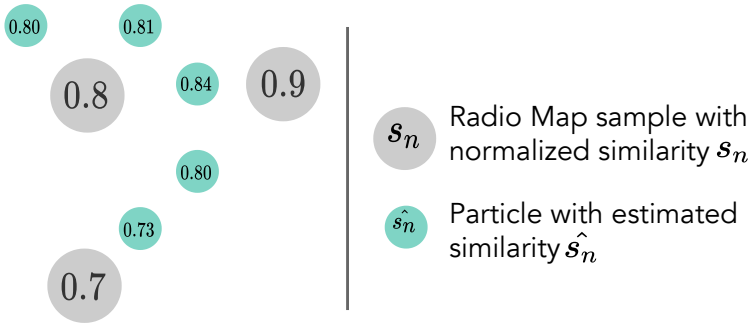


Figure 5.3: Estimated similarity of each particle.

5.4.3 Dynamic Alpha

As previously introduced in Section 4.6.6, the α parameter allows to define how much the particle's weight is influenced by the latest Wi-Fi sample. In the original Tight Coupling approach the expression $\alpha = 0.6 - 0.6 \times C$ was defined based on simulation results. In this version, the expression that defines α was refined so that α varies according to the distance between the particle and the nearest radio map Wi-Fi sample, being that when the distance is small, α is larger and vice versa. The rationale is that particles far from any radio map Wi-Fi sample are in positions not yet mapped and, therefore, their weights should not be significantly affected by the latest Wi-Fi sample. This allows the vehicles to map the space in areas where it has not been yet, without having a negative impact on the particles' weights that would cause them to be removed in resampling.

Figure 5.4 shows how α is obtained based on the distance (d) between the particle and the nearest radio map Wi-Fi sample. When the particle is close to the nearest radio map Wi-Fi sample ($d \leq 1\ m$), α is assigned α_{max}. For $1\ m < d < d_{max}$, a linear function is considered where α decreases as d increases. When $d \geq d_{max}$, α is assigned the α_{min} value.

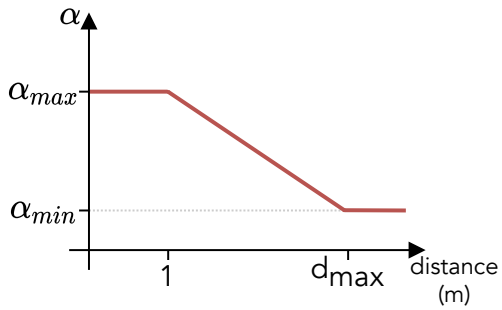


Figure 5.4: Dynamic α based on distance between particle and nearest radio map Wi-Fi sample.

5.4.4 Other Methods

The particles movement model, the resampling method, the confidence metric, and the method to obtain an estimated vehicle pose, are the same as the ones of the PF based on Tight Coupling, described in Section 4.6.

5.5 Vehicle (Self-healing) Radio Map

Vehicles contribute to build and maintain the vRM by adding annotated Wi-Fi samples to the vRM whenever the confidence of the position estimates is high enough. The vRM is different from a traditional radio map because Wi-Fi samples were collected automatically and have an associated error, as opposed to a manually created radio map where Wi-Fi samples have zero error. The main advantages of vRMs are as follows: no effort to build the radio map as it is an automatic process to map the radio environment; vRMs are self-healing since they overcome radio map degradation and adapt to the changes in the radio environment automatically; vRMs can be used for plain Wi-Fi fingerprinting-based positioning in other applications.

Upon receiving a new Wi-Fi sample, the vehicle checks if the confidence is high enough before deciding to add it to the vRM. When the confidence is equal or greater than a given threshold (C_{th}), the latest position estimate and confidence are assigned to the Wi-Fi sample, creating a vehicle Wi-Fi sample, defined as:

$$vws = (\rho_{vehicle}, ws, C) \tag{5.5.1}$$

where ws represents the signal strength values of detected APs, $\rho_{vehicle}$ represents the current position

of the vehicle and C represents the current confidence.

The quality of annotated Wi-Fi samples depends on the positioning performance of the PF. Higher precision and accuracy lead to a better vRM.

Being part of the radio map (considering that the full radio map is composed of the iRM+vRM), the vRM has a significant role in the method to update particles weights, given that it evolves dynamically as new samples are annotated and added. Newly annotated Wi-Fi samples are only added to the vRM after a 120 s time delay, in order to prevent a dragging effect that would cause the estimated trajectory to be delayed with respect to the true trajectory of the vehicle. Without this delay, the latest Wi-Fi samples (collected moments before) would be considered in the update weights method, which would cause the particles to have a high similarity with those Wi-Fi samples.

5.6 Real-world Experiments

Several experiments were conducted in the real-world to evaluate the enhanced PF. The purpose of these experiments is to assess whether it is possible to accurately track industrial vehicles to build/maintain self-healing radio maps under different floor plan scenarios with different radio map configurations.

Conducted experiments included four phases. First, the development of a mobile unit equipped with several sensors. Second, the setup at the testing space (Figure 5.5) with the mapping of RPs and the collection of Wi-Fi samples for the iRM used in some experiments. Third, collecting the test data. Fourth, offline processing of the collected data.

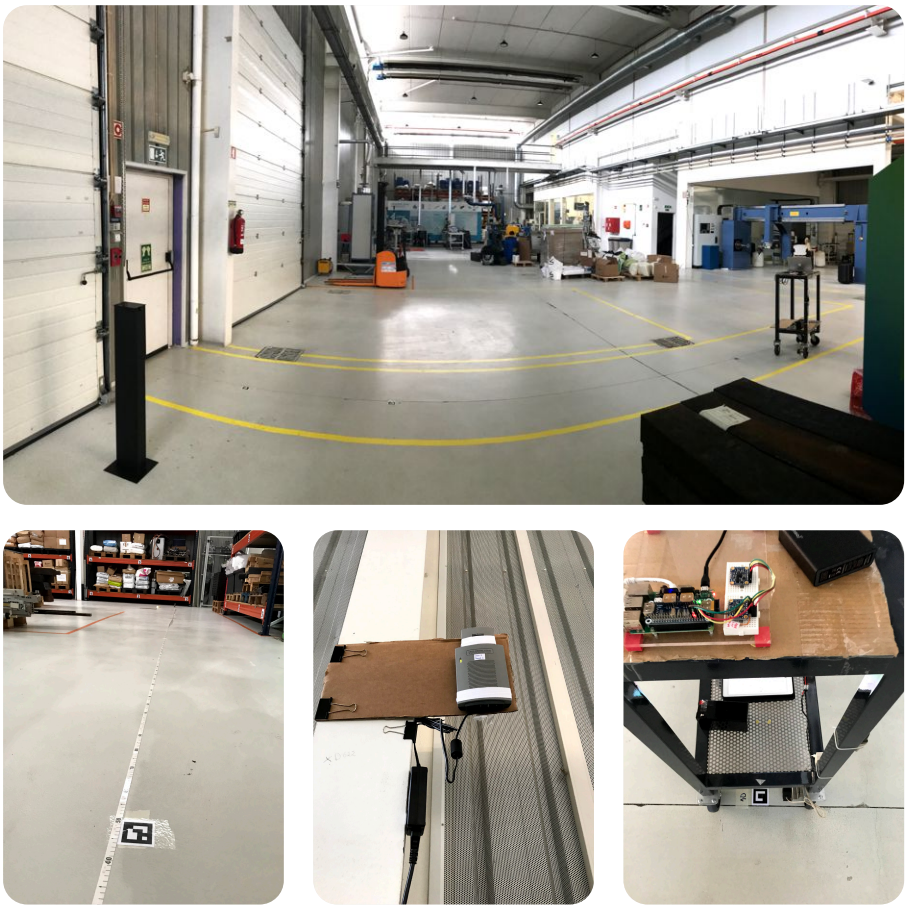


Figure 5.5: Setup at the PIEP building: mapping of the RPs (lower left); one of the installed APs (lower center); collection of Wi-Fi samples for the iRM (lower right).

5.6.1 Setup and Mobile Unit Prototype

Real-world experiments took place at the PIEP building, already described in Section 4.9.1. The setup process is depicted in Figure 5.5, showing the building, the process in the mapping of RPs, the installation of APs, and the site survey to collect Wi-Fi samples. The positions of APs available in the building are represented by the points in the salmon color in Figure 5.2.

The iRM, necessary in experiments that consider the full radio map (iRM+vRM), is a sparse radio map with $\approx$5 m between RPs (blue points in Figure 5.2). A site survey (Figure 5.5 (lower right)) was conducted to build the iRM with the collection of 40 Wi-Fi samples per RP, 10 in each direction (up, down, left, and right). Since the industrial vehicle used in previous experiments was not available, test data was collected

using a mobile unit, a manually pushed trolley to emulate an industrial vehicle (Figure 5.6).

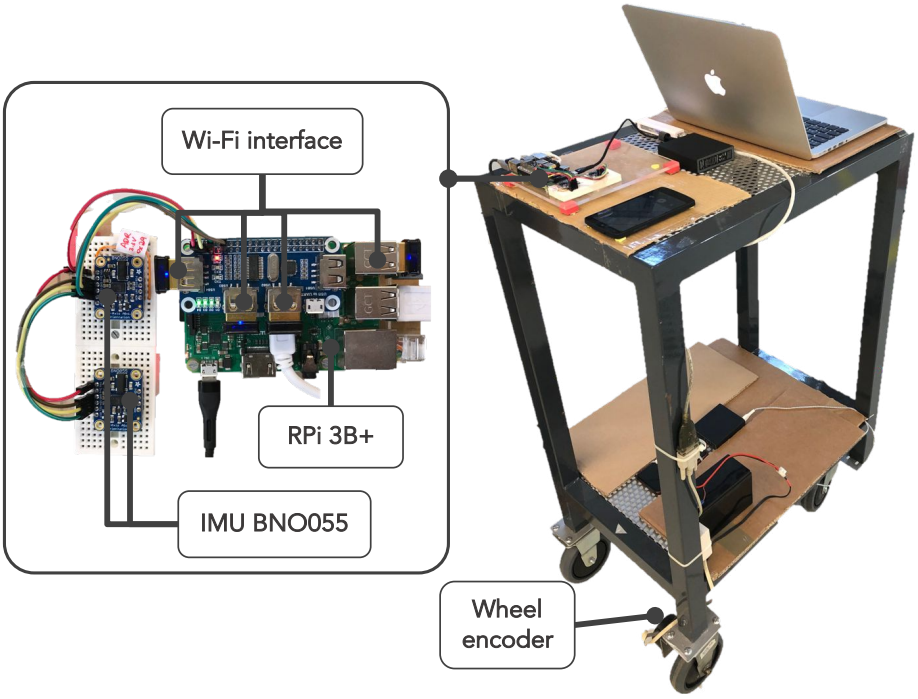


Figure 5.6: Mobile unit prototype with installed sensors.

The mobile unit is equipped with the positioning module, composed of: a Raspberry Pi Model 3B+; four external Wi-Fi interfaces (Edimax EW7811-Un); two IMUs (Adafruit BNO055) to measure the heading; and, an absolute encoder (US Digital A2) to measure the displacement. The Raspberry Pi is used as a computer to collect data from sensors. As previously stated, using multiple Wi-Fi interfaces leads to improvements in the positioning performance [28], therefore, this method was adopted to improve the accuracy of the proposed solution.

Figure 5.7 represents how sensors are connected to the Raspberry Pi, the communication interface of each sensor and the power supply specifications necessary to power the Raspberry Pi and the encoder sensor. The approximate retail cost of this prototype is 421 € considering: Raspberry Pi (35 €), encoder (255 €), IMUs (2x 30 €), Wi-Fi interfaces (4x 9 €), 12V battery to power encoder (15 €), 10,000 mAh PowerBank to power the Raspberry Pi (20 €). Prices were gathered from retailers web sites.

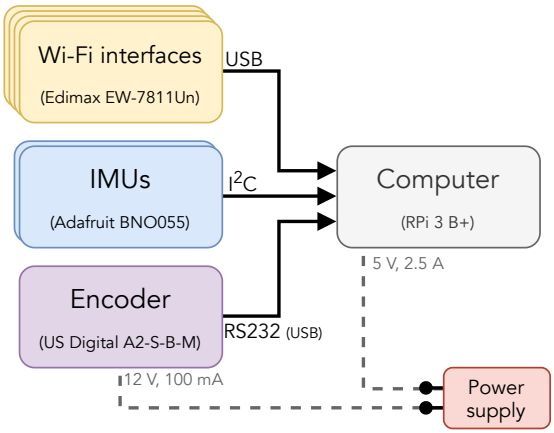


Figure 5.7: Hardware diagram of the positioning module of the mobile unit.

5.6.2 Low-cost IMU

In previous experiments (see Section 4.9), industrial-grade IMUs (Xsens MTi-300 AHRS) were used for obtaining the heading of the vehicles. Although these sensors are reliable and less prone to drift and magnetic perturbations, they are very expensive (each costing around 2900 €), making the IMU the most expensive part of the positioning module. Using lower-cost IMUs allows to drastically reduce the cost of the positioning module, but as a consequence, these sensors are noisier and less accurate than industrial-grade IMUs. In these experiments, the Adafruit BNO055 IMU was used as a substitute for the industrial-grade IMUs used in previous experiments. This sensor was chosen for several reasons. First, it is supported by the Raspberry Pi, therefore it can be easily integrated into this solution. Second, it is cheap and easily found in online stores. Third, it performs in-chip data fusion capable of producing absolute orientation in quaternion or Euler angle formats at an update rate up to 100 Hz. Fourth, it allows to collect raw measurements from the accelerometer, gyroscope and magnetometer which can be used in a sensor fusion algorithm to estimate the sensor's orientation. Fifth, the research community has revealed an increasing interest in this sensor for positioning and navigation, especially in recent years as shown by a search in the Scopus database with the query "(ALL(bno055) AND TITLE-ABS-KEY(positio* OR navigat*))".

Overall the Adafruit BNO055 IMU is a great sensor for research due to it being low-cost and simple to work with. A Madgwick filter [122] implementation was tested to perform sensor fusion from the low-cost

IMU raw measurements, but the obtained results were no better than the ones provided by the integrated filter, which had less noise and drift than the results achieved with the Madgwick filter, therefore the integrated filter of the IMU was used instead. In performed experiments, the Adafruit BNO055 sensors were configured with the NDOF operation mode with 50 Hz data output rate. With the NDOF mode[10], the absolute orientation data is calculated from the fusion of the accelerometer, gyroscope and magnetometer. The main advantages of combining all three sensors are a fast calculation, resulting in high output data rate, and high robustness from magnetic field distortions.

5.6.3 Particle Filter Parameters

Table 5.1 lists the parameters used in performed experiments. The same parameters were used in all experiments in order to assess the versatility of the PF with the same parameters in different floor plan scenarios and radio map configurations.

Table 5.1: Particle filter parameters.

Param	$\mathcal{N}$	r	n_d	n_θ	n_{ho}	
Value	3000	1 m	N(0, 0.012 m)	N(0, 1°)	N(0, 8°)	
Param	k_{uw}	α_{min}	α_{max}	d_{max}	w_{th}	C_{th}
Value	3000	0.05	0.3	2 m	0.2	0.75

5.6.4 Results

The mobile unit was manually pushed in five trajectories at a speed of $\approx$1 m/s, to construct the vRM. Test data was collected from the mobile unit's sensors and served as input for the proposed PF. Annotated Wi-Fi samples accumulated in the vRM and used from one trajectory to the next one. Therefore, the vRM used in the last trajectory includes all Wi-Fi samples that were annotated in previous trajectories.

The positioning error reported in this section was obtained by comparing estimated positions with ground truth data collected by a video camera mounted in the mobile unit. The video camera recorded

[10]https://www.bosch-sensortec.com/products/smart-sensors/bno055/

tags affixed to the floor (in known positions) when the mobile unit passes by them. The error metric is the Euclidean distance between the position estimates and the ground truth.

The PF was run for 9 experiments considering the combinations of 3 scenarios (cFP, pFP, and nFP) with 3 radio map configurations (VDR, vRM, and fRM). The PF was run three times for each trajectory, and the results include position estimates of all runs. The positioning results of performed experiments are reported in Table 5.2.

Table 5.2: Positioning results for different scenarios and radio map configurations (in meters).

	cFP			**pFP**			**nFP**		
	VDR	**vRM**	**fRM**	**VDR**	**vRM**	**fRM**	**VDR**	**vRM**	**fRM**
Mean	0.88	0.88	**0.67**	1.45	0.90	**0.62**	3.92	1.91	**0.63**
P_{75th}	1.17	1.20	**0.88**	1.96	1.11	**0.88**	6.03	2.70	**0.87**
P_{99th}	2.61	2.38	**1.71**	3.81	2.76	**1.54**	10.92	6.06	**1.64**
Max	2.89	4.27	**2.34**	3.84	3.90	**2.12**	24.70	7.56	**3.53**

cFP – complete Floor Plan; **pFP** – partial Floor Plan; **nFP** – no Floor Plan;
VDR – vehicle Dead Reckoning; **vRM** – vehicle Radio Map; **fRM** – full Radio Map.

As expected, the scenario with better positioning results is the one that uses the complete Floor Plan (cFP). In this scenario, the floor plan allows to remove particles that hit walls or obstacles improving estimates. In addition, this scenario is the one that better performs with VDR, showing that an iRM is not required to achieve good results.

The scenario with the partial Floor Plan (pFP) shows gradual improvements in the positioning accuracy as a richer radio map is used. As expected, the worst results are observed with VDR, then the configuration with vRM has improvements below the P99th, showing that this scenario benefits from Wi-Fi samples that are added to the radio map. This is particularly important in open areas that have no constrains from the floor plan, where drift in the heading can significantly affect the estimated trajectory.

