Time-based indoor positioning and context information using commodity WiFi chipsets

by

Maurizio Rea

in partial fulfillment of the requirements for the degree of Doctor in

Telematic Engineering

Universidad Carlos III de Madrid

Advisor: Domenico Giustiniano

May 2020

Time-based indoor positioning and context information using commodity WiFi chipsets

Prepared by:
Maurizio Rea, IMDEA Networks Institute, Universidad Carlos III de Madrid
contact: maurizio.rea@imdea.org

Under the advice of:
Domenico Giustiniano, IMDEA Networks Institute
Telematic Engineering Department, Universidad Carlos III de Madrid

This work has been supported by:

To my (almost) wife Gaia, for supporting me every day with her love.

To all my family, for making a real distance a small estimation.

Thank you!

Acknowledgements

I would like to use the first pages of this thesis to thank everyone that helped me during the last years along my PhD journey. I wish to express my first gratitude to my supervisor, Dr. Domenico Giustiniano, for his support during my PhD. He provided me constructive feedback and rigorous reviews of papers and thanks to him, I have been involved in many presentations between meetings, seminars, workshops and conferences. This made me believe more in myself and gave me the confidence to be able to introduce and present a research topic in front of other people. He was always pushing me but in a nice way. I cannot forget the late nights for paper submissions we spent together, either via skype or face to face (like Mobicom'19).

I would also like to thank Dr. Danilo De Donno for helping me during the first year of my PhD. In the first period of adaptation, he supported me with his experience and he gave many different suggestions during meetings.

I am very grateful to Dr. Holger Claussen and Dr. Traian Emanuel Abrudan who hosted me for 6 months during my internship in Nokia Bell Labs of Dublin. In particular, Dr. Traian Emanuel Abrudan brought out the best in me during that period in which he made me change my way of thinking and affording research problems. Let me also express my gratitude to Héctor Cordobés who helped me during a period of my PhD in different areas (besides jokes).

Furthermore, I would like to thank my colleagues at IMDEA Networks. They really made my PhD an extraordinary experience. I had very good times with all of them. Let me start to thank my "favorite" ones, Roberto and Dario, who started this experience with me and they always had time to joke, even during the working hours. Thanks to my old-desk mate Christian and my non-old-desk mate Mohamed for dealing nicely with my attacks. Finally, I would like to thank all IMDEA Networks team (professors, IT, administrative, human resources, support, etc) for helping and making everything easier.

Published Content

The ideas and investigations of this thesis resulted in the following refereed publications:

[1] Maurizio Rea, Aymen Fakhreddine, Domenico Giustiniano, Vincent Lenders. Filtering Noisy 802.11 Time-of-Flight Ranging Measurements from Commoditized WiFi Radios. Published in *IEEE Transaction on Networking, vol. 25, no. 4, pp. 2514-2527*, August 2017. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7932994&isnumber=8011587

- This work is fully included and its content is reported in Chapter 2.
- The author worked on the design, implementation and evaluation of the main adaptive filter for noisy Time-of-Flight (ToF) measurements. After the setup and the deployment of the localization system in different testbeds, showing the distance accuracy, the robustness to different environments and ranging capacity, the author also investigated a new version of the model calibration for the adaptive filter, to be used without requiring additional manual effort.

[2] Maurizio Rea, Domenico Garlisi, Héctor Cordobés de la Calle, Domenico Giustiniano. Location-Aware MAC Scheduling in Industrial-Like Environment. Published in *Broadband Communications, Networks, and Systems (BROADNETS 2018)*, 19-20 September 2018, Faro, Portugal. https://doi.org/10.1007/978-3-030-05195-2_20

- This work is fully included and its content is reported in Chapter 4.
- The author participated in writing several parts of this paper and his role in this work is focused on the design, implementation and experimentation of the proposed methodology.

[3] Maurizio Rea, Héctor Cordobés de la Calle, Domenico Giustiniano. Time-of-flight Wireless Indoor Navigation System for Industrial Environment. Published in *the 13th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & CHaracterization (ACM WiNTECH 2019)*, 16-20 October 2019, Los Cabos, Mexico. https://doi.org/10.1145/3349623.3355476

- This work is fully included and its content is reported in Chapter 4.

- The author participated in writing several parts of this paper and his role in this work is focused on the design, implementation and experimentation of the proposed system.

[4] Maurizio Rea, Traian Emanuel Abrudan, Domenico Giustiniano, Holger Claussen, Veli-Matti Kolmonen. Smartphone positioning with radio measurements from a single wifi access point. Published in *the 15th International Conference on Emerging Networking Experiments And Technologies (CoNEXT 2019)*, 9-12 December 2019, Orlando, Florida, U.S. https://doi.org/10.1145/3359989.3365427

- This work is fully included and its content is reported in Chapter 3.
- The author participated in writing several parts of this paper and his role in this work is focused on the design, implementation and evaluation of a single-Access Point (AP) system.

[5] Maurizio Rea, Domenico Giustiniano, Joerg Widmer. Virtual Inertial Sensors with Fine Time Measurements. [Under submission]

- This work is fully included and its content is reported in Chapter 5.
- The author participated in writing several parts of this paper and his role in this work is focused on the design and evaluation of the proposed algorithms.

[6] Maurizio Rea, Héctor Cordobés de la Calle, Domenico Giustiniano, Domenico Garlisi, Pierluigi Gallo, Spilios Giannoulis, Ingrid Moerman. Poster: Integration of WiFi ToF Positioning System in the Open, Flexible and Adaptive WiSHFUL Architecture. Published in *the 11th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & CHaracterization (ACM WiNTECH 2017)*, 16-20 October 2017, Snowbird, Utah, USA. https://doi.org/10.1145/3131473.3133334

- This work is partially included and its content is reported in Chapter 4.
- The author's role in this work is focused on the design and implementation of the architecture through Unified Program Interface (UPI)s.

[7] Maurizio Rea, Héctor Cordobés de la Calle, Domenico Giustiniano, Vincent Lenders. Robust WiFi Time-of-Flight Positioning System (Demo). Published in *the 15th ACM/IEEE International Conference on Information Processing in Sensor Networks - Microsoft Indoor Localization System (IPSN 2016)*, 11-14 April 2016, Vienna, Austria. http://eprints.networks.imdea.org/1192/

- This work is partially included in Chapter 4.

- The author's role in this work is focused on building the full demo.

[8] **Maurizio Rea**, Héctor Cordobés de la Calle, Domenico Giustiniano. TWINS: Time-of-flight based Wireless Indoor Navigation System (Demo). Published in *the 17th ACM/IEEE International Conference on Information Processing in Sensor Networks - Microsoft Indoor Localization System (IPSN 2018)*, 11-13 April 2018, Porto, Portugal. http://eprints.networks.imdea.org/1869/

- This work is partially included and its content is reported in Chapter 4.
- The author's role in this work is focused on building the full demo.

Abstract

In the last years indoor localization and applications that use positioning information have attracted a lot of attention from the research community as well as the industry. The achieved accuracy of an indoor localization system is the key to enable certain applications, such as navigation. In a scenario where the system runs exclusively on commodity hardware such as smartphones and even without installing any mobile app in the mobile device, location information may be exploited not only for navigation, but also for the benefit of the network itself or for the investigation of physical behaviours. In indoor areas, communication technologies such as Wireless Fidelity (WiFi) are gaining popularity due to the ever increasing availability and deployment of APs that can be used simultaneously as an infrastructure for networking and positioning. Despite being the most used technology for tracking devices in indoor environments, the research community has focused more intensively on approaches such as the Received Signal Strength Indicator (RSSI) or combining WiFi signals with inertial sensors as found in smartphones. These approaches are error prone, dependent on frequent calibrations, or they require specialized hardware or the user must be interested to run a specific application designed to run on the smartphone. Using only commodity WiFi chipsets, this thesis investigates location systems that are based on Time-of-Flight (ToF) echo technique, and do not require any calibration and user intervention, and it explores ranging and location data to improve network management and infer user behaviour.

ToF technique has been modestly exploited for indoor localization, whereas ToF is successfully used in Global Positioning System (GPS) in outdoor environments. The reason is that WiFi ToF measurements, mainly extracted from commodity chipsets, suffer from extensive device-related noise which makes it challenging to differentiate between direct path from non-direct path signal components when estimating the ranges. Existing multipath mitigation techniques tend to fail at identifying the direct path when the device-related Gaussian noise is in the same order of magnitude, or larger than the multipath noise. In order to address this challenge, we first propose in this thesis a new method for filtering ranging measurements, extracted from a commodity WiFi chipset, that is better suited for the inherent large noise as found in WiFi radios. Our proposed filter combines statistical learning and robust statistics and it does not require specialized hardware,

the intervention of the user, or cumbersome on-site manual calibration. This makes the method we propose as the first contribution of the present work particularly suitable for indoor localization in large-scale deployments using existing legacy WiFi infrastructures. We build and investigate a multi WiFi ToF-based APs localization system, where these filtered timing signals are used as ranging inputs of the multi-lateration problem for positioning. We then deploy and evaluate our system for indoor mobile tracking scenarios in many multipath-rich environments, across multiple testbeds which cover different surfaces and further test it in Microsoft indoor localization competitions, demonstrating that, despite these challenges, our technique can achieve distance accuracy comparable to other approaches proposed by the start of the art but does not share their aforementioned shortcomings.

The deployment of a multi-APs positioning system is targeted to specific areas, such as large offices and shopping malls. In order to bring indoor positioning also to homes and small businesses which typically have a single AP, the next step we envision towards a hybrid single-AP positioning system is a deep inspection and interaction of the Fine Time Measurements (FTM), a type of ToF echo technique standardized recently, and Channel State Information (CSI) for ranging and Angle-Of-Arrival (AOA) for angle estimates, respectively. We exploit Physical Layer (PHY) information to detect the number of paths and their directions and we use this information to derive a new method for filtering ranging measurements obtained with the FTM protocol. We achieve sub-meter distance estimation accuracy eliminating the adverse effect of multipath in FTM using calibrated inputs from CSI. We then evaluate the system in multipath-rich environments, demonstrating the capability of the combination of AOA estimation and the proposed FTM refinement approach to achieve reasonable positioning accuracy in areas comparable to typical flat sizes just using a commodity smartphone as target device.

All the contributions described so far and all the extracted location data may be exploited also in the network core to better allocate network resources based on the expected link performance. Thus, we take advantage of positioning data and more in general context information, to enable reliable mobile communications via advanced resource management policies and adaptive traffic engineering strategies. Of particular interest for this thesis is to investigate usage of positioning data in industrial environments, where the presence of metallic objects challenge the reliability of wireless communication. With the advent of the fourth Industrial revolution (Industry 4.0) such harsh environments have attracted high attention and, for this reason, of particular interest for this thesis is also the deployment of our multi-APs system in such industrial environment. The latter, due to blockage and strong reflections, is notorious for being adverse to wireless communication, impacting on the signal quality. We propose to exploit the knowledge of location to derive context information to dynamically allocate wireless resources in time and space to target devices. We exploit the spatial geometry of the APs and a statistical

model that maps the user position's spatial distribution to an angle error distribution to derive a hypothesis test to declare if the link is in Line-Of-Sight (LOS) or Non-Line-Of-Sight (NLOS). In order to avoid changes to the client side and operate with a single interface radio, we use the same wireless network both for positioning and scheduling. We experimentally show that context information applied to wireless resources protocol help increasing the network throughput in the aforementioned industrial-like scenario.

Finally, taking advantage of ranging information extracted from FTM, we propose to estimate the device movement without any access to physical inertial sensors in the mobile. The idea is to infer the movements of the mobile through radio measurements, a concept we call "virtual inertial sensors". We propose a method for estimating the user walking speed and a novel method for the rotation of a mobile device. We evaluate and demonstrate the proposed approaches with experiments, and we compare the rotation method with a possible solution that explores CSI measurements. While FTM works with only one single antenna, it achieves better performance than a CSI-based estimator that exploits four antennas and multiple sub-carriers at the AP, but are yet limited by the typical one single WiFi antenna at the smartphone side. This new concept of virtual inertial sensors can be leveraged by location systems and sensing mechanisms to improve localization accuracy, infer user behavior, and design better and more secure communication.

In conclusion, in this thesis we experimentally demonstrate that ToF data, extracted from commodity WiFi chipsets, can be used for tracking devices and provide context information in indoor environments without the need for environmental calibration and installations of mobile apps, but only with standard-compliant measurements performed from the WiFi infrastructure.

Table of Contents

List of Tables

List of Figures

List of Acronyms

AIC Akaike Information Criterion

AP Access Point

AOA Angle-Of-Arrival

ACK Acknowledgment

API Application Programming Interface

CDF Cumulative Distribution Function

COTS commercial off-the-shelf

CLU Central Location Unit

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance

DB Database

DC Direct Current

DCF Distributed Coordination Function

DS Direct Sequence

ECDF Empirical Cumulative Distribution Function

ESS Extended basic Service Set

EM Expectation-Maximization

EWMA Exponentially Weighted Moving Average

FFT Fast Fourier Transform

FTM Fine Time Measurements

FTM-I FTM initiator

FTM-R FTM responder

GMM Gaussian Mixture Model

GCP Global Control Program

GNSS Global Navigation Satellite System

GPS Global Positioning System

GMM Gaussian Mixture Model

HDOP Horizontal Dilution Of Precision

CSI Channel State Information

JCR Journal Citation Reports

LOS Line-Of-Sight

LLS Linear Least Squares

LBS Location-Based Services

LCP Local Control Program

LPE Localization and Positioning Engine

MPM Matrix Pencil Method

MAC Medium Access Control

MIMO Multiple Input Multiple Output

MUSIC MUltiple SIgnal Classification

NLLS Non-Linear Least Squares

NLOS Non-Line-Of-Sight

OpenFWWF Open FirmWare for WiFi networks

OS Operating System

PCIe Peripheral Component Interconnect express

PHY Physical Layer

QTNA Quantenna

QQ Quantile-Quantile

RF Radio Frequency

RGMII Reduced Gigabit Media-Independent Interface

RMSE Root Mean Square Error

RSSI Received Signal Strength Indicator

RTT Round Trip Time

SHM SHared Memory

SIFS Short InterFrame Space

SSID Service Set IDentifier

STA STAtion

TDMA Time-Division-Multiple Access

ToF Time-of-Flight

TCP Transmission Control Protocol

TWINS ToF-based Wireless Indoor Navigation System

UDP User Datagram Protocol

UE User Equipment

ULA Uniform Linear Array

UPI Unified Program Interface

WiFi Wireless Fidelity

WMP Wireless Mac Processor

Part I

Introduction

1 Introduction

Indoor localization has been an active area of research for more than a decade. The earliest indoor localization schemes were based on specialized hardware [9–12], achieving high accuracy but limiting their applications to dedicated deployments. Current localization trend goes towards Wireless Fidelity (WiFi) positioning, navigation and tracking, and the research community has focused more intensively on approaches such as the Received Signal Strength Indicator (RSSI) [13–17], the Angle-Of-Arrival (AOA) [18–20] or combining WiFi signals with inertial sensors as found in smartphones [21]. A drawback of using the RSSI is that the signal attenuation in indoor environments becomes not only depending on the distance but also highly dependent on the material of the obstacles between radio devices like walls, doors, etc. As a consequence, RSSI-based indoor localization systems require extensive site-specific calibration, in order to match RSSI measurements against the training data. Approaches that exploit the AOA require advancements in the hardware such as sophisticated array of antennas and they introduce practical challenges in commodity hardware. Other systems need input from the inertial sensors of the smartphones, requiring the active intervention of the mobile device (e.g. installation of an app). Also Time-of-Flight (ToF) echo techniques have been proposed as a way to estimate the distance between the WiFi Access Point (AP) and the STAtion (STA), and in a dense WiFi network of APs (three at least), the position of the STA can be then estimated applying a multi-lateration algorithm. As the ToF techniques do not require extensive manual on-site calibration, the need for intervention of the user, or specialized hardware, their applications for ranging mobile devices over WiFi radios have been suggested. But, despite the recent attention from the research community, there are still challenges that hinder the potential relevance of ToF for indoor localization. The main challenge with WiFi is that very precise signal propagation time estimates are required. At the speed of light, a measurement error of 1 μs already results in a distance estimation error of 300 meters. In order to achieve meter-level localization accuracy, a precision in the order of a few nanoseconds is therefore needed. Existing off-the-shelf chipsets were however not built with these strict timing requirements in place.

Another major challenge is the problem of multipath. In indoor environments where WiFi-based localization is sought to be used, Radio Frequency (RF) signals get easily reflected by obstructing objects and walls which renders ToF measurements very prone to errors when reflected signals superpose at the receiver or when there is simply no direct signal propagation path between devices. While dedicated hardware may be able to provide more accurate information of the direct path, commodity APs just synchronize to the strongest (and perhaps reflected) path. Thus, timing information available for ToF ranging estimation may be erroneous.

While different solutions in a dense network of APs for localizing a mobile device have been proposed, the problem of bringing indoor localization system to homes and small businesses, which typically have a single AP, remains open. Several techniques require different frequency bands to give the illusion of a wide-band radio [22, 23]. However, commodity smartphones do not have the feature to switch between different channels keeping the same technology for positioning and data communication. A ToF single-AP system using WiFi has been designed in [24] for localization in multipath environments, but it requires the access to inertial sensors of smartphones and extensive manual calibration per user. To the best of our knowledge, at the time of writing, there is no contribution from related work that with only one AP can localize commodity smartphone just using radio measurements.

Targets as mobile robots (machines in general) are becoming more autonomous, flexible and cooperative, and wireless networked solutions could ideally greatly help to increase the productivity in these environments. In this context, positioning data are gaining visibility but the accuracy, availability and reliability of the indoor positioning system results decisive for the correct operation of the context-awareness. Location information may be used not only as a service offered to customers, but also in the network core for advanced resource management policies and adaptive traffic engineering strategies. For instance, the Medium Access Control (MAC) protocol serves a vital role in every network and it is directly responsible for controlling access to the shared communication resources. In most cases, the network designer does not know about the network conditions and has to assume that they may change during operation. The usual approach in most MAC protocols to handle unknown or changing conditions is to include some adaptation mechanism in order to adjust the operation to the actual network load and signal-to-noise ratio, and recover from failures in data transmission.

For other applications such as dead-reckoning aided localization, fine-grained gesture and posture recognition, and mobile gaming, data from inertial sensors in smartphones has been exploited in the past years [21, 25–28]. Smartphone sensors such as gyroscopes, compass, and accelerometer have received particular attention and they can be leveraged by location systems and sensing mechanisms to improve localization accuracy, infer user behavior, and design better and more secure communication. However, the wireless

infrastructure does not have direct access to inertial sensors in mobile devices, calling for the installation of dedicated applications which requires the manual intervention of the user.

Driven by all the above observations, in this Thesis we present different ideas and solutions to solve the aforementioned problems, only using commercial off-the-shelf (COTS) WiFi hardware. We introduce two different ways for bringing positioning system in indoor environments, a multi WiFi ToF-based APs localization system and a single WiFi AP smartphone positioning. The first solution uses as inputs multiple ranging estimates (at least three), based on the ToF technique, from multiple APs, while the second proposed system combines two techniques, Fine Time Measurements (FTM) and Channel State Information (CSI), in order to provide the estimation of the position using a single AP. The FTM protocol has been standardized in the IEEE 802.11-2016 standard, with corresponding devices now entering the market. FTM allows to estimate the ranging between the mobile and the AP, even without any association. In both systems, multi and single AP, we use a single frequency band and we do not use inputs from inertial sensors. We deploy and evaluate these proposed systems across many testbeds in different areas, increasing the complexity of the scenarios, from typical office spaces to very harsh environment with metal obstacles and reflections. We show how to obtain reasonable positioning accuracy even in such industrial environment, and we also investigate how to dynamically allocate MAC resources in time to static and mobile users based on context awareness extracted from the multi legacy WiFi APs positioning system.

We finally introduce a new concept of virtual inertial sensors, estimating output of inertial sensors through radio measurements. More specifically, we propose methods for estimating the walking speed and the rotation of a user that uses only WiFi FTM. In particular, we show that for smartphones, AP side CSI is not suitable to estimate the rotation of mobile terminals. We evaluate and demonstrate the proposed approach to infer the rotation speed with experiments, using commodity 802.11ac APs for CSI and FTMs measurements. Together with walking speed estimation of a user, we can envision that virtual inertial sensors can be leveraged by location systems and sensing mechanisms to infer user behavior, to design better and more secure communication and to improve localization accuracy.

In the next subsection we present in detail the challenges of this works.

1.1. Challenges

This Thesis addresses practical challenges starting from the development of accurate indoor localization systems using commodity WiFi chipsets to the extraction of context-awareness used for resource management of the network and to infer the movements of the user. Let us analyze them below.

- *Working with commodity hardware:* The main challenges with commodity WiFi chipsets are the hardware imperfections, low-level functionalities and software information limitations. Specially for such commodity chipsets, the ToF measurements are affected by various sources of noise. For instance, its limited clock rate hinders the exact determination of the time of arrival of a signal.

- *Multipath reflections:* It is well known that signal propagation in complex indoor environments is subject to multipath effects in which multiple copies of the transmitted signal arrive at the receiver over different reflected paths. It is even possible that the direct component is entirely attenuated and the signal is received only over indirect paths.

- *Positioning with a single WiFi AP:* Achieving reasonable positioning accuracy with a single AP is challenging. The reason is that, for a given deployment scenario, the density of APs plays an important role, avoiding performance degradation as the distance from the AP increases.

- *Indoor positioning in industrial scenarios:* The industrial environment is notorious for being adverse to wireless communication, with traditional wireless resource mechanisms prone to errors.

1.2. Contributions

The main contributions of this thesis have been published in 6 publications and submitted in 1. More specifically, 1 has been published in *IEEE Transaction On Networking* (indexed in Journal Citation Reports (JCR)), and 1 publication in tier-1 conference (*ACM CoNEXT*) according to CORE2014[1] or ERA2010[2] datasets. Another publication has been published in tier-2 conference (BROADNETS), 1 has been published in a workshop (*WiNTECH*) and 1 more has been submitted. Also 2 demo papers have been presented in major conferences. In details,

Contribution 1. *Multi-APs ToF based indoor localization system.*
A solution for indoor mobile tracking has been investigated and evaluated in different scenarios [1]. A multi-APs system has been deployed in an industrial environment [3], after the investigation of a positioning algorithm that can achieve acceptable positioning accuracy in such environment.

- **Maurizio Rea**, Aymen Fakhreddine, Domenico Giustiniano, Vincent Lenders. Filtering Noisy 802.11 Time-of-Flight Ranging Measurements from Commoditized

[1] http://portal.core.edu.au/conf-ranks/
[2] http://www.conferenceranks.com/

WiFi Radios. Published in *IEEE Transaction on Networking, vol. 25, no. 4, pp. 2514-2527.*

- **Maurizio Rea**, Héctor Cordobés de la Calle, Domenico Giustiniano. Time-of-flight Wireless Indoor Navigation System for Industrial Environment. Published in *the 13th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & CHaracterization (ACM WiNTECH 2019)*, 16-20 October 2019, Los Cabos, Mexico.

Contribution 2. *Smartphone positioning with a single AP.*
Combining CSI and FTM, we have developed and evaluate an indoor localization system in different environments [4].

- **Maurizio Rea**, Traian Emanuel Abrudan, Domenico Giustiniano, Holger Claussen, Veli-Matti Kolmonen. Smartphone positioning with radio measurements from a single wifi access point. Published in *the 15th International Conference on Emerging Networking Experiments And Technologies (CoNEXT 2019)*, 9-12 December 2019, Orlando, Florida, U.S.

Contribution 3. *Location aware for wireless resources allocation in industrial environment.*
A ToF-based localization system has been deployed in an environment strongly affected by metallic objects [2]. Extracting location aware from the proposed positioning system, a new strategy for dynamic wireless resources allocation has been investigated.

- **Maurizio Rea**, Domenico Garlisi, Héctor Cordobés de la Calle, Domenico Giustiniano. Location-Aware MAC Scheduling in Industrial-Like Environment. Published in *Broadband Communications, Networks, and Systems (BROADNETS 2018)*, 19-20 September 2018, Faro, Portugal.

Contribution 4. *Virtual sensors*
Ideas to infer the movements of the mobile through radio measurements have been proposed as a new concept of virtual inertial sensors [5], estimating output of inertial sensors.

- **Maurizio Rea**, Domenico Giustiniano, Joerg Widmer. Virtual Inertial Sensors with Fine Time Measurements. [Submitted].

1.3. Outline of the thesis

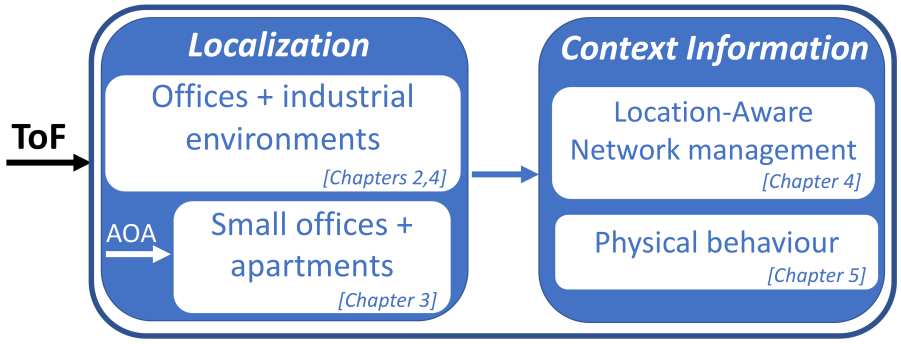


Figure 1.1: High-level illustration of the thesis.

The rest of the thesis is organized in different chapters detailing the contributions aforementioned in the previous subsection, as shown in Fig. 1.1.

A ToF-based solution for indoor localization and tracking of COTS WiFi devices is introduced in Chapter 2. This system, in a multi legacy 802.11 APs infrastructure, is evaluated in different office areas. Extracting AOA information from CSI measurements, the direction of the user can be estimated. With the latter added to the ToF information, a single-AP is able to estimate the position of the user. Doing so, a second solution for smartphone positioning using a single WiFi AP is introduced in Chapter 3. This system is proposed for small offices and apartments which typically have a single AP.

Chapters 4 and 5 take as input the extracted context beyond ranging and positioning information. More specifically, the main goal of Chapter 4 is the proposal of a smart location-aware network management, based on position estimates. The latter are obtained from the deployment of the multi-APs system in industrial environments. Chapter 5 then presents a new concept of virtual inertial sensors, based on radio measurements, explored to infer physical behaviour of a user.

Finally, Chapter 6 draws the most important conclusions of this research work.

Part II

Indoor Localization System

Positioning of mobile devices is at the core of a vast set of location-based services. Over the past years, we have seen an important increase of revenues in the mobile positioning market. For this reason, WiFi positioning systems have gained wide interest by the research as well as commercial providers. These WiFi-based positioning systems are envisioned to complement Global Navigation Satellite System (GNSS) in indoor and urban canyon ambient where the satellite signals largely fail but WiFi signals are widely available. Over the last years, many WiFi-based localization systems have been proposed. The majority of these systems rely on Received Signal Strength Indicator (RSSI) to estimate the position. It is known that using RSSI, signal attenuation in indoor environments becomes not only depending on the distance but also highly dependent on the material of the obstacles between radio devices like walls, doors, etc. An other technique for Wireless Fidelity (WiFi)-based positioning is localization with Time-of-Flight (ToF). In ToF localization systems, the distance between two devices is estimated using the time that the signal travels between two devices.

In part II of the thesis we investigate solutions where timing signals can be extracted from commodity WiFi chipsets and we propose two WiFi-based localization systems that run without the need for the intervention of the user. In Chapter 2 we present a multi-Access Point (AP)s ToF-based system. We develop an adaptive matched filter which manages to remove noise introduced by the devices and the multipath reflections in indoor environments. Given a set of filtered ranging measurements, we apply a multi-lateration algorithm in order to estimate the position of the target. We then evaluate the accuracy of the system for distance and localization in different scenarios. In chapter 3 we propose a positioning with radio measurements with a single AP. We exploit Fine Time Measurements (FTM), based on ToF, and Angle-Of-Arrival (AOA) extracted from Channel State Information (CSI) measurements. Exploiting Physical Layer (PHY) information we derive a method for filtering ranging measurements obtained with the FTM protocol. We then evaluate the system showing its accuracy in different scenarios.

“”

Brendan O’Brien (Aria Systems)

2 Multi Access Points

Localization using the Time-of-Flight (ToF) of Radio Frequency (RF) signals is one most popular techniques to track moving objects in outdoor environments. The most prominent usage of ToF is the Global Positioning System (GPS) which exploits signal propagation times from different satellites to localize mobile devices on earth. Also, radar systems commonly rely on the propagation time of RF signals to localize aircrafts in the airspace. While the ToF technique has been widely adopted in outdoor environments, its application for indoor localization in Wireless Fidelity (WiFi) environments has been relatively modest so far. When it comes to WiFi based localization, the research community has instead focused more intensively on different approaches such as the signal strength [13–17], the angle of arrival [18–20, 23] or combining WiFi signals with inertial sensors as found in smartphones [21, 24]. The problem with the latter approaches is that they either require extensive manual on-site calibration, the need for intervention of the user, or specialized hardware. These factors limit the deployment of these approaches at larger scale.