The scenario with no Floor Plan (nFP) is the most challenging, showing large errors with VDR. This scenario has significant improvements in the vRM configuration because of the annotated Wi-Fi samples. To annotate Wi-Fi samples with higher accuracy, it is necessary an iRM, because it allows to keep particles within the neighborhood of RPs which minimizes the maximum error.

All three scenarios (cFP, pFP and nFP) achieve the best results with the fRM configuration because

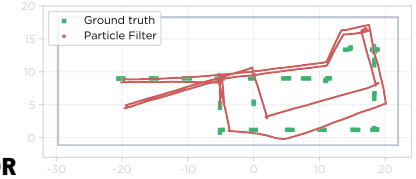
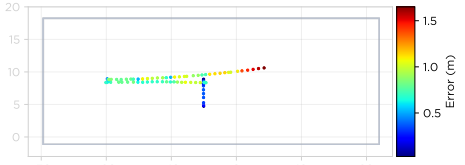
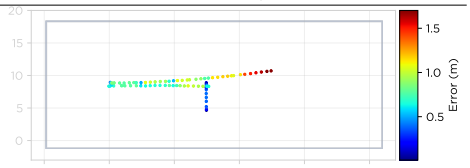
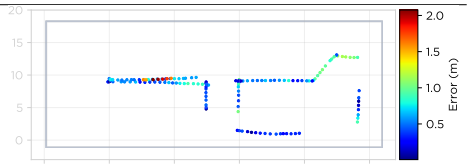
it allows to update particles' weights right from the beginning (particles initialization), which consequently allows to minimize large errors and improve the overall accuracy of the system. Since the overall performance of the PF is improved, the annotated Wi-Fi samples have a lower error associated. The scenario that benefits the most from using the fRM configuration is the nFP scenario because, without the floor plan, the only reference data used to correct the drift is the iRM and the vRM.

Estimated trajectories and annotated Wi-Fi Samples

Tables 5.3, 5.4, and 5.5 compare how the system is able to build the vRM in each scenario. Each table shows the PF estimated trajectory as well as the annotated Wi-Fi samples and the positioning error of each Wi-Fi sample. The same trajectory is considered in these tables to allow a direct comparison between the estimated trajectories and the number of annotated Wi-Fi samples.

Table 5.3 shows the estimated trajectories and annotated Wi-Fi samples for the scenario without floor plan (nFP). In the VDR and vRM configurations, it has difficulty in correcting the estimated trajectory, because there is no reference that favors particles that are in areas where it is more likely for the vehicle to be at. Besides, without the floor plan information to help correcting the trajectory, the confidence is high during the first iterations of the PF but after some time, it drops below the C_{th}, and no additional Wi-Fi samples are added to the vRM. As a result only a few samples are annotated and added to the vRM. When the full radio map is considered (fRM configuration), it allows to improve the estimated trajectory and consequently, the number of annotated Wi-Fi samples also increases because this allows to update particles' weights based on the Wi-Fi information from the iRM and vRM.

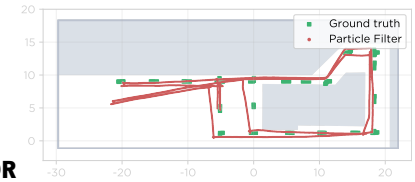
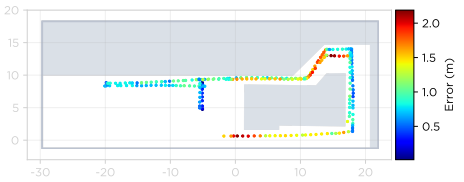
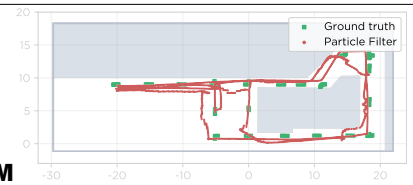
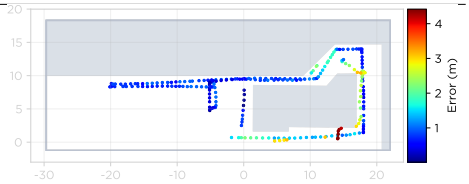
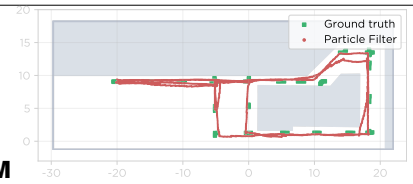
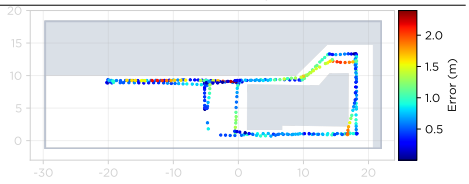
Table 5.3: Annotated Wi-Fi samples in trajectory 1 for the no Floor Plan (nFP) scenario.

	Estimated Trajectory	**Annotated Wi-Fi samples**
VDR		 87 Wi-Fi samples
vRM		 88 Wi-Fi samples
fRM		 151 Wi-Fi samples

VDR – vehicle Dead Reckoning; **vRM** – vehicle Radio Map; **fRM** – full Radio Map.

The benefit of using a partial floor plan (pFP scenario) is observed in Table 5.4. It allows to correct the trajectory when no radio map is available (VDR configuration). In this scenario, the vRM has an important role in reducing the drift present in the heading, as it can be seen in the estimated trajectory and in the number of annotated Wi-Fi samples. The drift in the estimated trajectory is not present in the estimated trajectory when the full radio map is considered. Again, the iRM information is essential to correct the trajectory in areas where the floor plan is not available. As a consequence, a significant amount of Wi-Fi samples are annotated in comparison with the VDR and vRM configurations.

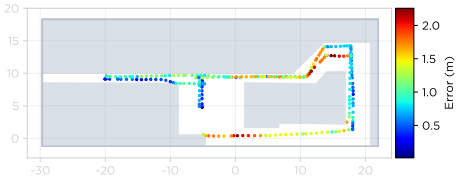
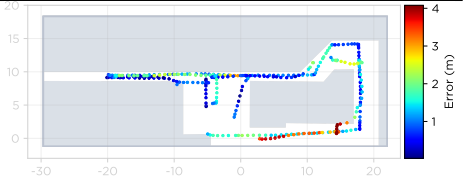
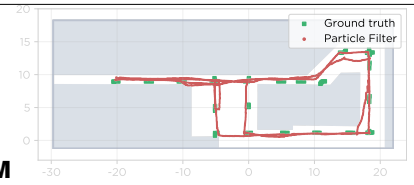
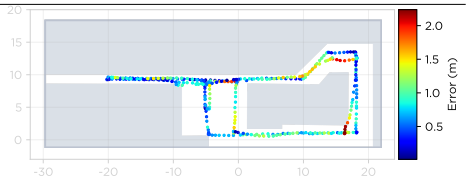
Table 5.4: Annotated Wi-Fi samples in trajectory 1 for the partial Floor Plan (pFP) scenario.

	Estimated Trajectory	**Annotated Wi-Fi samples**
VDR		 255 Wi-Fi samples
vRM		 265 Wi-Fi samples
fRM		 339 Wi-Fi samples

VDR – vehicle Dead Reckoning; **vRM** – vehicle Radio Map; **fRM** – full Radio Map.

A larger amount of Wi-Fi samples are annotated when the complete floor plan is available (cFP scenario), as shown in Table 5.5. The floor plan allows to correct the trajectory even when no radio map exists (in the VDR configuration) allowing to map the radio environment with 260 annotated Wi-Fi samples. In vRM and fRM configurations, the PF annotates even more Wi-Fi samples. Having the complete floor plan provides a significant advantage for building the radio map automatically because it helps to correct the estimated trajectory, which in turn leads to a larger number of annotated Wi-Fi samples.

Table 5.5: Annotated Wi-Fi samples in trajectory 1 for the complete Floor Plan (cFP) scenario.

	Estimated Trajectory	**Annotated Wi-Fi samples**
VDR		 260 Wi-Fi samples
vRM		 341 Wi-Fi samples
fRM		 383 Wi-Fi samples

VDR – vehicle Dead Reckoning; **vRM** – vehicle Radio Map; **fRM** – full Radio Map.

Figure 5.8 shows the confidence vs error of the position estimates obtained for all trajectories of the nFP scenario considering the vRM and fRM configurations. A higher density of position estimates is observed when $C > 0.6$ typically with errors lower than 2 m (red areas). When the confidence is lower than that, larger errors are observed, especially in the vRM configuration because the PF does not have reference data, and is not efficient in removing particles. When the iRM is provided (fRM config), confidence is high most of the time, except for a few cases (larger errors when $C < 0.6$).

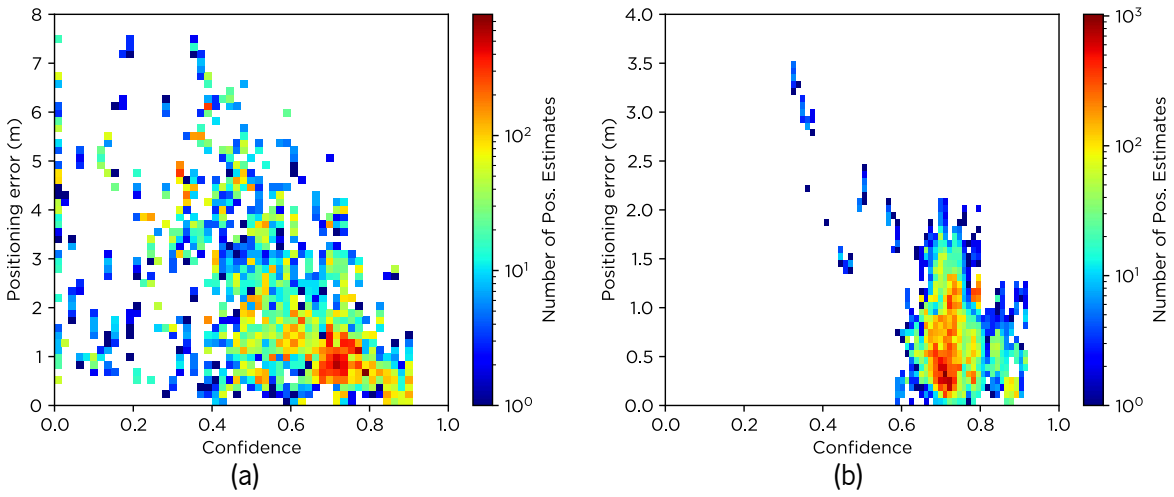


Figure 5.8: Confidence vs positioning error of all trajectories for the no Floor Plan (nFP) scenario: (a) vRM configuration; (b) fRM configuration.

In the vRM configuration (Figure 5.8 (a)), there is a statistically significant negative correlation between the confidence values and the positioning error, with $\mathcal{R}_{Pearson} = -0.64$, $p_{value} = 0.0$. These results are in accordance to those reported in Section 4.6.7 when the simulations were conducted to validate that the confidence function could be used as a reliability measure.

In the fRM configuration (Figure 5.8 (b)), the correlation between the confidence and the positioning error is $\mathcal{R}_{Pearson} = -0.122$, $p_{value} = 0.0$. The correlation coefficient close to zero suggests that there is no correlation in this particular case. This is justified by the low density of position estimates with large positioning errors. The maximum error is low when comparing to the other configuration, with 3.53 m (fRM configuration) vs 7.56 m (vRM configuration). The correlation coefficient is close to zero because the vast majority of position estimates has high confidence and low positioning error associated, thus, there is a low dispersion of samples in the density plot, which consequently leads to a low correlation coefficient.

The CDF curves in Figure 5.9 show that the best performance is achieved in fRM configurations, followed by the combinations pFP+vRM, cFP+vRM and cFP+VDR which have similar results. For these configurations, over 90% of all position estimates are below 2 m, therefore they are suitable for building and maintaining vRMs. The remaining configurations have larger errors, and should not be used for this application. Gradually worse results are observed for the combinations pFP+VDR, nFP+vRM and nFP+VDR, respectively. Overall, these results show that the best tracking of industrial vehicles is achieved in fRM

configurations, independently of the scenario. In case the iRM is not available, indoor tracking is still possible in the cFP and pFP scenarios. Therefore the system is capable of building the radio map from the ground up, only depending on the floor plan information.

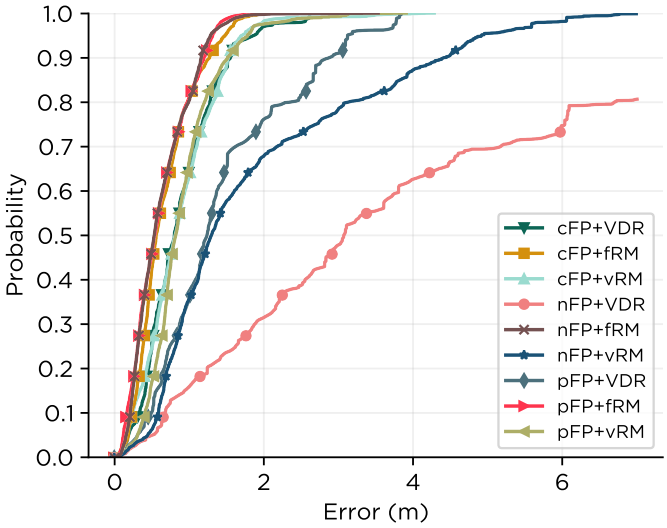


Figure 5.9: CDF curves of all scenarios and radio map combinations.

Computational Complexity - Optimization of Annotated Wi-Fi samples

In the previous experiments reported in this chapter, a large value for k_{uw} was used to consider all radio map samples when updating weights. This includes the vRM and the iRM samples, in cases where the iRM was used. As the number of annotated Wi-Fi samples increases, the number of samples used in the update weights process also increases, leading to a computational complexity of $O(n \times \log n)$. Since the number of particles in the PF remains constant, the computational complexity depends on k_{uw}, therefore it should be a constant value. When low values for k_{uw} were used (e.g. $k_{uw} = 3, 5, 10$), results were worse, therefore it is important to use an adequate k_{uw} value and to determine which Wi-Fi samples should be selected. In the experiments considering all Wi-Fi samples (i.e., large k_{uw}), results were better, showing that it is necessary to use Wi-Fi samples uniformly dispersed through space.

In order to keep the computational complexity constant, simple random sampling was implemented to limit the number of Wi-Fi samples when updating particles' weights. With this approach, every time

the particle's weights are updated, a limited number of Wi-Fi samples are sampled from the radio map. Wi-Fi samples are selected based on a uniform distribution, therefore each Wi-Fi sample has the same probability of being selected resulting in a set of Wi-Fi samples that are uniformly dispersed through space. The k_{uw} used in these experiments was the same as the number of Wi-Fi samples selected.

Table 5.6 contains the results of the nFP+fRM scenario using simple random sampling to select a limited set of Wi-Fi samples to be used when updating the particles' weights. This scenario was chosen because it is one of the most challenging scenarios, as it does not rely on the floor plan, and it is expected even better performance if the other scenarios were considered.

Table 5.6: Positioning results of nFP+fRM scenario using simple random sampling to limit the number of Wi-Fi samples (errors in meters).

	No. of Wi-Fi Samples						
	25	**50**	**100**	**150**	**200**	**300**	**400**
Mean	2.68	2.03	0.89	0.71	**0.63**	**0.63**	**0.63**
P_{75th}	4.15	3.05	1.02	0.99	0.89	0.87	**0.86**
P_{99th}	10.43	5.87	4.49	2.53	1.71	**1.66**	1.69
Max	15.98	8.07	6.23	5.89	4.37	3.74	**3.62**

Table 5.6 shows a clear improvement in the positioning performance is observed as the number of Wi-Fi samples increases. By limiting the number of Wi-Fi samples to 200 or higher, it is possible to achieve similar results to the ones that consider all Wi-Fi samples, as reported in Table 5.2. When the number of selected Wi-Fi samples is larger than 200, results seem to stabilize, and the main difference is observed in the maximum error which tends to decrease as the number of Wi-Fi samples increases. From 200 to 300 Wi-Fi samples, a major improvement is observed in the maximum error. From 300 to 400, no significant improvements are observed. Although the maximum error keeps improving when the number of Wi-Fi samples is larger than 200, the P99th is stabilized, meaning that the maximum error can be attributed to a rare occurrence, therefore, it seems that the cost-benefit between the computational effort and the positioning performance is best when 200 Wi-Fi samples are selected for updating particles' weights. Furthermore, this shows that the computational complexity will not be a hindrance to the positioning system.

A density of 200 WS$/1000\ m^2 = 0.2$ WS$/m^2$ is enough to ensure good positioning results because optimal positioning results were achieved with 200 Wi-Fi samples for a building with an area of about

1000 m^2 (50x20 m). A similar performance is expected in larger buildings as long as the same density of Wi-Fi samples is ensured in the process to update particles' weights.

Vehicle Radio Map vs Traditional Radio Map

As previously mentioned, Wi-Fi radio maps can be explored for many applications including the localization of people or assets in industrial environments. Therefore, vRMs can be of the utmost importance to easily support plain Wi-Fi fingerprinting applications. The only requirement is that a Wi-Fi-enabled device is used to allow indoor localization.

To evaluate how the vRM performs in a Wi-Fi fingerprinting scenario, the positioning performance was compared between the iRM and the vRM obtained with plain Wi-Fi fingerprinting using k-NN with $k = 5$. This experiment was performed for the three floor plan scenarios with the vRM configuration, namely, cFP+vRM, pFP+vRM and nFP+vRM configurations.

In this experiment, the Wi-Fi samples from trajectory 5 (the last one of the set of 5) were used as test data, and the Wi-Fi samples that were annotated in the previous 4 trajectories were used as the vRM. Therefore, the considered vRMs are composed of a total of 1401, 1216, and 247 Wi-Fi samples in cFP+vRM, pFP+vRM and nFP+vRM configurations, respectively.