The challenge with ToF is that very precise signal propagation time estimates are required. At the speed of light, a measurement error of $1\,\mu s$ already results in a distance estimation error of 300 meters which is intolerable for most indoor applications. In order to achieve meter-level localization accuracy, a precision in the order of a few nanoseconds is therefore needed. However, at this level the ToF is affected by various sources of noise [29–34]. For example, the relatively small bandwidth of WiFi signals and the limited clock rate of commercial off-the-shelf (COTS) WiFi radio receivers hinder the exact determination of the time of arrival of a signal. In echo techniques, the target may further add significant jitter before acknowledging the reception of a data. Specially indoors, the signal may additionally be obstructed and reflected over multiple paths which adds a positive bias to the estimated range.

In order to mitigate the impact of noise in ToF measurements, several techniques have been proposed in the literature. To combat the device-related Gaussian noise, unbiased estimators that rely on the mean or median of the collected ToF samples provide a good

estimate of the true ToF [34]. To combat the multipath noise, biased estimators that aim at identifying the direct path such as the MUSIC algorithm [35,36] can provide good results as well. Other approaches such as the expectation-maximization algorithm have been successfully applied to signal strength-based localization [16], and we could envision their application to combat the multipath noise of ToF measurements. However, we show in this work that, in WiFi ToF, the noise follows a Gaussian mixture model with each Gaussian component being in the same order of magnitude as the multipath bias, and we demonstrate that these estimators fail to provide an accurate estimate of the true ToF as we need for ranging.

In this chapter, we therefore present a new filter that is specially designed for estimating the range in WiFi indoor environments based on noisy ToF measurements. Our filter relies on a combination of statistical learning techniques to train an environmental-specific linear regression model for the multipath bias and robust statistics for range estimation. Our approach does not require any specialized hardware and relies exclusively on the ToF information that can be extracted from COTS WiFi radios. In addition, our environmental-specific linear regression model can be learned in-situ and thus does not require any manual calibration.

We have used our filter to estimate the location of WiFi radio devices in four different experimental testbeds. Our results show that our filtering technique is able to significantly reduce the median ranging error over classical estimators. Our main contributions in this chapter are the following:

- We present a firmware-based ranging architecture for round-trip time measurements running in the Medium Access Control (MAC) processor of WiFi chipsets, and we quantify and compare the amount of noise that comes from the WiFi devices and the noise from the WiFi radio signal propagation over the wireless channel.

- We propose a filter based on statistical learning and robust statistics to estimate the distance range from a series of noisy ToF measurements. Our filter does not require any manual calibration and manages to estimate the distance to a remote WiFi device with median error between 1.7 and 2.4 meters.

- Finally, we evaluate our system for indoor mobile device tracking. We demonstrate across four different experimental testbeds that mobile devices associated to the WiFi infrastructure can be tracked continuously using our filter with a median accuracy between 2.0 and 3.4 meters.

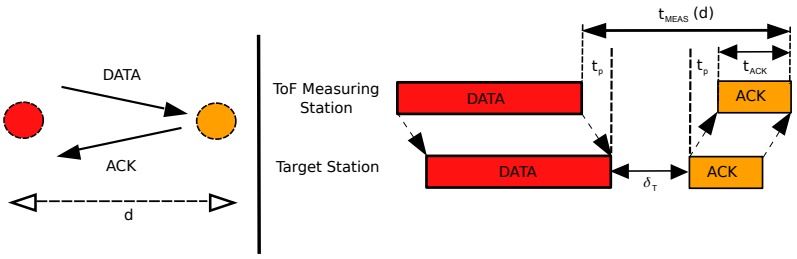


Figure 2.1: Principles of WiFi ToF echo technique. In the example, offset δ_T is originated at the target device, and it depends on the hardware delay to schedule the ACK at the target device. The time offset δ_L (not shown in the figure) arises at the measuring device.

2.1. ToF Echo Technique

This section describes the principles of ToF ranging and the noise effects that arise when applying this technique to off-the-shelf WiFi devices.

While traditional ToF-based echo techniques as employed in radar systems rely on encoded RF signals and their reflections, WiFi echo techniques use regular frames of communication [29–34]. In WiFi communication, every DATA frame is acknowledged by the receiver with an Acknowledgment (ACK) frame. Since the interframe time between the DATA and ACK frames is fixed by the 802.11 standard (the Short InterFrame Space (SIFS) time), the delay between DATA and ACK frames can be used to infer the distance between two nodes.

In reality, sources of time offset exist at the local and target stations and that affects the accuracy of the ranging measurement. Let us refer to Fig. 2.1. If d is the true distance between a local station and a target, the local station measures the time $t_{MEAS}(d)$ between a sent DATA frame and a received ACK frame:

$$t_{MEAS}(d) = 2 \cdot t_P(d) + t_{ACK} + \delta, \tag{2.1}$$

where $t_P(d)$ is the signal propagation time between the transmitter of the DATA frame and the target (channel reciprocity is assumed), t_{ACK} is the time needed to transmit the ACK, and δ is the measurement offset.

The offset δ is the sum of the offset due to the local station δ_L and the offset due to the target station δ_T:

$$\delta = \delta_L + \delta_T. \tag{2.2}$$

The local offset δ_L arises at the measuring device due to the local imprecision in the timing information extraction, and the target offset δ_T is equal to the SIFS time plus any device-dependent deviation from this number. The presence of multipath effect causes a positive offset to single measurements and increases δ_L and/or δ_T.

From eq. (2.1), we can then define the ToF measurement as

$$t^{ToF}(d) = \frac{t_{MEAS}(d) - t_{ACK} - \delta}{2}. \tag{2.3}$$

It follows that the distance from the measuring station to the target device can be computed for the generic sample[1] m as

$$d_m = c \cdot t_m^{ToF}(d), \tag{2.4}$$

where c is the speed of signal propagation which is close to the speed of light in air.

In the more general case, let $\mathcal{L}$ denote the set of links to a mobile station and $\mathcal{M}$ the set of samples. We can then express the generic sample as:

$$d_{i,m} \qquad i \in \mathcal{L}, m \in \mathcal{M}.$$

Once M samples have been collected for a link $i \in \mathcal{L}$, the distance $\widehat{d}_i$ between the local station and the mobile device is estimated as follows:

$$\widehat{d}_i = f(d_{i,1}, d_{i,2}, \ldots, d_{i,M}), \quad i \in \mathcal{L} \tag{2.5}$$

where f is an estimator of the distance that aims at filtering out the noise of individual measurements. The major contribution of this work is *the development of a robust estimator f that estimates the distance to a target device using WiFi ToF, in the presence of rich multipath as found in common indoor environments, and severe noise as found in commodity WiFi chipsets.*

2.2. ToF on COTS WiFi Radios

The design of a robust estimator f requires first to characterize the noise sources of WiFi ToF. For this reason, we implement a ToF ranging system using commodity hardware. To alleviate any unnecessary source of noise or instability from the operating system, ToF measurements have to be performed as close as possible to the radio hardware. Rather than in the driver [34] or upper layers [31], the best method is therefore implementing the code for ToF measurements in the firmware of the WiFi radio chipset. To measure $t_{MEAS}(d)$ in the firmware, we have customized the open-source 802.11 Open FirmWare for WiFi networks (OpenFWWF)[2]. This firmware is written in assembler and runs on off-the-shelf 802.11 Broadcom chipsets, such as the ones widely used in Linksys Access Point (AP)s. Our customized firmware reports $t_{MEAS}(d)$ for each successful DATA-ACK frame pair. The timing is regulated by the general purpose timer, running

[1] A sample is collected per 802.11 DATA/ACK handshake.

[2] http://www.ing.unibs.it/openfwwf/

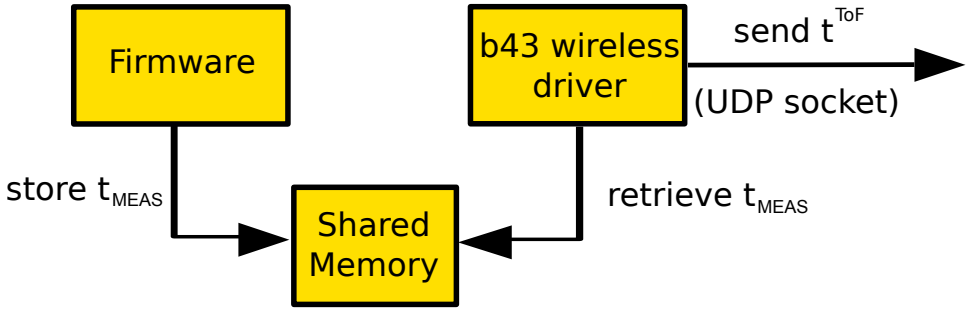


Figure 2.2: ToF measurement architecture. The figure shows the interaction between firmware, driver and user space. Measured ToF data from the firmware is transferred to the driver through SHM. Buffered ToF measurements are then sent to user space.

on the wireless card's internal clock at a rate of 88 MHz. The timer starts to count clock cycles just after the 802.11 processor sets up a register to indicate that a frame has been sent. Once the ACK frame has been received (or the ACK timeout has elapsed), another register gets updated and the timer gets stopped. Every time a measurement is made, the firmware writes $t_{MEAS}(d)$ into a defined address of the SHared Memory (SHM). The architecture is shown in Fig. 2.2. Since the driver has also access to the shared memory block, it can retrieve the measurement every time an ACK is received[3]. In the driver, we gather additional data about the incoming ACK such as the data rate, the AP MAC address, etc, and store them all in a buffer. Once this buffer is full or a timeout has elapsed, the data is transferred to the user space with the help of User Datagram Protocol (UDP) sockets.

2.2.1. System Setup

We have built a prototype ToF system that consists of COTS APs operating as ToF measuring stations. In our deployments, the APs are static stations. The APs use Soekris net5501 embedded machine with a 500 MHz AMD Geode LX single chip processor. The APs are part of the same 802.11 Extended basic Service Set (ESS) and broadcast the same service set ID (Service Set IDentifier (SSID)). The APs are equipped with Broadcom AirForce54G 4318 mini PCI type III cards and an omnidirectional antenna. The Broadcom chipset is operated with our customized firmware and the b43 driver presented above. The APs are connected over Ethernet to the location computing unit, that is responsible to process the raw data and compute the position.

As target station we use unmodified Dell Inspiron 5150 laptops equipped with Broadcom AirForce54G 4318 mini PCI type III cards and the integrated antenna of the laptop. In our deployments, the target station may be static or mobile according to the type of experiment (see Section 2.2.2 on deployment scenarios for the details). The

[3]We operate in promiscuous mode which allows us to know in the driver when an ACK has been received and thus a new data is available in the shared memory block.

target station is associated to the ESS network connecting to one of the APs. At any point in time, the target device is associated to only one AP of the ESS, as in typical 802.11 wireless networks.

In order to perform ranging measurements, the APs use regular DATA frames that are acknowledged by ACK frames from the targets. The ranging measurements are performed in a round-robin fashion among the APs. For every successful DATA-ACK frame pair, $t_{MEAS}(d)$ is measured by the AP from the end of the transmission to the end of the reception of the corresponding ACK. Since the target is associated to a single AP, the other APs send their DATA frames with the source MAC address of the AP to which the target is associated. In order to configure the source MAC address of the APs, we rely on raw sockets and PCAP library[4]. The latter allows us to generate a custom MAC header with any MAC address as the source.

2.2.2. Deployment Scenarios

We consider two types of scenarios for the deployment. The first scenario considers controlled experiments, where we avoid any environmental effects by connecting two stations using coaxial cables. The cables are of length 13.5 m and are based on the standard RG-58. In the second scenario, we perform experiments with transmissions over the air in four indoor testbeds, namely Testbed I, II, III and IV. The maps of the four indoor testbeds are shown in Fig. 2.3. Testbed IV uses the same environment as in Testbed I, but APs are partially placed at different locations.

We deploy 9 APs in Testbed I, 9 APs in Testbed II, 10 APs in Testbed III, and 9 APs in Testbed IV. Testbed I, II and IV are office environments.

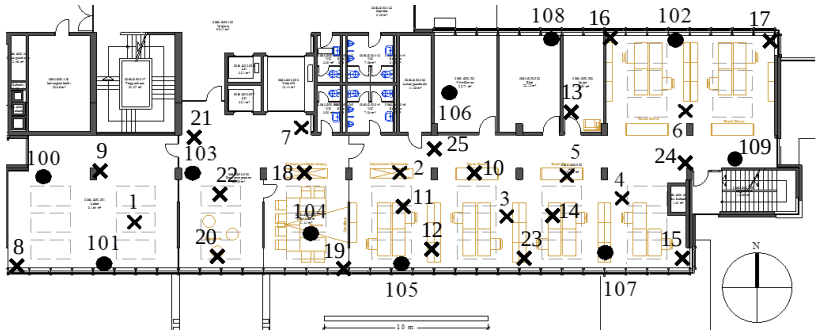
- The environment of Testbed I is shown in Fig. 2.3(a). It covers a surface of almost 1000 m^2. We use 25 randomly selected locations (marked as a cross) to test our algorithms. We conducted tests over two different days, with some positions repeated again with different locations of some furniture, and collect a total of 207 wireless links.

- Testbed II is depicted in Fig. 2.3(b). It features 180 links and it covers a smaller space of around 200 m^2. The target station is placed in 20 different positions.

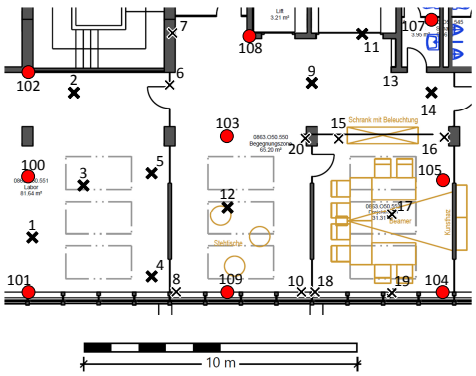
- Testbed III is shown in Fig. 2.3(c). It has been deployed at the facilities of the IEEE/ACM IPSN 2014 - Microsoft Indoor Localization Competition [37]. The testbed has 200 links and it covers 320 m^2. The target device is placed at 20 positions in two rooms and a hallway.

- Testbed IV is deployed to study the performance of mobile device tracking. In this testbed, we measure the position when a mobile device is moving from position

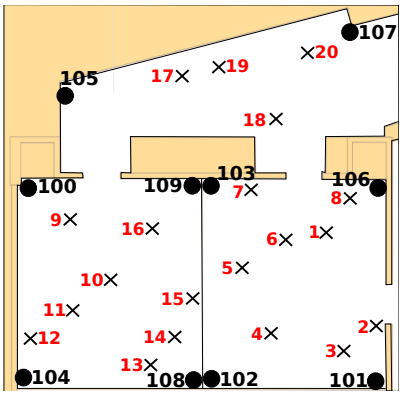
[4] http://www.tcpdump.org/



(a) Testbed I.



(b) Testbed II.



(c) Testbed III.

(d) Testbed IV (mobile tracking).

Figure 2.3: Testbeds to assess the ranging and positioning capabilities of our system in static conditions. Circles correspond to AP locations and crosses to target locations.

1 to 19 as marked in Fig. 2.3(d). For these mobility tests, a user moves at a speed of approximately 0.4 m/s. At the time that the user passes at one of the marked positions, we estimate the position with our filter and record the value. We then repeat the tests in the following setups: when the user stops at the marked locations for 2 and 5 seconds, respectively.

In all the testbeds, a mixture of Line-Of-Sight (LOS) and Non-Line-Of-Sight (NLOS) wireless links are present. During all experiments, people are moving within the testbed areas. The testbeds also contain several obstructions, including concrete walls, tables and glasses. All experiments are conducted with other active WiFi networks in the neighborhood. We operate the testbeds on a fixed frequency channel of the 2.4 GHz ISM band. The Physical Layer (PHY) automatic selection rate is active, such that the measurements include DATA sent at different PHY rates.

2.3. Sources of Noise

While the firmware-based approach introduced in Section 2.2 can allow us to measure the ToF with the best precision using commoditized WiFi radios, the ToF measurements are still affected by large noise coming from severe sources of noise which we discuss in the following in more details.

Target ACK delay. The 802.11 standard specifies the SIFS time between the reception of a DATA and the transmission of an ACK at the receiver as a fixed interval. In 802.11b, for example, this time is specified as $10\,\mu s$ [38]. However a relatively high tolerance of 1 μs is tolerated which can result in significant noise and distance estimation errors up to 300 meters if the target would fully exploit this specified tolerance level. While most chipsets may not fully exploit this tolerance, the dispersion is still quite significant. To illustrate this, Fig. 2.4 on the left represents the resulting dispersion of a typical Broadcom WiFi chipset. The shown histogram was obtained by collecting sequences of samples $\{d_{i,m}\}$ according to Eq. (2.4) for $10,000$ packets. To avoid any dispersion from environmental effects, the measurements were performed over a coaxial cable of 13.5 meters. As we can see, there are sources of noise in the measurement setup that can lead to distance estimations that range from 0 to 25 meters, even under ideal signal propagation conditions such as cables.

Multipath reflections. It is well known that signal propagation in complex indoor environments is subject to multipath effects in which multiple copies of the transmitted signal arrive at the receiver over different reflected paths. It is even possible that the direct component is entirely attenuated and the signal is received only over indirect paths. Since signals that travel over indirect paths will take longer time to arrive at the receiver, they introduce a non-Gaussian error in the distance estimation when considering the ToF. This situation is shown in Fig. 2.4 in the middle and on the right where the same

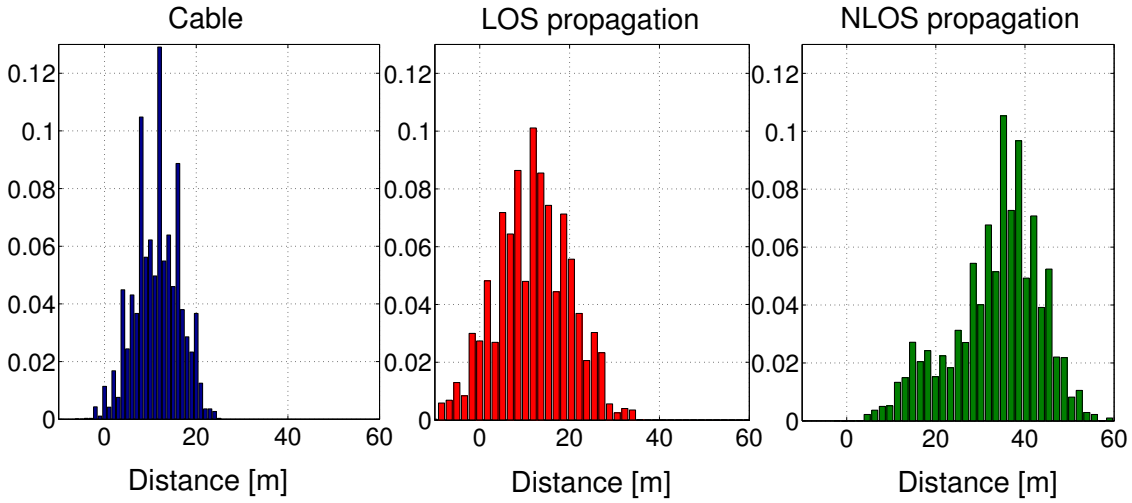


Figure 2.4: Noise introduced by ToF ranging. Tests in a controlled environment (Cable) show that there are sources of large dispersion in the estimation. Tests over the air (LOS and NLOS propagation) show that the distribution greatly depends on the channel conditions. All the tests are in addition subjected to quantization noise and other spurious noise sources.

experiment as on the left was repeated but for a LOS and NLOS signal propagation link over omnidirectional antennas. In the LOS experiment, there is a visible connection between the measuring station and the target, while in the NLOS experiment, they are obstructed. The dispersion spans a range of 40 and 60 meters for the LOS and NLOS links, respectively, and the large noise in the measurement can also result in negative $d_{i,m}$ samples (see Fig. 2.4 in the middle), especially when the true distance d_i is relatively small. In addition, the NLOS link shows a non-Gaussian distribution. Multipath effects must therefore be taken into consideration in order not to overestimate the distance when dealing with reflected signal propagation paths. Finally, multipath may also happen in LOS links, and thus a method robust to the propagation conditions must be designed.

Quantization and measurement uncertainty. Off-the-shelf WiFi chipsets have not been designed to provide accurate ToF measurements. A main source of noise comes from the coarse clock resolution of the radios. For example, the Broadcom chipset operates with a reference clock of 88 MHz, corresponding to a maximal distance resolution of 1.7 meters. In addition to this quantization noise, off-the-shelf chipsets introduce all sorts of considerable additional noise. As we could see in the histograms of Fig. 2.4, the shape of the distribution is far from being smooth despite using $10,000$ samples to create the histograms, suggesting that the radios must have some bias when measuring the time. Other sources of noise must therefore also be factored in order to estimate the distance. On the other hand, the effects of the clock drift are negligible. See details in [34].

Congestion and interference. Wireless congestion and interference have no impact on the ACK delay (that is, δ_T) since the 802.11 channel is reserved during the SIFS period

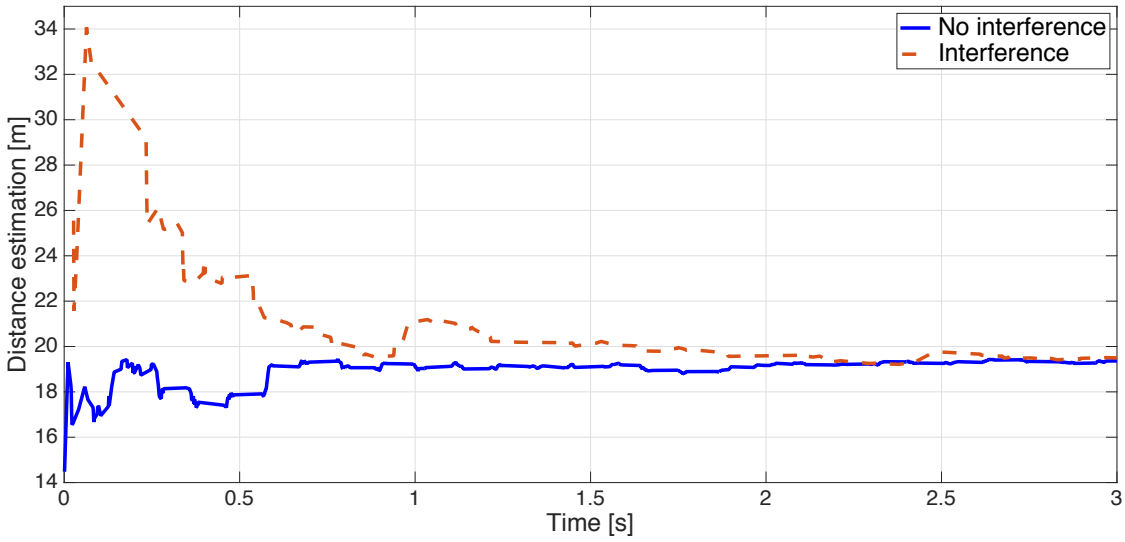


Figure 2.5: ToF distance estimation in a LOS link of 19 meters in absence and presence of interfering traffic. In the experiment, the ToF measuring station sends DATA at PHY rate of 1 Mb/s.

to the receiving node (i.e. the target station) as dictated by the 802.11 Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol [38]. However, collisions occur on the wireless medium, causing data retransmissions. Only acknowledged data frames are considered as valid ToF samples, while unacknowledged frames do not generate any ToF measurement. Since the wireless resources are shared, the presence of congestion can increase the time required to obtain a sufficient number of samples for ranging. We study the problem of wireless congestion in an open space room (to reduce as much as possible the multipath, and focus on the impact of interference), with one link of 19 meters in LOS. In this setup, the AP transmits DATA at PHY rate of 1 Mb/s. The AP computes the distance averaging over the collected ToF samples. We then repeat the ranging measurement for the same link, adding 802.11 traffic from another wireless station that saturates the channel with interfering traffic (UDP traffic of 4 Mb/s) sent at PHY rate of 1 Mb/s. The results are presented in Fig. 2.5. The Figure shows the average of the distance estimation, both in absence and presence of interference. While the long-term ranging accuracy is the same in absence of interference and in presence of interference saturating the channel, the latter causes a longer time to converge to the true distance d_i. We can conclude that, in order to efficiently handle interference, it is important to design a system that requires only a few samples M for the ranging estimation.

In Section 2.4 we provide an analysis about the target and measuring noise. We then describe how to deal with the Non-Gaussian noise resulting from the multipath propagation in Section 2.5 to finally describe in details the robust shortest-path estimator we propose as a solution to mitigate the effects of the main sources of noise in Section 2.6.

2.4. Device-related Noise Analysis

In this section, we analyze in details the noise originated at the measuring station and at the target station. The goal of this analysis is to quantify the device-related noise and determine whether the non-Gaussian noise we observe in ToF measurements is only due to environmental effects such as multipath or the local and target devices also add non-Gaussian noise to the measured values.

In order to address this question, we use a low-noise oscilloscope to measure the offset distribution of $\{\delta_T\}$. This high-end wideband oscilloscope is an Infiniium 90000A model with a fast sampling rate of 10 GS/s [39]. The internal noise of the oscilloscope can be regarded as negligible compared to the noise introduced by the WiFi radios and therefore provides us a mean to analyze the target offset δ_T in isolation.

On the left of Fig. 2.6, we show the histogram of the target offset δ_T as measured with the oscilloscope using approximately 300 samples, and the resulting Gaussian fitting function. We run the Lilliefors test and find that the hypothesis that the distribution of δ_T is Gaussian *cannot be rejected*, with a high p-value equal to 0.43 while setting the significant level to 0.05. Therefore it is safe to assume that δ_T can be approximated with a Gaussian distribution $\mathcal{N}$ with standard deviation equal to σ_{δ_T}.

We then statistically compare the distribution of $\{\delta_T\}$ measured with the oscilloscope with the distribution $\{t_{MEAS}(d)\}$ measured at the local station with our firmware-based approach presented in Section 2.2. For the comparison, we collect samples of $t_{MEAS}(d)$ in LOS links with limited multipath at $d = \{1, 15, 60\}$ m and draw the Quantile-Quantile (QQ) plots, a graphical method for comparing two probability distributions. If the distributions are linearly related, the points in the QQ plot will approximately follow a line.

The resulting QQ plots are shown on the right of Fig. 2.6. We observe a linear pattern, which indicates that $\{\delta_T\}$ *and* $\{t_{MEAS}(d)\}$ *have very similar noise distributions in presence of limited multipath.* We also observe that the dispersion of the two distributions is very similar, with a standard deviation of $4.0 - 4.1$ WiFi clock cycles in both cases. As a result of these investigations, we conclude that:

(i) The noise of $t_{MEAS}(d)$ in LOS links with limited multipath is largely dominated by the Gaussian noise of the target offset δ_T.

(ii) The dispersion of the local offset δ_L in LOS links has a negligible impact on the distribution of $t_{MEAS}(d)$, and our approach to implement the ToF measurement in the firmware does not add a significant dispersion.

(iii) The non-Gaussian ToF noise that we observe in many indoor links of our testbed is not related to the local or target noise of the WiFi devices but rather to environmental effects such as multipath.

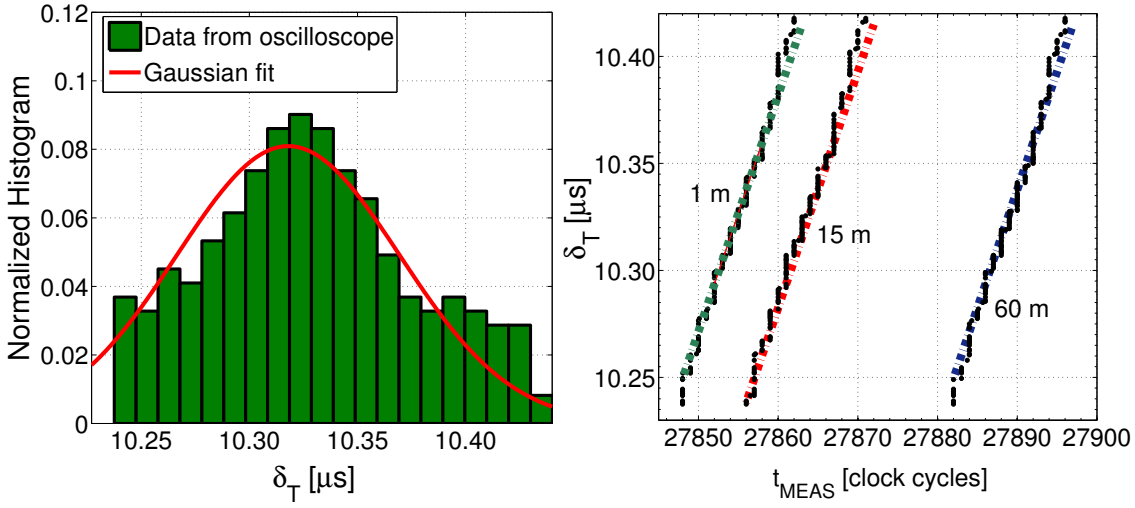


Figure 2.6: On the left: distribution of $\{\delta_T\}$, as measured with our oscilloscope, and its Gaussian fitting function. On the right: QQ plots of δ_T versus $t_{MEAS}(d)$.

2.5. Dealing with WiFi Multipath Noise

Existing multipath-resistant estimators assume that the non-Gaussian noise from the reflected signal path components are the dominant sources of noise. However, as we have shown in the previous Section, the device-related Gaussian noise is extensive in WiFi radios. In order to deal with these combined sources of noise, the focus of this section is to analyze different robust estimators f to estimate the true distance d for multipath-rich indoor environments.

2.5.1. Noise Model

While the device-related noise can be approximated as an environmental-independent Gaussian distribution, this is not the case for the noise that arises from the effect of multipath reflections. The amount of multipath is very much dependent on the environment and the exact position of the devices. Under multipath propagation between two WiFi nodes, the distribution of ToF measurements will be non-Gaussian, as a result of those data frames where the 802.11 transceiver synchronizes to a delayed copy of the signal.

Estimating the true distance using this noisy data is not trivial with WiFi radios. In fact, we cannot directly measure the individual multipath components at the signal-level on off-the-shelf WiFi radios since the WiFi chipset only synchronizes to the strongest path, resulting in a positive or null bias $b_{i,m} \geq 0$.

In order to design a reliable estimator f of the distance, we model the ToF noise as follows. Let us consider a link $i \in \mathcal{L}$. The true distance d_i of this link is affected by an additive Gaussian noise $\mathcal{N}$, generated by the target station, with the standard deviation

equal to σ_{δ_T} (cf. Section 2.4). d_i is further affected by a distance bias $b_{i,m} \geq 0$ caused by the absence ($b_{i,m} = 0$) or presence ($b_{i,m} > 0$) of multipath. Therefore, for each sample, we can model the distance $d_{i,m}$ as follows:

$$d_{i,m} = d_i + b_{i,m} + \mathcal{N}(0, \sigma_{\delta_T}), \tag{2.6}$$

where d_i is the true distance.

Grouping together the samples with the same bias after clock quantization (that limits the distance resolution per sample to 1.7 m, cf. Section 2.2), we have a finite Gaussian Mixture Model (GMM) with a small number of modes. One of these modes corresponds to the samples synchronized to the direct path (the samples with $b_{i,m} = 0$), while the others correspond to the samples synchronized to any of the reflected paths. In general, no component is dominant and we can conclude that the Gaussian mixture is non-Gaussian distributed.

2.5.2. Reliability of the Estimation of the Gaussian Components

A fundamental difference with respect to classical GMM applications is that the multipath noise superimposes to the large noise σ_{δ_T} added by the target device. The deviation due to σ_{δ_T} can be much stronger than the bias error caused by synchronizing to a delayed copy in a multipath propagation[5]. Current techniques for learning the modes of the GMM model need instead that the modes are either sufficiently apart or consider that only a small number of samples is received as reflected paths representing a few outliers [40].

In order to verify how well current learning mechanisms for a GMM model could be applied to WiFi ToF, we generate in simulations $1,500$ samples using the GMM model and suppose that there exist only two modes, one mode representing the direct path and one mode representing the reflected path. We then apply the Expectation-Maximization (EM) algorithm [41] (the details of the algorithm are presented in Section 2.7.1) to cluster the samples. In the scenario, we consider that the direct path has a distance of 10 m and that the reflected path has a traveled distance of 26 m. The mixture weights of the Gaussian components are equal to 0.12 and 0.88, respectively. This indicates that the 802.11 WiFi chipset synchronizes more often to the reflected path, which is in general the stronger signal component. We first make the (ideal) assumption that δ_T is much smaller than in real measurements. We call it "Simulation of Ideal Case". In the second scenario, we consider that δ_T is equal to the one empirically measured in Section 2.4. We call it

[5]For instance, in the example in Section 2.4, we have a standard deviation of 4 clock cycles of the noise of the target device in LOS conditions. This results in a 99-percentile dispersion of the error of approximately $2.58 * 4$(deviation in clocks) $* 2 * 1.7$(conversion of clock to ToF distance) $= 35.1$ m, higher than a typical error caused by synchronizing to a delayed copy of the 802.11 signal in a multipath propagation in indoor scenario.

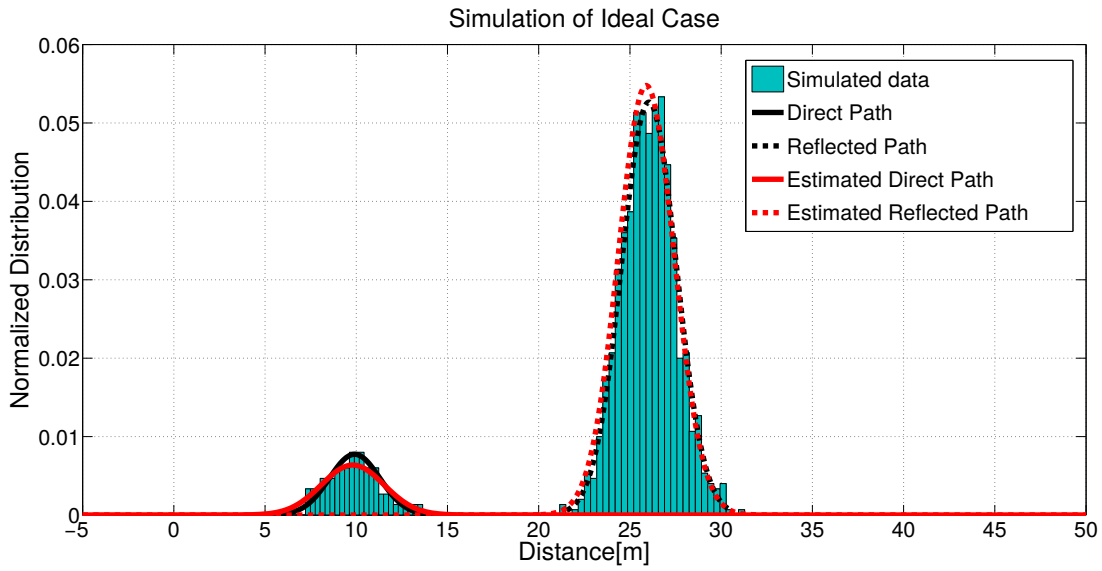


Figure 2.7: Generation of a direct path and a reflected path in simulation using the GMM model, with small Gaussian noise of the target station. As expected, the EM algorithm can reliably estimate the two modes.

"Simulation of Real Case".

We plot the results of "Simulation of Ideal Case" in Fig. 2.7. We observe that the EM algorithm estimates the parameters of the components of the Gaussian mixture almost perfectly: the estimated mean of the direct path's distribution is equal to 9.8 m and the mean of the reflected path's distribution is 25.9 m. It follows that we are able to estimate the distance of the direct path with an error of only 0.2 m. We then plot the results of "Simulation of Real Case" in Fig. 2.8. Clearly visible from the Figure, we do not have anymore a clear separation of the two Gaussian components. The parameters are here overestimated: the estimated mean of the direct path's distribution is 19.3 m and the mean of the reflected path's distribution is 29.1 m. Concluding, we have a high error in the estimation of the direct (and shortest) path, used for ranging, of 9.3 m.

2.6. Robust Shortest Path Estimator

The results of the previous section have shown that classical algorithms fail to estimate the distance in WiFi echo techniques. In this section, we explore a different methodology for the estimation of the distance. First of all, in WiFi ToF, we are only interested in the direct path's distance or, in general, the shortest path's distance. Indeed, we do not need an algorithm (as the EM algorithm) that estimates the distance of all the paths, including the longer ones. In GMM links with only a direct path ($\forall m \in \mathcal{M}, b_{i,m} = 0$), the median or the mean would be a reliable estimator of the direct path's distance. However, as a result

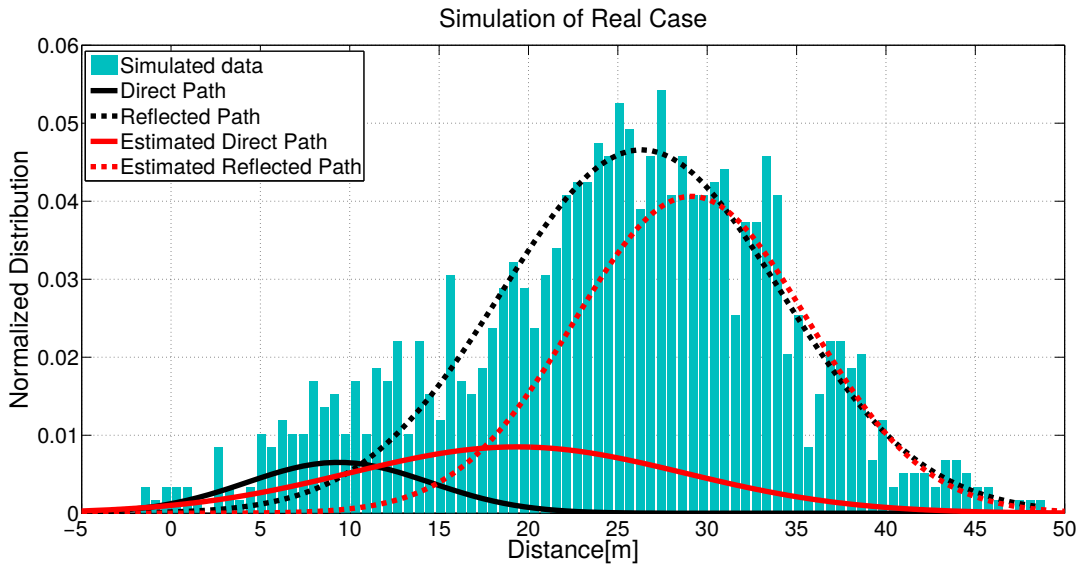


Figure 2.8: Generation of a direct path and a reflected path in simulation using the GMM model, with Gaussian noise of the target station as found in our experiments. The EM algorithm fails to reliably estimate the two modes.

of the GMM model in WiFi ToF, the sequence is in general mixed with samples where the local station synchronizes to a delayed copy of the WiFi signal ($\exists\ m \in \mathcal{M} : b_{i,m} > 0$). In the latter case, using the median of the sequence, e.g. the 50% percentile of the distribution[6] as estimator would result in an over-estimation of the distance. *We then conjecture that, by taking a percentile below 50%, we can counterbalance the biased values* $b_{i,m} > 0$ *in the estimation process.* In the example shown in Fig. 2.9, a percentile equal to 9 would be effective to counterbalance the biased value and report the true distance.

Formally, let us consider a link $i \in \mathcal{L}$ and collect a sequence of M measurements $\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}$. We define the optimal percentile ${p_i}^{opt}$ as the percentile that provides, for each link i, the minimum absolute distance estimation error with respect to the true distance d_i:

$$ {p_i}^{opt} = \underset{0 \le p \le 0.5}{\operatorname{argmin}} \ |d_i - \widehat{d_i(p)}|, \tag{2.7} $$

where $\widehat{d_i(p)}$ is the estimated distance using the p-percentile of the distribution $\{d_{i,m}\}$. The goal is to design a statistical estimator f that estimates the optimal percentile ${p_i}^{opt}$ based on some observables and gives as output the estimated distance $\widehat{d_i({p_i}^{opt})}$.

We train various options for the observables in the estimator f in an extensive evaluation in one of our testbeds (Testbed I, Fig. 2.3(a)) with all links. As candidate observables, we consider the first three moments (median, standard deviation, and

[6]The p-th percentile of a distribution is a number such that p percent (p%) of the values in the distribution are equal to or less than that number.

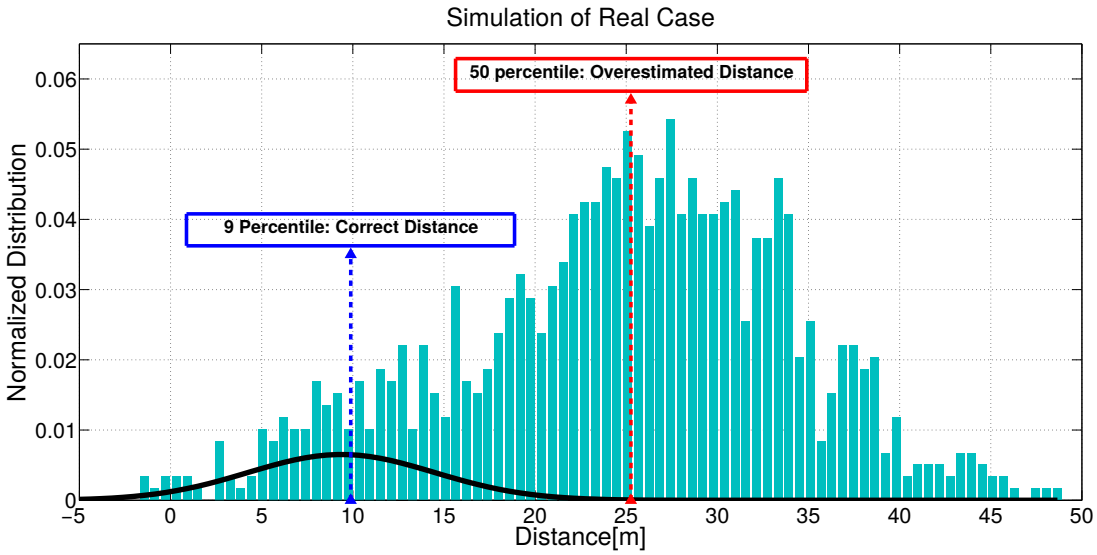


Figure 2.9: Same simulation data set as in Fig. 2.8: We conjecture that, by taking a percentile below 50%, we can counterbalance the biased values $b_{i,m} > 0$ in the estimation process and estimate the mean of the direct path's Gaussian distribution.

skewness) of the ToF as well as of the Received Signal Strength Indicator (RSSI). All three moments could in principle be indicators of multipath. For example, when the median of the ToF is low (or the median of the RSSI is high), the two ranging devices are close to each other and hence likely to have a short line-of-sight connection between each other with little multipath delay. An increased standard deviation of the ToF or the RSSI may indicate that signals are received over several paths with different propagation delays and/or attenuations. The skewness of the ToF and RSSI distributions will also be intuitively larger over links with multiple propagation paths. For each of these six observables, we evaluate two variants, leading to a total of twelve candidate estimators. In the first variant, we determine the moments directly on the raw samples $\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}$. In the second variant, we attempt to pre-filter obvious outliers that arise from the device-related measurement noise (cf. Section 2.3) prior to determining the moments. These outliers are filtered out applying the Thompson Tau technique [42], a statistical method for deciding whether to keep or discard samples based on the expected value and the expected deviation of the sequence of samples.

We evaluate the precision of these estimators by determining their correlation to the a priori known optimal percentile ${p_i}^{opt}$. For this analysis, we compute ${p_i}^{opt}$ for a set of links with known distance, i.e. placing the mobile device at known positions. To quantify the correlation between the different moments and ${p_i}^{opt}$, we rely on the Pearson correlation coefficient [43]. The Pearson correlation coefficient ρ is an indicator of the linear correlation of the variables, where absolute values close to zero indicate a low

Table 2.1: The absolute value of Pearson correlation coefficient ρ between different moments of the RSSI and ToF versus the optimal percentile p_{opt} (0=no correlation, 1=maximal correlation). It is worth noting that the ranging is always performed using ToF measurements. For example, the use of moments of RSSI is limited to infer the optimal percentile, it should not be confused with RSSI-based ranging.

	unfiltered	**pre-filtered**
median of RSSI	0.62	0.63
standard deviation of RSSI	0.04	0.05
skewness of RSSI	0.23	0.24
median of ToF	0.76	0.76
standard deviation of ToF	0.19	0.21
skewness of ToF	0.20	0.51

correlation and absolute values close to one represent a high linear dependence of two variables. A value close to one thus indicates that a moment is a good estimator to predict the percentile that will filter out the multipath noise effectively.

We consider the entire set of links in one of our testbeds (Testbed I). Table 2.1 shows the resulting correlation coefficient ρ for all twelve variants. *The best correlation is provided by the median of the ToF*, followed by the median of the RSSI and the skewness of the ToF. All other moments have a correlation coefficient below 0.5 which indicates a low correlation. We note that not all moments profit from pre-filtering to remove the outliers. For example, the median of the RSSI and ToF perform worse when outliers are pre-filtered. On the other hand, the skewness of the ToF increases from 0.20 to 0.51 and is therefore considerably better when pre-filtering the outliers.

One may ask why the skewness of the ToF has a worse correlation than the median ToF. Intuitively, the skewness of the distribution should be a good indicator of the multipath, given that links with strong reflected (delayed) components are left-skewed, with p_{opt} smaller than for right-skewed link. Our results suggest that the combined device-related noise of the receiver and the measuring station have a strong negative effect on the correlation on the skewness. This is reflected in the pre-filtered version of the skewness which has a considerably better correlation than the unfiltered version. In contrast, the median of the ToF is much more robust to any device-related noise and therefore outperforms the skewness. The reason for the median ToF working best can be associated to the tendency of having longer reflected paths (and thus longer delays) for links with longer distances and vice versa. In other words, when the ToF is small, the devices are close to each other and the multipath noise will likely not affect much the ToF measurements. On the other hand, when the median ToF is large, the devices are further apart and the links may be affected more severely by the noise from reflected paths.

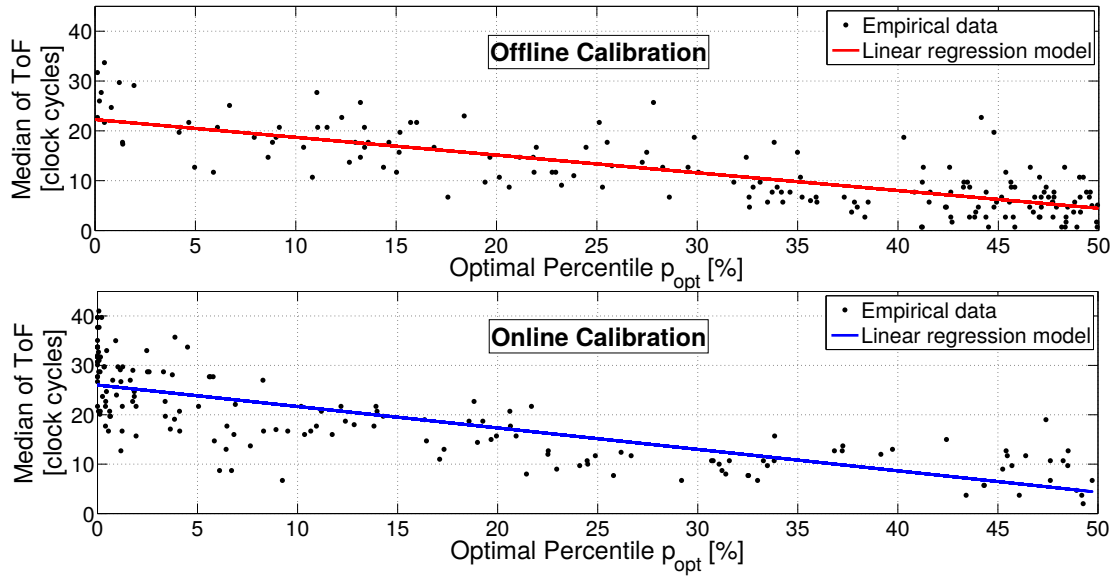


Figure 2.10: Median ToF versus optimal percentile p_{opt} for all links of Testbed I - Linear regression using offline and online calibration

2.6.1. Proposed Filter Design

Provided the good correlation ρ between the median ToF and the optimal percentile ${p_i}^{opt}$, we design a linear model for the estimator f that relies on this correlation to estimate the percentile from ToF measurements. The model is specific to the environment and we therefore train a model for each testbed. The training works as follows. For each environment, we compute the median ToF and the corresponding ${p_i}^{opt}$ as in the tests in Section 2.6. Note that this requires extensive training at many locations, but we will show how to train a model without manual efforts in the next Section by training using data collected only between the APs. The empirical distribution of the median ToF versus ${p_i}^{opt}$ for all the 207 links of Testbed I is shown in the top of Fig. 2.10.

We note that ${p_i}^{opt}$ is widely distributed between 0 and 50%. Therefore, it does not exist one value of percentile p that it is optimal for all the links, but it rather changes from link to link. Nevertheless, there is a clear linear trend which we will exploit in our estimator f. To get the model, we perform a linear regression on these data points and obtain an environment-dependent linear model, as represented by the continuous line in the figure. Formally, let $\widehat{{p_i}^{opt}}$ be the estimated optimal percentile p_i^{opt} for a link $i \in \mathcal{L}$. $\widehat{{p_i}^{opt}}$ is computed with the following linear regression equation:

$$\bar{d_i} = a \cdot \widehat{{p_i}^{opt}} + c \qquad a \leq 0, c \geq 0 \tag{2.8}$$

where $\bar{d_i} = d_i(\widehat{p = 0.5})$ is the median ToF (estimated distance using the median), and the regression parameters a and c are trained by environment calibration on a given set

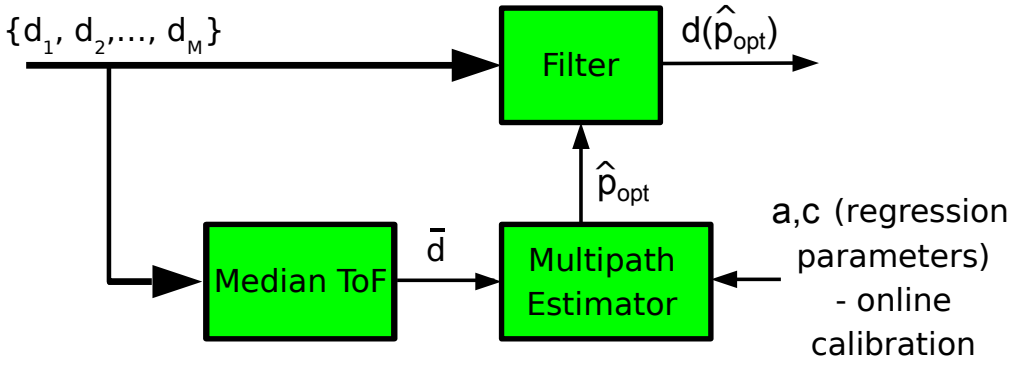


Figure 2.11: Adaptive filter design. A sequence of ToF measurements $\{d_1, d_2, \ldots, d_M\}$ are filtered by selecting a percentile p of the distribution according to the median of the measurements $\bar{d}$ which serves as an estimate of the amount of multipath on the link. The multipath estimator is trained using a linear regression model of the median ToF versus the optimal percentile that minimizes the error from ToF measurements between the APs.

of known distances, minimizing the sum of squared residuals. Applying classical linear regression theory [44], we can then state that a ρ^2 ratio of the total variation of ${p_i}^{opt}$ can be explained by looking at the median ToF $\bar{d}_i$. Small values of $\widehat{{p_i}^{opt}}$ indicate that stronger multipath is statistically expected using the median ToF for the distance estimation, while values closer to $p = 0.5$ indicate smaller multipath delay.

2.6.2. Automatic Model Calibration

In real deployments, it is desirable to avoid any manual offline calibration to estimate the regression parameters a and c of the linear regression model. We therefore propose to run the calibration methodology based on ToF measurements between pairs of APs from which we know the exact positions. This can be therefore performed online without human intervention. The results of this online calibration between APs of Testbed I are shown in the bottom of Fig. 2.10. We find that the calibration results in a very similar linear regression. This suggests that *in order to calibrate the model for a particular environment, the ToF measurements between each pair of APs are sufficient.*

We illustrate our filter in Fig. 2.11. The individual steps are as follows:

1. First, we train a linear regression model from online measurements between the APs and determine the parameters a and c of eq. (2.8). The model characterizes the relationship between the median ToF and the optimal percentile ${p_i}^{opt}$ as defined in eq. (2.7).

2. For each link $i \in \mathcal{L}$, we take a sequence of M measurements $\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}$ and determine the median ToF, $\bar{d}_i$.

3. The median ToF $\bar{d}_i$ is used to estimate the multipath delay using the linear regression model from step 1. The output of the estimator is a percentile value $\widehat{p_i{}^{opt}}$.

4. Finally, for each link $i \in \mathcal{L}$, we apply a linear interpolation to the sequence of ToF measurements $\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}$, select the $\widehat{p_i{}^{opt}}$-percentile of that sequence, and estimate the distance as $d_i(\widehat{p_i{}^{opt}})$.

2.7. Evaluation Methodology

This section presents the algorithms for the distance estimation evaluation used in this work to evaluate the performance of our estimator.

2.7.1. Algorithms for the distance estimation evaluation

For the evaluation, we study the performance of the following filters:

- our filter that uses the median ToF to estimate the optimal percentile (unfiltered[7]),

- two other versions of the filter that rely on the median of the RSSI (pre-filtered) and the skewness of the ToF (pre-filtered), as they provided the second- and third-best correlation coefficients in Section 2.6, Table 2.1.

Recall that these different metrics all estimate the optimal percentile $p_i{}^{opt}$ of the multipath filter and that all three variants rely on the ToF for the final distance estimation. For each of the testbeds in Fig. 2.3, we compute the linear regression coefficients of each moment above with the $p_i{}^{opt}$ (cf. Section 2.6.1).

In order to compare our filter against other approaches proposed in the literature, we further implement the MUltiple SIgnal Classification (MUSIC) algorithm [35], the EM algorithm for GMM model and CAESAR [34]. CAESAR has been specifically designed for WiFi ranging while the MUSIC and the EM algorithms represent popular approaches to mitigate the impact of multipath in wireless localization. In what follows we briefly describe these three algorithms.

2.7.1.1. MUSIC algorithm

First, in order to apply MUSIC to WiFi ToF, we consider that the ToF sequence of samples $\{t_{i,1}^{ToF}, t_{i,2}^{ToF}, \ldots, t_{i,M}^{ToF}\}$ characterizes the wireless link $i \in \mathcal{L}$ between the WiFi AP i and the receiver. We see this sequence as equivalent to the time impulse response of the propagation radio channel. The purpose here is to estimate the P multipath delays;

[7]Since the correlation coefficient of the median ToF versus the optimal percentile is not improved by pre-filtering outliers, we apply this model on the raw, unfiltered samples.

including the first one (the direct path or, in general, the shortest path). So we apply the MUSIC algorithm to $\{t_{i,1}^{ToF}, t_{i,2}^{ToF}, \ldots, t_{i,M}^{ToF}\}$ in the same way it is applied in [35] to the impulse response. The MUSIC algorithm relies on the eigen-decomposition of the autocorrelation matrix of the frequency domain channel response, which will analogically be in our case the Fast Fourier Transform of $\{t_{i,1}^{ToF}, t_{i,2}^{ToF}, \ldots, t_{i,M}^{ToF}\}$ in N_{FTT} points. In our evaluation we set $N_{FTT} = 100$. The frequency domain signal is then decomposed by projection into two orthogonal subspaces. The signal subspace generated by the eigenvectors corresponding to the P largest eigenvalues of its autocorrelation matrix and the noise subspace generated by the eigenvectors corresponding to the remaining N_{FTT}-P smallest eigenvalues. The problem is reduced finally to finding the P multipath delays that maximize the inverse of the norm of the projection vector into the noise subspace followed by applying a peak detection algorithm to find the positions of the local maxima. The position of the first peak corresponds then to the estimated ToF for the direct path component for distance computation.

2.7.1.2. EM algorithm

Second, the EM algorithm [41] estimates the parameters π_k, μ_k, and σ_k of a Gaussian mixture $g(x) = \sum_{k \in \kappa} \pi_k * g_{\mathcal{N}(\mu_k, \sigma_k)}(x)$, where κ indicates the set of modes. It relies on the iterative maximization of the log-likelihood function weighted by the conditional probability that a considered sample belongs to the mode $k \in \kappa$. In our evaluation we decompose the sequence $\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}$ in two modes (with the first one indicating the shortest path)[8] in order to estimate the mean of the first Gaussian distribution, used to compute the distance estimate. The initialization of the parameters π_k, μ_k and σ_k is done as follows: i) $\pi_k = 1/card(\kappa)$, where $card(\kappa)$ refers to the cardinality of the set of modes κ (i.e. we start by assigning equal weights to the Gaussian components of the Gaussian Mixture distribution), ii) for μ_k we randomly select $card(\kappa)$ points from the overall set of data to serve as the initial means, and finally, iii) concerning the initial values of the standard deviations of the Gaussian components, we simply start from the standard deviation of the overall GMM sequence as follows: $\forall k \in \kappa, \sigma_k = \sqrt{var\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}}$.

2.7.1.3. CAESAR

Third, CAESAR relies on comparing the standard deviation σ_i of $\{d_{i,1}, d_{i,2}, \ldots, d_{i,M}\}$ to a threshold t_s to decide whether the shortest path component is predominant for the considered link or not. It exploits the fact that the presence of severe multipath results in a higher σ_i with respect to links with only a direct path. So a correction factor γ_s is subtracted to reduce the overestimated distance caused by the multipath. $\gamma_s = 0$ if

[8] Higher order modes did not show better performance.

$\sigma_i < t_s$ which means that no correction is needed and $\gamma_s = \sigma_i/2$ if $\sigma_i \geq t_s$ to mitigate the multipath effect on a sequence of samples partially received via a reflected path.

2.7.2. Localization and tracking

In addition to the ranging error, we also consider the localization error in static and mobile settings. For device positioning, we define a coordinate system on a two-dimensional map. To ease the notation, we consider only one target and a number of links equal to the number of APs in range, denoted as L. Let $\mathbf{s_i} = (x_i, y_i)$ be the position of the AP i, $\mathbf{p} = (x, y)$ the position of the target device, and $r_i = \|\mathbf{s_i} - \mathbf{p}\| = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ the distance between the AP i and the target device. Let $\hat{d}_i$ further denote the estimate of the distance from AP i to the target, as computed using eq. (2.5). Our objective is to find the coordinates that solve the following weighted least square optimization problem:

$$\hat{\mathbf{p}} = (\hat{x}, \hat{y}) = \underset{x,y}{\operatorname{argmin}} \sum_{i=1}^{L} \frac{(\|\mathbf{s_i} - \mathbf{p}\| - \hat{d}_i)^2}{w_i} \tag{2.9}$$

where w_i is a weighting function of the samples $\{\hat{d}_i\}$. We explain in Section 2.8.3.1 how the weights w_i are determined. For solving eq. (4.1), we apply the Newton-Gauss method with line-search for the stepsize, a well-known method for this problem. As for the initial position for the computation, we use the one given by a linear least square algorithm that linearizes the system above by subtracting a constraint having the advantage of being computationally efficient.