The iRM is a sparse radio map, composed of 840 Wi-Fi samples with 40 samples collected at each of the 21 RPs, as depicted in Figure 5.2.

Table 5.7 presents the plain Wi-Fi fingerprinting results using the vRM of each scenario. As expected, the results of cFP and pFP are better than the results of the nFP, because they have more Wi-Fi samples and have lower positioning error associated. Results of cFP and pFP are also better than the ones achieved with the iRM. The results achieved with the iRM are slightly worse than the ones achieved with the vRMs in cFP and pFP scenarios, which is expected because the vRMs have more samples with a lower distance between neighboring samples. Worse results are observed in the nFP scenario because there is an area of the building without any annotated Wi-Fi samples. In this configuration (nFP+vRM), there is no floor plan to assist in removing particles and there is no iRM, therefore it is not able to annotate Wi-Fi samples in some areas because there is not enough confidence in position estimates. As previously stated, in order to properly construct a vRM in a scenario without a floor plan, it is necessary to use the iRM. For

that configuration (nFP+fRM), Wi-Fi fingerprinting results are on par with the ones achieved with the other configurations with a mean error of 1.42 m, P99th of 5.54 m and a maximum error of 6.28 m. It is worth noting that these Wi-Fi fingerprinting results are over-optimistic, probably because the training and test data were collected on the same day, and the iRM was collected on the previous day. Usually, Wi-Fi fingerprinting achieves a mean positioning error between 2 and 4 m with a presence of sporadic large errors higher than 10 m [114].

Table 5.7: Plain Wi-Fi fingerprinting results (in meters) of each scenario in comparison to an iRM.

	cFP+vRM	**pFP+vRM**	**nFP+vRM**	**iRM (grid 5x5m)**
Mean	1.32	**1.24**	6.66	1.50
P_{75th}	1.69	**1.59**	8.34	2.24
P_{99th}	**3.57**	4.18	33.93	5.40
Max	**4.98**	6.04	34.31	7.51

cFP – complete FP; **pFP** – partial FP; **nFP** – no FP; **vRM** – vehicle RM; **iRM** – initial RM.

Comparison with Similar Systems

In Table 5.8 a comparison is made between this approach and other solutions for the construction and maintenance of radio maps. It compares the techniques for localization, the sensors used, if the system depends on the user's feedback, whether a radio map is obtained from a site survey or interpolation, and the performance of Wi-Fi fingerprinting achieved.

The proposed system achieved the best Wi-Fi fingerprinting results with the automatically built vRMs. The advantage of using dedicated sensors (i.e., IMU and wheel encoder) to track the vehicle's movement is that it leads to lower error in annotated Wi-Fi samples, which consequently leads to better Wi-Fi fingerprinting results. Using Wi-Fi fingerprinting, the proposed solution achieved 1.32 m, 1.24 m and 1.42 m mean error in cFP+vRM, pFP+vRM and nFP+fRM scenarios, respectively. In [85] an unmanned ground vehicle is used to map the space and collect SOP from Wi-Fi, magnetometer and light sensors. It achieved a mean error of 1.89 m with Wi-Fi fingerprinting using the automatically built radio map. This solution is more expensive than the proposed one, as it requires several sensors, including a LiDAR. The advantages of this solution are that it does not depend on the initial pose of the device and the floor plan, since it uses the LiDAR

Table 5.8: Comparison with similar systems for the construction of radio maps.

Solution	Technology	Sensor(s) / Device(s)	UF*	Type	Initial RM	Mean Error (m)
Proposed solution	Particle Filter	IMU, wheel encoder, Wi-Fi	No	Site survey	No No Yes	1.32 (cFP+vRM) 1.24 (pFP+vRM) 1.43 (nFP+fRM)
[85]	IMLE	LiDAR, magnetometer, light sensor, Wi-Fi	No	Site survey	No	1.89
[42]	Visual Odometry + Particle Filter	Kinect camera, sonars, wheel encoder	No	Site survey	No	5.2
[6]	GP-LVM	Laptop (Wi-Fi)	No	Interpolation	No	3.97
[82]	Wi-Fi fingerprinting	Smartphone (Wi-Fi)	Yes	Site survey	Yes	Between 2 and 4 m
[81]	PDR	Smartphone (Wi-Fi+IMU)	No	Site survey	No	6.9

***UF** – User feedback; **cFP** – complete FP; **pFP** – partial FP; **nFP** – no FP; **vRM** – vehicle RM; **fRM** – full RM.

sensor to generate the map of the space. In [42], the mobile robot equipped with sensors is capable of locating itself using DR and QR codes which are observed by the camera, thus it does not need an iRM. As a disadvantage, it explores several sensors such as sonars, a wheel encoder, and a camera that are more expensive than the ones used in our system. In [6] Gaussian process latent variable models are applied to estimate the locations of Wi-Fi samples collected by a person walking through a building. This process allows to estimate signal strength levels in the building similar to the ones obtained in an iRM. Systems that depend on users' feedback [21, 82], are susceptible to malicious users that can feed the system with wrong information hindering performance and they also require the iRM. The solution described in this chapter does not depend on users' feedback since it self-assesses whether the estimated position is reliable enough in order to annotate a Wi-Fi sample, besides, it does depend on a radio map to function, but it is basic with low density. Without any training (calibration) cost and only relying on PDR, the system in [81] depends on users' daily routine to build radio maps, achieving an average error of 6.9 m with Wi-Fi fingerprinting.

5.7 Radio Map Construction and Maintenance - effort

As previously mentioned, the construction and maintenance of radio maps is probably the main disadvantage of Wi-Fi-based indoor positioning solutions since it requires time and effort to collect Wi-Fi samples. There are some techniques for optimizing or automating the radio map construction process (presented in Section 2.4), however many systems still use manually collected radio maps. This section compares the time and effort it takes to manually construct a building's radio map against the time it takes to construct the radio map using the solution presented in this chapter, considering the building used in the experiments as an example.

The first step in the construction of a radio map consists of mapping RPs by measuring and determining the coordinates of each RP. 21 RPs were mapped at the PIEP building for a grid of ≈5 m between neighbor RPs. Then, the surveying process comprises the collection of Wi-Fi samples at the position of each RP. It is important to highlight that many IPSs based on radio maps use denser grids with much more RPs, therefore they take a longer time to complete.

It took 5 man/hours to map RPs at PIEP, and 0.6 man/hours to survey the building. The site survey process considered 21 RPs $* 80\ s = 28$ min, with the collection of 40 Wi-Fi samples per RP (time interval between consecutive Wi-Fi samples is around 2 s), additional time is necessary to move from the one RP to the next. The total time spent on the manual construction of the iRM at the PIEP building is 5.6 man/hours, although it is important to note that the duration of these tasks is proportional to the building size because larger buildings require more RPs.

In the performed experiments, it took only 53 minutes (accumulated run time of all trajectories) to obtain the vRM of the building with zero effort because it takes advantage of vehicles that are already operating in the factory to map the space. Contrarily to radio maps that have to be manually updated, this solution automatically keeps the radio map up to date without additional effort, thus it saves costs in future maintenance. As a drawback, this solution depends on the vehicles operating indoors, therefore, they can only map the areas they visit while working.

5.8 Summary

In this chapter, an enhanced PF based on Tight Coupling was proposed for the automatic building and maintenance of radio maps. This version of the PF explores a new way of updating particles weights by estimating the Wi-Fi similarity at the position of each particle, improving the previous PF version introduced in Section 4.6. The dynamic alpha was also refined to allow a better functioning of the PF in different scenarios. This approach is self-healing since it overcomes radio map degradation by automatically building and maintaining radio maps over time. The IPS deployment effort is significantly reduced because this solution does not depend on any calibration, installation of infrastructure, or human effort to build the radio map. In addition, it is flexible and dynamic since it works in scenarios without an initial Radio Map given that a floor plan is provided, and it also works in scenarios where the floor plan information is missing (no Floor Plan or partial Floor Plan), given that an initial Radio Map with just a few RPs is provided.

In comparison to the industrial vehicle used in experiments described in the previous chapter, the mobile unit prototype used to evaluate this solution is low-cost by replacing the industrial-grade IMUs with the low-cost Adafruit BNO055 IMU sensors. Conducted experiments have shown that this solution performs better than more expensive systems based on LiDAR or sonar sensors. Wi-Fi fingerprinting results revealed that vRMs perform better than a traditional radio map (grid of $\approx$5 m). VRMs can be used to locate people and improve safety, essential in an industrial context where vehicles and people interact frequently, as well as to locate assets, enhance operations and logistics processes by tracking vehicles. For instance, tracking vehicles allows to gather data that can be used to optimize routes and assets' locations to improve efficiency.

The main contributions of the research presented in this chapter are published in [12].

Chapter 6

Conclusions and Future Work

This chapter closes this work. It comprises a summary of the main contributions, the main learnings from this work and the future work topics that are of interest for this line of research.

6.1 Summary of the Main Contributions

The overall objective of this work was to develop an IPS for industrial vehicles capable to overcome the radio map degradation in industrial environments. First, an analysis of the long-term radio map degradation was made to assess how radio maps degrade over time and what causes it. Second, a positioning solution was developed and iteratively refined to improve several aspects of the sensor fusion algorithm, aiming to achieve accurate localization and tracking of vehicles, and also a means of building and maintaining Wi-Fi radio maps in industrial environments. The main contributions from this thesis are summarized next.

The first contribution is the collection and analysis of long-term datasets to measure and quantify the radio map degradation using two metrics, the mean positioning error with Wi-Fi fingerprinting and the Radio Map Degradation Ratio (RMDR). This study provided answers to several challenges, namely: being able to determine whether the radio map degrades over time; quantifying the degradation over time; and, identifying which events cause significant radio map degradation.

The second contribution is the IPS based on Loose Coupling and Tight Coupling for industrial vehicles. These approaches are distinct in the way they perform the sensor fusion. The Loose Coupling approach

merged a position estimate from Wi-Fi fingerprinting with data from motion sensors, and the Tight Coupling approach combines the similarities from Wi-Fi samples with data from motion sensors. Several aspects were improved in the Tight Coupling approach, namely, the integration of the floor plans, and the method to update the particles' weights, which considers all radio map information and uses the similarity instead of a Wi-Fi position estimate, as it is done in the Loose Coupling approach. Another contribution regarding the Tight Coupling implementation is the confidence metric, which is a novel concept introduced in this work to improve the reliability of the system. Based on the dispersion of particles from the PF, the confidence metric allows to determine when there might be a large positioning error associated with the estimated position. The simulation tool to generate synthetic data was essential to tune the parameters of the PF and also contributed to the Dioptra tool, described in Appendix B. Thanks to the improvements in the Tight Coupling approach relative to the Loose Coupling approach, an improvement of $\approx$24% and $\approx$118% is observed in the mean and maximum error, respectively.

The third contribution is the self-healing radio maps solution which explores industrial vehicles to build and maintain the radio map automatically. This solution is based on a novel approach in the process to update the particles' weights, where the Wi-Fi similarity is used to determine the weight of each particle using the weighted k-NN algorithm. The above-mentioned confidence metric takes a crucial part in this solution because new Wi-Fi samples are only annotated and added to the radio map when the confidence value is higher than a given threshold. In addition, it allows to build and maintain radio maps in several deployment scenarios depending on the existing information, e.g, when the floor plan or an initial radio map is available.

6.2 Conclusions

In this section, the main conclusions of this work are discussed. First, the radio environment of an indoor space varies over time, and it affects the positioning performance when the Wi-Fi infrastructure suffers alterations (removal of APs, changing their positions, or adding new ones). Therefore, radio maps should be updated after these changes. Thanks to this study, it was possible to quantify the radio map degradation in long-term through the applied methodology.

Second, using a PF to perform the sensor fusion of Wi-Fi and motion sensors has shown to be an

appropriate tool for indoor positioning and tracking of industrial vehicles as it allows to mitigate errors from Wi-Fi and drift in the heading, correcting the trajectory. The Tight Coupling approach performs better than the Loose Coupling approach, because it considers all available radio map information to compute the similarity between a new Wi-Fi sample and the ones from the radio map when updating particles' weights, and also because it takes advantage of the building's floor plan. The Loose Coupling approach is more suited for situations where a radio map exists and the floor plan information is not available, as it does not depend on a floor plan to work. Regarding the reliability of the system, the Tight Coupling approach included a confidence metric designed purposefully to determine whether the estimated position is reliable. This is crucial for vehicle tracking in industrial environments when the ultimate goal is to use the estimated position to annotate new Wi-Fi samples and build self-healing radio maps, but also to comply with safety and accuracy requirements.

Third, it is possible to create self-healing radio maps using the enhanced PF based on Tight Coupling. This solution performs an automatic unsupervised mapping of the radio environment in several deployment configurations. The practical aspects of this solution are discussed below:

- **Minimum Requirements, Deployment and Cost**: Either the floor plan (partial or complete) or an initial radio map are required for the system to work since it serves as reference information that allows the PF to converge. In addition to the floor plan information or an iRM, the initial pose of the vehicle should be provided to allow the system to start mapping as soon as it is initialized. The deployment costs are mostly associated with the hardware installed in the vehicles, and the effort to design the floor plan, or alternatively, to build the iRM. This solution is low-cost because it explores vehicles operating in the industrial context, and the positioning module is composed of a low-cost computer (Raspberry Pi), and motion sensors, including the low-cost IMU which make the positioning module cheap and easily reproducible in other vehicles. A server should be used to collect and store the Wi-Fi samples annotated by vehicles.
- **Adaptability and Scalability**: The system works in different floor plan scenarios with different radio map configurations, therefore it is versatile for different types of industrial spaces, e.g., factories with several production or assembly lines, warehouses, manufacturing facilities, etc. Each space may have different restrictions, either large open spaces or the floor plan may be incomplete. The best results were achieved with the floor plan scenarios (cFP, pFP and nFP) considering the iRM.

Although this is the best combination in terms of positioning performance, it depends on having the iRM. In cFP and pFP scenarios, the system is able to build the self-healing radio maps just using the floor plan (without the iRM) and the Wi-Fi samples it annotates during operation. In nFP scenarios, since there is no floor plan information, the iRM is essential to allow the PF to remove particles and maintain a low positioning error over time. Furthermore, the system is scalable for larger buildings. In experiments conducted at PIEP, it took only 53 minutes to build the radio map automatically with just one vehicle (mobile unit prototype). This represents a density of one vehicle for $\approx 1000\ m^2$.

- **Maintenance**: Since the vehicles contribute to the self-healing radio map while they operate, the system does not require maintenance or re-calibration. The system was designed to dynamically adapt to the radio environment keeping itself updated over time, hence it performs self-maintenance.
- **Multi-purpose Radio Maps**: The evaluation of self-healing radio maps has shown that they perform better than traditional radio maps with Wi-Fi fingerprinting, therefore, self-healing radio maps are suitable for indoor localization of other vehicles, human operators, among other devices as long as they are equipped with a Wi-Fi interface.

Although this thesis work was conducted during one of the most difficult periods of recent history, limiting the access to the University's campus, the proposed objectives were accomplished. To conclude, the participation in other activities related to indoor positioning, namely the IPIN competition [9, 10] and the Dioptra project [17], have contributed to this thesis work in two ways. First, it allowed working with a team of knowledgeable people interested in the same research area. Second, it allowed to apply the knowledge obtained in this thesis, and also to learn new concepts that would not be explored in this thesis otherwise.

6.3 Future Work

This thesis contributes to a promising field, especially with the advent of Industry 4.0 and 5G mobile networks, where it is expected interconnectivity between devices in industrial spaces, and improvements in the industrial processes aiming to increase productivity and enhance the control of operations. Along the course of this thesis, several topics of interest were identified as possible future work.

Regarding the radio map degradation study, several aspects could be explored in the future. First, the

development of a solution to automatically detect radio map degradation. Second, publishing the long-term datasets in an open-access database for the research community. Third, analyze the variation of the day-to-day positioning performance to evaluate how short-term variations in the radio environment affect the results.

Integrating the proposed solution in a real factory environment is the next logical step in the future, to measure the positive impact it has in day-to-day operations. The ability to locate and track industrial vehicles can be used to improve the factory's operations, increase productivity and optimize processes. Monitoring the positions of the vehicles can also improve security in the industrial space. In addition, it may contribute to the IoT area, since the positioning system may be used to enable communications between these vehicles, which were previously disconnected from the factory's communication systems. The ability to track vehicles as they operate can be used to track the materials they transport along the supply chain, thus enabling to monitor the raw materials and/or finished goods inside the industrial space.

The main drawback of the self-healing radio maps solution using industrial vehicles is that they annotate Wi-Fi samples only in navigable areas, leaving the other areas of the factory without coverage of the system. A way to overcome this issue is to combine this strategy with the Monitoring Devices collecting Wi-Fi samples in known positions. These two approaches complement each other, vehicles collect samples in areas where they usually operate, while Monitoring Devices can be installed in areas where vehicles have restricted access. If Raspberry Pi Zero devices (low-cost and low-power consumption) are used as the Monitoring Devices, it would be cheap to deploy several of these devices in areas where vehicles cannot operate. In cases where the density of Monitoring Devices is low, there are several techniques that may be applied to interpolate Wi-Fi samples in nearby locations, e.g., RBF, IDW, LDPL. RBF is one of the most straightforward techniques since it does not require the positions of APs and the floor plan, whilst the LDPL model depend on the positions of APs as well as the floor plan and building materials to work properly. In Appendix C, a study was conducted to evaluate the feasibility of RBF to interpolate the building's radio map from Monitoring Devices collected data. Obtained Wi-Fi fingerprinting results with interpolated radio maps are similar to the ones from traditional radio maps, hence this technique can be used to complement the self-healing radio maps solution.

The scope of this work considers only encoder and IMU sensors to measure the displacement and heading, respectively. In the future, one might explore new ways of measuring the displacement without

the encoder attached to the wheel. This could be achieved with inertial sensors, with the challenge being the integration of acceleration is usually noisy and may lead to large errors. Furthermore, the sensor fusion methods based on PFs can be extended to support other types of sensors, e.g., barometer to detect floor changes using atmospheric pressure, obstacle detection sensors (infrared, sonar, LiDAR, etc.) to eliminate the need for floor plan also enabling the SLAM of the space. Another aspect that may be explored in the future is the use of two low-cost IMUs simultaneously. Sensor fusion techniques may be applied in raw measurements from IMUs to minimize drift and magnetic perturbations that affect the estimated heading.

Finally, while vehicles operate, numerous Wi-Fi samples are annotated and added to the radio map in a matter of hours, therefore, the radio map should be optimized in such a way that allows to keep the latest Wi-Fi samples while removing the old Wi-Fi samples from locations where there is a higher density of samples.

Bibliography

[1] P. Bahl and V. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM 2000. Conf. Comput. Commun. Ninet. Annu. Jt. Conf. IEEE Comput. Commun. Soc. (Cat. No.00CH37064)*, vol. 2, IEEE, 2000, pp. 775–784. doi: 10.1109/INFCOM.2000.832252. arXiv: 1106.0222.

[2] F. Evennou and F. Marx, "Advanced Integration of WiFi and Inertial Navigation Systems for Indoor Mobile Positioning," *EURASIP J. Adv. Signal Process.*, vol. 2006, no. 1, p. 086 706, Dec. 2006. doi: 10.1155/ASP/2006/86706.

[3] J. Biswas and M. Veloso, "WiFi localization and navigation for autonomous indoor mobile robots," in *2010 IEEE Int. Conf. Robot. Autom.*, IEEE, May 2010, pp. 4379–4384. doi: 10.1109/ROBOT.2010.5509842.

[4] K. Witrisal, P. Meissner, E. Leitinger, *et al.*, "High-Accuracy Localization for Assisted Living: 5G systems will turn multipath channels from foe to friend," *IEEE Signal Process. Mag.*, vol. 33, no. 2, pp. 59–70, Mar. 2016. doi: 10.1109/MSP.2015.2504328.

[5] L. Yin, Q. Ni, and Z. Deng, "A GNSS/5G Integrated Positioning Methodology in D2D Communication Networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 2, pp. 351–362, Feb. 2018. doi: 10.1109/JSAC.2018.2804223.

[6] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM using Gaussian process latent variable models," in *Proc. 20th Int. Jt. Conf. Artifical Intell.*, 2007, pp. 2480–2485.

[7] Hyojeong Shin, Yohan Chon, and Hojung Cha, "Unsupervised Construction of an Indoor Floor Plan Using a Smartphone," *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.*, vol. 42, no. 6, pp. 889–898, Nov. 2012. doi: 10.1109/TSMCC.2011.2169403.

[8] X. Chai and Q. Yang, "Reducing the Calibration Effort for Probabilistic Indoor Location Estimation," *IEEE Trans. Mob. Comput.*, vol. 6, no. 6, pp. 649–662, Jun. 2007. doi: 10.1109/TMC.2007.1025.

[9] F. Potorti, S. Park, A. Crivello, *et al.*, "The IPIN 2019 Indoor Localisation Competition—Description and Results," *IEEE Access*, vol. 8, pp. 206 674–206 718, 2020. doi: 10.1109/ACCESS.2020.3037221.

[10] F. Potorti, J. Torres-Sospedra, D. Quezada-Gaibor, *et al.*, "Off-line Evaluation of Indoor Positioning Systems in Different Scenarios: The Experiences from IPIN 2020 Competition," *IEEE Sens. J.*, pp. 1–1, 2021. doi: 10.1109/JSEN.2021.3083149.

[11] I. Silva, C. Pendao, J. Torres-Sospedra, *et al.*, "TrackInFactory: A Tight Coupling Particle Filter for Industrial Vehicle Tracking in Indoor Environments," *IEEE Trans. Syst. Man, Cybern. Syst.*, pp. 1–12, 2021. doi: 10.1109/TSMC.2021.3091987.

[12] I. Silva, C. Pendao, and A. Moreira, "Real-World Deployment of Low-Cost Indoor Positioning Systems for Industrial Applications," *IEEE Sens. J.*, vol. XX, no. XX, pp. 1–1, 2021. doi: 10.1109/JSEN.2021.3103662.

[13] J. Torres-Sospedra, A. Moreira, G. M. Mendoza-Silva, *et al.*, "Exploiting Different Combinations of Complementary Sensor's data for Fingerprint-based Indoor Positioning in Industrial Environments," in *2019 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Sep. 2019, pp. 1–8. doi: 10.1109/IPIN.2019.8911758.

[14] I. Silva, A. Moreira, M. J. Nicolau, *et al.*, "Floor Plan-free Particle Filter for Indoor Positioning of Industrial Vehicles," in *ICL-GNSS Int. Conf. Localization GNSS*, Tampere, Finland, 2020.

[15] J. Torres-Sospedra, F. J. Aranda, F. J. Alvarez, *et al.*, "Ensembling Multiple Radio Maps with Dynamic Noise in Fingerprint-based Indoor Positioning," in *2021 IEEE 93rd Veh. Technol.*

Conf., Helsinki, Finland: IEEE, Apr. 2021, pp. 1–5. doi: 10.1109/VTC2021-Spring51267.2021.9448947.

[16] I. Silva, C. Pendão, J. Torres-Sospedra, *et al.*, "Quantifying the Degradation of Radio Maps in Wi-Fi Fingerprinting," in *2021 Int. Conf. Indoor Position. Indoor Navig. IPIN 2021*, 2021.

[17] C. Pendão, I. Silva, A. Moreira, *et al.*, "Dioptra – A Data Generation Application for Indoor Positioning Systems," in *2021 Int. Conf. Indoor Position. Indoor Navig. IPIN 2021*, Lloret de Mar, 2021.

[18] J. Torres-Sospedra, I. Silva, L. Klus, *et al.*, "Towards Ubiquitous Indoor Positioning : Comparing Systems across Heterogeneous Datasets," in *2021 Int. Conf. Indoor Position. Indoor Navig. IPIN 2021*, 2021.

[19] A. Moreira, M. J. Nicolau, I. Silva, *et al.*, "Wi-Fi Fingerprinting dataset with multiple simultaneous interfaces," version 1.0, Sep. 2019. doi: 10.5281/zenodo.3342526.

[20] A. Moreira, I. Silva, and J. Torres-Sospedra, "The DSI dataset for Wi-Fi fingerprinting using mobile devices," version 1.0, Apr. 2020. doi: 10.5281/zenodo.3778646.

[21] A. Moreira and F. Meneses, "Where@UM - Dependable organic radio maps," *2015 Int. Conf. Indoor Position. Indoor Navig. IPIN 2015*, no. October, pp. 1–9, 2015. doi: 10.1109/IPIN.2015.7346751.

[22] J. S. Seybold, *Introduction to RF Propagation*, 1-3. Hoboken, NJ, USA: John Wiley & Sons, Inc., Sep. 2005, vol. 14, pp. 145–173. doi: 10.1002/0471743690.

[23] R. Flickenger, A. Corinna, and S. Büttrich, *Wireless Networking in the Developing World Second Edition: A Practical Guide to Planning and Building Low-Cost Telecommunications Infrastructure*, 2nd, CreateSpace Independent Publishing Platform, Ed. 2007, p. 426.

[24] H. Hashemi, "The indoor radio propagation channel," *Proc. IEEE*, vol. 81, no. 7, pp. 943–968, Jul. 1993. doi: 10.1109/5.231342.

[25] J. Jun, Y. Gu, L. Cheng, *et al.*, "Social-Loc: Improving Indoor Localization with Social Sensing," in *Proc. 11th ACM Conf. Embed. Networked Sens. Syst. - SenSys '13*, New York, New York, USA: ACM Press, 2013, pp. 1–14. doi: 10.1145/2517351.2517352.

[26] E. Mok and B. Cheung, "An Improved Neural Network Training Algorithm for Wi-Fi Fingerprinting Positioning," *ISPRS Int. J. Geo-Information*, vol. 2, no. 3, pp. 854–868, Sep. 2013. doi: 10.3390/ijgi2030854.

[27] S. He and S.-H. G. Chan, "Wi-Fi Fingerprint-Based Indoor Positioning: Recent Advances and Comparisons," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 1, pp. 466–490, 2016. doi: 10.1109/COMST.2015.2464084.

[28] A. Moreira, I. Silva, F. Meneses, *et al.*, "Multiple simultaneous Wi-Fi measurements in fingerprinting indoor positioning," in *2017 Int. Conf. Indoor Position. Indoor Navig.*, Sapporo, Japan: IEEE, Sep. 2017, pp. 1–8. doi: 10.1109/IPIN.2017.8115914.

[29] S. S. Saab and Z. S. Nakad, "A Standalone RFID Indoor Positioning System Using Passive Tags," *IEEE Trans. Ind. Electron.*, vol. 58, no. 5, pp. 1961–1970, May 2011. doi: 10.1109/TIE.2010.2055774.

[30] A. Motroni, A. Buffi, P. Nepa, *et al.*, "Sensor-Fusion and Tracking Method for Indoor Vehicles With Low-Density UHF-RFID Tags," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–14, 2021. doi: 10.1109/TIM.2020.3027926.

[31] M. Kok, J. D. Hol, and T. B. Schon, "Indoor Positioning Using Ultrawideband and Inertial Measurements," *IEEE Trans. Veh. Technol.*, vol. 64, no. 4, pp. 1293–1303, Apr. 2015. doi: 10.1109/TVT.2015.2396640.

[32] Z. Yin, X. Jiang, Z. Yang, *et al.*, "WUB-IP: A High-Precision UWB Positioning Scheme for Indoor Multiuser Applications," *IEEE Syst. J.*, vol. 13, no. 1, pp. 279–288, Mar. 2019. doi: 10.1109/JSYST.2017.2766690.

[33] C. Cipolli, I. Fagioli, M. Mazzetti, *et al.*, "Incorporation of presleep stimuli into dream contents: evidence for a consolidation effect on declarative knowledge during REM sleep?" *J. Sleep Res.*, vol. 13, no. 4, pp. 317–326, Dec. 2004. doi: 10.1111/j.1365-2869.2004.00420.x.

[34] R. Faragher and R. Harle, "Location Fingerprinting With Bluetooth Low Energy Beacons," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2418–2428, Nov. 2015. doi: 10.1109/JSAC.2015.2430281.

[35] M. Hazas and A. Ward, "A novel broadband ultrasonic location system," *UbiComp 2002 Ubiquitous Comput.*, vol. 2498, no. September, pp. 264–280, 2002. doi: 10.1007/3-540-45809-3_21.

[36] P. Bolliger, "Redpin - adaptive, zero-configuration indoor localization through user collaboration," in *Proc. first ACM Int. Work. Mob. entity localization Track. GPS-less Environ. - MELT '08*, New York, New York, USA: ACM Press, 2008, p. 55. doi: 10.1145/1410012.1410025.

[37] Dhruv Pandya, Ravi Jain, and E. Lupu, "Indoor location estimation using multiple wireless technologies," in *14th IEEE Proc. Pers. Indoor Mob. Radio Commun. 2003. PIMRC 2003.*, IEEE, 2003, pp. 2208–2212. doi: 10.1109/PIMRC.2003.1259108.

[38] Lyu-Han Chen, E. H.-K. Wu, Ming-Hui Jin, *et al.*, "Intelligent Fusion of Wi-Fi and Inertial Sensor-Based Positioning Systems for Indoor Pedestrian Navigation," *IEEE Sens. J.*, vol. 14, no. 11, pp. 4034–4042, Nov. 2014. doi: 10.1109/JSEN.2014.2330573.

[39] R. Liu, C. Yuen, T. N. Do, *et al.*, "Cooperative positioning for emergency responders using self IMU and peer-to-peer radios measurements," *Inf. Fusion*, vol. 56, no. October 2019, pp. 93–102, 2020. doi: 10.1016/j.inffus.2019.10.009.

[40] U. Yayan, H. Yucel, and A. Yazıcı, "A Low Cost Ultrasonic Based Positioning System for the Indoor Navigation of Mobile Robots," *J. Intell. Robot. Syst.*, vol. 78, no. 3-4, pp. 541–552, Jun. 2015. doi: 10.1007/s10846-014-0060-7.

[41] J. Park, Y. K. Cho, and D. Martinez, "A BIM and UWB integrated Mobile Robot Navigation System for Indoor Position Tracking Applications," *J. Constr. Eng. Proj. Manag.*, vol. 6, no. 2, pp. 30–39, Jun. 2016. doi: 10.6106/JCEPM.2016.6.2.030.

[42] P. Mirowski, R. Palaniappan, and T. K. Ho, "Depth camera SLAM on a low-cost WiFi mapping robot," in *2012 IEEE Int. Conf. Technol. Pract. Robot Appl.*, vol. 1, IEEE, Apr. 2012, pp. 1–6. doi: 10.1109/TePRA.2012.6215673.

[43] Z. Wu, M. Wen, G. Peng, *et al.*, "Magnetic-Assisted Initialization for Infrastructure-free Mobile Robot Localization," in *9th IEEE Int. Conf. Cybern. Intell. Syst. Robot. Autom. Mechatronics*, Nov. 2019. arXiv: 1911.09313.

[44] H.-S. Kim, D.-R. Kim, S.-H. Yang, *et al.*, "An Indoor Visible Light Communication Positioning System Using a RF Carrier Allocation Technique," *J. Light. Technol.*, vol. 31, no. 1, pp. 134–144, Jan. 2013. doi: 10.1109/JLT.2012.2225826.

[45] W. Mao, H. Xie, Z. Tan, *et al.*, "High precision indoor positioning method based on visible light communication using improved Camshift tracking algorithm," *Opt. Commun.*, vol. 468, no. January, p. 125 599, Aug. 2020. doi: 10.1016/j.optcom.2020.125599.

[46] A. Olivier, G. Bielsa, I. Tejado, *et al.*, "Lightweight Indoor Localization for 60-GHz Millimeter Wave Systems," in *2016 13th Annu. IEEE Int. Conf. Sensing, Commun. Netw.*, IEEE, Jun. 2016, pp. 1–9. doi: 10.1109/SAHCN.2016.7732999.

[47] X. Han, J. Wang, W. Shi, *et al.*, "An Indoor Precise Positioning Algorithm Using 60GHz Millimeter-Wave Based on the Optimal Path Search," in *2017 IEEE GlobecomWork. (GCWkshps)*, IEEE, Dec. 2017, pp. 1–5. doi: 10.1109/GLOCOMW.2017.8269057.

[48] M. Vari and D. Cassioli, "mmWaves RSSI indoor network localization," in *2014 IEEE Int. Conf. Commun. Work.*, IEEE, Jun. 2014, pp. 127–132. doi: 10.1109/ICCW.2014.6881184.

[49] X. Wang, Z. Zhang, N. Zhao, *et al.*, "Indoor Localization and Trajectory Tracking System based on Millimeter-Wave Radar Sensor," in *2021 IEEE 10th Data Driven Control Learn. Syst. Conf.*, IEEE, May 2021, pp. 1141–1147. doi: 10.1109/DDCLS52934.2021.9455492.

[50] H. Liu, H. Darabi, P. Banerjee, *et al.*, "Survey of Wireless Indoor Positioning Techniques and Systems," *IEEE Trans. Syst. Man Cybern. Part C (Applications Rev.*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007. doi: 10.1109/TSMCC.2007.905750.

[51] B. B. PETERSON, C. KMIECIK, R. HARTNETT, *et al.*, "Spread Spectrum Indoor Geolocation," *Navigation*, vol. 45, no. 2, pp. 97–102, Jun. 1998. doi: 10.1002/j.2161-4296.1998.tb02374.x.

[52] N. S. Correal, S. Kyperountas, Q. Shi, *et al.*, "An UWB relative location system," in *2003 IEEE Conf. Ultra Wideband Syst. Technol. UWBST 2003 - Conf. Proc.*, Reston, Virginia, USA: IEEE, 2003, pp. 394–397. doi: 10.1109/UWBST.2003.1267871.

[53] S. J. Ingram, D. Harmer, and M. Quinlan, "Ultra wide band indoor positioning systems and their use in emergencies," *Rec. - IEEE PLANS, Position Locat. Navig. Symp.*, pp. 706–715, 2004. doi: 10.1109/PLANS.2004.1309063.

[54] R. Merz, F. Chastellain, C. Botteron, *et al.*, "An Experimental Platform for an Indoor Location and Tracking System," in *proc. Eur. Navig. Conf. (ENC-GNSS 08)*, France, 2008.

[55] J. González, J. L. Blanco, C. Galindo, *et al.*, "Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach," *Rob. Auton. Syst.*, vol. 57, no. 5, pp. 496–507, 2009. doi: 10.1016/j.robot.2008.10.022.

[56] B. P. Nissanka, "The Cricket Indoor Location System," PhD thesis, 2005, p. 199.

[57] A. Harter, A. Hopper, P. Steggles, *et al.*, "The anatomy of a context-aware application," in *Proc. 5th Annu. ACM/IEEE Int. Conf. Mob. Comput. Netw. - MobiCom '99*, vol. 8, New York, New York, USA: ACM Press, 1999, pp. 59–68. doi: 10.1145/313451.313476.

[58] C. Laoudias, D. G. Eliades, P. Kemppi, *et al.*, "Indoor Localization Using Neural Networks with Location Fingerprints," in *Proc. Int. Conf. Artif. Neural Networks*, 2009, pp. 954–963. doi: 10.1007/978-3-642-04277-5_96.