To capture the localization error intrinsic to our filter and avoid including measurement data with high error due to a bad deployment of the AP positions, we calculate the Horizontal Dilution Of Precision (HDOP) values. The HDOP relates to the multiplicative effect of AP geometry on positional measurement precision error. We only consider the positions with an HDOP value smaller or equal to five.

For mobile device tracking, we apply an Exponentially Weighted Moving Average (EWMA) filter of the position:

$$\begin{aligned} \bar{\mathbf{p}}_\mathbf{k} &= (1 - \alpha)\bar{\mathbf{p}}_{\mathbf{k-1}} + \alpha\hat{\mathbf{p}}_\mathbf{k}, \\ s.t. \quad \bar{\mathbf{p}}_\mathbf{1} &= \hat{\mathbf{p}}_\mathbf{1} \end{aligned} \tag{2.10}$$

where $\bar{\mathbf{p}}_\mathbf{k}$ is the current position estimate, $\hat{\mathbf{p}}_\mathbf{k}$ is the last measured position and α is the filter weight. An EWMA filter puts more or less weight on historical values according to the filter weight α. A good choice of α depends on the mobility of the target devices. The filter weight α is computed by means of experiments. Details are provided in Section 2.8.3.3.

2.8. Evaluation

We analyze the performance of our statistical filter for ranging and mobile device position tracking.

2.8.1. Distance Ranging

In this section we evaluate the distance estimation error of our filter.

2.8.1.1. Distance Ranging Accuracy

We first evaluate the ranging accuracy using all links of Testbed I and offline calibration for our filters (the impact of online calibration is evaluated later). Similar experiments are performed in Testbed II and III. For each link, we compute the distance estimation error with 20 samples, and then calculate the average error using 500 sequences. We consider our filter that uses the median ToF, the median of the RSSI and the skewness of the ToF for estimating the optimal percentile. In addition, we also provide the error for CAESAR, the MUSIC algorithm, the GMM model using the EM algorithm, as well as classical estimators such as the mean and median of the ToF.

From Fig. 2.12 we clearly see that our three new estimators outperform the mean and the median metrics by comparing the Empirical Cumulative Distribution Function (ECDF) of the ranging errors. We then compare our best performing filter, based on the median of ToF, against the EM algorithm, CAESAR and the MUSIC algorithm. We show in Figure 2.13 that our filter outperforms the other evaluated approaches. The best performance is achieved by our filter using the median ToF. We obtain a median error of 2.4 m and a 80-percentile error of 5.3 m. The filter that uses the median RSSI slightly outperforms the skewness of the ToF. This is not surprising since Table 2.1 shows higher correlation coefficients of the filter with median RSSI.

The mean and median have roughly equal estimation error. Their median error is approximately 4.5 m and the 80-percentile error is approximately 11.5 m. CAESAR gets a median error of 4.1 m and a 80-percentile error of 7.3 m. Our evaluation of CAESAR is also very consistent to the one recently presented in the indoor evaluation of [24]. With regard to the MUSIC algorithm, this approach is ineffective as a result of the large noise introduced by the target station (cf. Section 2.4), which is not taken into account in the model used by the MUSIC approach. The 80-percentile of the MUSIC algorithm shows better performance than the mean and the median metrics, but still worse than our new estimators. Finally, the accuracy of the EM algorithm is better than CAESAR and MUSIC. However, the median error of 3.9 m and an 80-percentile error of 7.1 m shows its inefficiency with respect to our filter based on the median of ToF.

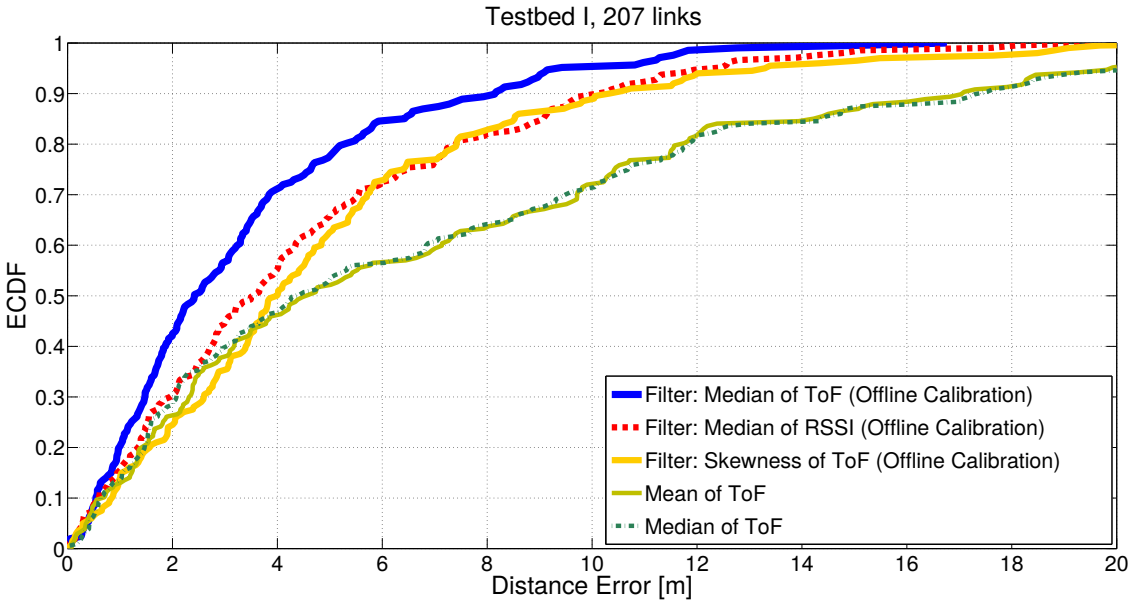


Figure 2.12: ECDF of the distance estimation error for our adaptive filter compared to adaptive filters based on other metrics and classical estimators based on the mean and the median of ToF (Testbed I). For each estimator, we use sequences of 20 samples.

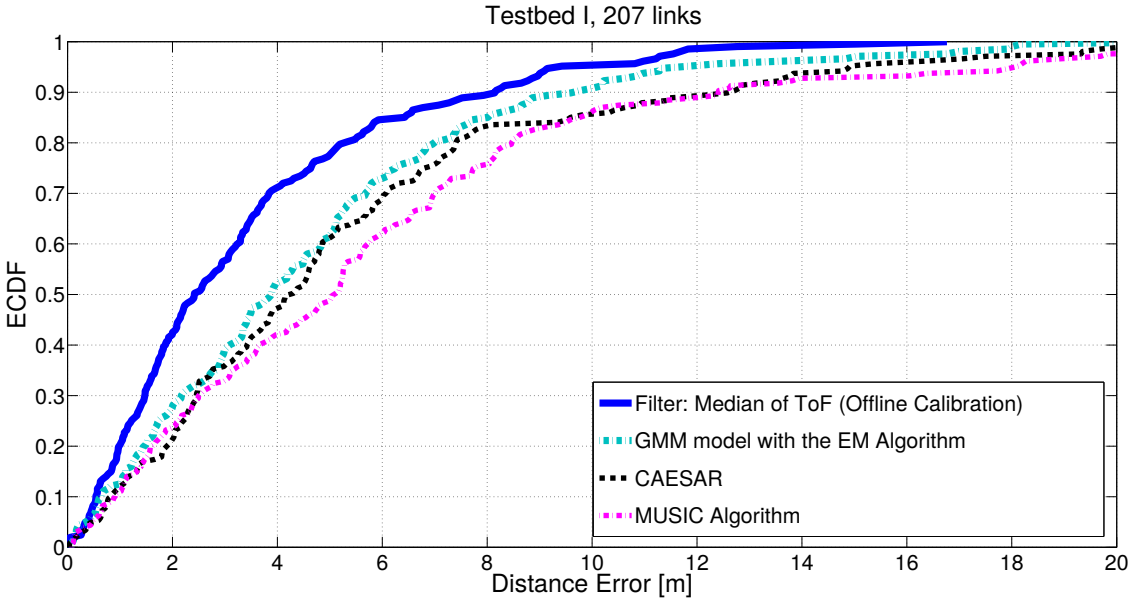


Figure 2.13: ECDF of the distance estimation error for our adaptive filter compared to other algorithms (Testbed I). For each scheme, we use sequences of 20 samples.

2.8.1.2. Robustness to Different Environments/Testbeds

Fig. 2.14 shows the median and 80-percentile of the distance error for the three different testbeds using sequences of 20 samples and offline calibration. As shown in the x-label of the figure, we measure a high Pearson correlation coefficient between the median of ToF

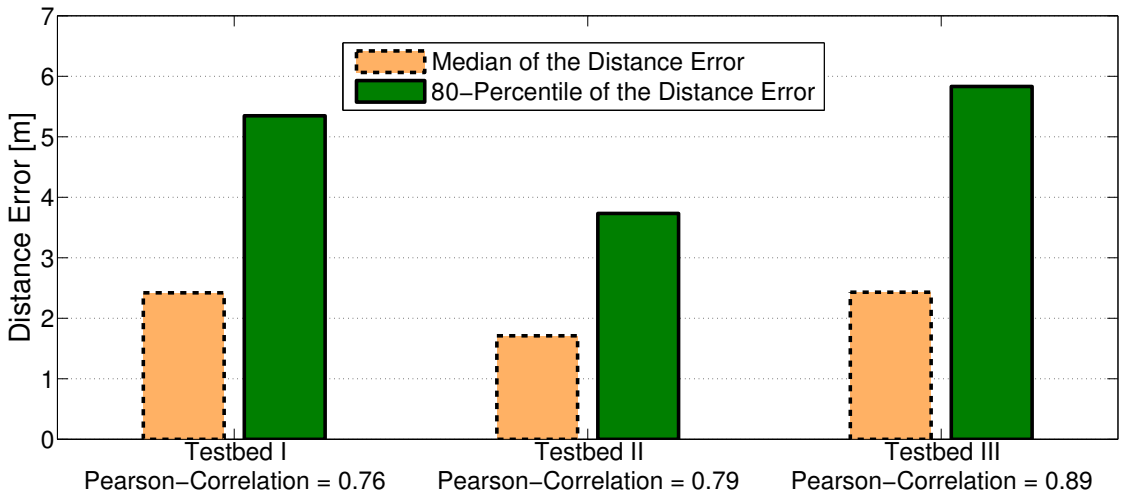


Figure 2.14: Distance accuracy (50- and 80-percentile distance error) in Testbed I, II and III.

and the optimal percentile, in the range of $0.76 - 0.89$. The median distance error is in the range $1.7 - 2.4$ m and the 80-percentile error is in the range $3.7 - 5.8$ m. Concluding, our filter is robust across different environments.

2.8.1.3. Impact of Number of ToF Samples

We evaluate the number of ToF measurement samples M that are used for the estimation of the distance based on our filter. Figure 2.15 shows the error for our filter that relies on the median of the ToF as a function of the number of samples. The error is stable with ten or more samples for the median of the distance error. Only the 80-percentile of Testbed III can clearly benefit from a higher number of samples.

2.8.1.4. Ranging Capacity

We study the capacity of the WiFi ranging technique, defined as the time C required to collect M samples in a WiFi network of N users. In order to find the capacity of the ToF ranging method, we apply Little's formula [45]:

$$C = \frac{M \cdot N}{S/P}, \tag{2.11}$$

where S is the throughput and P indicates the payload bits of the single frame. We conduct the analysis in saturation conditions, where we can apply the well-known Bianchi's formula to compute S [46], and we consider that the traffic for ranging does not have data content (see the details in Section 2.2.2) and it only consists of MAC overhead. Considering $M = 20$ samples for the ranging estimation, the results versus the number

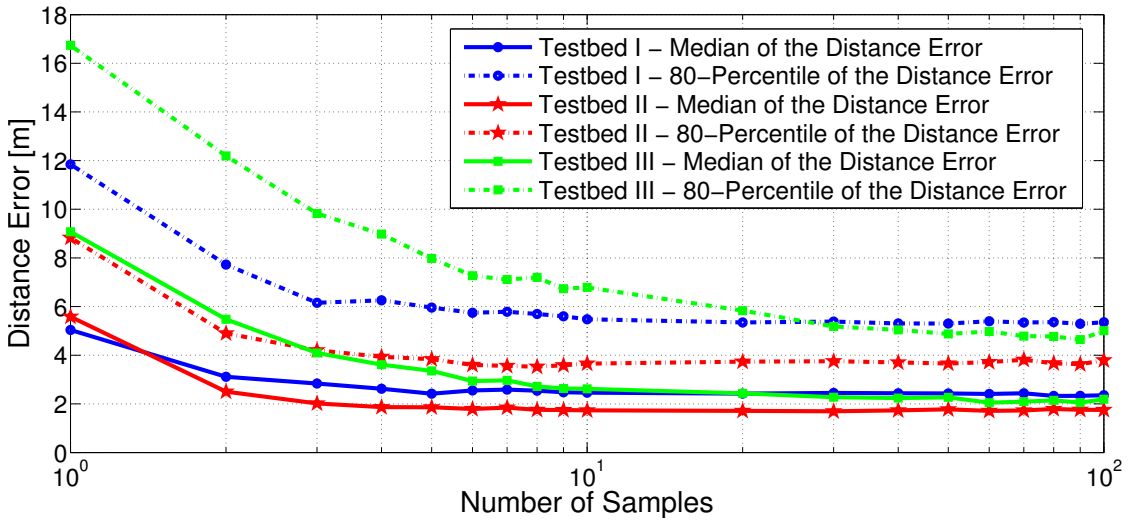


Figure 2.15: Median and 80-percentile error of our estimator that filters the noise based on the median of ToF. Results are provided for different number of samples for Testbed I, II and III.

of users in the system are shown in Fig. 2.16 , considering different PHY rates for the communication. As expected, increasing the PHY rate helps reduce the time C required to compute ranging estimates for N users. For instance, approximately 0.25 s are needed to compute the ranges to 30 users with a PHY rate of 11 Mb/s, while 0.1 s is needed with a PHY rate of 18 Mb/s.

2.8.2. Online vs offline calibration

Our proposal for online calibration is to train our model for the adaptive filter using the links between the fixed APs. Since the APs are at fixed known locations, this type of calibration does not require any additional manual effort and can therefore be performed online.

We compare online versus offline model calibration for Testbed I in Fig. 2.17. The online calibration achieves similar results with respect to the offline tests. Thus, the model calibration can be executed online without significant performance loss. Furthermore, online calibration outperforms the results we could achieve applying the linear regression achieved in Testbed II and III to Testbed I, which implies that it is better to calibrate online than using the calibration coefficients of other environments.

We also find that the regression parameters $\{a, c\}$ of Testbed I and Testbed IV are very similar. This is mapped to very similar performance observed in Fig. 2.17 using the offline calibration of Testbed IV. These testbeds use the same environment but different placements of the APs. This result suggests that the calibration is largely AP-position independent, but rather a feature of the environment.

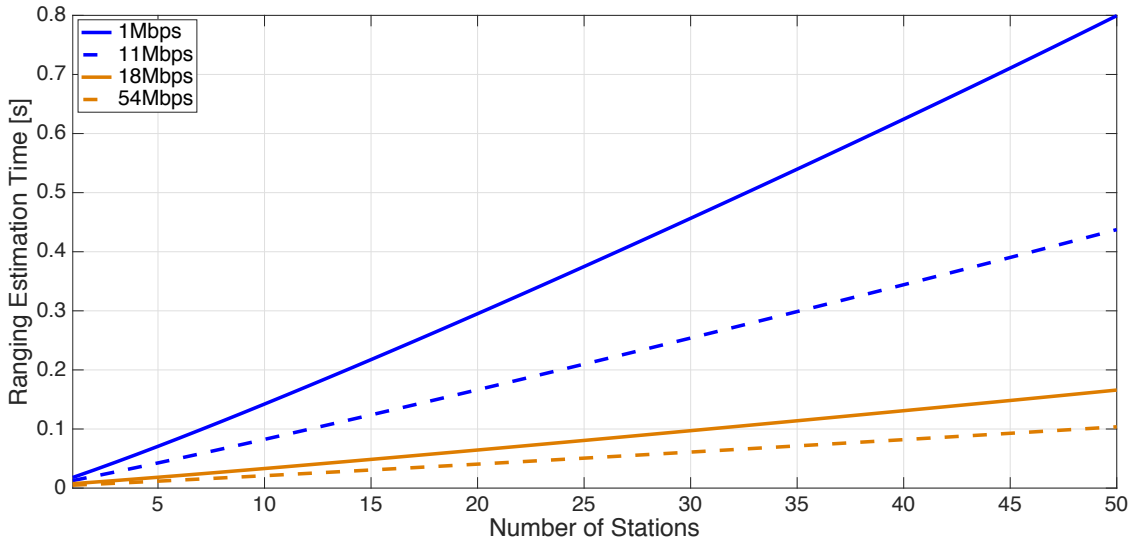


Figure 2.16: Capacity analysis for ranging, considering N target stations and $M = 20$ samples for the ranging estimation.

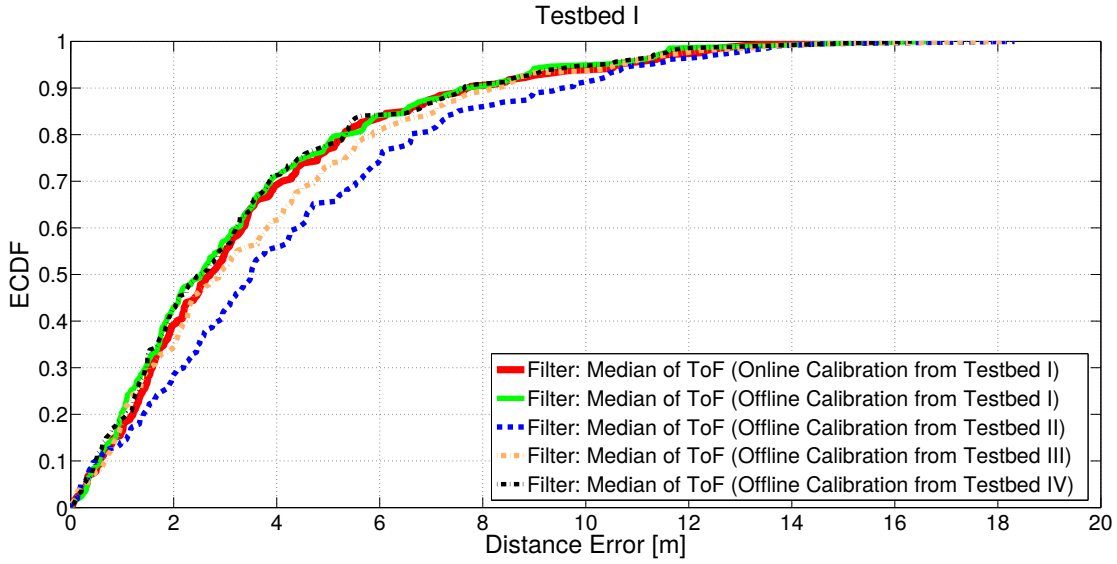


Figure 2.17: Comparison of offline and online calibration for Testbed I, and comparison with different placements of the APs (i.e., Testbed IV).

We then study the variability of the coefficients of the automatic model calibration over time. These tests rely on experiments with 5 APs across an area of 250 m^2 in an office space. We perform the online calibration each hour for a total of 48 consecutive hours. The results of this investigation, in terms of the evolution of the coefficients a and c (see eq. (2.8)) versus the time are shown in Fig. 2.18. From the Figure, we can clearly see a dynamic trend during the working hours of the day and a static one during out-of-office hours. As a result of this experiment, we can conclude that frequent updates, in order of

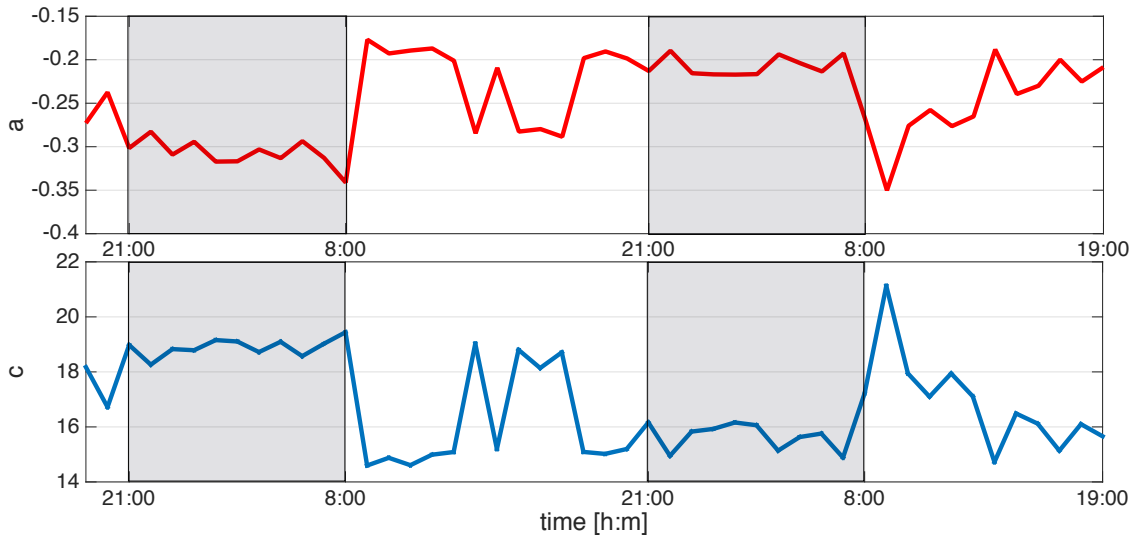


Figure 2.18: Evolution of the coefficients of the linear regression model used for online calibration.

every hour, may be required during the day due to the presence and mobility of people as well as objects moved around, both increasing the dynamics of the environment.

2.8.3. Localization and Tracking

Following the ranging accuracy, we investigate next the positioning and tracking performance.

2.8.3.1. ToF weighting

Previous work has shown that weighting $\{w_j\}$ according to the standard deviation of the measured samples is a useful indicator for the goodness of the measurement [47]. However, we find this approach to be ineffective in our work. In particular, we compare different weighting function and compute the position using our best performing filter (median of ToF) using all the three static testbeds. The results are summarized in Table 2.2. We observe that weighting the distance generally helps with respect to no weighting at all, especially in terms of 80-percentile and Root Mean Square Error (RMSE). Since weighting based on the median ToF gives the best performance in terms of 80-percentile and RMSE, we use this weighting function for positioning the device[9].

2.8.3.2. Positioning Accuracy

For the evaluation of positioning and tracking error, we study the position accuracy for Testbed I, II and III, and summarize the results in Fig. 2.19. The figure shows that

[9]The weight based on the median ToF uses a slope coefficient of 0.09. Other slopes in the range of $0.05 - 0.20$ achieve very similar performance.

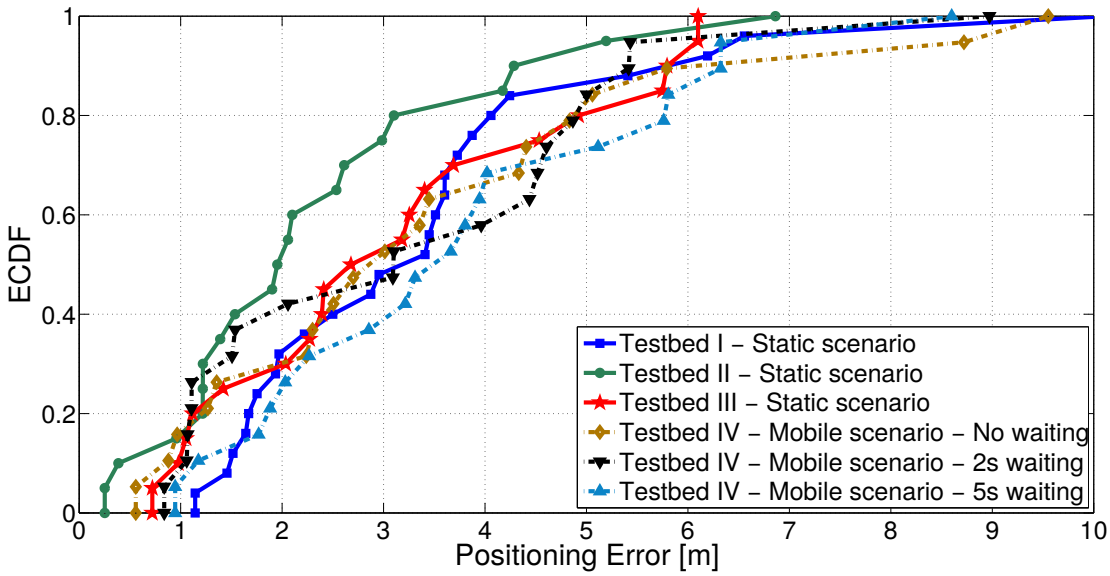


Figure 2.19: Positioning error in static (Testbed I, II and III) and mobile scenarios (Testbed IV).

we have a median positioning error between 2.0 and 3.1 m, and 80-percentile positioning error between 3.2 and 4.8 m, depending on the testbed. Compared to ranging, the median positioning accuracy is slightly worse ($1.7-2.4$ m for ranging). However, the 80-percentile error is slightly better ($3.7-5.8$ m for ranging).

2.8.3.3. Mobile Tracking

To determine the best filter weight α for the EWMA in mobile settings, we first explore how the RMSE of the positioning error is affected by α using distance ranging sequences of $M = 20$ samples. The tracking error versus α is shown in Fig. 2.20. $\alpha = 0.03$ performs well and is fairly robust to differences in waiting times. We therefore apply $\alpha = 0.03$ in the remaining tests.

Fig. 2.19 shows the ECDFs for the different waiting times in Testbed IV at each

Table 2.2: Positioning error analysis - impact of weight w_j using all the positions of our static testbeds (Testbed I, II and III). The median of RSSI here is only used to compute the weight w_j but the ranging is always based on ToF samples.

	50-percentile	**80-percentile**	**RMSE**
median ToF	2.6 m	4.5 m	3.7 m
median RSSI	2.8 m	5.1 m	4.2 m
average $\{\hat{d}_i\}$	2.4 m	4.6 m	3.8 m
standard deviation $\{\hat{d}_i\}$	2.8 m	5.3 m	4.3 m
no weighting	2.7 m	5.4 m	4.3 m

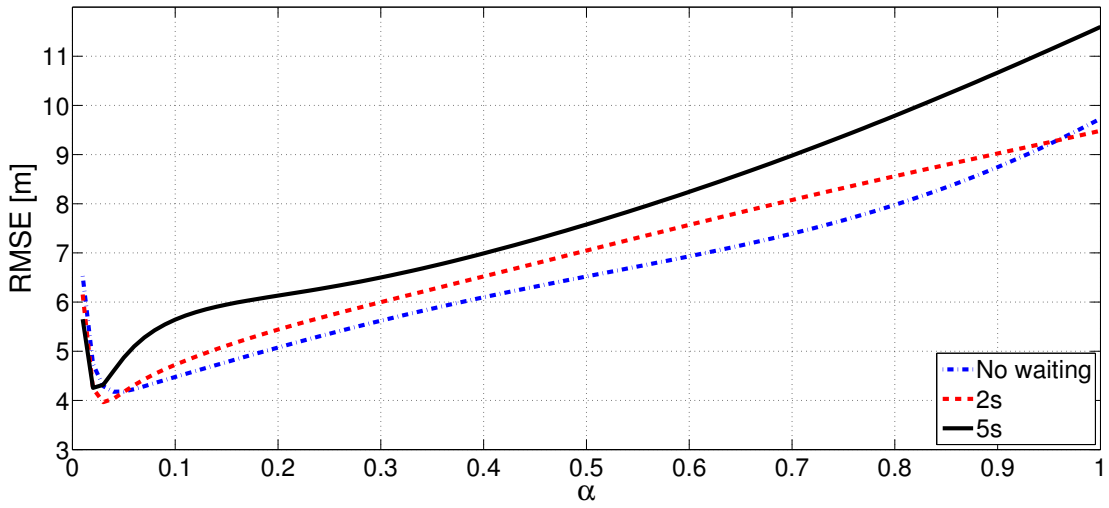


Figure 2.20: RMSE of the positioning error as function of α for different setups of Testbed IV.

location, no waiting, two seconds and five seconds waiting. We observe that the tests under full mobility (label 'no waiting') give comparable positioning accuracy to the cases where the user waits for a few seconds to collect more data at a given position (label '2s' and '5s'), with a median error in the range $2.6 - 3.4$ of meters. The reason for this result is that mobility increases the performance in tracking mode because the moving device experiences different multipath constellations which may cancel out more effectively than when the device remains fixed. This effect compensates for the increased tracking error which arises when measurements are slightly lagging behind due to the measurement delays when the device constantly moves ahead.

2.9. Discussion

We have developed a new filter based on statistical learning and robust statistics to improve the ranging accuracy in the presence of noisy ToF measurements. Our filter does not require additional information form the devices or the user besides the ToF values which makes it applicable to a wide range of applications. We have shown how to apply our filter for indoor localization and tracking of COTS WiFi devices with legacy 802.11 Access Point infrastructures. In indoor deployments with multipath, our filter outperforms conventional ToF based range estimators by a factor of more than two. We have demonstrated the accuracy of the filter to estimate the position in static and mobile settings. In static conditions, the filter achieved a median error of $2.0 - 3.1$ meters. In mobile settings, the error was only slightly higher with a median error of $2.6 - 3.4$ meters. We have shown that the performance of our filter can be achieved with online model calibration, and hence does not require any cumbersome onsite pre-calibration efforts.