[59] Y. Xu and Y. Sun, "Neural Network-Based Accuracy Enhancement Method for WLAN Indoor Positioning," in *2012 IEEE Veh. Technol. Conf. (VTC Fall)*, IEEE, Sep. 2012, pp. 1–5. doi: 10.1109/VTCFall.2012.6399107.

[60] Chao-Lin Wu, Li-Chen Fu, and Feng-Li Lian, "WLAN location determination in e-home via support vector classification," in *IEEE Int. Conf. Networking, Sens. Control. 2004*, vol. 2, 2004, pp. 1026–1031. doi: 10.1109/ICNSC.2004.1297088.

[61] D. Pandya, "A Hierarchical Location Tracking System," M.S. thesis, Imperial College, London, 2002.

[62] P.-Y. Gilliéron and B. Merminod, "Personal Navigation System for Indoor Applications," in *11th IAIN World Congr.*, Berlin, Germany, 2003, pp. 1–15.

[63] Z. Xiao, H. Wen, A. Markham, *et al.*, "Lightweight map matching for indoor localisation using conditional random fields," in *IPSN-14 Proc. 13th Int. Symp. Inf. Process. Sens. Networks*, IEEE, Apr. 2014, pp. 131–142. doi: 10.1109/IPSN.2014.6846747.

[64] W. Elmenreich, "An introduction to sensor fusion," *AustriaVienna Univ.Technol.*, no. February, pp. 1–28, 2002.

[65] M. Kam, Xiaoxun Zhu, and P. Kalata, "Sensor Fusion for Mobile Robot Navigation," *Proc. IEEE*, vol. 85, no. 1, pp. 108–119, Jan. 1997. doi: 10.1109/JPROC.1997.554212.

[66] A. Koch and A. Zell, "RFID-enabled location fingerprinting based on similarity models from probabilistic similarity measures," in *2016 IEEE Int. Conf. Robot. Autom.*, vol. 2016-June, IEEE, May 2016, pp. 4557–4563. doi: 10.1109/ICRA.2016.7487656.

[67] Z. Chen, H. Zou, H. Jiang, *et al.*, "Fusion of WiFi, Smartphone Sensors and Landmarks Using the Kalman Filter for Indoor Localization," *Sensors*, vol. 15, no. 1, pp. 715–732, Jan. 2015. doi: 10.3390/s150100715. arXiv: 1007.0085.

[68] H. Zou, Z. Chen, H. Jiang, *et al.*, "Accurate indoor localization and tracking using mobile phone inertial sensors, WiFi and iBeacon," *4th IEEE Int. Symp. Inert. Sensors Syst. Inert. 2017 - Proc.*, no. 1, pp. 1–4, 2017. doi: 10.1109/ISISS.2017.7935650.

[69] S. Hilsenbeck, D. Bobkov, G. Schroth, *et al.*, "Graph-based data fusion of pedometer and WiFi measurements for mobile indoor positioning," in *Proc. 2014 ACM Int. Jt. Conf. Pervasive Ubiquitous Comput. - UbiComp '14 Adjun.*, 2014, pp. 147–158. doi: 10.1145/2632048.2636079.

[70] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer (Long. Beach. Calif).*, vol. 34, no. 8, pp. 57–66, 2001. doi: 10.1109/2.940014.

[71] B. Silva, Z. Pang, J. Akerberg, *et al.*, "Experimental study of UWB-based high precision localization for industrial applications," in *2014 IEEE Int. Conf. Ultra-WideBand*, IEEE, Sep. 2014, pp. 280–285. doi: 10.1109/ICUWB.2014.6958993.

[72] "ISO/IEC 18305:2016 Information technology — Real time locating systems — Test and evaluation of localization and tracking systems," Tech. Rep., 2016.

[73] G. M. Mendoza-Silva, J. Torres-Sospedra, F. Potorti, *et al.*, "Beyond Euclidean Distance for Error Measurement in Pedestrian Indoor Location," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021. doi: 10.1109/TIM.2020.3021514.

[74] A. Rácz-Szabó, T. Ruppert, L. Bántay, *et al.*, "Real-Time Locating System in Production Management," *Sensors*, vol. 20, no. 23, p. 6766, Nov. 2020. doi: 10.3390/s20236766.

[75] J. Fernandez-Madrigal, E. Cruz-Martin, J. Gonzalez, *et al.*, "Application of UWB and GPS technologies for vehicle localization in combined indoor-outdoor environments," in *2007 9th Int. Symp. Signal Process. Its Appl.*, IEEE, Feb. 2007, pp. 1–4. doi: 10.1109/ISSPA.2007.4555416.

[76] R. G. Yudanto and F. Petre, "Sensor fusion for indoor navigation and tracking of automated guided vehicles," in *2015 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Oct. 2015, pp. 1–8. doi: 10.1109/IPIN.2015.7346941.

[77] G. Vasiljević, D. Miklić, I. Draganjac, *et al.*, "High-accuracy vehicle localization for autonomous warehousing," *Robot. Comput. Integr. Manuf.*, vol. 42, pp. 1–16, Dec. 2016. doi: 10.1016/j.rcim.2016.05.001.

[78] P. Mpeis, T. Roussel, M. Kumar, *et al.*, "The Anyplace 4.0 IoT Localization Architecture," in *2020 21st IEEE Int. Conf. Mob. Data Manag.*, vol. 2020-June, IEEE, Jun. 2020, pp. 218–225. doi: 10.1109/MDM48529.2020.00045.

[79] X. Ma and T. Liu, "The application of Wi-Fi RTLS in automatic warehouse management system," in *2011 IEEE Int. Conf. Autom. Logist.*, IEEE, Aug. 2011, pp. 64–69. doi: 10.1109/ICAL.2011.6024685.

[80] Nguyen Dinh-Van, F. Nashashibi, Nguyen Thanh-Huong, *et al.*, "Indoor Intelligent Vehicle localization using WiFi received signal strength indicator," in *2017 IEEE MTT-S Int. Conf. Microwaves Intell. Mobil.*, IEEE, Mar. 2017, pp. 33–36. doi: 10.1109/ICMIM.2017.7918849.

[81] Y. Kim, Y. Chon, and H. Cha, "Smartphone-Based Collaborative and Autonomous Radio Fingerprinting," *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.*, vol. 42, no. 1, pp. 112–122, Jan. 2012. doi: 10.1109/TSMCC.2010.2093516.

[82] Y. Luo, Y. P. Chen, and O. Hoeber, "Wi-Fi-Based Indoor Positioning Using Human-Centric Collaborative Feedback," in *2011 IEEE Int. Conf. Commun.*, IEEE, Jun. 2011, pp. 1–6. doi: 10.1109/icc.2011.5963278.

[83] J. Ledlie, J.-g. Park, D. Curtis, *et al.*, "Molé: a scalable, user-generated WiFi positioning engine," *J. Locat. Based Serv.*, vol. 6, no. 2, pp. 55–80, Jun. 2012. doi: 10.1080/17489725.2012.692617.

[84] J. Leonard and H. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," in *Proc. IROS '91IEEE/RSJ Int. Work. Intell. Robot. Syst. '91*, vol. No. 91TH03, IEEE, 1991, pp. 1442–1447. doi: 10.1109/IROS.1991.174711.

[85] J. Tang, Y. Chen, L. Chen, *et al.*, "Fast Fingerprint Database Maintenance for Indoor Positioning Based on UGV SLAM," *Sensors*, vol. 15, no. 3, pp. 5311–5330, Mar. 2015. doi: 10.3390/s150305311.

[86] R. Liu, S. H. Marakkalage, M. Padmal, *et al.*, "Collaborative SLAM Based on WiFi Fingerprint Similarity and Motion Information," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 1826–1840, Mar. 2020. doi: 10.1109/JIOT.2019.2957293. arXiv: 2001.02759.

[87] J. Krumm and J. Platt, "Minimizing Calibration Efforts for an Indoor 802.11 Device Location Measurement System," *Microsoft Res.*, pp. 1–9, 2003. doi: 10.1.1.145.8006.

[88] C. Laoudias, P. Kemppi, and C. G. Panayiotou, "Localization Using Radial Basis Function Networks and Signal Strength Fingerprints in WLAN," in *GLOBECOM 2009 - 2009 IEEE Glob. Telecommun. Conf.*, IEEE, Nov. 2009, pp. 1–6. doi: 10.1109/GLOCOM.2009.5425278.

[89] C. Laoudias, C. G. Panayiotou, and P. Kemppi, "On the RBF-based positioning using WLAN signal strength fingerprints," in *2010 7th Work. Positioning, Navig. Commun.*, IEEE, Mar. 2010, pp. 93–98. doi: 10.1109/WPNC.2010.5653403.

[90] H. Zhao, B. Huang, and B. Jia, "Applying kriging interpolation for WiFi fingerprinting based indoor positioning systems," in *2016 IEEE Wirel. Commun. Netw. Conf.*, vol. 2016-Septe, IEEE, Apr. 2016, pp. 1–6. doi: 10.1109/WCNC.2016.7565018.

[91] Y. Wang, W.-j. Cui, D.-m. Wang, *et al.*, "Fingerprint Space Building Algorithm with Kriging for Large Positioning Regional Environment," Iccset 2014, 2015, pp. 437–442. doi: 10.2991/iccset-14.2015.98.

[92] S.-S. Jan, S.-J. Yeh, and Y.-W. Liu, "Received Signal Strength Database Interpolation by Kriging for a Wi-Fi Indoor Positioning System," *Sensors*, vol. 15, no. 9, pp. 21 377–21 393, Aug. 2015. doi: 10.3390/s150921377.

[93] P. M. Bartier and C. Keller, "Multivariate interpolation to incorporate thematic surface data using inverse distance weighting (IDW)," *Comput. Geosci.*, vol. 22, no. 7, pp. 795–799, Aug. 1996. doi: 10.1016/0098-3004(96)00021-0.

[94] Z. Han, J. Liao, Q. Qi, *et al.*, "Radio Environment Map Construction by Kriging Algorithm Based on Mobile Crowd Sensing," *Wirel. Commun. Mob. Comput.*, vol. 2019, pp. 1–12, Feb. 2019. doi: 10.1155/2019/4064201.

[95] M. Lee and D. Han, "Voronoi Tessellation Based Interpolation Method for Wi-Fi Radio Map Construction," *IEEE Commun. Lett.*, vol. 16, no. 3, pp. 404–407, Mar. 2012. doi: 10.1109/LCOMM.2012.020212.111992.

[96] Sheng-Po Kuo and Yu-Chee Tseng, "Discriminant Minimization Search for Large-Scale RF-Based Localization Systems," *IEEE Trans. Mob. Comput.*, vol. 10, no. 2, pp. 291–304, Feb. 2011. doi: 10.1109/TMC.2010.67.

[97] H. Wang, L. Ma, Y. Xu, *et al.*, "Dynamic Radio Map Construction for WLAN Indoor Location," in *2011 Third Int. Conf. Intell. Human-Machine Syst. Cybern.*, vol. 2, IEEE, Aug. 2011, pp. 162–165. doi: 10.1109/IHMSC.2011.110.

[98] Jie Yin, Qiang Yang, and L. Ni, "Learning Adaptive Temporal Radio Maps for Signal-Strength-Based Location Estimation," *IEEE Trans. Mob. Comput.*, vol. 7, no. 7, pp. 869–883, Jul. 2008. doi: 10.1109/TMC.2007.70764.

[99] Y. Ji, S. Biaz, S. Pandey, *et al.*, "ARIADNE," in *Proc. 4th Int. Conf. Mob. Syst. Appl. Serv. - MobiSys 2006*, New York, New York, USA: ACM Press, 2006, p. 151. doi: 10.1145/1134680.1134697.

[100] Atreyi Bose and Chuan Heng Foh, "A practical path loss model for indoor WiFi positioning enhancement," in *2007 6th Int. Conf. Information, Commun. Signal Process.*, IEEE, 2007, pp. 1–5. doi: 10.1109/ICICS.2007.4449717.

[101] M. Lott and I. Forkel, "A multi-wall-and-floor model for indoor radio propagation," in *IEEE VTS 53rd Veh. Technol. Conf. Spring 2001. Proc. (Cat. No.01CH37202)*, vol. 1, IEEE, 2001, pp. 464–468. doi: 10.1109/VETECS.2001.944886.

[102] C. Pendao and A. Moreira, "FastGraph Enhanced: High Accuracy Automatic Indoor Navigation and Mapping," *IEEE Trans. Mob. Comput.*, vol. 20, no. 3, pp. 1027–1045, Mar. 2021. doi: 10.1109/TMC.2019.2955653.

[103] C. Luo, L. Cheng, M. C. Chan, *et al.*, "Pallas: Self-Bootstrapping Fine-Grained Passive Indoor Localization Using WiFi Monitors," *IEEE Trans. Mob. Comput.*, vol. 16, no. 2, pp. 466–481, Feb. 2017. doi: 10.1109/TMC.2016.2550452.

[104] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proc. Sixt. Annu. Int. Conf. Mob. Comput. Netw. - MobiCom '10*, 2010, p. 173. doi: 10.1145/1859995.1860016. arXiv: arXiv:1011.1669v3.

[105] M. M. Atia, A. Noureldin, and M. J. Korenberg, "Dynamic Online-Calibrated Radio Maps for Indoor Positioning in Wireless Local Area Networks," *IEEE Trans. Mob. Comput.*, vol. 12, no. 9, pp. 1774–1787, Sep. 2013. doi: 10.1109/TMC.2012.143.

[106] R. Montoliu, E. Sansano, O. Belmonte, *et al.*, "A New Methodology for Long-Term Maintenance of WiFi Fingerprinting Radio Maps," in *2018 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Sep. 2018, pp. 1–7. doi: 10.1109/IPIN.2018.8533825.

[107] C. Wu, Z. Yang, and C. Xiao, "Automatic Radio Map Adaptation for Indoor Localization Using Smartphones," *IEEE Trans. Mob. Comput.*, vol. 17, no. 3, pp. 517–528, Mar. 2018. doi: 10.1109/TMC.2017.2737004.

[108] K. Kaemarungsi and P. Krishnamurthy, "Properties of indoor received signal strength for WLAN location fingerprinting," in *First Annu. Int. Conf. Mob. Ubiquitous Syst. Netw. Serv. 2004. MOBIQUITOUS 2004.*, IEEE, 2004, pp. 14–23. doi: 10.1109/MOBIQ.2004.1331706.

[109] J. Torres-Sospedra, P. Richter, G. Mendoza-Silva, *et al.*, "Characterising the Alteration in the AP Distribution with the RSS Distance and the Position Estimates," in *2018 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Sep. 2018, pp. 1–8. doi: 10.1109/IPIN.2018.8533791.

[110] J. Luo and X. Zhan, "Characterization of Smart Phone Received Signal Strength Indication for WLAN Indoor Positioning Accuracy Improvement," *J. Networks*, vol. 9, no. 3, pp. 739–746, Mar. 2014. doi: 10.4304/jnw.9.3.739-746.

[111] S. Eisa, J. Peixoto, F. Meneses, *et al.*, "Removing useless APs and fingerprints from WiFi indoor positioning radio maps," in *Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Oct. 2013, pp. 1–7. doi: 10.1109/IPIN.2013.6817919.

[112] Sunhong Park and S. Hashimoto, "Autonomous Mobile Robot Navigation Using Passive RFID in Indoor Environment," *IEEE Trans. Ind. Electron.*, vol. 56, no. 7, pp. 2366–2373, Jul. 2009. doi: 10.1109/TIE.2009.2013690.

[113] S. Fu, Z.-g. Hou, and G. Yang, "An indoor navigation system for autonomous mobile robot using wireless sensor network," in *2009 Int. Conf. Networking, Sens. Control*, 2009, pp. 227–232. doi: 10.1109/ICNSC.2009.4919277.

[114] J. Torres-Sospedra, A. Jiménez, A. Moreira, *et al.*, "Off-Line Evaluation of Mobile-Centric Indoor Positioning Systems: The Experiences from the 2017 IPIN Competition," *Sensors*, vol. 18, no. 2, p. 487, Feb. 2018. doi: 10.3390/s18020487.

[115] E. Ivanjko and I. Petrovic, "Extended Kalman filter based mobile robot pose tracking using occupancy grid maps," in *Proc. 12th IEEE Mediterr. Electrotech. Conf. (IEEE Cat. No.04CH37521)*, vol. 1, IEEE, 2004, pp. 311–314. doi: 10.1109/MELCON.2004.1346851.

[116] S. Thrun, D. Fox, W. Burgard, *et al.*, "Robust Monte Carlo localization for mobile robots," *Artif. Intell.*, vol. 128, no. 1-2, pp. 99–141, May 2001. doi: 10.1016/S0004-3702(01)00069-8.

[117] F. Duvallet and A. Tews, "WiFi position estimation in industrial environments using Gaussian processes," in *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, IEEE, Sep. 2008, pp. 2216–2221. doi: 10.1109/IROS.2008.4650910.

[118] E. Bachmann, Xiaoping Yun, and C. Peterson, "An investigation of the effects of magnetic variations on inertial/magnetic orientation sensors," in *IEEE Int. Conf. Robot. Autom. 2004. Proceedings. ICRA'04. 2004*, vol. 2004, IEEE, 2004, 1115–1122 Vol.2. doi: 10.1109/ROBOT.2004.1307974.

[119] M. Euston, P. Coote, R. Mahony, *et al.*, "A complementary filter for attitude estimation of a fixed-wing UAV," in *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, IEEE, Sep. 2008, pp. 340–345. doi: 10.1109/IROS.2008.4650766.

[120] K. Feng, J. Li, X. Zhang, *et al.*, "A New Quaternion-Based Kalman Filter for Real-Time Attitude Estimation Using the Two-Step Geometrically-Intuitive Correction Algorithm," *Sensors*, vol. 17, no. 9, p. 2146, Sep. 2017. doi: 10.3390/s17092146.

[121] R. Mahony, T. Hamel, P. Morin, *et al.*, "Nonlinear complementary filters on the special linear group," *Int. J. Control*, vol. 85, no. 10, pp. 1557–1573, Oct. 2012. doi: 10.1080/00207179.2012.693951.

[122] S. O. Madgwick, "An efficient orientation filter for inertial and inertial/magnetic sensor arrays," University of Bristol, Tech. Rep., 2010.

[123] S. Ludwig, K. Burnham, A. Jimenez, *et al.*, "Comparison of attitude and heading reference systems using foot mounted MIMU sensor data: basic, Madgwick, and Mahony," in *Sensors Smart Struct. Technol. Civil, Mech. Aerosp. Syst. 2018*, H. Sohn, Ed., SPIE, Mar. 2018, p. 96. doi: 10.1117/12.2296568.

[124] J. Borenstein and Liqiang Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE Trans. Robot. Autom.*, vol. 12, no. 6, pp. 869–880, 1996. doi: 10.1109/70.544770.

[125] A. Heinrich, "An Optical Flow Odometry Sensor Based on the Raspberry Pi Computer," Master Thesis, Czech Technical University, 2017.

[126] C. McCarthy and N. Bames, "Performance of optical flow techniques for indoor navigation with a mobile robot," in *IEEE Int. Conf. Robot. Autom. 2004. Proceedings. ICRA'04. 2004*, vol. 2004, IEEE, 2004, pp. 5093–5098. doi: 10.1109/ROBOT.2004.1302525.

[127] A. A. Panyov, A. A. Golovan, and A. S. Smirnov, "Indoor positioning using Wi-Fi fingerprinting pedestrian dead reckoning and aided INS," in *2014 Int. Symp. Inert. Sensors Syst.*, IEEE, Feb. 2014, pp. 1–2. doi: 10.1109/ISISS.2014.6782540.

[128] Z. Wu, E. Jedari, R. Muscedere, *et al.*, "Improved particle filter based on WLAN RSSI fingerprinting and smart sensors for indoor localization," *Comput. Commun.*, vol. 83, pp. 64–71, Jun. 2016. doi: 10.1016/j.comcom.2016.03.001.

[129] J. Torres-Sospedra, P. Richter, A. Moreira, *et al.*, "A Comprehensive and Reproducible Comparison of Clustering and Optimization Rules in Wi-Fi Fingerprinting," *IEEE Trans. Mob. Comput.*, vol. X, no. 10, pp. 1–1, 2020. doi: 10.1109/TMC.2020.3017176.

[130] J. Torres-Sospedra and A. Moreira, "Analysis of Sources of Large Positioning Errors in Deterministic Fingerprinting," *Sensors*, vol. 17, no. 12, p. 2736, Nov. 2017. doi: 10.3390/s17122736.

[131] N. Marques, F. Meneses, and A. Moreira, "Combining similarity functions and majority rules for multi-building, multi-floor, WiFi positioning," in *2012 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Nov. 2012, pp. 1–9. doi: 10.1109/IPIN.2012.6418937.

[132] G. Retscher and J. Joksch, "Comparison of Different Vector Distance Measure Calculation Variants for Indoor Location Fingerprinting," *Proc. 13th Int. Conf. Locat. Serv.*, no. November, pp. 14–16, 2016.

[133] J.-A. Fernández-Madrigal and J. L. Blanco Claraco, "Resampling Algorithms," in *Simultaneous Localization Mapp. Mob. Robot. Introd. Methods*, ser. Advances in Computational Intelligence and Robotics, 2. IGI Global, Ed., IGI Global, 2013, ch. Appendix B, pp. 407–411. doi: 10.4018/978-1-4666-2104-6.

[134] R. Liu, C. Yuen, T.-N. Do, *et al.*, "Indoor positioning using similarity-based sequence and dead reckoning without training," in *2017 IEEE 18th Int. Work. Signal Process. Adv. Wirel. Commun.*, vol. 2017-July, IEEE, Jul. 2017, pp. 1–5. doi: 10.1109/SPAWC.2017.8227641. arXiv: 1705.04934.

[135] C. Pendao and A. Moreira, “FastGraph - Organic 3D Graph for Unsupervised Location and Mapping,” in *2018 Int. Conf. Indoor Position. Indoor Navig.*, IEEE, Sep. 2018, pp. 206–212. doi: 10.1109/IPIN.2018.8533746.

[136] C. Pendão, “FastGraph - unsupervised location and mapping in wireless networks,” PhD, University of Minho, Aveiro and Porto, 2018.

[137] N. Benoudjit and M. Verleysen, “On the kernel widths in radial-basis function networks,” *Neural Process. Lett.*, vol. 18, no. 2, pp. 139–154, 2003. doi: 10.1023/A:1026289910256.

Appendices

Appendix A

Long-term Collection of Wi-Fi Samples using Monitoring Devices

Wi-Fi-enabled devices can collect annotated Wi-Fi samples, characterized by a set of RSSI values of detected APs, the time instant and the position where they were was collected. This information can be gathered from several devices deployed throughout the building and used to detect variations in the radio map over time. This appendix describes the solution that was developed to enable the long-term collection of Wi-Fi samples from Monitoring Devices deployed in a building.

A.1 Hardware

A low-cost and low-power device is ideal for collecting Wi-Fi samples because it does not require much computing power and it is necessary to deploy several Monitoring Devices throughout the building, thus it should be cheap. The Raspberry Pi device is an optimal solution for monitoring the radio environment because it is a single-board computer that is low-cost and does not have demanding power requirements. Additionally, the Raspberry Pi is small enough to be installed in virtually any place of a building as long as it is connected to a power source. There are several models with different specifications and prices, the most expensive ones have integrated Bluetooth and Wi-Fi as well as more computing power, while the

cheaper ones might not have integrated Wi-Fi and have a lower computing power. The Raspberry Pi Zero W is one of the cheapest models and costs around 12 € and has an integrated Wi-Fi interface. The Raspberry Pi 3B+, presented in Figure A.1, has more computing power than the Raspberry Pi Zero W, an integrated IEEE 802.11 b/g/n/ac wireless and it costs around 38 €.

Figure A.1: Raspberry Pi 3 B+.
Source: https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/

The Raspberry Pi 3B+, launched in March 2018, has a quad-core 1.4 GHz 64-bit processor and 1 Gb of RAM. In terms of connectivity, it supports dual-band 2.4 and 5 GHz wireless LAN, Bluetooth 4.2 (BLE) and Gigabit Ethernet over USB 2.0 (up to 300 Mbps throughput). To be properly powered, the Raspberry Pi 3B+ requires a (5 V, 2.5 A) power supply. In contrast with previous models that do not support dual-band Wi-Fi, the Raspberry Pi 3B+ can be used to explore the variability of Wi-Fi signals both from 2.4 and 5 GHz WLANs.

Raspbian is the most common operating system for normal use on a Raspberry Pi, it is a Debian-based operating system optimised for the Raspberry Pi hardware. Wi-Fi samples can be obtained from the Wi-Fi interface of the Raspberry Pi using the iwlist command, for example, the command "sudo iwlist wlan0 scanning" allows to obtain the information regarding the detected APs of 2.4 and 5 GHz WLANs.

A.2 Software

The architecture of the solution for the long-term collection of Wi-Fi samples is shown in Figure A.2, including the main modules and how they are connected between each other. The architecture is divided into two main parts, the Monitoring Device that is responsible for the collection of Wi-Fi samples and the server that gathers samples from many Monitoring Devices and analyzes them.

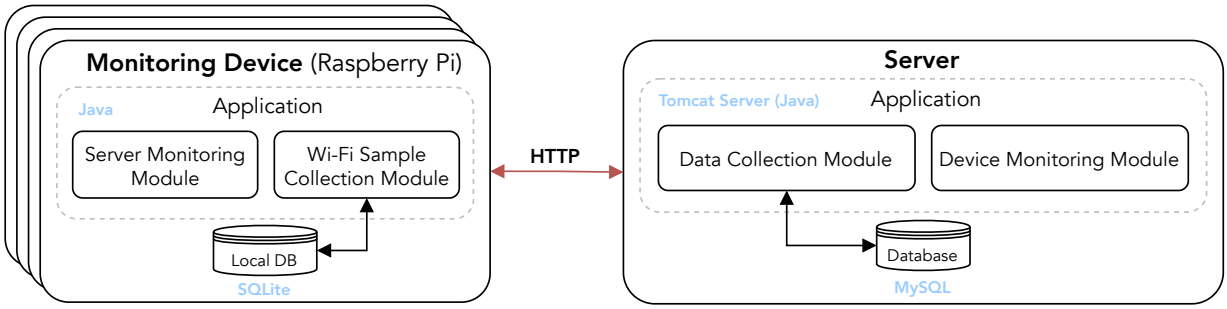


Figure A.2: Architecture of the solution for the long-term collection of Wi-Fi samples.

A.2.1 Monitoring Device Application

Several Monitoring Devices are used to collect Wi-Fi samples in known positions inside the building. Each device runs an application that comprises two modules, one responsible for obtaining Wi-Fi samples and another responsible for monitoring the connection with the central server. A local database is used by the application to store all samples.

When the application starts, it runs both modules simultaneously. In order to work properly, the application loads a configuration file that allows to configure several parameters:

- Wi-Fi sample polling - time interval between consecutive scans to obtain Wi-Fi samples, defined in seconds;
- location coordinates - latitude and longitude coordinates of the location where the Monitoring Device is deployed;
- location description - textual description of the location where the Monitoring Device is deployed;
- e-mail - the e-mail address to send alerts;

- number of unsent samples - an alert e-mail will be sent when the number of unsent Wi-Fi samples is higher than this value.

The server monitoring module checks the number of Wi-Fi samples that were not successfully sent to the server and sends an alert e-mail when the number of unsent samples is higher than the value specified in the configuration file. In case the connection to the server is not successful, a new e-mail will be sent every eight hours.

A new Wi-Fi sample is obtained by the Wi-Fi sample collection module in time intervals defined by the polling time defined in the configuration file. When a new Wi-Fi sample is collected, it is sent to the server and saved in the local database. In case some Wi-Fi samples fail to send to the server, they are sent later by the server monitoring module when it has a connection to the server.

In regard to the local database, the long-term collection of Wi-Fi samples will cause a significant use of storage. For instance, considering the collection of samples every 60 seconds during one month, represents a total of 43200 Wi-Fi samples that occupy around 95 MBytes in the SQLite database file. Performing queries in SQLite databases will take longer the larger the database is, therefore, a new database is created every month in order to guarantee that the SQLite performance is not affected over time. The old database remains stored in the Monitoring Device, as a backup. In addition, the SQLite database serves only as a backup because all Wi-Fi samples should be successfully sent to the server where they are kept in the main database.

The following information can be obtained for each detected AP in a Wi-Fi sample obtained from the Raspberry Pi 3B+:

- Service Set ID (SSID) - the name of the network that is broadcast by the AP;
- Basic Service Set ID (BSSID) - the MAC address of the AP wireless interface;
- Received Signal Strength Indicator (RSSI) - the signal level in dBm;
- Channel - the frequency channel of the AP;
- Link quality - aggregate value that represents the overall quality of the link.

A.2.2 Server Application

Two sub-modules constitute the server application, namely, the data collection module that receives and stores the Wi-Fi samples, and the device monitoring module responsible for sending alerts whenever a Monitoring Device has stopped sending Wi-Fi samples for over a period of time

Several parameters can be used to configure the Device Monitoring Module, as presented next:

- e-mail - the e-mail address to send alerts;
- time without receiving data from device (hours) - when the device has not send new Wi-Fi samples for over a certain number of hours, an e-mail alert will be sent;
- polling time for device monitoring (seconds) - time interval in which the device monitoring module checks the latest device's connections.

Data Collection Module

The server has a RESTful interface that is used by the Monitoring Devices to send Wi-Fi samples. When the Wi-Fi sample collection module receives a new sample, it processes it and stores it as a wifi_sample in the database. Therefore, samples from all devices are properly stored in the database. The model of the implemented database that supports the continuous sample collection is presented in Figure A.3. The place table, represents a place that is associated with each fingerprint. A place may have a parent place depending on its type. There are five types of places: operator, defines the main entity that owns a set of areas; area, represents a geographical area with several buildings; building, represents a building that may have a set of floors; floor, represents a floor that can have several rooms; room, an indoor part of a floor; position, a place inside a room represented by a set of geographic coordinates. For example, in order to represent a location inside the research lab LID3 from the Department of Information Systems at University of Minho, the following places must be considered:

- Operator: University of Minho;
- Area: Azurém Campus;
- Building: School of Engineering;
- Floor: Floor 1;
- Room: LID3;

- Position: lat = 41.453345271696, lon = -8.288865185421.

Regarding the Wi-Fi data that is collected in each sample, it is properly stored in the *wifi_sample* and *ap* tables. Every time a new Wi-Fi scan is made, the set of observed APs is obtained as well as its signal levels.

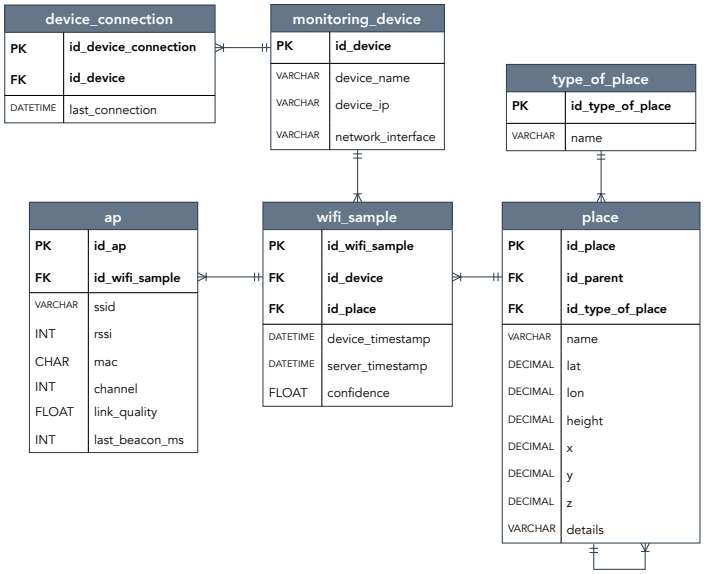


Figure A.3: Wi-Fi samples collection: Entity Relationship Diagram.

Device Monitoring Module

In order to guarantee that all Monitoring Devices are working properly, the device monitoring module sends e-mail alerts whenever a Monitoring Device has stopped sending samples to the server. The *device_connection* table holds the last time that the device sent data. When a certain period of time has gone since the last connection from the device, an alert is sent via e-mail. The number of hours that trigger an alert can be set in the configuration file of the application. The last connection of each device is checked at regular intervals by the device monitoring module.

A.3 Summary

This appendix described the solution for the long-term Wi-Fi data collection to evaluate the degradation of radio maps over time.

The hardware and software components were described as well as the operation principle of each sub-module. Monitoring Devices collect Wi-Fi samples and send them to a central server that processes and stores the information in a database. The collected data can later be used to evaluate the degradation of the radio environment, or as a radio map, by using an interpolation method as described in Appendix C.

The system works autonomously and only requires attention when the server or a Monitoring Device is down. In case it happens, it alerts the user via e-mail, so that the situation is fixed as soon as possible.

Appendix B

Dioptra: An Open Access, Open Source, Synthetic Data Generator

In this appendix, a brief description of the Dioptra tool [17] is made. Dioptra is a synthetic data generator for indoor positioning, which allows to perform fair comparison between IPSs since different algorithms and positioning techniques can be tested with Dioptra generated datasets.

B.1 Introduction

The need to create synthetic data to test and evaluate IPSs, as described in Section 4.6.7, showed that other researchers also depend on similar solutions to generate synthetic data. Currently, there is a lack of such solutions that allow users to easily generate datasets to test their algorithms. Performing real-world experiments is a complex task due to several aspects, for example, finding a building/scenario, prepare the setup for the data collection, collect the training dataset (usually the radio map), and the test dataset. Although simulation has some limitations (e.g. it uses simple models for signal propagation or to represent the indoor space), it is the simplest way of testing a system without much effort.

Dioptra[11] was created for researchers on indoor positioning to easily create synthetic datasets to develop and evaluate their positioning systems. The idea is to simulate an indoor environment where different

[11] http://dioptra.dsi.uminho.pt

actors move through the indoor space. The indoor space is characterized by a space with indoor elements such as walls or obstacles made of different building materials. Multiple types of actors (i.e., drone, pedestrian, industrial vehicle, etc.), sensors, transmitters, motion models and propagation models are supported. An actor may carry sensors and/or transmitters and follows a motion model. Different motion models are supported, for example, random motion (actor moves randomly avoiding walls and obstacles), grid motion (used to map the space, i.e., to build the radio map), or graph-based motion where the actor's movement is defined by an undirected graph. Supported sensors are: AHRS, Wi-Fi/BLE scanner, odometer, step counter and a reference sensor (virtual sensor that provides ground truth pose). Currently, Wi-Fi and BLE are the transmitter technologies supported, therefore, users can add Wi-Fi APs or BLE beacons to the space. In order to account to signal propagation indoors, an extended LDPL model was implemented which accounts for the attenuation caused by walls and obstacles.

Dioptra was designed with these practical aspects in mind:

- It can run on multiple platforms, such as Windows, Linux or macOS because it is coded in Java. A Graphical User Interface (GUI) was developed to allow users to follow the progress and control most aspects of a work session (simulation). Work sessions are based on configuration files that provide a simple way to reproduce results and can be easily adapted to implement other use cases.
- Integration of new modules and features. Dioptra's framework allows the integration of new modules into the application. New modules can be new motion models, sensors, propagation models or transmitters. To ensure abstraction and compatibility between the new modules and the Dioptra engine, the new modules must follow the defined Java Interfaces[12].