3 Single Access Point

Indoor positioning has attracted a lot of attention from the research as well as industry communities. Due to the limitation and complexity of the indoor environment, the problem of bringing indoor positioning to homes and small businesses which typically have a single AP remains open. Accurate indoor positioning can be achieved using different approaches such as the signal strength [13–17], Angle-Of-Arrival (AOA) [18–20, 23], time-based ranging [29, 30, 34, 48] or combining WiFi signals with inertial sensors as found in smartphones [21, 24]. While localizing a client with a dense network of APs has been solved [23], early attempts to operate with a single AP assume that the WiFi chipset in the AP can continuously change its frequency of operation [22, 49]. However, commodity smartphones do not have this feature, which is not supported by any 802.11 standard. [24] operates with a single AP, but requires the access to inertial sensors of smartphones and extensive manual calibration per user. All the above factors limit the deployment of these approaches at larger scale.

We introduce SPRING, Smartphone Positioning with Radio measurements from a sINGle wifi access point. First, it takes advantage of the 802.11ac standard, exploiting in 80 MHz bandwidth the PHY layer information called Channel State Information (CSI), obtained from multiple antennas in a single AP. These chipsets come with problems of calibration before being used to extract high-resolution AOA. Second, recently, IEEE 802.11-2016 standardized [50] the Fine Time Measurements (FTM) protocol to supports time-based WiFi ranging techniques. So far, there have been few studies of such ranging system, with limit investigation of methods to alleviate the multipath. Based on corrected input data, SPRING measures the angle and the distance from a single AP to a COTS smartphone (Google Pixel 3), showing for the first time the feasibility to locate a smartphone in typical indoor environments only with measurements performed by the single AP.

The chapter is organized as follows. First, we describe how to obtain AOA estimations from CSI. We then discuss our proposed solution to avoid the phase calibration issue related to the WiFi chipset, showing how well it alleviates the phase error. Next, as FTM

is a new feature from the 802.11 standard, no model to treat the noise has been presented so far. We analyze the key factors and parameters that affect the ranging performance using the FTM protocol and introduce a model to treat the noise in FTM measurements. We mitigate the impact of multipath in FTM measurements, developing an estimator that receives calibrated inputs from CSI. We also compare our estimator with other techniques, showing performance improvements. Finally, we evaluate SPRING and we demonstrate using only commodity hardware that the target device can be localized across two different experimental testbeds with a median accuracy between 0.9 and 2.15 meters in areas up to 125 m^2.

3.1. Channel State Information

This section presents how to obtain CSI measurements with the proposed hardware, presented in Section 3.4.1, and how to estimate the AOA.

3.1.1. From CSI to AOA estimate

We obtain the AOA from the CSI, available in the PHY layer. The reported CSI is a matrix containing one complex number per subcarrier and per received antenna at the AP. We use MUSIC algorithm [51] to identify the strongest angle of arrival. This technique performs the subspace decomposition of the autocorrelation matrix.

3.1.2. Measurements in Anechoic Chamber

Our first experimental setup consists of a custom lab prototype as AP and a raspberryPi with an external dongle for supporting the 802.11ac standard configured as STAtion (STA). We start with the anechoic chamber as indoor scenario, in order to avoid multipath. More in details, the STA is connected to the AP through channel 112 at 5.56 GHz as carrier frequency. The device gives the ability to configure the number of subcarriers (N_s) sending data, typically used by the Compressed Beamforming Feedback Matrix in 802.11ac for beamforming. We select the highest number of subcarriers (234), which allows us to get the highest resolution in the analysis of the spectrum. In this setup, we fix the STA in the middle of the room and rotate the AP in order to span an angle of π, from $\frac{-\pi}{2}$ to $\frac{+\pi}{2}$, where 0° corresponds to the normal direction of the antenna array elements, which are placed as Uniform Linear Array (ULA) with a distance of $\lambda/2$ between antennas. More specifically, we conduct 36 experiments, every 5°, and for each of them we collect around 165 CSI samples. Fig. 3.1 shows the true AOA versus the estimated AOA in degrees, and illustrates a calibration issue that introduces a bias in all estimations. This phenomenon is due to a phase shift in each antenna for all N_s subcarriers.

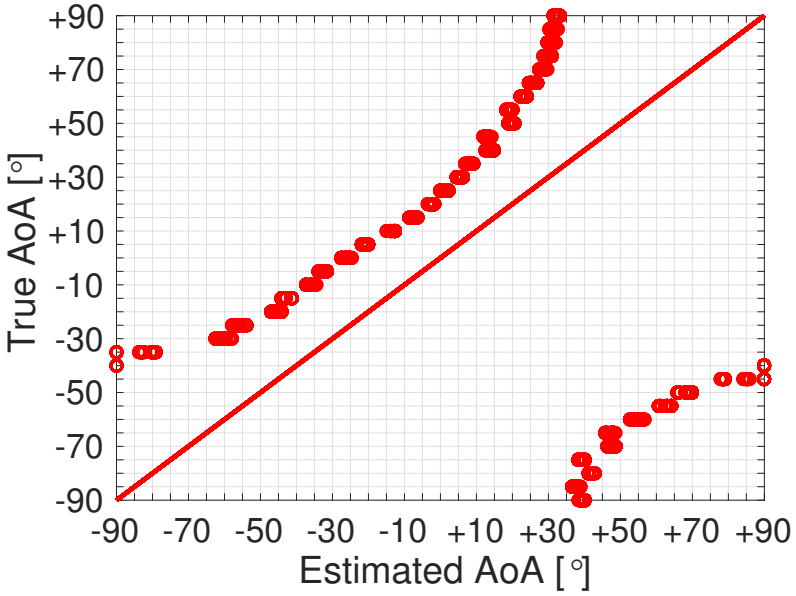


Figure 3.1: Before phase calibration

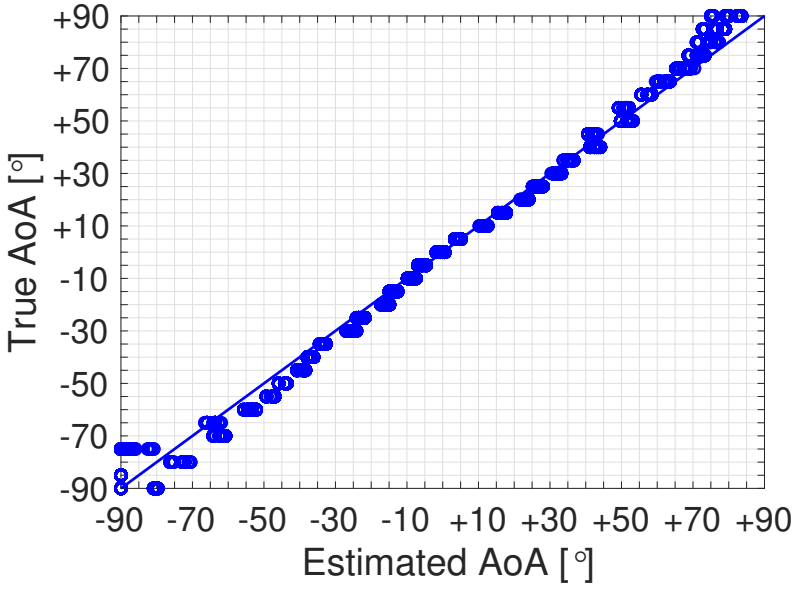


Figure 3.2: After phase calibration

Figure 3.3: MUSIC estimations in the anechoic chamber.

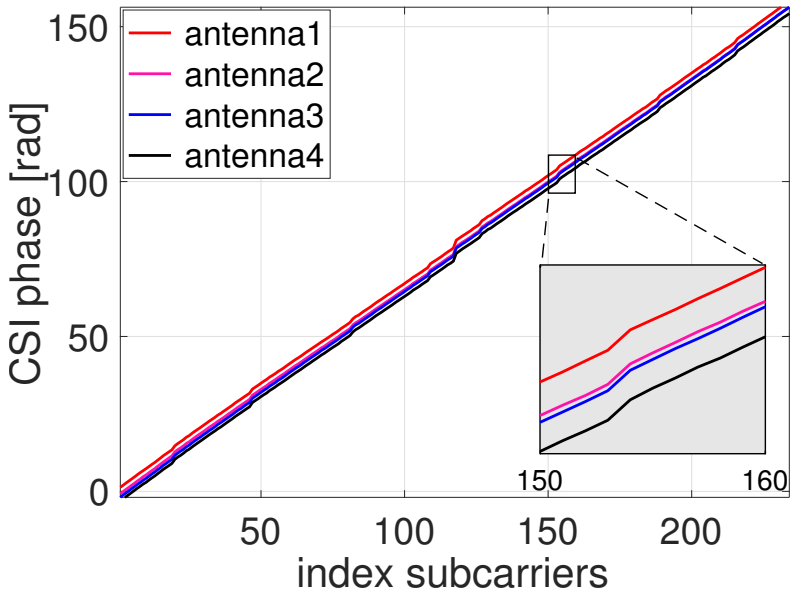


Figure 3.4: Before phase calibration

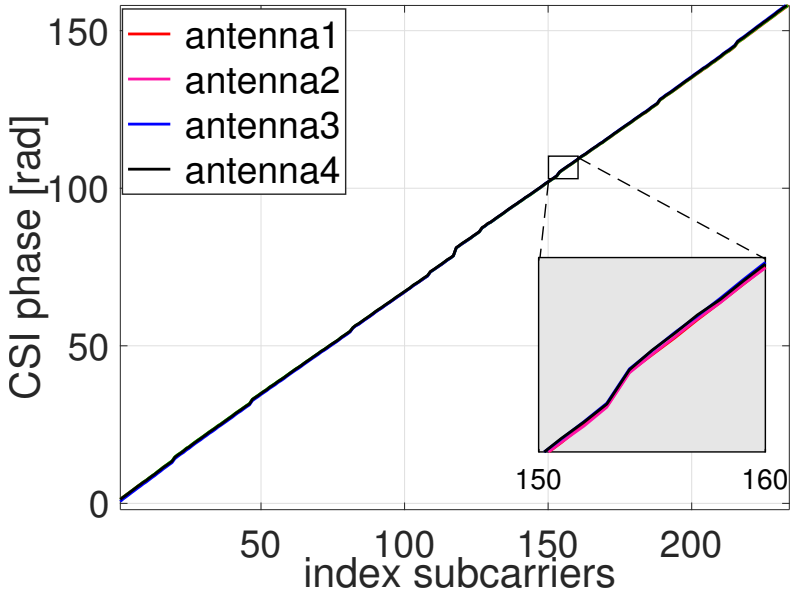


Figure 3.5: After phase calibration

Figure 3.6: A snapshot of CSI unwrapped phases for each antenna, at 0°.

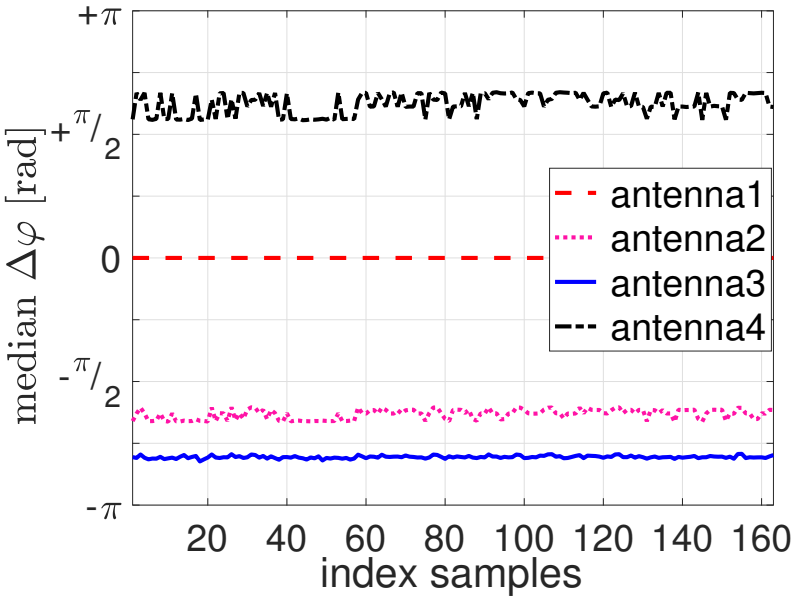


Figure 3.7: Median of $\Delta\varphi$ for each antenna.

3.1.3. Phase Calibration

In this subsection, we propose a solution to correct the bias shown in the previous subsection, without involving any theoretical model. The principle behind is that, due to the far field assumption of a received single path CSI, at 0° measurements all the phases are supposed to be overlapped, for all antennas and over all N_s subcarriers. As shown in Fig. 3.4, the CSI unwrapped phases of all 4 antennas are shifted, and for this reason we estimate another AOA ($\sim 24°$).

To correct this shift, we evaluate the phase difference between all antennas and all sub-carrier with respect to the measurements at the first antenna at 0° only, given by the vector

$$\Delta\varphi_m = \angle(\mathbf{CSI}_m \cdot \overline{\mathbf{CSI}}_1), \tag{3.1}$$

where m is the antenna index and $\overline{\mathbf{CSI}}_1$ is the complex conjugate of vector $\mathbf{CSI}_1$.

For each antenna and subcarrier, we then calibrate the CSI as follows:

$$\textbf{calibrated } \mathbf{CSI}_m = \mathbf{CSI}_m \cdot e^{-j\Delta\varphi_m}. \tag{3.2}$$

Doing so, we obtain the required overlap, as shown in Fig. 3.5, where the difference between the unwrapped phases of all antennas is close to zero. In Fig. 3.7, the solid line shows the evolution of the median of $\Delta\varphi$ for each antenna over the set of samples $\mathcal{S}$ (one sample is collected per packet). Based on eq. 3.2, the median of $\Delta\varphi_1$ is 0 for each packet,

and the median over all packets of the median of $\Delta\varphi$ across all subcarriers is

$$\Delta\varphi_m = [0, -1.976, -2.532, 2][rad], with\ m \in \{1, .., 4\}.$$

Fig. 3.2 shows the true AOA versus the "calibrated" AOA estimation, highlighting the benefit of the proposed CSI phase calibration. Doing so, the estimated AOA well matches the true AOA. This procedure is necessary only once.

In this section we provide the background on the FTM protocol of the IEEE 802.11-2016 standard and we model the detected FTM noise.

3.2. IEEE 802.11mc Background

IEEE 802.11-2016 standardized the FTM protocol to enable a pair of WiFi chipsets to estimate the distance between them. A FTM initiator (FTM-I) is a STA that initiates the FTM process by sending a FTM Request to a corresponding AP. An AP that supports the FTM procedure as a responding device is called a FTM responder (FTM-R).

If the AP agrees to perform the measurements, it starts to send FTM message and wait for its ACK. The Round Trip Time (RTT) is estimated based on the transmission timestamp of the FTM message and the reception timestamp of its ACK. In the computation, the protocol subtracts STA processing time from the total RTT. Yet, as all localization systems performing active wireless measurements, data retransmissions are possible due to collisions. It follows that interference tends to increase the time required to obtain a sufficient number of messages for localization.

3.2.1. FTM Sources of Noise

Signal propagation in rich indoor environments is subject to multipath effects where multiple coherent copies of the transmitted signal arrive at the receiver over different reflected paths. It is even possible that the direct component is severely attenuated and the signal is received mostly over reflected paths. Since signals that travel over reflected paths will take longer time to arrive at the receiver, they introduce an error in the distance estimation when considering the time-of-flight.

We define the following function y for a path $i \in \mathcal{L}$, where $\mathcal{L}$ denotes the set of paths to the target station (FTM-I):

$$y = \log_{10}(d_i) + \mathcal{N}(0, \sigma_{\mathcal{N}}) \qquad d_i \geq d_0. \tag{3.3}$$

$\mathcal{N}(0, \sigma_{\mathcal{N}})$ represents an additive Gaussian noise $\mathcal{N}$, with a standard deviation $\sigma_{\mathcal{N}}$, and d_0 is equal to 1 m. This expression is inspired by the path loss model with the log-normal distribution that represents the shadowing effect.

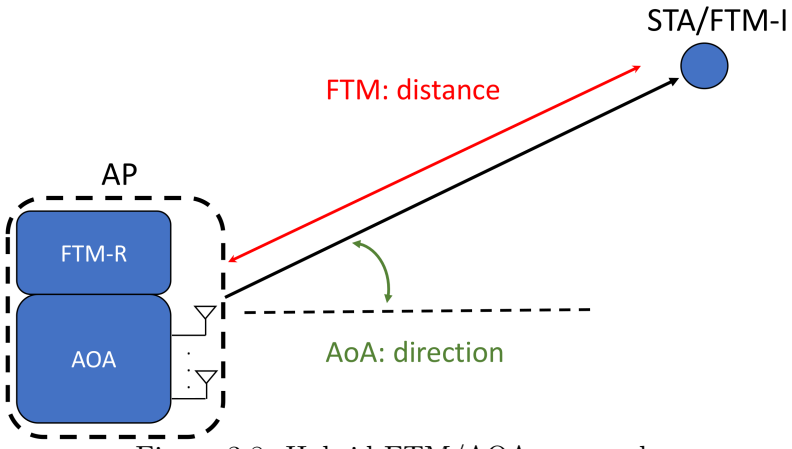


Figure 3.8: Hybrid FTM/AOA approach.

Now, let $\mathcal{S}$ be the set of samples. We express the generic FTM sample as:

$$d_{i,s} \qquad i \in \mathcal{L}, s \in \mathcal{S}. \tag{3.4}$$

Based on the model in eq. (3.3), in Section 3.3.1 we will introduce the first path estimator f to mitigate the effects of the main sources of noise in the channel. The estimator will operate in the log domain, i.e., $log_{10}(d_{i,s})$.

Furthermore, the RTT for short distances returns negative values, underestimating the true distance. Since multipath can lead to longer paths, we believe that this effect, that allows the signal to arrive earlier than expected, is due to a post-processing in the firmware. We correct the offset using the experimental value in [52]. However, even after correcting the fixed offset, the ranging system may give $d_{i,s}$ smaller than d_0. As such, we add a constant factor for the purpose of operating in the log domain, such that it is larger than d_0 for a sequence of samples.

3.3. Hybrid FTM/AOA system

In Fig. 3.8 we show the high-level illustration of our hybrid FTM/AOA localization system. While being associated to the WiFi chipset performing AOA measurements, the STA sends FTM requests to the FTM-R (using a different WiFi chipset). This is possible since the protocol allows the initiator to exchange FTM packets without the need to be associated with a WiFi AP. We collect AOA and distance measurements from AP AOA and FTM-R, respectively, and post-process the results.

3.3.1. FirstPath Estimator for FTM

This subsection describes in details the estimator f of the first path.

In eq. (3.4), each FTM sample $d_{i,s}$ is affected by a distance bias caused by the absence

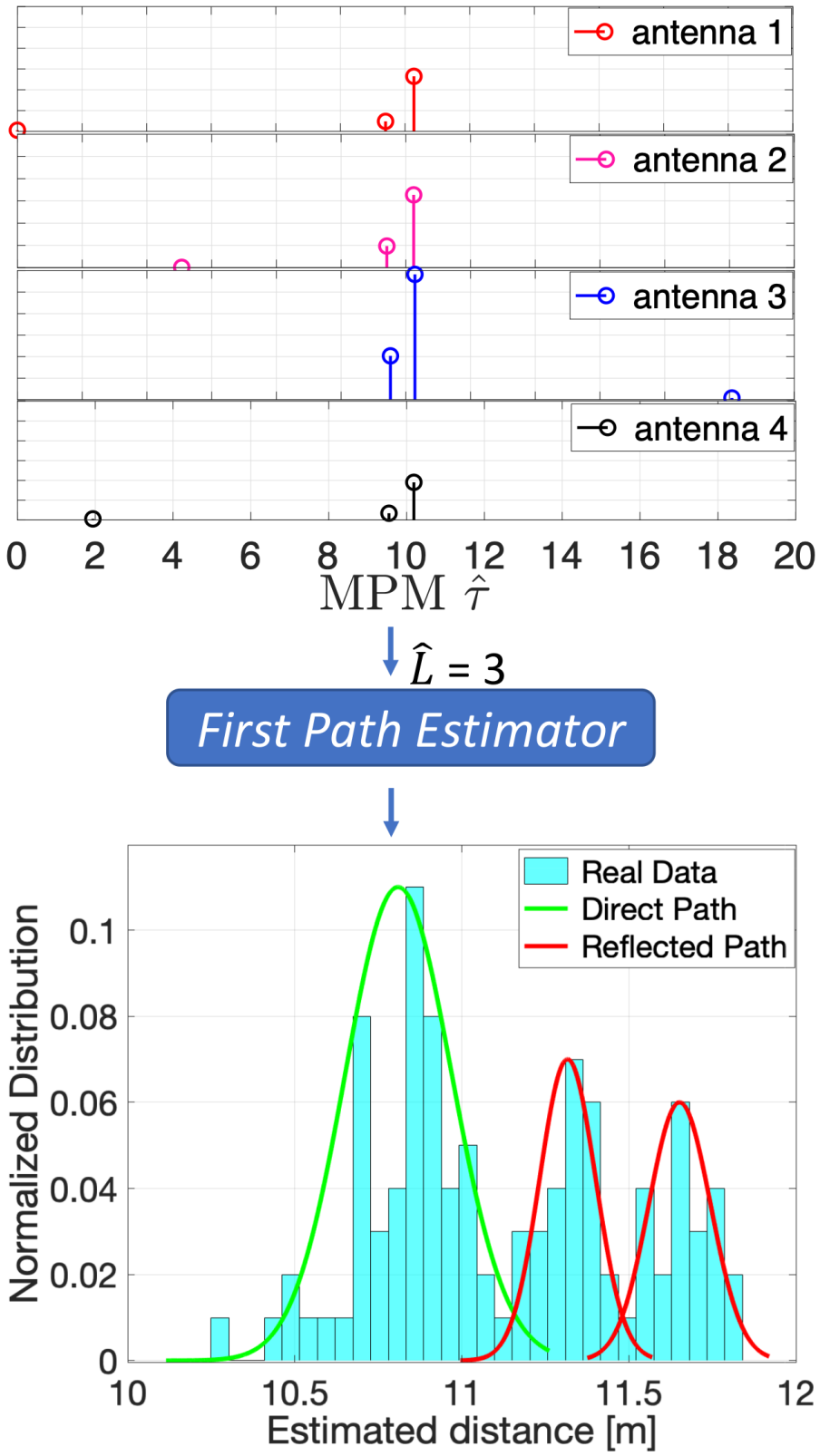


Figure 3.9: Example of the estimator f for a real case where the real distance is 10.73 m. The number of paths, $\hat{L}$, is the mode of the estimations provided by the Matrix Pencil Method (MPM) for each antenna, and it is given as input to the estimator f that calculates only the mean of the first log-Gaussian.

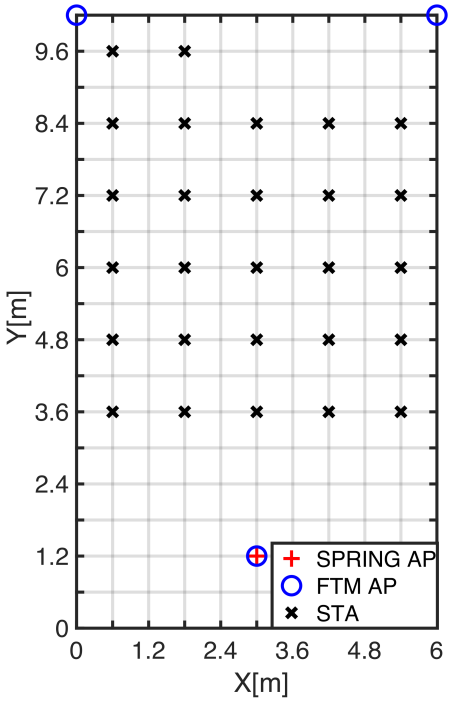


Figure 3.10: Testbed I.

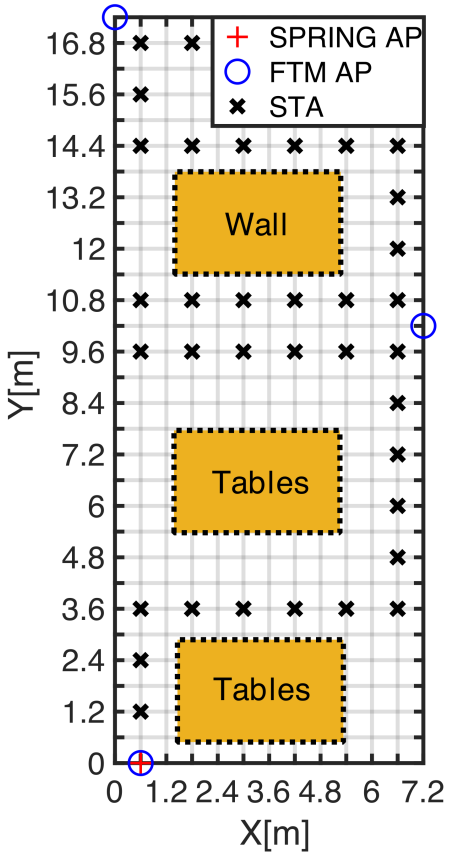


Figure 3.11: Testbed II.

Figure 3.12: Testbeds to assess the direction, ranging and positioning capabilities of SPRING and baseline. Yellow box in Testbed II corresponds to blockage.

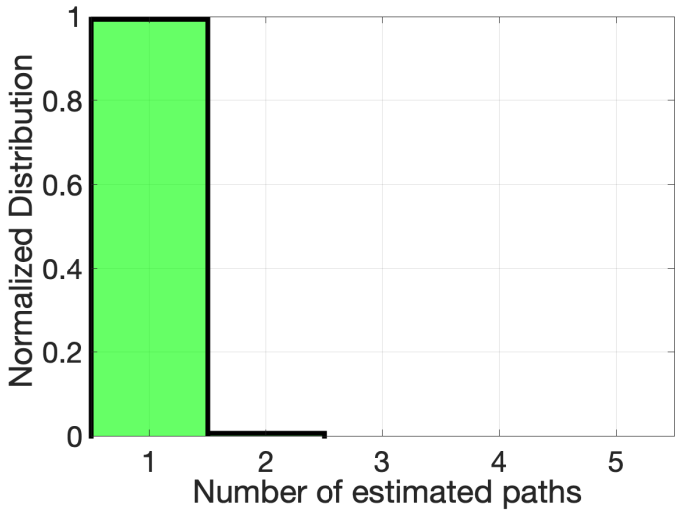


Figure 3.13: Testbed I.

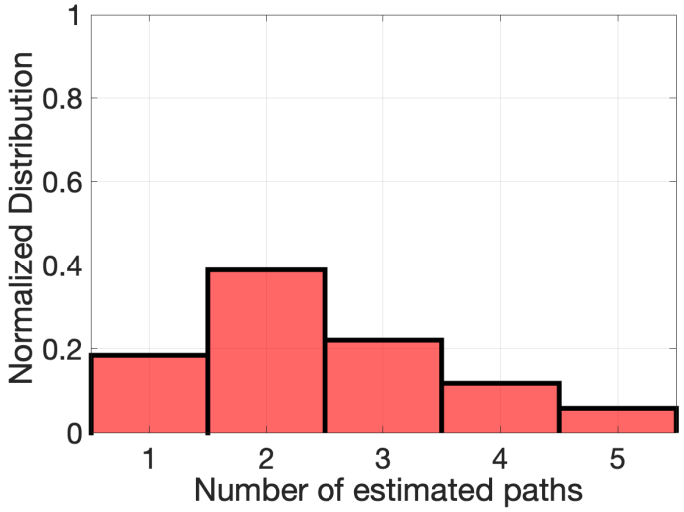


Figure 3.14: Testbed II.

Figure 3.15: Normalized histograms of the number of estimated paths for Testbed I and II.

or presence of multipath. Grouping together the samples with the same bias results in a finite GMM in the log domain with a small number of modes. One of these modes corresponds to the samples received through the direct path, while the others correspond to the samples received through to any of the reflected paths.

Then, knowing the number of estimated paths, we can separate all the Gaussian components, and the median or the mean of the first Gaussian would be a reliable estimator of the direct path's distance. For this purpose, we exploit the PHY layer information. But we do not invoke MUSIC algorithm as we do for AOA estimation. In fact, even in a single path scenario, the AOA MUSIC spectrum might present side lobes and furthermore, in the presence of multipath, the incoming signals are coherent and therefore the rank of the covariance matrix does not increase in order to be able to separate them. Consequently, having two incoming paths well separated in angular domain, we might only detect a single path. We instead resort to the MPM [53]. Knowing that the multipath channels are usually modeled in time-domain as a weighted sum of delayed impulse functions, for each antenna a, let L_a be the number of delayed paths, and $\tau_{l,a}$ and $h_{l,a}$ the propagation delay and the complex gain of the l-th path, respectively. MPM uses as input the calibrated CSI values from eq. (3.2), it operates on the frequency response of the channel and it calculates with high accuracy the estimated values $\hat{L}_a, \hat{\tau}_{l,a}$ and $\hat{h}_{l,a}$. We then calculate the estimated number of paths as $\hat{L} = Mo(\hat{L}_1, \hat{L}_2, \hat{L}_3, \hat{L}_4)$ where Mo indicates the most frequent value (mode).

Fig. 3.9 shows an example of a real case. We observe that the estimator f estimates the parameters of the components of the Gaussian mixture very well: the direct path has a traveled distance of 10.73 m and the estimated mean of the direct path's distribution is equal to 10.82 m. It follows that we are able to estimate the distance of the direct path with an error of only 0.09 m. In Section 5.3 we show a comparison between the proposed filter and other metrics.