B.2 Architecture

Dioptra's architecture, depicted in Figure B.1, includes the modules and their interactions. It follows a modular structure to be easily extensible to support new sensors, motion models, transmitter types or propagation models.

The User Interface allows the user to interact with the tool during a work session (simulation) through the command line (CMD) and the GUI, which enables users to follow the progress and control most aspects

[12]https://docs.oracle.com/javase/tutorial/java/concepts/interface.html

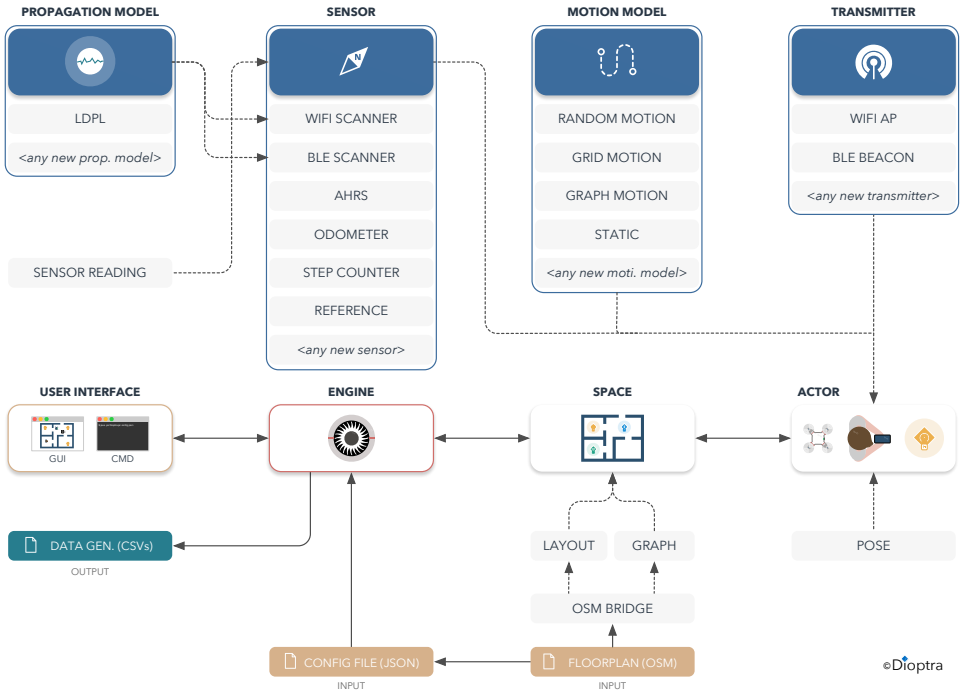


Figure B.1: Architecture of Dioptra [17].

of a work session. Before starting a work session, the user has to specify the configuration file (JSON) as the input for the Engine. The Config File allows to specify the floor plan of the space, the actors (with their motion models and sensors), the transmitters, and other simulation parameters.

The floor plan file (OpenStreetMap (OSM) format) is loaded by the Engine using the OSM Bridge module, creating a new Space. Additional details (warnings, errors, info) are logged to a file that can be accessed by the user. The space layout and the transmitters' positions can also be exported.

The Space module includes the layout and navigation topologies that are loaded and created by the Engine, based on the information from the input floor plan file (OSM). The navigation topologies are used as input for the graph-based motion model.

The Engine module sets up and controls the operations in a work session. It loads the configuration file (JSON), creates all actors based on the configuration file and adds them to the Space. The Actor module supports different types of actors, namely, Fixed Transmitters, Pedestrians, Robots, Industrial Machines, Drones, among others. Motion Models define the movement of an Actor during a work session. Different parameters can be configured for each type of Motion Model, and each Motion model instance that can

be associated with an Actor can have different values for the available parameters.

The Sensor module allows the implementation of different types of sensors, for instance, the Wi-Fi and BLE scanner, the AHRS, the odometer, the step counter and the reference sensor (provides a ground truth pose). Actors can carry any type of sensor. Each sensor has a specific set of parameters that can be configured. For example, sensors such as the Wi-Fi or BLEs scanners measure virtual signals from the transmitters based on a Propagation Model, that can also be configured.

B.3 Graphical User Interface

Dioptra was designed to be user friendly hence, most of the controls for a simulation session are available through the GUI (Figure B.2). Once the user selects the configuration file, the floor plan of the space is loaded and displayed in the GUI. The user has a view control tool to pan, zoom and reset the original view of the space. This tool can be used before and during the simulation session. The simulation speed can be configured on the toolbar button. Several aspects can be configured in the configuration panel: enable/disable the dark mode; show/hide a grid inside the space area; and, set the simulation speed to max, running the simulation at the maximum speed allowed by the computer.

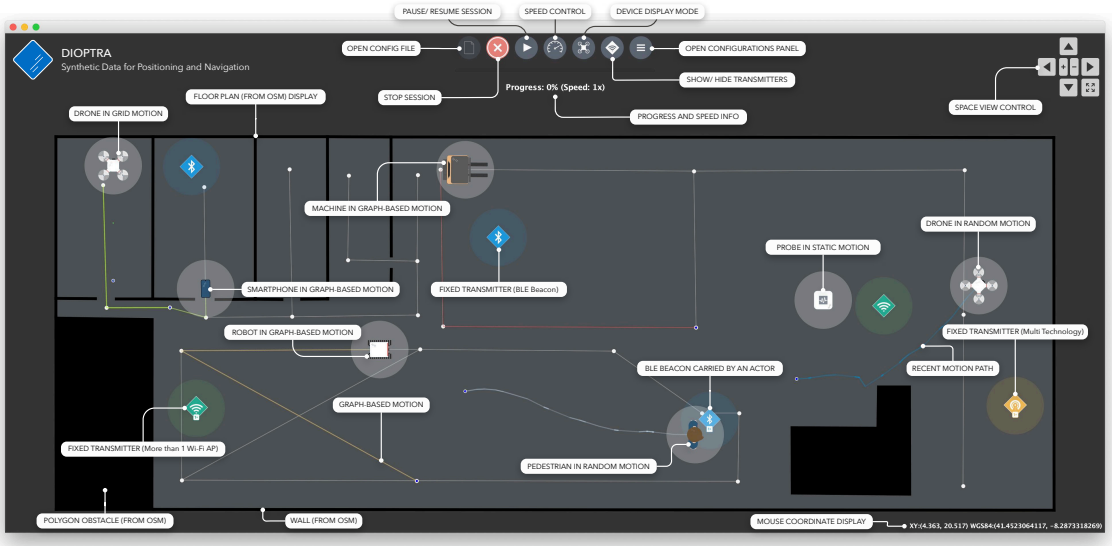


Figure B.2: Graphical user interface of Dioptra [17] (dark mode enabled).

During a session, users have controls to pause, resume or end the simulation, and to change the view parameters. Figure B.2 shows a work session running with six transmitting devices, three fixed Wi-Fi APs, two fixed BLE Beacons and a BLE Beacon carried by a moving actor. The session includes seven actors, with distinct parameters and sensors including: two pedestrians (one in random motion and another in graph-based motion), a machine and a robot in graph-based motion, and three drones (one in grid motion, one in random motion, and a Probe (fixed scanner) in static mode).

B.4 Generating a Radio Map for Pedestrian Tracking

A use-case scenario for Dioptra is the generation of synthetic radio maps and test datasets for assessing fingerprinting methods. Following, the steps necessary to generate a radio map are described:

- **Set the simulation parameters:** The main parameters of the simulation should be set first, e.g.: the simulation duration, output directory of the folder where data is saved, the simulation title/description.
- **Set the space:** The information regarding the building and floor, including the floor height (in m) and the corresponding floor plan (path to the OSM file), must be provided. The attenuation parameters regarding the building materials should also be provided.
- **Define the actors:** The actors involved in the work session must be identified and configured accordingly. Actors can be anything from fixed transmitters and autonomous vehicles to pedestrians. It is suggested that all actors involved in the infrastructure should be added first, e.g., the Wi-Fi APs, and the actors involved in the data collection should appear later (e.g. a pedestrian carrying a smartphone to collect Wi-Fi fingerprints in the RPs over a grid).

The type of signals broadcast by the actors acting as emitters must be identified. For Wi-Fi APs (type: "wifi_ap"), the static motion model is preferred as the APs are usually fixed to walls or ceilings. Technology specific parameters include the MAC address, the channel, the transmission power, and the wireless network SSID. An actor can emit multiple signals emulating, for instance, an AP broadcasting multiple virtual APs with different parameters.

For actors that act as receivers, it is necessary to specify the type of signals it receives and its corresponding motion model. To generate a radio map, an actor should collect data, for instance over a regular

grid. At each RP of the grid, the actor will collect data for a few seconds (configurable in the motion model). The propagation model is set on the sensor's side to allow the generation of synthetic data with different propagation scenarios in the same work session.

The result of the steps described above is a configuration file as shown in Figure B.3.

```
{
"settings": {
  "sim_duration": 1,
  "random_seed": -1, //With -1 a random seed will be used
  "output_folder": "Dioptra Data",
  "output_space_model": false,
  "description": "Generate a Wi-Fi Radio Map."},
"space": {
  "floors": [
    {.../// Only the first space/floor is parsed
      "building_id": "0","floor_id": "0","floor_height": 0.0,
      "osm_file_name": "scenarios/factory_scenario.osm"
    }.../// Multi-building Multi-floor will come in future versions
  ],
  "materials": [
    { "material": "Red Brick", "material_id": "red_brick",
     "attenuation_db": 4.4},... // More materials
  ]},
"actors": [
  { // Actors emitting Wi-Fi probes (APs)
    "actor_id": "WIFI AP 1",
    "type": "fixed_transmitter",
    "status": "enabled",
    "motion_model": {  "model_name": "static",
                       "x": 20.036, "y": 4.490, "z": 3.0,
                       "yaw": 0},
    "transmitters": [
      {  "transmitter_id": "AP 1",
        "type": "wifi_ap", "status": "enabled",
        "mac": "00:00:00:00:00:00", "channel": 1,
        "power": -21, "ssid": "Network X"
      },.../// More networks if allows Virtual nets
    ]
  },.../// More Wi-Fi APs
  { // Actor collecting the radio map
    "actor_id": "RadioMap",
    "type": "uav",
    "status": "enabled",
    "motion_model": {
      "model_name": "grid", "z": 1.0,
      "grid_size": 3.0, "time_per_point": 10.0,
      "mode":  "mapping_mode"},
    "sensors": [
      {  "sensor_id": "WIFI1",
        "sample_frequency_hz": 1,
        "sensor_type": "WIFI",
        "noise_sigma": 4, "sensitivity": -90,
        "beacon_rec_prob_function": "g0",
        "propagation_model": {
            "model_name":"ldpl", "fs_path_loss": 2},
        "status": "enabled"},
      {  "sensor_id": "REF1",
        "sample_frequency_hz": 1, "sensor_type": "POSI",
        "status":"enabled"}]}]}
```

Figure B.3: Example of configuration file for radio map collection.

Contributions

Dioptra was created in collaboration with Cristiano Pendão, who had the idea of creating the tool and already had a previous contribution in this subject [136] with the simulation engine and the LDPL model that considers attenuation caused by obstacles. The full team is composed of Adriano Moreira, Cristiano Pendão, Ivo Silva, and Joaquín Torres-Sospedra. The main contributions from this dissertation to Dioptra are:

- Conceptualization ideas;
- Software Programming;
- Space models support (definition of indoor spaces based on OSM format and display of floor plans in the application);
- Grid and graph motion models;
- Support for BLE beacons as transmitters.

Dioptra is a free and open-access tool, soon to be open-source. It was designed to be easily extensible to support other sensors, motion models and propagation models so that the research community can contribute to the project. Dioptra contributes to the fair comparison between IPSs and research reproducibility, as researchers can share their synthetic datasets with the community.

Many times, researchers abstain from sharing real-world datasets because they are tied to their funding institution, and they contain important information regarding a real-world scenario, which may have privacy concerns. Furthermore, real-world datasets assume several tasks to be performed namely, the preparation of the setup, the collection of data and the preparation of the data to be shared. These are some of the reasons why researchers often abstain from sharing their datasets. Dioptra solves these issues as it provides a simple and easy way of generating datasets, without any privacy concerns, which can be easily shared with the community. The datasets themselves do not need to be shared, just the configuration file necessary to run the simulation session from which the datasets are generated.

Appendix C

Interpolation of Radio Maps

Traditional methods to construct radio maps rely on the manual measurement of the intensity of radio signals, a site-survey that is expensive in time and human-resources, hence the larger the area to survey the longer it takes to complete building the radio map. Interpolating radio maps is an alternative to the use of industrial vehicles to build and maintain radio maps. Interpolation of radio maps aims to reduce or even eliminate the need for human intervention in the construction of radio maps. RSS measurements, obtained from site-surveying or from Monitoring Devices, at a few different locations are used as input for mathematical models that produce interpolated RSS values for all other locations, creating the radio map. This process can also be used to keep the radio map updated over time without additional effort.

The interpolation of radio maps can complement the self-healing radio maps solution since it allows to estimate RSS values in areas where vehicles cannot operate, and maintains the radio map updated. As disadvantages, it depends on a dedicated infrastructure (Monitoring Devices) to collect Wi-Fi samples, and some interpolation techniques depend on knowing the APs positions, the floor plan and in some cases the building materials of walls and obstacles.

Although, a brief introduction of radio map interpolation methods was made in Section 2.4.3, this appendix presents a more detailed analysis of some of these methods. In addition, it includes an RBF implementation to demonstrate how this method can be explored for building and maintaining radio maps.

C.1 Radio Map Interpolation Methods

Some of the most well-known techniques for the interpolation of radio maps are: the Kriging method, the Inverse Distance Weighted (IDW) method, and the Radial Basis Functions (RBF) method. These techniques allow to estimate RSS values based on mathematical models. The interpolation of RSS values depends on a set of RPs also known as observation points, in which the RSS values of APs were measured. Then, the RSS values are estimated for each unobserved point based on the information that was previously observed at each RP.

C.1.1 Kriging Method

The Kriging method, used in [90, 91, 92], interpolates the RSS levels at a given position by creating a spatial structure that can be represented by a mathematical function to provide continuous values. The Kriging method assumes that the RSS levels from neighboring locations are more correlated than the RSS levels from locations that are farther away, hence it considers that a spatial random variable is spatially dependent. The semivariogram between the RP and the unobserved points to be estimated is necessary in order to estimate the RSS values. The semivariogram represents the half the average squared difference between two points. The experimental semivariance $\hat{\gamma}\,(d)$ is mathematically estimated by the following equation:

$$\hat{\gamma}\,(d) = \frac{1}{2\nu(d)} \sum_{\nu(d)} \left[q\,(u_i) - q\,(u_j) \right]^2 \tag{C.1.1}$$

where d is the distance between points u_i and u_j, $\nu(d)$ is the number of pairs of sample points separated by d, and $q(u)$ is the measured RSS at point u.

The main purpose of the Kriging method is to provide an unbiased estimation with minimum error variance at unobserved locations, using a formulated function based on known measurements. The function is determined by the semivariogram. Existing Kriging estimators are based on the basic linear regression

estimator $\hat{q}(u_0)$, expressed as:

$$\hat{q}\left(u_0\right) = \sum_{i=1}^{n} \lambda_i q\left(u_i\right) \tag{C.1.2}$$

where u_0 is the location whose RSS will be estimated, $q(u_i)$ is the RSS measurement obtained at point u_i, n represents the total number of known measurements, and λ_i is the Kriging weight of $q(u_i)$. The λ_i value is estimated by the equation below:

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} \gamma_{11} & \cdots & \gamma_{1n} \\ \vdots & \ddots & \vdots \\ \gamma_{n1} & \cdots & \gamma_{nn} \end{bmatrix}^{-1} \begin{bmatrix} \gamma_{10} \\ \vdots \\ \gamma_{n0} \end{bmatrix} \tag{C.1.3}$$

where γ_{ij} is the semivariogram between $q(u_i)$ and $q(u_j)$, and γ_{i0} is the semivariogram between the observed RSS value and the unobserved value $q(u_0)$.

With only a few calibration points, the Kriging method is able to create a complete radio map of the building. This method shows good interpolation results because, theoretically, the estimation errors show minimum variance.

Universal Kriging is used in [90] to interpolate the radio map in the offline phase. For that, it considers that RSS measurements at adjacent locations have a spatial correlation, thus it uses the Kriging method to interpolate the RSS average and standard deviation. This system requires an initial site survey to obtain annotated Wi-Fi samples which are then used by the interpolation method to generate a certain number of interpolation points and associated fingerprints.

C.1.2 Inverse Distance Weighted

The IDW method follows a deterministic model to interpolate unknown points. RSS values can be estimated for unobserved locations by calculating a weighted average of the RSS values available at the known locations. IDW uses spatial autocorrelation that gives more emphasis to values from points that are closer while values from points farther away have less effect.

Based on the generic equation for IDW interpolation, presented in [93], the following equation can be used to estimate the RSS at unobserved locations:

$$\hat{z} = \frac{\sum_{i=1}^{n} \omega_i \times RSSI_i}{\sum_{i=1}^{n} \omega_i} \tag{C.1.4}$$

where $\hat{z}$ is the estimated RSS value for the considered point, $RSSI_i$ represents the signal strength at reference point i, ω_i is a weight value that is based on the distance between the RP and the point to be estimated.

The ω value assigned to each observed location is obtained from:

$$\omega = \frac{1}{d_i^{\varepsilon}} \tag{C.1.5}$$

where ε is the power parameter defined by the user and d is the distance between the point to be estimated and the RP. As shown by Equation C.1.5, this method favours the RPs that are closer to the point to interpolate because it directly relates the distance to the weight value, the further a RP is from the point to be interpolated the lower will its assigned weight.