3.3.2. Deployment Scenarios

We perform experiments in two indoor testbeds, namely Testbed I and II. Both are office environments and the maps are shown in Fig. 3.12. Testbed I, in Fig. 3.10, covers a surface of almost $65\,m^2$. We use 27 selected locations (marked as cross) to test our system, and the propagation is mainly over a LOS path. Deploying a single AP, the number of links is equal to the number of target STA locations. Testbed II is depicted in Fig. 3.11. It covers a space of around $125\,m^2$ and the User Equipment (UE) is placed in 35 different positions. In Testbed II, a mixture of LOS and NLOS wireless links are present. Testbed II also contains several obstructions, including concrete walls and tables (yellow boxes in Fig. 3.11) and it is surrounded by glasses. All experiments are conducted with other active WiFi networks in the neighborhood. The PHY information are obtained on a fixed frequency channel of the 5 GHz band. For the evaluation of our system SPRING

we deploy a single AP ('SPRING AP', red marker in Fig. 3.12), while for the evaluation of the proposed baseline we deploy three FTM-Rs ('FTM AP', blue circle in Fig. 3.12).

Furthermore, we highlight the difference in the scenarios complexity showing the number of estimated paths, using MPM, in Fig. 3.15. In Testbed I MPM estimates a single path almost 100% of the time, while in Testbed II it estimates a variable number of paths (from 1 to 5), due to the mixture of LOS and NLOS wireless links.

3.4. Evaluation

In this section, first we introduce the experimental platform and then we analyze the performance of the AOA estimator, our proposed method for computing ranges and finally positioning of the STA in Testbed I and II. We deploy the Google Pixel 3 as target STA in all marked positions, shown in Fig. 3.12.

3.4.1. Experimental Platform

An illustration of the commodity hardware we use for the AP is shown in Fig. 3.16. The AP is composed by two different commodity chipsets (although we envision that, in the future, manufacturers will implement them in the same chipset). As mentioned in Section 3.3, the STA is associated to the chipset performing AOA, while FTM does not require association of the STA. Therefore, it is possible to use the STA for both type of measurements. Three fundamental components are needed to deploy SPRING: a multi-antennas AP which enables collection of Multiple Input Multiple Output (MIMO) CSI, the FTM-R and the FTM-I, described in details in what follows. In order to collect CSI we use a QHS8405S4-RDK device, the Quantenna (QTNA) 4x1 ULA. QTNA supports Peripheral Component Interconnect express (PCIe), Reduced Gigabit Media-Independent Interface (RGMII) and 802.11a/n/ac protocol. The frequency range is from 5.15 GHz to 5.85 GHz and it supports 20/40/80 MHz bandwidth. QTNA enables rapid collection of precise high-order MIMO CSI. The spatial diagnostics interface is supported on QTNA's BBIC4 based platform and it supports extracting up to 4x1 channels with bandwidth up to 80MHz, with CSI data from the driver accessed over a Transmission Control Protocol (TCP) socket. Any WiFi device can be used as the STA.

Regarding the FTM protocol, we choose the fitlet2-CJ3455 platform as responder (FTM-R) since it is an integrated solution in compact form, which includes the WiFi Intel 8265 chipset. We use the WiFi Indoor Location Device (WILD) configuration tool which enables all the configuration files for a FTM-R.

As STA, we use the Google Pixel 3 phone with Android Pie (API Level 28) that supports the FTM protocol. The phone operates as FTM-I for time measurements. The device must have location-based services enabled at the system level to access the FTM protocol. We use the android-WifiRttScan application to initiate the measurements. We

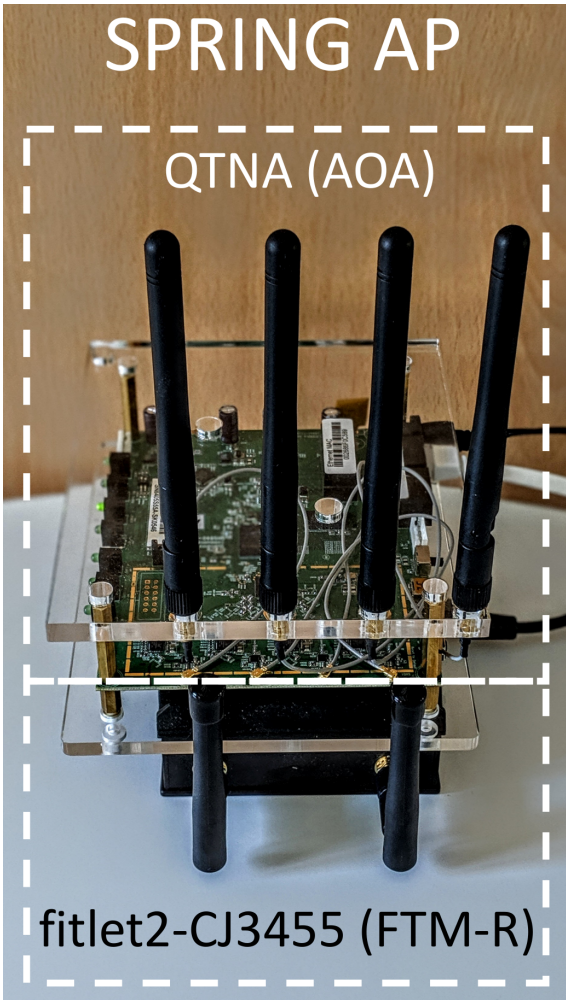


Figure 3.16: SPRING AP used for measurements.

modify its code to facilitate the data collection, and we configure it to receive a distance measurement per packet. Its main activity lists all APs using the WifiManager. By selecting an AP that supports FTM-R, another activity is launched and RangingRequest initiated via the WifiRttManager. The activity displays and stores many of the details returned from the FTM-R including the distance reported between the AP and the smartphone.

3.4.2. AOA

We collect CSI data once the phone Google Pixel 3 is associated to the QTNA device. We estimate the AOA according to the methodology presented in subsection 3.1.1. For the estimation of the strongest path we consider the highest peak in the MUSIC spectrum. We then evaluate the AOA estimation error in Testbed I and II. From Fig. 3.17 we see the ECDF of the AOA error in degrees. Estimating the strongest path does not mean

we are always able to estimate the direct path: the median and 80-percentile error are around 4.8° and 21° for Testbed I, respectively, while 14° and 36.5° for Testbed II. As shown in Fig. 3.17, the error is reduced significantly with respect to a naive approach that considers the estimation of AOA without calibration.

3.4.3. Distance

In order to collect FTM measurements, the FTM-R sends FTM message to the phone and waits for the ACK. For this evaluation, we compute the distance estimation error with $M = 20$ samples for each link. Fig. 3.19 shows the ECDF of the ranging errors, for Testbed I and II. We obtain a median error of 0.49 m and an 80-percentile error of 0.86 m in Testbed I, while in Testbed II a median and 80-percentile error of 0.73 m and 1.5 m, respectively. As shown in Fig. 3.18, we also compare the obtained results with both the median and the Akaike Information Criterion (AIC). The latter is commonly applied to identify the optimal number of clusters in GMM. We use the lowest AIC to infer the optimal number of paths [54] and then we use the path with the least positive mean as ranging estimate. Furthermore, the results shown in Fig. 3.19 highlight the difference on the scenarios complexity. In fact, in Testbed I where the target is in LOS with the AP, the difference between the proposed first path estimator and the simple median is not so relevant. In Testbed II the complexity increases having a mixture of LOS and NLOS wireless links, and so our solution greatly outperforms other estimators.

3.4.4. Positioning

In addition to the direction and ranging error, we also study the localization error. We define a coordinate system on a two-dimensional map. Considering a single AP system, let $s = (x_{AP}, y_{AP})$ be the position of the AP, $\hat{d}$ the estimate of the distance from AP to the target and $\hat{\theta}$ the estimated direction between the AP and the target. We find the coordinates of the estimated position as follows:

$$\hat{p} = (\hat{x}, \hat{y}) = (x_{AP} + \hat{d} \cdot \cos\hat{\theta}, y_{AP} + \hat{d} \cdot \sin\hat{\theta}). \tag{3.5}$$

We study the position accuracy of our proposed system, SPRING, and we compare the obtained results with a baseline composed by three FTM APs. We consider this a fair comparison, as we aim to compare the performance with the minimum number of APs needed to perform positioning: SPRING needs only one AP, while 3 FTM APs are required for multilateration. For the latter, using the estimated distance from each FTM-R, we make an initial position estimate with the Linear Least Squares (LLS) multilateration algorithm. Then the algorithm makes use of the Non-Linear Least Squares (NLLS) technique to compute the position from this initial value. We summarize the results in Fig. 3.20, where solid lines indicate SPRING, while dashed lines correspond

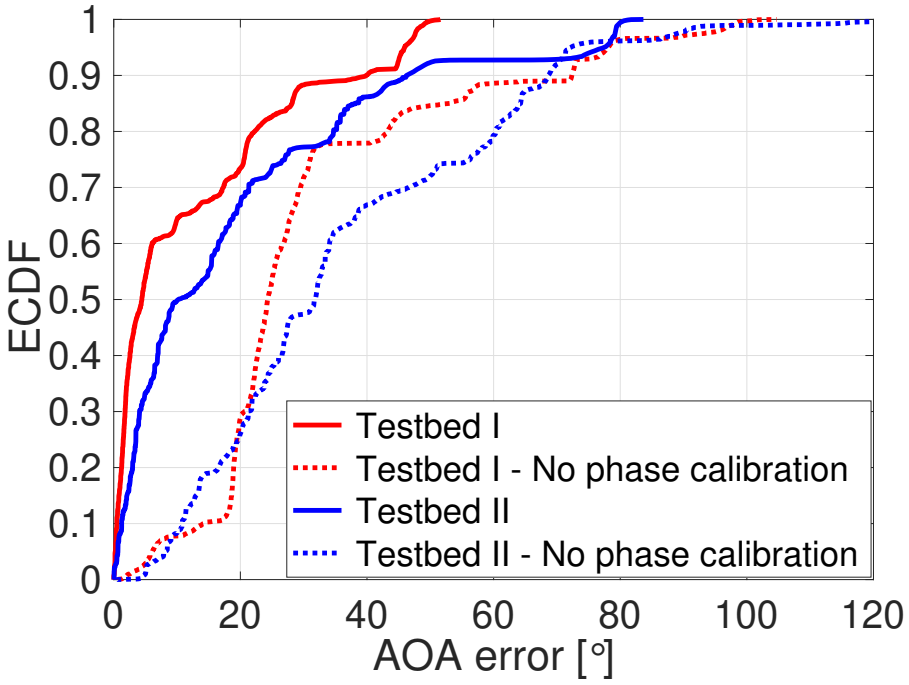


Figure 3.17: AOA.

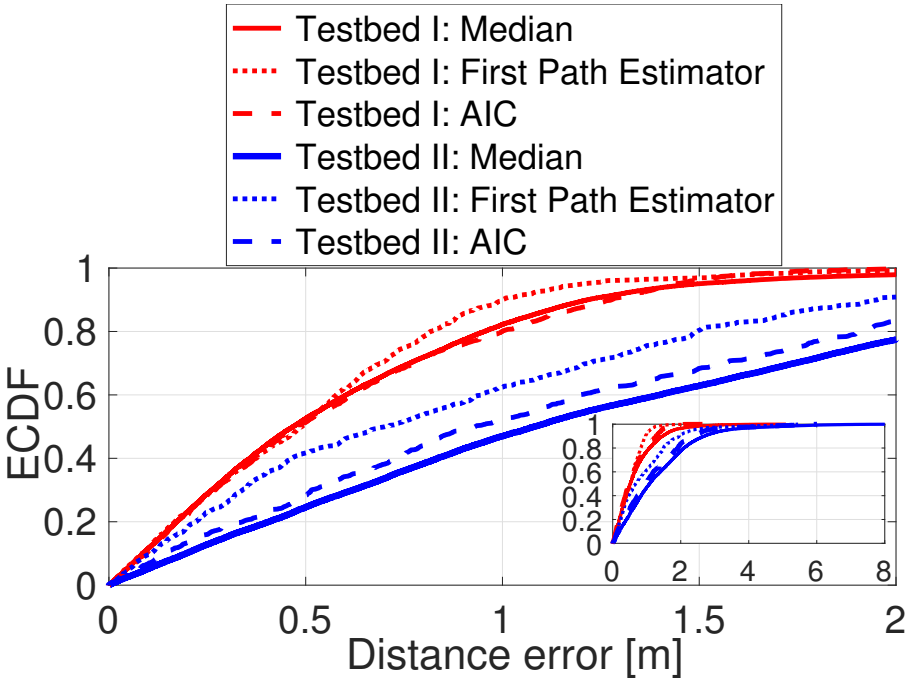


Figure 3.18: Distance.

Figure 3.19: ECDF of AOA estimation error in degrees with and without phase calibration (3.17) and distance estimation error (3.18) for our estimator compared to the median and AIC.

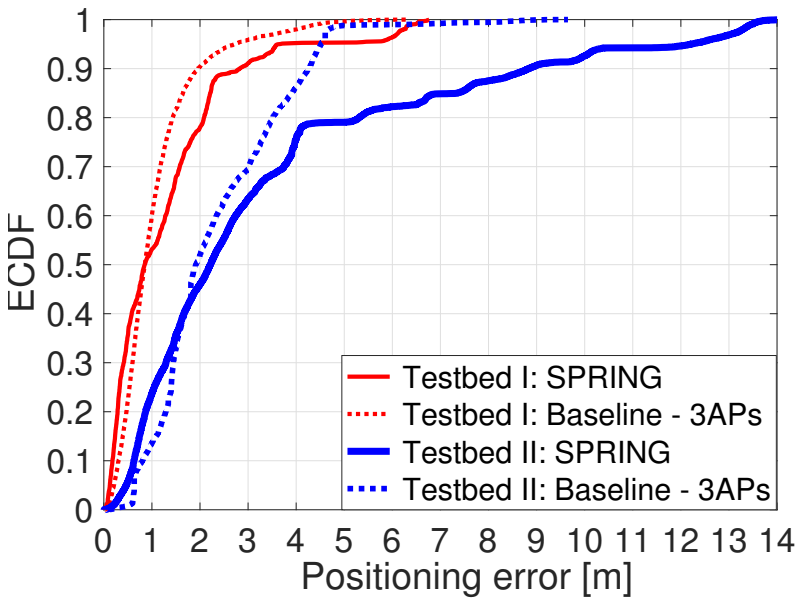


Figure 3.20: Positioning error in Testbed I and II. Solid lines indicate SPRING, while dashed lines correspond to the baseline.

to the baseline. For Testbed I, SPRING achieves a median and 80-percentile positioning error of about 0.9 m and 2 m, respectively, while 2.15 m and 5 m for Testbed II. On the other hand, we observe that the baseline system accuracy tends to increase compared with SPRING, above 60 percentile. In fact, for Testbed I, the baseline achieves a median and 80- percentile positioning error of about 0.9 m and 1.4 m, respectively, while 2 m and 3.5 m for Testbed II. The reason is that, for a given deployment scenario, the density of APs plays an important role, avoiding performance degradation as the distance from the AP increases. Nevertheless, SPRING, with just one AP, has similar median positioning error with the respect to a system that requires three FTM APs.

3.5. Discussion

We have presented a hybrid system that methodically addresses many of the challenges towards practical WiFi-based positioning using only a single AP. We have exploited PHY layer information to estimate the direction between the AP and the client using a 802.11ac WiFi chipset that uses 4 linear antennas, operates at 80 MHz and gives access to CSI per sub-carrier. We have presented a phase calibration technique to correct the estimation error introduced by the hardware in the AOA estimations. We have shown how to apply our calibration and the results in the anechoic chamber. Then we have developed an AOA estimator based on channel state information and we have proposed a methodology to alleviate the effect of multipath in FTM-based ranging using inputs from CSI. The estimator achieves a median error of 0.5-0.7 meters. Finally we have combined both

information, direction and distance between the AP and the STA and have shown for the first time the ability to position a COTS smartphone only with WiFi measurements performed by one single AP that uses commodity hardware. The experiments have shown a median 2-D positioning error between 0.9 and 2.15 meters in two different testbeds, in areas up to 125 m^2 and in presence of strong blockage and multipath.

Part III

Context Awareness and applications

We have seen so far how reasonable positioning accuracy in indoor environment is achievable using different techniques. At this point, we wonder what possible benefits of an indoor localization system beyond applications such navigation service in museum, shopping centers and hospitals to provide. Localization may be exploited not only as a service offered to customers, but also in the network core to support anticipatory networking, enabling reliable mobile communications via advanced resource management policies and adaptive traffic engineering strategies. Position could provide a much greater benefit to network applications than what done so far. For this reason, in this part of the thesis we investigate how context awareness could give the opportunity to explore benefits of the network itself or physical behaviours of the user.

Nowadays, the advent of the fourth Industrial Revolution (Industry 4.0) requires wireless networked solutions to connect machines. However, the industrial environment is notorious for being adverse to wireless communication, with traditional wireless resource mechanisms prone to errors. In a very harsh environment radio communication has notorious difficulties, as metallic objects create a strong blockage component and surfaces are highly reflective. In chapter 4, we consider an environment strongly affected by the presence of metallic objects and we revise the Time-of-Flight (ToF)-based wireless indoor navigation system, presented in chapter 2. We provide a robust system architecture for near real-time positioning, solving stability problems for distance and positioning estimates in industrial scenarios, showing acceptable positioning accuracy. Based on these results, we also propose a location-aware Medium Access Control (MAC) protocol, that leverages hypothesis test to declare if the link is in Line-Of-Sight (LOS) or Non-Line-Of-Sight (NLOS), to alleviate blockage and severe multipath in industrial-like scenarios. We finally integrate both localization system and the location-aware MAC protocol in the same architecture, which fully supports hybrid (centralized and distributed) control and network intelligence. Doing so, we investigate how to dynamically allocate MAC resources to target devices, based on location data. The latter can be also used to estimate device movements, as done by physical inertial sensors embedded in mobile devices, such as accelerometers and gyroscopes.

As inertial sensors have shown great potential, in chapter 5, we propose to estimate the device movement without any access to physical inertial sensors in the mobile. The idea is to infer the movements of the mobile through radio measurements, a concept we call "virtual inertial sensors". We show that for smartphones, Access Point (AP) side Channel

State Information (CSI) is not suitable to estimate the rotation of mobile terminals. We then propose a method for estimating the rotation of a user that uses only Wireless Fidelity (WiFi) Fine Time Measurements (FTM) to infer the rotation speed. While FTM works with only one single antenna, it achieves better performance than a CSI-based estimator that exploits four antennas and multiple sub-carriers at the AP, but are yet limited by the typical one single WiFi antenna at the smartphone side. Together with walking speed estimation of a user, we can envision that virtual inertial sensors can be leveraged by location systems and sensing mechanisms to improve localization accuracy, infer user behavior, and design better and more secure communication.

In this part of the thesis, as for part II, we carry out performance evaluations using experimental studies and commodity hardware only.

4 Location-aware Wireless Resource Allocation

The advent of Industry 4.0 solutions to automate manufacturing technologies [55] is challenging the way machines execute complex tasks. Machines are becoming more autonomous, flexible and cooperative, and wireless networked solutions could ideally greatly help to increase the productivity in these environments. However, environments strongly affected by the presence of metallic objects, such as Industry 4.0 deployments, challenge the reliability of wireless communication. In turn, the mere usage of wireless protocols that have been designed for more traditional environments (home, office, shops, etc.) results in networked solutions that are prone to error in harsh environments. As of today, there is a limited investigation of tailored wireless networked solution for challenging and harsh environments.

Pervasive positioning is a cornerstone to enable several data analytics and applications, and in this work we investigate its potential for networked solutions in harsh enviroments. While Location-Based Services (LBS) providers are ready to exploit new and better position information for data analytics and personalized services, the potential for networks applications of positioning data remains largely unleashed. Positioning data may be exploited not only as a service offered to customers, but also in the network core to better allocate network resources based on the expected link performance.

The idea is to take advantage of positioning data, and more in general context information, to enable reliable mobile communications via advanced resource management policies and adaptive traffic engineering strategies. As of today, experimental evaluation in this research field is still in its early stage, as it is requires the integration of several network and positioning software and hardware components involving a large scientific and engineering effort. As a result, there is limited experimental understanding of what is possible to do coupling position and communication.

Of particular interest for this chapter is the Medium Access Control (MAC) protocol, that serves a vital role in every network. It is directly responsible for controlling access to the shared communication resources. In most cases, the network designer does not know about the network conditions and has to assume that they may change during

operation. The traditional approach in MAC protocols to handle unknown or changing conditions is to provide an adaptation mechanism in order to adjust the operation to the actual network load and signal-to-noise ratio (for instance, using a different modulation scheme), and recover from failures in data transmission (for instance, re-transmitting after a longer back-off period). However, MAC designers typically ignore the possibility that the wireless channel could undergo total blockage, and recovery from such cases happen only at higher level through an inefficient handover procedure that attempts to connect to the client with another Access Point (AP) in the network.

We investigate how to dynamically allocate MAC resources to target devices, based on location data extracted from a positioning and communication system. In order to avoid changes to the client side and operate with a single interface radio, we consider essential to use the same wireless network both for positioning and scheduling. Our positioning solution uses uses two-way Time-of-Flight (ToF) measurements to compute the ranges from APs to targets, and a first version was presented in [56].

For such very harsh environment, our contributions are:

- We present our positioning system ToF-based Wireless Indoor Navigation System (TWINS), that improves the version first presented in 2 and provides a robust system architecture for near real-time positioning. Our technical innovations with respect to our past work are: (i) time scheduler strategy for APs to increase the convergence time of positioning algorithm; (ii) ranging estimation technique that estimates the first path and solves stability problems of the distance estimate over time caused by over-estimating the number of paths; (iii) weighting function designed to prioritize APs, and provide stable distance estimates.

- We exploit the spatial geometry of the deployed APs and a statistical model to map the spatial likelihood of target device position to an angle error distribution. We derive a hypothesis test to declare if the link is in Line-Of-Sight (LOS) or Non-Line-Of-Sight (NLOS), and we propose a location-aware MAC protocol that leverages our hypothesis test to alleviate blockage and severe multipath in industrial-like scenarios.

- We integrate both TWINS and our location-aware MAC protocol in the WiSHFUL architecture [57], which fully supports hybrid (centralized and distributed) control and network intelligence, and whose environment presences strong metallic objects as in industrial scenarios.

Our experimental results show that TWINS can track simultaneously 4 mobile robots, with median error between 1.8 and 3.8 m. TWINS greatly outperforms the best average error of state-of-the-art systems in the same environment, that was 6-7 m of average error for a single target device. Our statistical angular information can help increase the network throughput in industrial-like scenarios for mobile (robots)

devices. The experimental results show a throughput gain of 15% and 40% for Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and Time-Division-Multiple Access (TDMA) protocols, respectively.

4.1. Motivation

The environment under study (w.iLab.2 testbed[1] [57]) is full of metal objects which block Radio Frequency (RF) signals and cause strong reflections, impacting on the quality of wireless communication. We target the analysis of MAC scheduling strategies in this environment and want to study whether location information can be used to optimize MAC scheduling decisions. The fundamental questions we want to investigate are:

- Can we achieve acceptable positioning accuracy with legacy wireless communication technology in an industrial environment?
- Can we improve network performance integrating positioning data in the MAC scheduler in mobile scenarios?
- There is very limited experimental work on MAC scheduling strategies that exploit a prototype location system. How well experiments can help us designing better MAC schedulers?
- What gain is expected with respect to a classical approach without location and context knowledge, given that location information is far from perfect and it is subjected to position error?

We start introducing in Section 4.2 the positioning system used in this work. Section 4.3 describes our context-aware MAC protocol. We then present the whole system architecture in Section 4.4, and the testbeds in Section 4.5. We show the experimental evaluations in Section 4.6, and we draw the conclusion in Section 5.4.

4.2. TWINS

In this section we introduce TWINS, our legacy Wireless Fidelity (WiFi) positioning system, starting with the AP orchestration and then presenting the position algorithm.

4.2.1. Access Point orchestration

Our system uses commercial off-the-shelf (COTS) APs with customized firmware operating in the core of the 802.11 MAC state machine of a low-cost WiFi chipset. TWINS can estimate the position of any 802.11 standard-compatible devices. TWINS can also

[1] http://doc.ilabt.imec.be/ilabt-documentation/

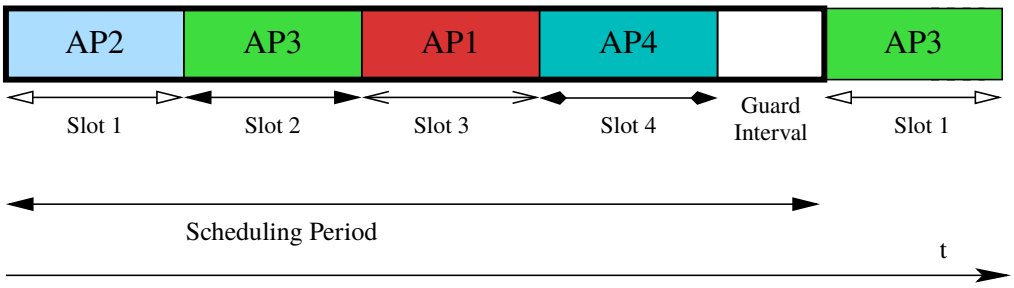


Figure 4.1: Measurement scheduler and slot allocation

be used for communicating to WiFi devices, as it does not require any changes to the WiFi protocol. As we have seen in chapter 2, the APs are equipped with Broadcom WiFi chipsets that run our customized version of the 802.11 Open FirmWare for WiFi networks (OpenFWWF) firmware.

We measure the ToF to the mobile device(s) with two-way ranging measurements using Probe Responses, an 802.11 standard compliant solution. In particular, the target devices are associated to any of the ranging APs by means of a specific Service Set IDentifier (SSID) common to the whole set of APs. No modification to the clients are required in TWINS. ToF measurements are passed to the open-source b43 driver[2] running in the AP, where they are pre-filtered. Data are then sent to the Central Location Unit (CLU) that orchestrates the network. The CLU computes the mobile position and store the results to a database.

Central Location Unit. The positioning system is orchestrated by the CLU, which manages the APs, generates traffic to which target devices respond, and processes location data. APs operate in the same radio frequency channel as the client, and APs may not be in range with each other. For this reason, the measurements are scheduled by the CLU to alleviate network collisions and hidden nodes among transmissions from different APs. As shown in Fig. 4.1, the CLU uses a time division scheduler to issue measurements rounds where only one AP at each time has the token to measure the ToF to target devices. Together with the time division scheduler, we further randomize the AP slot assignment, within the scheduling period to compensate potential asymmetries due, for instance, to the target movement or hidden nodes effects caused by other networks. Because APs use the listen-before-talk 802.11 protocol, they cannot transmit with guaranteed delay. As such, we allocate a final guard interval to drain the measurement queue of all APs.

A report of the measurement round (series of ToF values per target) is sent by each AP at the end of the scheduling period through kernel socket. Once a full round of measurements is performed by each AP, they are sent to the CLU. Finally, the CLU estimates the distance from each AP to each intended target (Section 4.2.2), and then estimate the position of the target (Section 4.2.3).

[2]http://linuxwireless.sipsolutions.net/en/users/Drivers/b43/

4.2.2. Ranging measurements

The ToF range is computed using regular 802.11 Probe Responses sent by the APs and acknowledged by the target device via 802.11 Acknowledgment (ACK)s after a Short InterFrame Space (SIFS) time. We use Probe Responses rather than normal Data as we experimentally observe that the target device replies only to the Data of its associated AP, but not from other APs of TWINS. In contrast, the target replies reliably to the short Probe Responses sent by any AP, which is a requirement of our positioning system. In this way, ToF ranges can be computed from multiple APs to estimate the mobile position. In addition, in order to guarantee the highest reliability of range traffic from APs at any distance from the target, Probes are sent at the basic rate of 1 Mb/s.

In order to minimize the duration of Probes, we limit the payload size of Probe Response to 30 Bytes (a small SSID and no extra extensions used). There are 12 bytes in fixed fields (timestamp, beacon interval and capabilities), and 18 tagged, of which 9 bytes are taken by the SSID, 6 by the B mode data rates and 3 by the Direct Sequence (DS) parameters (channel used).

Distance estimation. For estimating the range to the target device on a given instant, we assume that, given N measurements, some of them follow the LOS path (or the shortest NLOS path in case the LOS path does not exist), and others have one or more NLOS paths. For each single path, the noise in the measurement is mainly due to the target device [58]. We consider a Gaussian distribution for the noise generated by the target replying with ACKs as experimental demonstrated in [56].

We model the sum of all these multipath components as a *Gaussian Mixture Model (GMM)* [59]. A key aspect of the model is to identify the number of dominant paths (clusters) κ. Increasing the maximum number of clusters increases the probability that the GMM identifies clusters without physical meaning. Based on our experience, this, in turn, causes instability in the ranging estimates over time and, hence, in the positioning algorithm. Therefore, we limit the number of clusters to 3. We generate all the GMM models for each $\kappa \in \{1, \ldots, 3\}$, by using the iterative Expectation-Maximization (EM) algorithm initialized by a K-means++ run. We infer the optimal κ for the GMM statistical model selecting the model with the lowest Akaike Information Criterion (AIC) for the N measurements [54]. We finally consider the mean of each of the k paths. Our estimator rejects the paths with negative means, as they do not correspond to a physical propagation path, and uses the *path with the least positive mean* as ranging estimate. No offline calibration is required by our positioning system.

4.2.3. Position estimation

Using the estimated distances, a position-wise suboptimal but computationally efficient Linear Least Squares (LLS) multilateration algorithm is used to make an initial

position estimate. Then the algorithm makes use the Weighted Non-Linear Least Squares (NLLS) technique to compute the position from this initial value.

We define a coordinate system on a two-dimensional map. To ease the notation, we consider only one target and a number of links equal to the number of APs in communication range, denoted as L. Let $\mathbf{s_i} = (x_i, y_i)$ be the position of the AP_i, $\mathbf{p} = (x, y)$ the position of the target device, and $r_i = \|\mathbf{s_i} - \mathbf{p}\| = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ the distance between the AP_i and the target device. Let $\hat{d}_i$ further denote the estimate of the distance from AP_i to the target. Our objective is to find the coordinates that solve the following weighted least square optimization problem:

$$\hat{\mathbf{p}} = (\hat{x}, \hat{y}) = \underset{x,y}{\operatorname{argmin}} \sum_{i=1}^{L} \frac{(\|\mathbf{s_i} - \mathbf{p}\| - \hat{d}_i)^2}{w_i} \tag{4.1}$$

where w_i is a weighting function of the samples $\{\hat{d}_i\}$. To solve eq. (4.1), we apply the Newton-Gauss method with line-search for the step size.

The weight is computed as the distance with respect to the previously estimated position, that is:

$$w_i = \|\mathbf{s_i} - \hat{\mathbf{p}}[k-1]\| - \hat{d}_i[k] \tag{4.2}$$

where k indicates the iteration step. The idea of this approach is that a higher weight is an indication of instability in the link between the AP and the target device, likely meaning that the link is affected by multipath. This is especially important for industrial environments, where metals can completely block the LOS path and the estimated distance will be higher.

4.3. Context-aware resource allocation

Once the positioning system is laid out, we use location data to elaborate mechanisms to allocate the network resources more efficiently. An illustration of the concept is presented in Fig. 4.2, where we consider a system where the network has access to the estimated User Equipment (UE) location for allocating the wireless network resources in time and space. We also consider that the location of metallic objects in the environment is known (for instance through access to the map). Each AP should avoid to transmit to a UE if it can anticipate that the UE is behind a metallic blockage as the link would be totally disrupted in these conditions.

In order to make decisions for allocating the network resources, in the ideal case of perfect knowledge of location position, the AP should "draw a line" between the AP and the UE, and verify if the metallic blockage is in-between. Since noise and obstacles affect the positioning system, the only line between the AP and the estimated UE location does not ensure that the UE is affected by blockage. For this reason, it is convenient to *map*

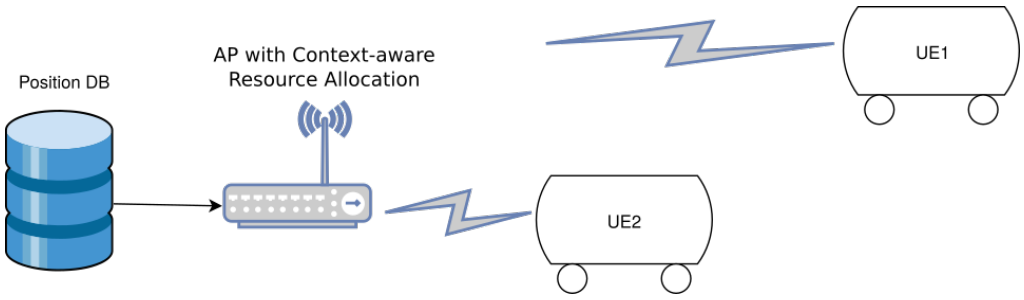


Figure 4.2: High-level illustration to exploit context-aware decisions in the allocation of network resources.

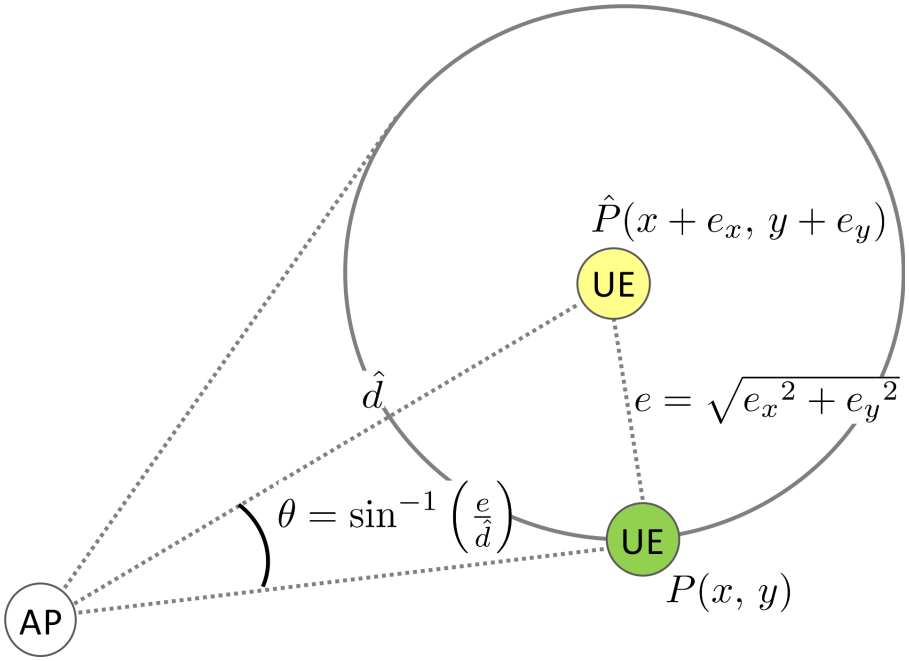


Figure 4.3: Mapping user position error to angle error.

position information, including the error, into angular information, as presented in the next sub-section.

4.3.1. Angle error model

We derive a statistical model of angle errors from the estimated location, spatial geometric information and the desired confidence level. The model will be used in Section 4.3.2 to detect if the UE is affected by blockage, going behind a metallic object.

From user position error to angle error. Let us refer to Fig. 4.3 where we consider a scenario with a fixed AP and a mobile UE. We assume a 2D Cartesian coordinate system. Extension to the 3D case is straightforward. The AP position is known and equal to $\mathbf{p}^{\mathrm{AP}} \in \mathbb{R}^{2\times 1}$. Given $\hat{x} = x + e_x$ and $\hat{y} = y + e_y$, the UE's real and estimated positions are $\mathbf{p}^{\mathrm{UE}} = P(x,\, y) \in \mathbb{R}^{2\times 1}$ and $\hat{\mathbf{p}}^{\mathrm{UE}} = \hat{P}(\hat{x},\, \hat{y}) \in \mathbb{R}^{2\times 1}$, respectively. The terms

e_x and e_y represent the location errors on the x- and y-axis, respectively.

Let us also define $\hat{d} = \|\mathbf{p}^{\mathrm{AP}} - \hat{\mathbf{p}}^{\mathrm{UE}}\|$ as the estimated distance from the AP to the UE, and $e = \sqrt{{e_x}^2 + {e_y}^2}$ as the UE position error. We can then introduce the angular region of width 2θ, given by

$$\theta = \sin^{-1}(e/\hat{d})\,. \tag{4.3}$$

Let us now define $p \in [0, 1)$ as the level of confidence of the position error. A location error e_p can be defined, which maps, in turn, to an angle error:

$$\theta_p = \sin^{-1}(e_p/\hat{d})\,, \tag{4.4}$$

which holds for $e_p \leq \hat{d}$, i.e., $\theta_p \leq \frac{\pi}{2}$. From eq. 4.4, a low p reduces the width of the angular region, but it increases the probability that the link detection fails.

Closed-form expression of location angle error. We derive a closed-form expression of the location angle error as a function of parameters that can be estimated by the positioning infrastructure and of the desired confidence level. Considering ranges with normal distribution (c.f. Section 4.2.2), we can assume that the position error is a bi-variate normal distribution, where the statistical processes e_x over the x-axis and e_y over the y-axis have no correlation, and that e_x and e_y have zero mean. We also define $\mathrm{dRMS} = \sqrt{{\sigma_x}^2 + {\sigma_y}^2}$ as the distance root-mean-square error, where σ_x^2 and σ_y^2, are the variance of e_x and e_y, respectively. Assuming that the sources of error in the x- and y-axes have the same statistical distribution, the statistical processes e_x and e_y have identical normal distributions with $\sigma = \sigma_x = \sigma_y$. It follows that $\mathrm{dRMS} = \sqrt{2\sigma^2}$. We can then resort to the error modulus $e = \sqrt{e_x^2 + e_y^2}$ to describe the position error, modeled as a Rayleigh Cumulative Distribution Function (CDF):

$$F_E(e) = Pr(E \leq e) = 1 - e^{-\frac{e^2}{2\sigma^2}} = 1 - e^{-\frac{e^2}{\mathrm{dRMS}^2}}\,. \tag{4.5}$$

The dRMS is often referred to as "63% error distance", meaning that 63% of errors fall within a circle of radius dRMS, i.e., $Pr(E \leq \mathrm{dRMS}) \approx 0.63$ [60].

The distance error CDF in eq. 4.5 can be mapped to the location angle error CDF as follows:

$$\begin{aligned} F_\Theta(\theta) &= Pr(\Theta \leq \theta) = Pr(\sin^{-1}(E/\hat{d}) \leq \theta) \\ &= Pr(E \leq \hat{d}\sin\theta) = F_E(\hat{d}\sin\theta)\,, \end{aligned} \tag{4.6}$$

which based on eq. 4.5 can be expressed as:

$$F_\Theta(\theta) = \begin{cases} 1 - e^{-\left(\frac{\hat{d}}{\mathrm{dRMS}}\sin\theta\right)^2} & \theta \leq \frac{\pi}{2} \\ 1 & \theta > \frac{\pi}{2}\,. \end{cases} \tag{4.7}$$

We remark that the angular error in eq. 4.7 is *not* a Rayleigh function because of the trigonometric function sin.

Computing $F_{\Theta}^{-1}(\theta)$, we can derive a closed-form expression of the location angle error θ_p as a function of geometric parameters and the desired confidence level $p \in [0, 1)$:

$$\theta_p = \sin^{-1}\left(\frac{\text{dRMS}}{\hat{d}}\sqrt{-\ln(1-p)}\right). \tag{4.8}$$

We can remove the dependence on dRMS by resorting to the Horizontal Dilution Of Precision (HDOP), used in geomatics engineering to measure the multiplicative effect of the geometry of the APs on the positioning accuracy based on the (known) AP coordinates [61, 62].

For the formal definition of HDOP, we denote

$$\mathbf{P}^{\text{AP}} = \begin{bmatrix}\mathbf{p}_1^{\text{AP}} & \mathbf{p}_2^{\text{AP}} & \dots & \mathbf{p}_N^{\text{AP}}\end{bmatrix} \in \mathbb{R}^{3\times N}$$

the matrix containing the AP coordinates. We can then define the matrix A containing the unit vectors of the direction between each AP and the UE:

$$\mathbf{A} = \begin{bmatrix}(\mathbf{p}_1^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}})/\|\mathbf{p}_\mathbf{1}^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}}\| & -1 \\ (\mathbf{p}_2^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}})/\|\mathbf{p}_\mathbf{2}^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}}\| & -1 \\ \vdots & \vdots \\ (\mathbf{p}_N^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}})/\|\mathbf{p}_\mathbf{N}^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}}\| & -1\end{bmatrix} \tag{4.9}$$

and formulate the matrix $\mathbf{Q} = (\mathbf{A}^T\mathbf{A})^{-1} \in \mathbb{R}^{4\times 4}$. The HDOP can be then computed as:

$$\text{HDOP} = \sqrt{\mathbf{Q}_{11} + \mathbf{Q}_{22}}\,. \tag{4.10}$$

For instance, when the visible APs are in the same line as the UE or close (as it would occur measuring the distance from multiple antennas in the same AP), the geometry is unfavorable for positioning and the HDOP value is high. In contrast, in the ideal case of perfect spatial geometry (for instance APs distributed in the corners of a square), the HDOP is close to one for most of UE locations. Using the HDOP, the dRMS can be expressed as follows [60]:

$$\text{dRMS} = \text{HDOP} \cdot \sigma_{\hat{d}}\,, \tag{4.11}$$

where $\sigma_{\hat{d}}$ is the standard deviation of the estimated distances for a specific location. Substituting eq. 4.11 into eq. 4.4 and computing $F_{\Theta}^{-1}(\theta)$, we can derive the following closed-form expression of the angle error θ_p:

$$\theta_p = \sin^{-1}\left(\text{HDOP} \cdot \frac{\sigma_{\hat{d}}}{\hat{d}}\sqrt{-\ln(1-p)}\right). \tag{4.12}$$

Estimation process of the angle error. From the above analysis, the estimation process of the angle error θ_p for a given confidence value p with respect to an AP, AP_n, operates as follows:

- Estimate the UE position $\hat{\mathbf{p}}^{\text{UE}}$;
- Compute the estimated distance $\hat{d} = \|\mathbf{p}_n^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}}\|$, where $\mathbf{P}^{\text{AP}} = \left[\mathbf{p}_1^{\text{AP}} \quad \mathbf{p}_2^{\text{AP}} \quad \dots \quad \mathbf{p}_N^{\text{AP}}\right] \in \mathbb{R}^{2\times N}$ is the matrix containing the AP coordinates, and $\sigma_{\hat{d}}$ the standard deviation over the observation period.
- Calculate the HDOP based on $\mathbf{P}^{\text{AP}}$ and $\hat{\mathbf{p}}^{\text{UE}}$ according to eq. 4.9;
- Derive θ_p based on eq. 4.12.

4.3.2. Blockage Detection

We then introduce a simple criterion to infer the link state. Knowing the real position of the metallic obstacles, let us define the hypotheses H_1 and H_2 as:

$$\begin{cases} H_1 : \text{“LOS”} \\ H_2 : \text{“NLOS”} \end{cases}$$

The test is as follows. Accept H_2 if both conditions below are fulfilled:

- The position of an obstacle falls within the angular portion $2\theta_p$;
- The estimated distance $\hat{d} = \|\mathbf{p}^{\text{AP}} - \hat{\mathbf{p}}^{\text{UE}}\|$ is higher than the radius from the AP to the metallic object.

Stay with H_1 otherwise. Given the strong link quality degradation in presence of metallic blockage, we allocate the wireless network resources only for those links that satisfies the hypothesis H_1 only. An example of the impact of the choice of the desired confidence level p is shown in Fig. 4.4. In fact, choosing a low $p = 0.1$ reduces the width of the angular region, but it may not be sufficient to detect the blockage (false negative detection of blockage). In contrast, a larger confidence level, as $p = 0.6$, results in a larger angular region, which includes the real UE position and hence the metallic obstacle.

4.4. System architecture

We integrate both TWINS (cf. Section 4.2) and our location-aware network resource allocation (cf. Section 4.3) in the WISHFUL architecture to investigate how location information can help MAC protocols. WISHFUL fully supports hybrid (centralized/ distributed) control and network intelligence [57], provided as an open-source solution, and it integrates several type of devices.

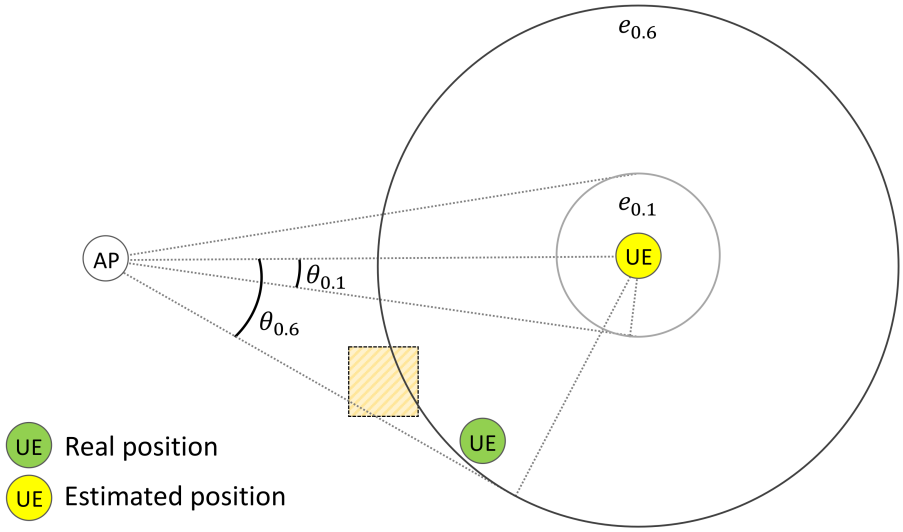


Figure 4.4: Impact of the choice of two different levels of confidence for the estimated position on the blockage management. Choosing $\theta_{0.1}$ ($p = 0.1$) leaves out the blockage (yellow box), causing false negative detection of blockage. A larger confidence level, as $p = 0.6$, results in a larger angular region, which includes the real UE position and hence the metallic obstacle.

The WiSHFUL control framework is based on a two-tier architecture which enables local, global and hierarchical control programs. Nodes can be monitored and controlled individually or in clusters, by exploiting control services devised to coordinate through Unified Program Interface (UPI) calls, a programming interface that abstracts from the physical device [63]. The UPI allows to make controlling programs independent from the device brand, model or even technology. Another significant aspect of WiSHFUL is the aim in reproducibility of results, as every experiment is programatically controlled in its entirety, even robot paths. As such, by integrating the positioning system, we can create reproducible location-based experiments for MAC scheduling.

WiSHFUL integrates multiple experimentation platforms for which a software architecture devised to simplify MAC or Physical Layer (PHY) protocol prototyping was already available. In this work we use the provided Wireless Mac Processor (WMP) platform [64]. The WMP platform was developed exposing an Application Programming Interface (API) for controlling the driver, by enabling the possibility to specify the configuration parameters of the WiFi chipset in a declarative language. The API also supports a time-based channel access scheme based on functionality developed under the API to enable a TDMA-based scheme. This is performed by specifying the time intervals (slots) in which nodes, specifically packet flows, are allowed to transmit running the usual Distributed Coordination Function (DCF) scheme. We enhance the TDMA mechanism in this work to allow for finer scheduling decisions, as explained next.

Control of TDMA resources allocation. In order to allocate MAC resources in

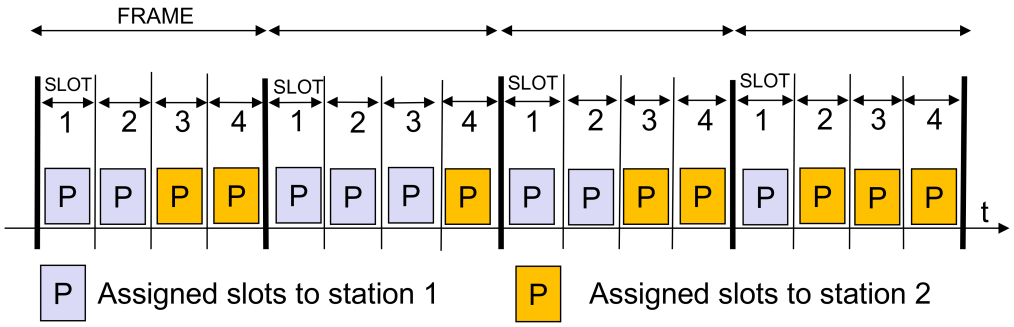


Figure 4.5: WMP TDMA access scheme with pattern slots definition.

time based on context awareness, we implement both global and local control program that use the WMP platform. The WMP implements both the standard 802.11 CSMA/CA as well as TDMA access protocol or radio programs. For both protocols, communication occurs in the unlicensed 2.4 GHz band to *unmodified* target devices.

In the TDMA radio program, we divide the channel access in periodic frames, and each frame in time slots. TDMA is a proven mechanism that can provide high throughput in high-dense environments. We activate each radio program after an explicit signaling from the control program used to transmit parameters to configure channel access scheme. The TDMA radio program has three main parameters:

- TDMA_SUPER_FRAME_SIZE - Duration of periodic frames used for slot allocations in us;
- TDMA_NUMBER_OF_SYNC_SLOT - Number of slots included in a super frame;
- TDMA_ALLOCATED_MASK_SLOT - Pattern of used slots in frame;

Figure 4.5 shows an example of 4 TDMA frames where two stations are active and each frame has 4 slots (pattern:"xxxx"). For instance, in the first frame, the TDMA_ALLOCATED_MASK_SLOT parameter of the station 1 is configured to use the slots 1 and 2 (pattern:"1100"), while the station 2 is configured to use the slots 3 and 4 (pattern:"0011"). The logic for activating the TDMA protocol and setting the relative mask pattern is embedded into the experiment control program.

Integration of TWINS. We integrate the positioning system to the WiSHFUL testbed (cf. Fig. 4.6). The CLU is our main process, which runs, separate from the APs, on a server. Upon a command from the CLU, the APs send probe packets to the target based on the measurement scheduler introduced in Fig. 4.1 and report the ToF to the CLU. Probes are generated in user space and sent to the virtual device driver operating in monitor mode through the pcap interface. The APs run the WMP firmware, which enables them to perform ToF measurements and pre-filter out invalid measurements such

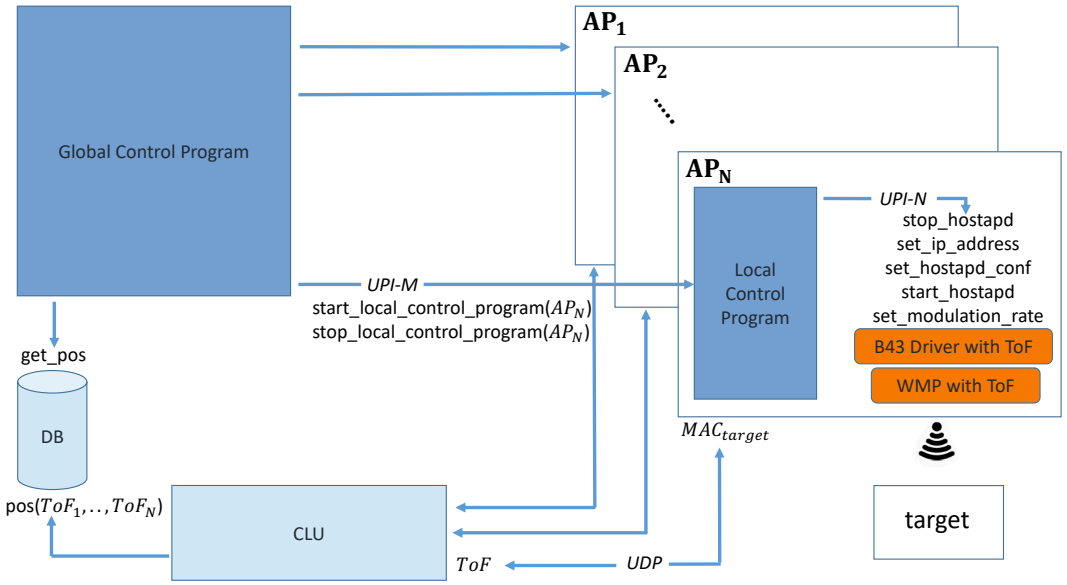


Figure 4.6: System architecture: Positioning system integrated in the WiSHFUL framework.

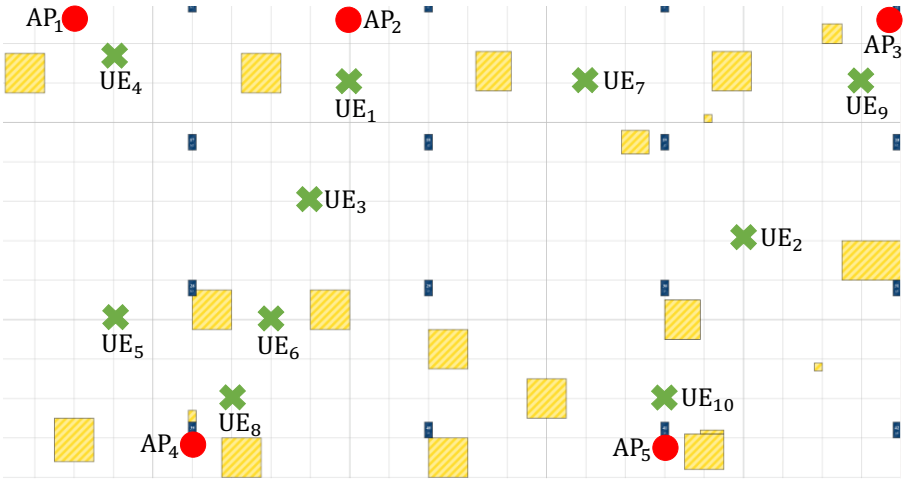
as values that can be considered outliers. The target is associated to one of the APs by means of a specific SSID common to the whole set of APs. As mentioned in Section 4.2.2, being connected to one of the APs does not refrain the device from responding on the probes from the rest of APs.

Once the CLU receives all the measurements, it runs a series of algorithms to generate position estimates for the target as presented in Section 4.2. The CLU stores positioning data on a Database (DB), so that they can be exploited by third parties. In our case, the context-aware MAC resource mechanism. We control the whole system by a Global Control Program (GCP) deployed in the same server as the CLU, and we run a Local Control Program (LCP) in each of the APs to handle the elements through UPI calls.

4.5. Testbeds

We perform experiments in the Testbed I that can be considered representative of an industrial indoor environment. A picture of the testbed is shown in Fig. 4.7(a). Testbed I presents open spaces surrounded by metal obstacles where radio communication has notorious difficulties. One of the key aspects is the fact that metallic objects create a strong blockage component. In Fig. 4.7(b) we show a scenario of the industrial testbed, where yellow squares indicate metallic objects. Additional metallic objects are tubes presented in the area (c.f. Fig. 4.7(a)). All APs use Alixes boards and UEs are all robots based on the Turtlebot II Robotic platforms. For further performance assessments, we also deploy and test our positioning system TWINS in another testbed (Testbed II).

(a) Real picture.



(b) Scenario I.

Figure 4.7: Testbed I: industrial environment. Yellow box corresponds to blockage.

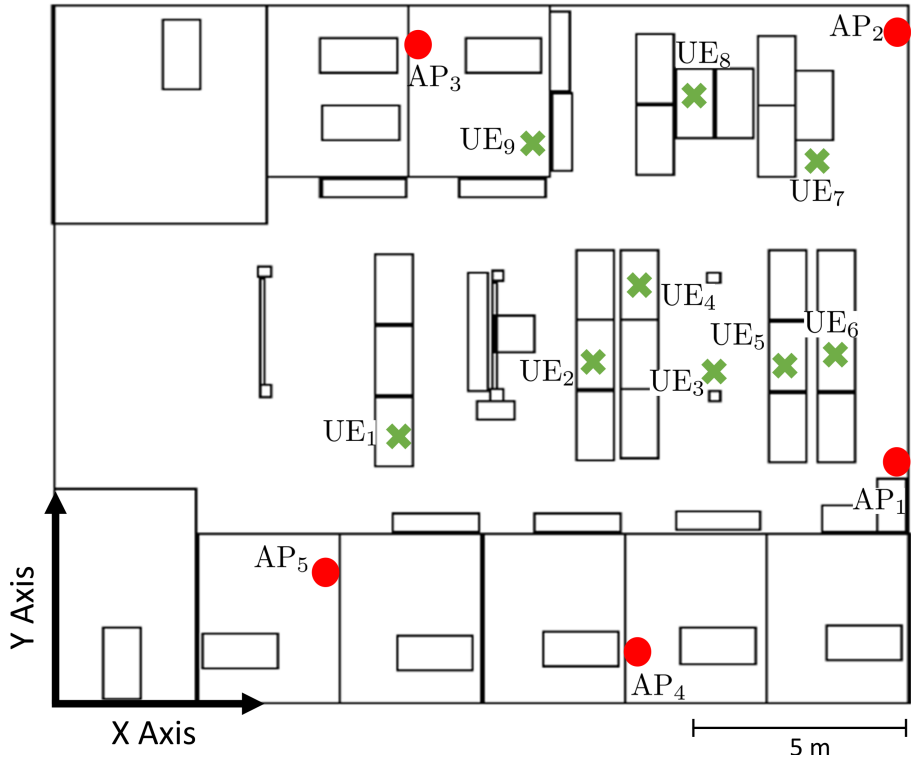


Figure 4.8: Testbed II: office environment.

The latter is an office environment covering a total area of 300 m^2, where concrete walls separate the office from the open area, and significant multipath is present. The map of the proposed scenario is shown in Fig. 4.8.

4.6. Evaluation

In this section, we analyze TWINS performance in static and mobile scenarios, we experimentally validate the angular error model, introduced in Section 4.3.1, and the criterion for blockage detection, introduced in Section 4.3.2. We then finally evaluate the exploitation of context awareness with CSMA/CA and TDMA MAC resource allocation.

4.6.1. Injection rate

The frequency to issue measuring messages is finite and is affected by different factors such as delay in communications (CLU towards AP, AP Operating System towards 802.11 firmware, collisions). To assess a good value for the injection frequency, we issue a series of tests, in which we observe the overall behavior at different injection rates. As seen in Fig. 4.9, smaller pauses between injections imply much larger delays since the measurement packet is injected (at Operating System (OS) level) until the report of ToF measurements is received. The drawback of this approach is that the CLU has less

control on the time it needs to wait before receiving a sufficient number of measurements to perform range estimates. We also observe from Fig. 4.9 that, under such conditions, it is not possible to keep up with the injection rate. If the AP cannot send all packets in its assigned slot, it will start using the slot assigned to other APs. This, in turn, will increase the probability of collisions, and, especially of hidden nodes.

Despite we could consider the 4 ms scenario the safest of all, 3 ms injection period is more practical. In fact, Fig. 4.9(b) reports a success rate of Probe-ACKs of 98.6% against 99.1%, shown in Fig. 4.9(c). Furthermore, Fig. 4.10 shows the ECDF of the delay for three different injection rates and, more specifically, it reports a 90%-ile delay of 97 ms and 90 ms for 3 ms and 4 ms injection period, respectively. Hence, the difference between those two cases is not significant in terms of delay and success rate, but in in case of 3 ms injection period we are able to collect more samples in a fixed time window. This is an advantage as TWINS aims to track users in real time.