Since the IDW method only considers the distance when interpolating RSS values, it will not take into account the indoor layout of the building, i.e., the walls, the doors, building materials, etc. Therefore, it might have errors in the interpolated RSS values. In many cases, the Kriging method is preferred over IDW, because it provides a more robust model in relation to the IDW method.

Two variants for the IDW method can be used, one that considers all RPs and another that only considers the RPs that are within a certain radius of the point to be interpolated.

C.1.3 Radial Basis Functions

RBF networks are a class of artificial neural networks that were explored in [87, 88] to interpolate RSS values of a radio map from observations at known RPs. Generally, RBF networks have an input layer, a hidden-layer with radial basis functions and an output layer.

A radial basis function takes a set of measured RSS values at different reference points as input and produces the estimated signal strength values for the point to be estimated, therefore, the set of signal

strength values ws for a location $\rho(x, y)$ can be defined as:

$$\rho\left(ws\right) = \sum_{i=0}^{M-1} \omega_i K\left(\|ws - ws_i^*\|\right) \tag{C.1.6}$$

where $K(r)$ is a kernel function, M represents the number of basis functions, ws_i^* are the RSS vectors of the reference points, ω_i are the network weights based on calibration data, and $\|ws - ws_i^*\|$ is the Euclidian distance between the test signal strength vector ws and kernel center ws_i^* in signal strength space. The selected kernel function for localization is the Gaussian kernel function:

$$K(d) = exp\left(- \frac{d^2}{2r_0^2} \right) \tag{C.1.7}$$

where d represents the Euclidean distance between the two vectors and r_0 is a scale factor.

The interpolation accuracy using Gaussian basis functions can be very sensitive to the r_0 parameter. Generally, the r_0 value should be larger than the typical distance between points and smaller than the "outer scale" of the function to interpolate [137]. Therefore, some experimentation is required to achieve an optimal r_0 for high interpolation accuracy.

RSS values at a given location can be estimated using RBF without knowing the position of APs and without the indoor layout such as the walls and building materials, hence it can be useful specially in situations where the building's floor plan is not available and there are many APs whose positions are difficult to map.

C.2 Radio Map Interpolation using Radial Basis Functions

RBF was selected as the method to interpolate the radio map as a proof of concept that this method can be used to complement the self-healing radio maps solution based on industrial vehicles. As previously mentioned, the main advantages of RBF are that it does not depend on the knowledge of the APs' positions and the building's floor plan and building materials. It also does not depend on information gathered from crowdsourced data and, most importantly, it does not require any human intervention to construct the radio map. The current section describes how the RBF method is used to estimate the radio map and how

it performs in a real-world scenario.

C.2.1 Inputs

Signal strength values from Wi-Fi samples collected at known positions (RPs) are used as input values for the radial basis functions. One can merge Wi-Fi samples collected at the same RP and average RSSI values in order to reduce the impact of sudden variations that can occur in the received signal strength.

For instance, estimating a radio map with $radioMapDuration = t_1 - t_0$ considers Wi-Fi samples collected between time instants t_0 and t_1. A time interval can be used for merging Wi-Fi samples ($mergeInterval$), as represented in Figure C.1, where Wi-Fi samples are grouped into four merging intervals. Both $radioMapDuration$ and $mergeInterval$ are parameters that can be configured appropriately.

In the example of Figure C.1, the result of the merging process are four Wi-Fi samples where the RSSI values of each detected AP are averaged, as shown next:

$$\begin{aligned} ws^{m_0} &= \langle m_0 \rangle, m_0 = \{ws^0, ws^1, ws^2\} \\ ws^{m_1} &= \langle m_1 \rangle, m_1 = \{ws^3, ws^4\} \\ ws^{m_2} &= \langle m_2 \rangle, m_2 = \{ws^5, ws^6, ws^7\} \\ ws^{m_3} &= \langle m_3 \rangle, m_3 = \{ws^8, ws^9, ws^{10}\} \end{aligned} \tag{C.2.1}$$

where ws^m contains averaged RSSI values from Wi-Fi samples represented by the merging interval m_i set.

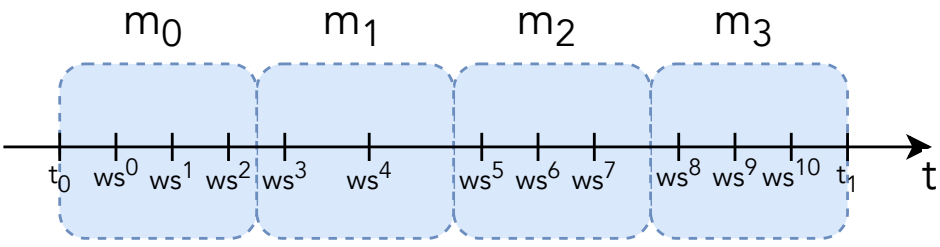


Figure C.1: Wi-Fi samples merging.

Since there are several RPs, the set of merged Wi-Fi samples at a given merge interval i, is defined by:

$$WS^i = \left\{ws_1^i, ws_2^i, ..., ws_M^i\right\} \tag{C.2.2}$$

where WS^i represents the list of merged Wi-Fi samples from all RPs (considering a total of M reference points), for merge interval i.

In cases where an AP is not detected at a particular RP it is assigned a default RSSI value defined by the constant $RSSI_{min}$. Furthermore, APs whose signal strength is weak, i.e., lower than $RSSI_{min}$, are also assigned with the constant $RSSI_{min}$.

When using the Gaussian kernel function in the RBF method, interpolated values far from the RPs tend rapidly to zero, due to the form of the Gaussian function. Since signals from APs are negative values, it is necessary to convert RSSI values in a way that when RSSI values are low, they tend to zero. Since the lowest possible RSSI value is defined by $RSSI_{min}$, one can convert the RSSI using the following expression:

$$RSSI_{conv} = RSSI - RSSI_{min} \tag{C.2.3}$$

After the RBF method is applied, the RSSI estimated values can be converted back to the original domain using:

$$RSSI = RSSI_{conv} + RSSI_{min} \tag{C.2.4}$$

A grid layout defines the output points where RSSI values are estimated by the RBF method. Although it is not necessary to have any knowledge of the floor plan to use the RBF method, it is necessary to know the bounds of the building to create the grid of points where the signal strength values are estimated. Thus, the building's bounded box is necessary to determine the limits of the grid. In addition, the grid size is a parameter that is directly linked with the resolution of the radio map because it defines the distance between neighboring points. The list of output points is defined as RP.

C.2.2 Algorithm

Algorithm 5 describes how the RSSI values are estimated for one AP. It has to be repeated for each AP detected in Wi-Fi samples. The algorithm takes WS and RP as inputs. WS represents the set of merged Wi-Fi samples for a total of n merge intervals and RP represents the list of grid points where RSSI will be estimated.

Algorithm 5 Estimate RSSI values of an AP using RBF.

Input

WS - set of merged Wi-Fi samples from n merging intervals
RP - set of reference points where RSSI values will be estimated

Output

E - the set of RSSI estimates for the AP

1: **procedure** RBF to estimate AP RSSI values
2: $E = \{\}$
3: $i = 1$
4: **while** $i \leq n$ **do**
5: $aux^i = \text{convertRssi}(WS^i, RSSI_{min})$
6: $rbf = \text{createRbfModel}(aux^i, RP)$
7: $E^i = \text{estimateRssi}(rbf, RP)$
8: $E^i = \text{convertRssiInverse}(E^i, RSSI_{min})$
9: $E = E \cup E^i$
10: $i = i + 1$

For each cycle iteration, the AP's signal strength values are interpolated for all points that compose the grid. For that to be accomplished the algorithm runs as follows: first, Equation C.2.3 is used to convert RSSI values and assign the $RSSI_{min}$ constant when AP is not detected, then the resulting RSSI values are stored in aux^i; second, the RBF model is created using the merged RSSI values measured by monitoring devices (aux^i); third, interpolated RSSI values for all RP points are obtained from the RBF model and then, stored in E^i; fourth, RSSI values are converted back to the original form, using Equation C.2.4; fifth, values from E^i are added to E which contains all estimated RSSI values associated with each iteration of the cycle. Once the cycle has finished, E contains all RSSI values associated with each merged value.

C.2.3 Real-world validation

A real-world experiment was performed to evaluate the positioning performance of RBF interpolated radio maps for Wi-Fi fingerprinting. The testing setup, already described in Section 3.2, includes several Monitoring Devices deployed in known locations on two floors of a building.

Monitoring Devices continuously collected Wi-Fi samples (every 60 s) during 24 h. These Wi-Fi samples were then used to create the radio map using RBF. Using a grid size of 1 m translates into 1782 RP points, for which RSSI values were interpolated.

To evaluate the positioning error, test data was manually collected at 49 locations throughout the first floor, and 20 Wi-Fi samples were collected at each location. Once the test data was collected, a position estimate was obtained for each test Wi-Fi sample using the Wi-Fi fingerprinting technique with the k-NN algorithm using the Manhattan distance as the dissimilarity function. In addition, in cases where an AP was not detected in a Wi-Fi sample when obtaining the similarity between two Wi-Fi samples, the AP was ignored, i.e., it was not considered in the similarity function. Performed tests have shown that using this method does not result in worse positioning performance in comparison with cases where all APs in both Wi-Fi samples are considered, and if the AP is not detected in one of the Wi-Fi samples, its RSSI is replaced with a default value (i.e., -90 dBm).

The following parameters were used for testing the RBF method:

- $RSSI_{min}$: -90 dBm;
- $radioMapDuration$: 24h (considers the Wi-Fi samples collected immediately before starting the collection of test data);
- $mergeInterval$: 2h, 4h, 8h, 12h and 24h (meaning that all RSSI values obtained during this time interval were averaged into one);
- k: 3, 5, 10, 15, and 20 (values used to test the k parameter of the k-NN algorithm);
- $gridSize$: 1 m;
- r_0: 9 (defined after experimenting different values).

The r_0 value was tuned based on experiments where different r_0 values were used to obtain distinct radio maps interpolated with RBF. Then these radio maps were evaluated with Wi-Fi fingerprinting, and the positioning performance was compared. The values that achieved better performance are presented

in Table C.1. Given that the results are similar, it was defined $r_0 = 9$ for the remaining experiments.

Table C.1: Positioning results for different r_0 values (in meters).

	r_0=8	r_0=8.5	r_0=9	r_0=10	r_0=12.8
Mean	6.05	5.89	5.90	6.03	6.74
Median	4.24	4.05	4.08	4.31	5.69
P_{95th}	15.72	15.86	15.40	15.57	15.84
P_{99th}	20.02	19.56	20.52	19.38	19.20
Max	88.30	89.92	88.87	89.72	94.23

Figure C.2 depicts the interpolated RSSI values obtained for all points of the grid using the RBF algorithm, for a particular AP. As expected, RSSI values are low (equal to $RSSI_{min}$) in locations far from the AP, where it was not detected. Conversely, in areas where it was detected RSSI values follow a Gaussian curve, with higher RSSI values closer to the centre of the curve. RSSI values rapidly decrease as the distance to the centre of the Gaussian curve increases, thus RSSI values tend to $RSSI_{min}$ in these areas. Figure C.2 also includes the RSSI values that were measured at each test position for comparison with the interpolated values.

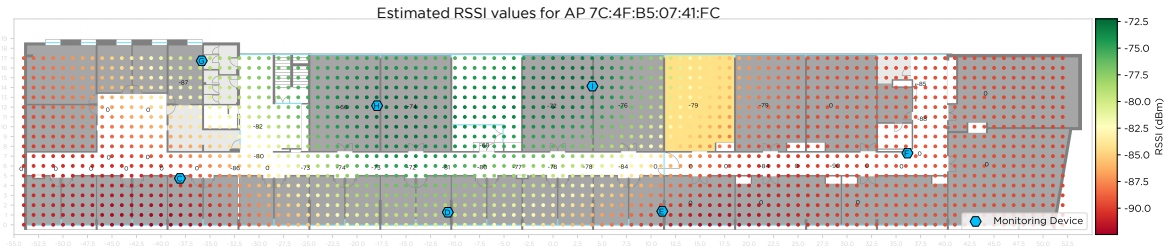


Figure C.2: Estimated RSSI values for a given AP using the RBF technique.

Table C.2 shows the positioning error using different merging intervals for the interpolated radio maps. The results are similar for merging intervals of 2h, 4, 8h, and 12h, but a clear improvement is observed when the merging interval is 24h.

Another test was conducted to evaluate how the k parameter from the k-NN algorithm might influence the results. In this case, two different merging intervals were considered, 2h and 24h, because with a 2h merging interval there are 12 Wi-Fi samples for each grid point of the radio map (denser radio map),

Table C.2: Positioning results for different merging intervals (in meters).

	Merging interval				
	2h	**4h**	**8h**	**12h**	**24h**
Mean	6.60	6.61	6.72	6.76	5.77
Median	4.24	4.26	4.38	4.40	4.21
P_{95th}	15.10	15.23	14.91	14.76	14.02
P_{99th}	91.92	92.41	91.55	92.54	20.30
Max	97.60	96.18	96.47	97.03	96.54

and with a 24h merging interval, there will be only one Wi-Fi sample for each grid point of the radio map. Results of this test are presented in Table C.3.

Table C.3: Positioning results for different k values (in meters).

	k = 3	**k = 5**	**k = 10**	**k = 15**	**k = 20**	**Merging Interval**
Mean	6.75	6.68	6.63	6.60	6.61	
Median	4.42	4.33	4.27	4.24	4.20	
P_{95th}	15.71	15.51	15.15	15.10	15.07	2h
P_{99th}	92.82	93.20	92.21	91.92	92.20	
Max	97.99	98.63	97.94	97.60	96.95	
Mean	5.86	5.81	5.80	5.77	5.73	
Median	4.22	4.15	4.27	4.21	4.21	
P_{95th}	14.96	15.17	14.47	14.02	13.80	24h
P_{99th}	20.39	20.20	20.20	20.29	20.54	
Max	99.24	97.81	97.45	96.54	96.53	

Overall, the positioning performance is better for a 24h merging interval when compared with results of a 2h merging interval, independently of the k parameter. Varying the k parameter does not lead to significant changes in the positioning error, however, higher k values lead to minor improvements in positioning results. No significant variation in the positioning error is noticed because the radio map is interpolated using the RBF method where interpolated RSSI values follow a Gaussian curve, therefore, the estimated RSSI values will always be similar around the same location, which explains why the k value does not have a significant impact on the positioning error. In reality, walls and other obstacles affect the propagation of

radio signals which do not follow a Gaussian curve, hence the interpolated radio map does not represent the real radio environment of the building floor, but is an approximation. If a manual radio map was used in this case, one would expect a higher variation in the positioning results, when testing different k values.

The reported results show very large errors above the 95th percentile which are uncommon with Wi-Fi fingerprinting. To help ascertain the cause of these errors, Figure C.3 depicts the average positioning error at each testing location for the merge interval of 24h and $k = 15$. Higher errors exist in test points near the end of a corridor, where there are no Monitoring Devices to cover that particular area. This causes a few APs to be detected with strong signals in two distinct areas of the building, causing large errors in 3 test points, as shown in Figure C.3. Despite that, the mean errors achieved with this technique are in line with the typical values achieved with Wi-Fi fingerprinting. The overall mean error in this test is 5.77 m, with 14.02 m and 20.29 m for the 95th and 99th percentiles, respectively.

It is important to note that only 7 Monitoring Devices were considered to interpolate the radio map in this scenario that measures 106 $\times$ 19 m, hence there is a Monitoring Device per $\approx$288 m^2. A better performance is expected if the density of Monitoring Devices per square meter increases.

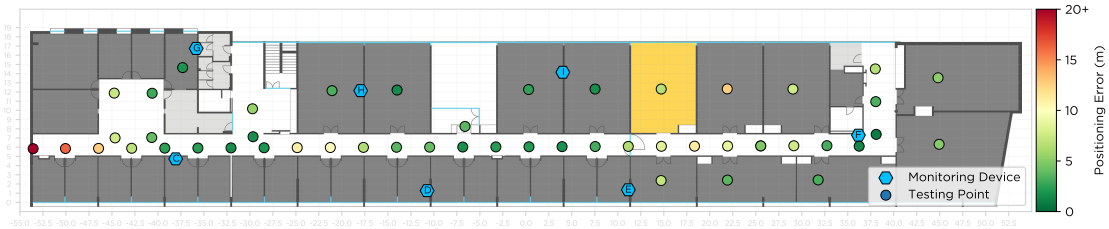


Figure C.3: Average positioning error at each testing location.

C.3 Summary

This appendix presented a deeper analysis of several methods to interpolate radio maps, namely, the Kriging method, the IDW method, and the RBF method. It also included an implementation of the RBF method and its validation in the real world. Several parameters were evaluated to understand how they might have an influence over the results.

The RBF approach is flexible because it does not depend on the positions of APs and the floor plan infor-

mation to interpolate radio maps, but as consequence, it is not as accurate as other methods, such as the LDPL model. The advantages of using RBF for interpolation of radio maps are: setup and configuration are simple since it just requires installation of Monitoring Devices; performs similarly to standard Wi-Fi fingerprinting solutions with a low density of Monitoring Devices; and, can be used in combination of the industrial vehicles self-healing radio maps solution. Conversely, the drawbacks of this approach are that it requires installation of infrastructure and a dedicated server to gather Wi-Fi samples from all Monitoring Devices, and it has very large maximum errors, that can be mitigated by adding more Monitoring Devices. The overall cost of the infrastructure can be reduced by using cheaper hardware, e.g., Raspberry Pi Zero devices.