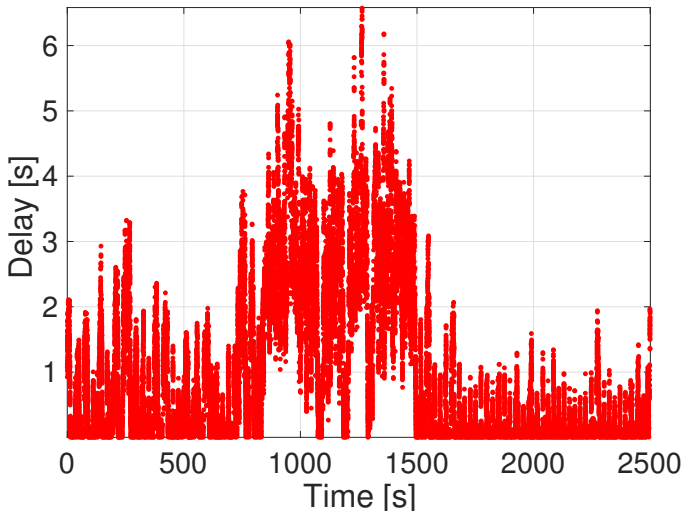
For the implementation, TWINS sends $N = 20$ Probes in each measurement round [56], which results in a slot time of $20 \cdot 3 = 60$ ms. In the example of Fig. 4.1, the Scheduling Period is $60 \cdot 4 + 30 = 270$ ms, where 4 is the number of APs and 30 ms is the fixed final guard interval (11% of the scheduling period in this example). The final guard interval is allocated to drain the measurement queue of all APs. Finally the experimental results from Fig. 4.9(b) show that 70% of the samples fall below the slot time of 60 ms, but the remaining 30% will surely occur after the allocated slot for the subsequent AP (but not the other APs).

4.6.2. Static Node

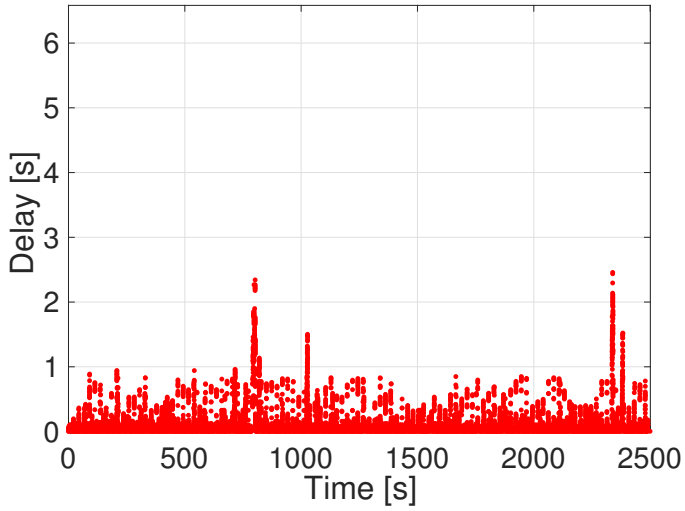
In this subsection we focus on static nodes only, showing the results for positioning, angular error model and blockage detection validation.

4.6.2.1. Positioning

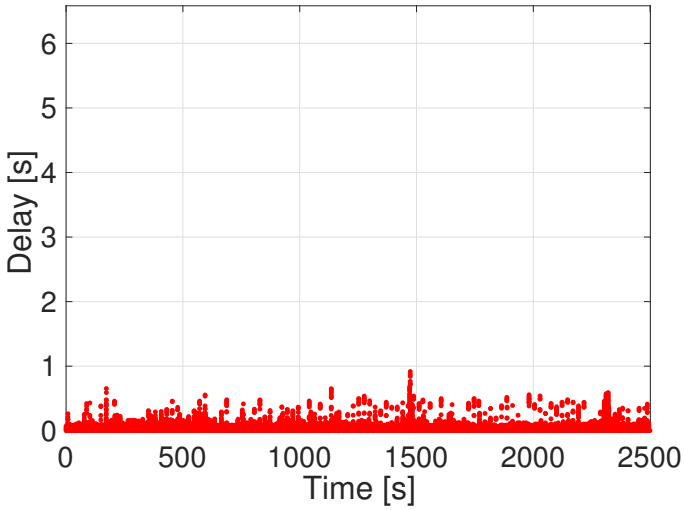
We first analyze TWINS performance in a static scenario of the proposed industrial environment, shown in Fig. 4.7(b). We locate the UE in 10 randomly selected locations (marked with crosses) and we use five APs to test our localization system. We study the position accuracy and summarize the results in Fig. 4.11. The red dotted line shows the ECDF of the positioning error for all the marked positions of Testbed I. The X-axis indicates the measured positioning error, and the Y-axis shows the cumulative probability (0.5 being the median). We obtain a median and 80-percentile positioning error about 4 m and 5 m. Furthermore, we compare the performance in such industrial environment with the one obtained in a normal office environment, Testbed II. The scenario is shown in Fig. 4.8, where we use five APs, as for Testbed I, and we locate the UE in 9 random locations. The blue solid line in Fig. 4.11 shows the ECDF of the positioning error for



(a) Period: 2ms, success rate: 91.7%.



(b) Period: 3ms, success rate: 98.6%.



(c) Period: 4ms, success rate: 99.1%.

Figure 4.9: Effect of different injection rates. Each figure shows, for an experiment in time (X-axis), the delay for the received packet in that instant (Y-axis).

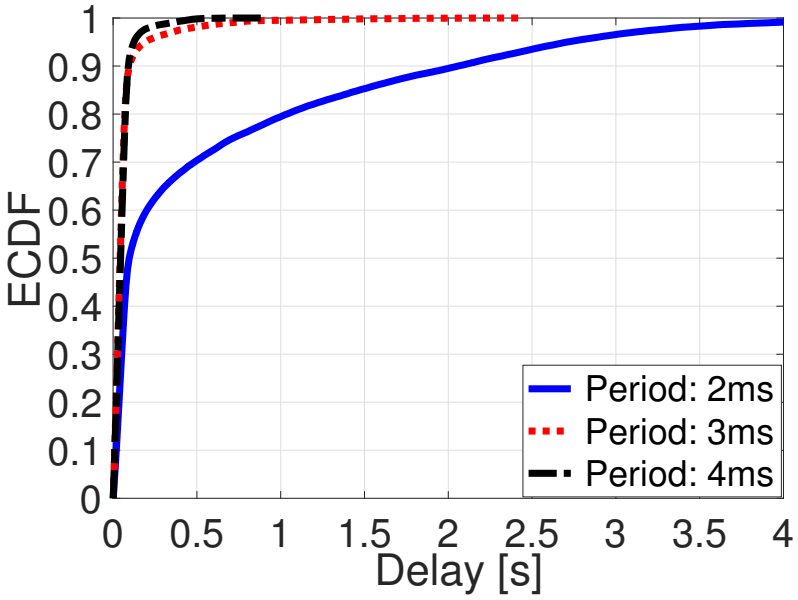


Figure 4.10: Empirical Cumulative Distribution Function (ECDF) of the delay for different injection rates.

all the marked positions of Testbed II. We obtain a median and 80-percentile positioning error about 2.1 m and 2.8 m, highlighting an improvement of about 50% with respect to the industrial environment.

In order to investigate more in detail TWINS performance in an industrial environment, we use a subset of this data to analyze the ranging accuracy under different propagation conditions. We show in Fig. 4.12 the ECDF of the distance error of one specific location (UE_4) with respect to the five APs. From the map in Fig. 4.8, we can observe that three links are LOS (UE_4 with AP_1-AP_2-AP_4) and two are NLOS (UE_4 with AP_3-AP_5). The results in Fig. 4.12 show that the blockage effects in the links AP_3-AP_5 to the target largely increases the median error of the distance estimate, without significantly changing the overall distribution. This effect is unique of industrial-like scenarios full of metal objects, as studied in this work.

4.6.2.2. Context-Aware Validation

After performing extensive WiFi localization measurements in Testbed I and II, we use these data sets to analyze the location angle error model, introduced in Section 4.3.1, and the criterion for blockage detection, proposed in Section 4.3.2.

Angle Error Model. We use the data set of Testbed II to analyze and validate the location angle error model introduced in Section 4.3.1. We first show in Fig. 4.13 the CDFs of the angle error at UE_1 and UE_9 locations using the model in eq. 4.12. For the study, we use as input i) the estimated AP-UE distance $\hat{d}$ from real experiments, ii) the standard deviation $\sigma_{\hat{d}}$ over the observation period, and iii) HDOP computed based on

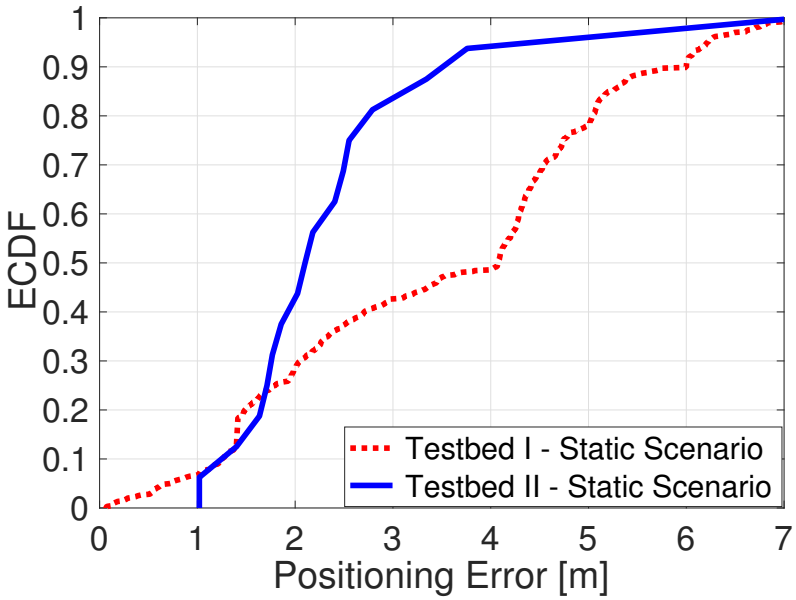


Figure 4.11: ECDF of positioning error in static conditions using 5 APs, 9 and 10 randomly selected locations, for Testbed I and II, respectively. Testbed I represents an industrial environment, while Testbed II an office environment.

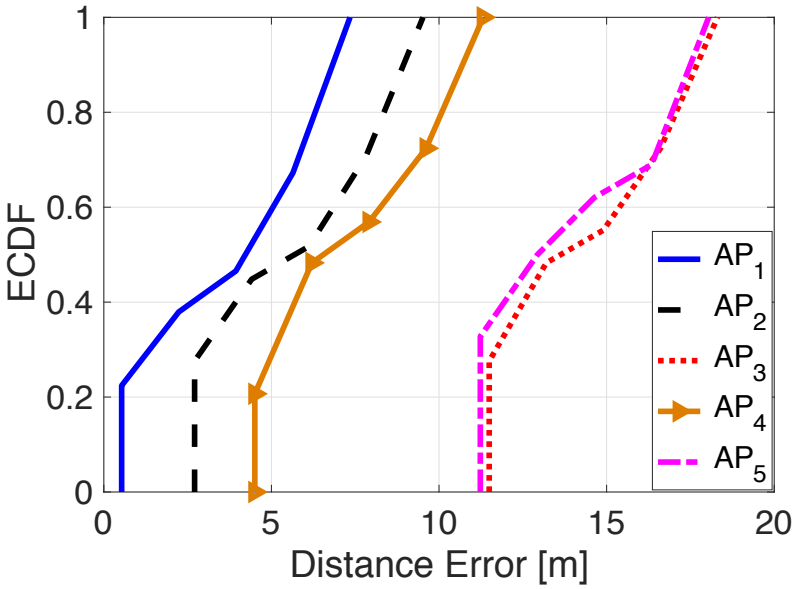
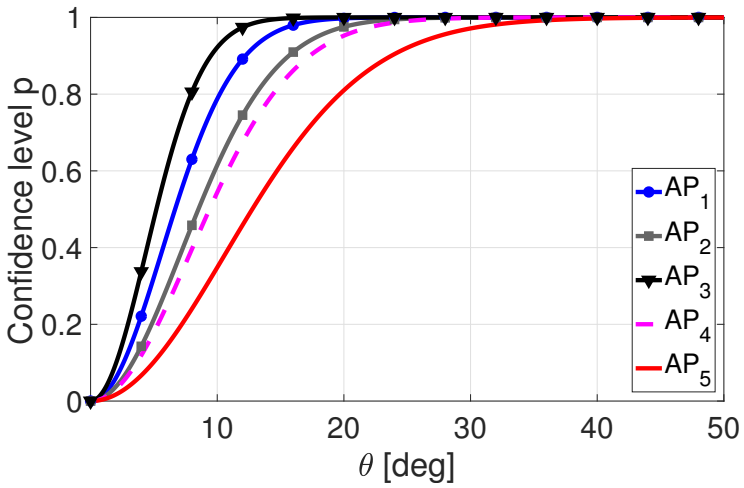
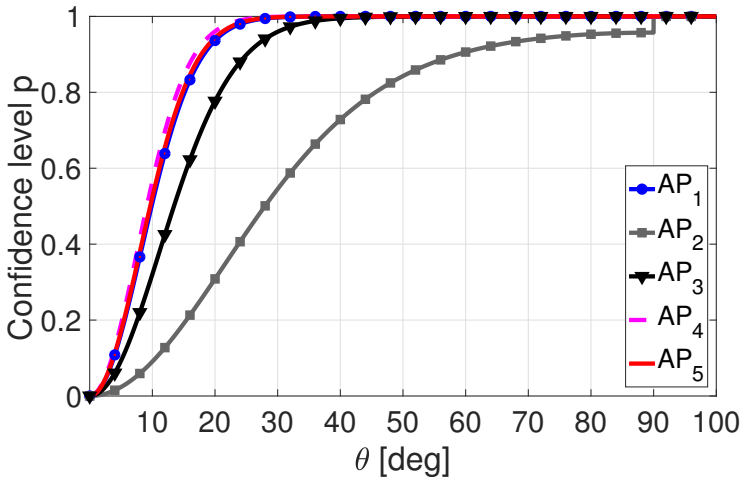


Figure 4.12: ECDF of distance error in static conditions for UE_4 only, in Testbed I.



(a) Model for UE_1



(b) Model for UE_9

Figure 4.13: Model CDF of the angle error, in Testbed II.

the known position of the five APs. The plots in Fig. 4.13 confirm the dependence of the CDF on $\hat{d}$ (c.f. eq. 4.7): the larger $\hat{d}$, the smaller the angle θ required to achieve a desired level of confidence. The results in Fig. 4.13 can be used to obtain θ_p. By selecting on the y-axis a desired level of confidence p for the position estimate, we can obtain the corresponding θ_p on the x-axis.

We then assess the validity of this approach for each AP and each p, where we compare the theoretical values computed using eq. 4.12 (as in Fig. 4.13) with the experimental distribution of θ_p. Using this data set, we then compute the median of the angle error between the experimental and the theoretical outputs across all UE positions, normalized with respect to the experimental results, and plot it in Fig. 4.14. We can observe that our location angle error model matches very well the experimental findings for a p level higher than 0.1. In absolute terms, our statistical model provides a median location angle

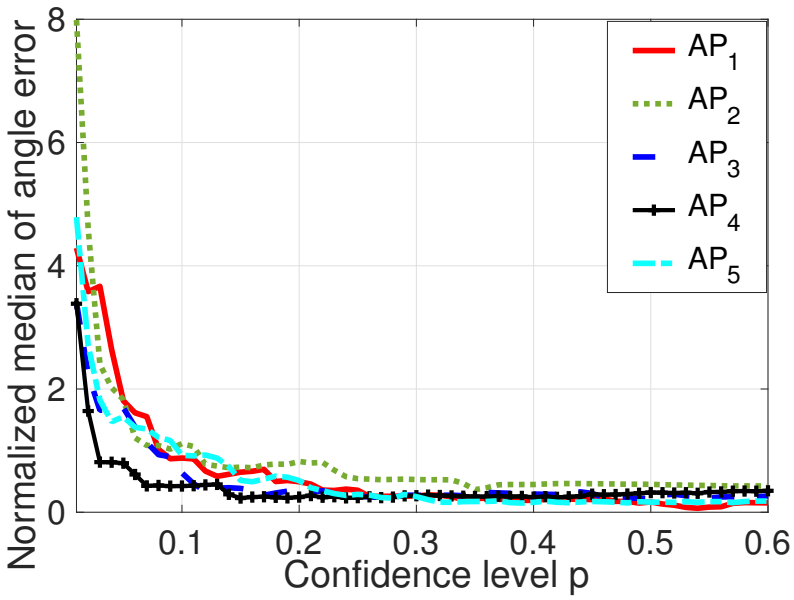


Figure 4.14: Location angle error in Testbed II for all confidence levels: low error is observed between experimental results and theoretical outcomes.

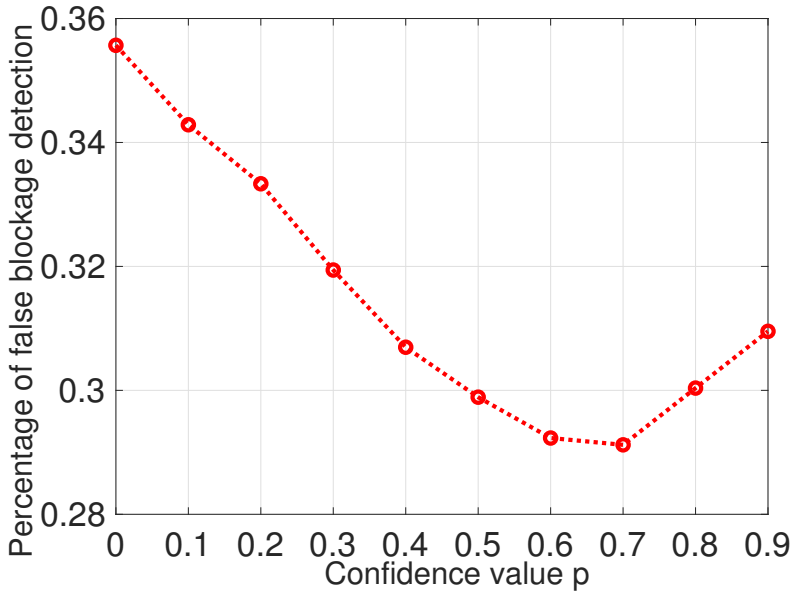


Figure 4.15: Percentage of false blockage detection for all confidence levels, in Testbed I.

error of 1.44° with $p = 0.1$ and 2.21° with $p = 0.4$ with respect to the measured one. This shows that multipath is efficiently handled by TWINS, and it does not affect significantly the validity of the model.

Blockage Detection. We use the data set of Testbed I to analyze the proposed criterion for blockage detection, introduced in Section 4.3.2. For this study, we use as input the estimated positions and their theoretical CDFs of the angle error, based on eq. 4.12. We then experimentally evaluate the impact of the choice of the confidence level p, based on the percentage of false blockage detection. The latter is shown in Fig. 4.15 for all confidence levels p. We can observe that reducing the level p, and so the width of the angular region, we increase the probability to have false negative detection of blockage. In contrast, increasing the level p, we include the metallic obstacle within a larger angular region, decreasing the probability to have false blockage detection. More specifically, a confidence level $p = 0.7$ is the best choice for blockage detection in Testbed I. For this reason, we use this value for the dynamic allocation of MAC resources in Sec. 4.6.3.3.

4.6.3. Mobile Node

In this subsection we focus on mobile nodes in Testbed I, showing the results of TWINS for distance and positioning. We then evaluate the exploitation of context awareness for CSMA/CA and TDMA allocation.

4.6.3.1. Distance

We perform ranging measurements in mobile conditions. In the experiment, the robot moves toward the AP_1 at a speed of 0.16 m/s. The results are summarized in Fig. 4.16, where the x-axis shows the measurement round number for which the estimate is made, while the y-axis corresponds to the distance in meter. Each round is comprised of a small set of measurements (N=20) for which we consider the target is at a semi-static position. The dotted line indicates the ground truth, and different statistics and filtered results are shown, namely median and our GMM-based estimation. We notice that regardless of the selection of a maximum of three clusters to run the GMM algorithm (cf. Section 4.2.2), the GMM can still under-estimate the distance more significantly with respect to the median estimator. As the estimate is measured on a relative small number of samples (20), a filter is necessary to improve the estimate. We then apply an Exponentially Weighted Moving Average (EWMA) filter of the distance:

$$\begin{aligned} \bar{d_k} &= (1-\alpha)\bar{d}_{k-1} + \alpha\hat{d_k}, \\ s.t. \quad \bar{d_1} &= \hat{d_1} \end{aligned} \tag{4.13}$$

where $\bar{d_k}$ is the current distance estimate, $\hat{d_k}$ is the last measured distance and α is the filter weight. An EWMA filter puts more or less weight on historical values according

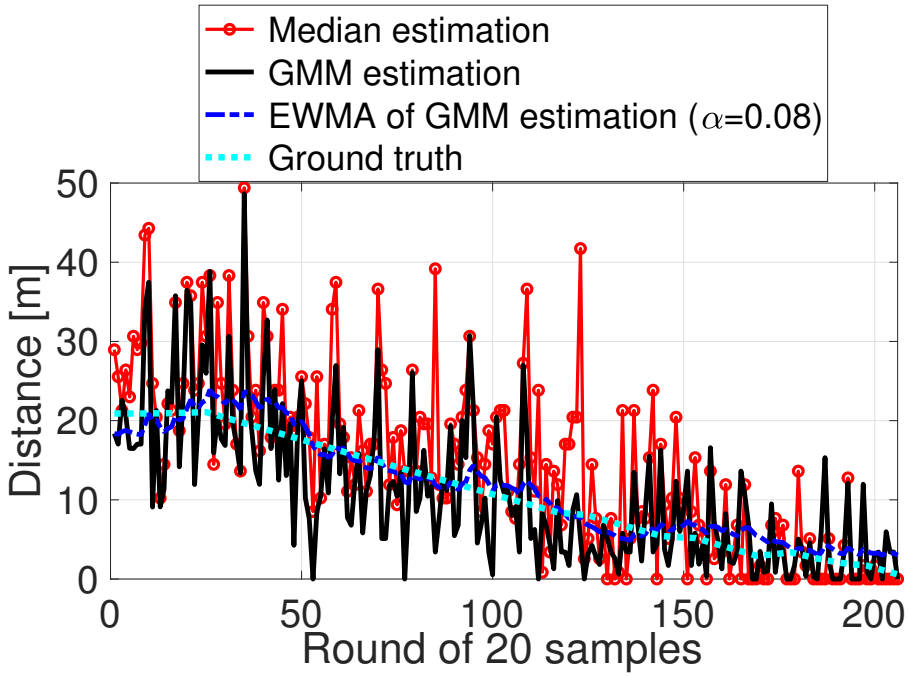


Figure 4.16: Ranging tests with mobile target.

to the filter weight α, which depends on the mobility of the target. The Fig. 4.17 shows the Root Mean Square Error (RMSE) of the GMM-based distance estimation versus α weight. At a speed of 0.16 m/s, α=0.08 is the best weight for our EWMA filter. We then can see in Fig. 4.16 that the EWMA metric estimates applied to the output of the GMM algorithm match very well to the ground truth. Even with the small time-delay caused by the EWMA filtering and the position mismatch (as we have a moving target, our estimate might be slightly delayed), the ranging error is limited, with the maximum error below 4 m. Hence the selection of a maximum of three clusters works well in practice with a low-pass filter. These values are very similar to those obtained previously by our system in other less challenging testbeds [56].

4.6.3.2. Positioning

We then make experimental tests for estimating the position of mobile robots. We run the experiment for four robots (9, 11, 13, 15) used as targets, along four different trajectories. Fig. 4.18 shows the ground truth (marked on blue continuous lines) and the position estimates (marked in red) for each trajectory.

Fig. 4.19 shows the ECDF for each of the given paths in the test areas presented above. We notice that the positioning error is within the given error figures given above. Intuitively it might seem that in some cases the error is higher than expected, but we remark that the scale may be different for each figure. The median error in the positioning estimate is below 4 m in the worst case, and in most cases below 3 m, for an area of 11 m

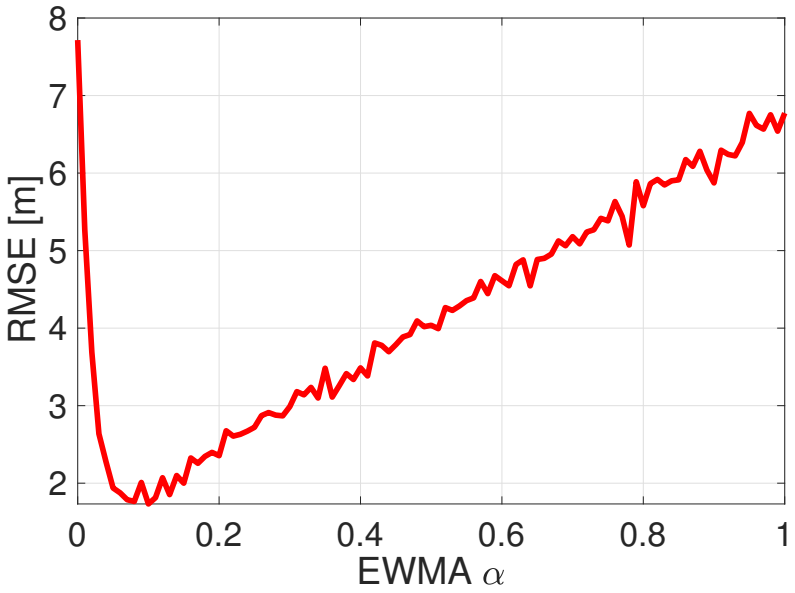


Figure 4.17: RMSE of GMM-based distance estimations versus EWMA α weight.

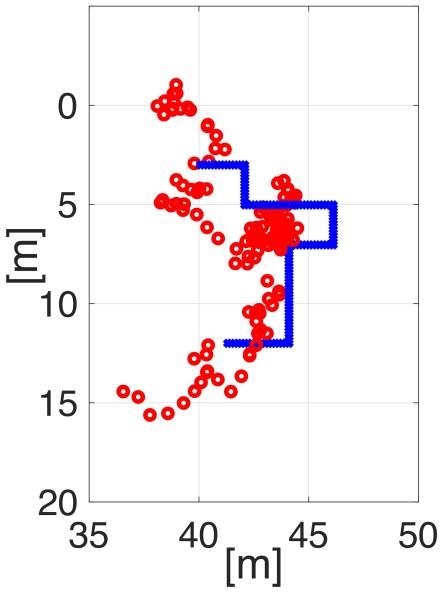
x 22 m. We observe that the *accuracy tends to increase compared with the static case*. The reason is that moving devices experience different multipath constellations, and w_i as defined in eq. (4.2) tends to increase for links that are more affected by multipath with respect to static links. The result is that the positioning algorithm makes better decisions in the weighting of each links and positioning computation. This effect compensates for the increased tracking error which arises when estimates are slightly lagging behind due to the measurement process delay, irrelevant when the device is at a static position.

Multiple Mobile Nodes. We finally run an experiment where we track all the robots simultaneously, on the given paths shown in Fig. 4.18. As main consequence of running the localization system with different targets at the same time, the latency of the system itself increases. Comparing the ECDF at 80% in Fig. 4.20 with Fig. 4.19, we can see a small degradation of the performances in terms of positioning error, which is to be expected as there are multiple targets.

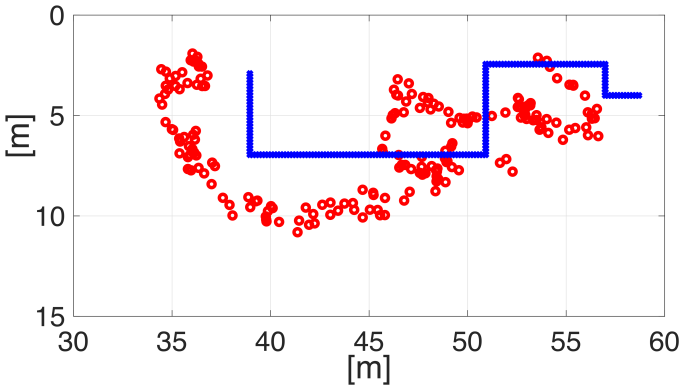
4.6.3.3. Exploitation of context awareness

Let us see now the effect of taking actions in the allocation of MAC resources.

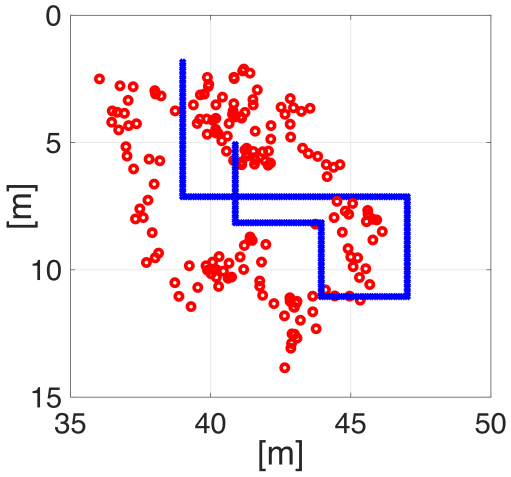
CSMA/CA. The first scenario of Testbed I considers a MAC based on CSMA/CA with two mobile robots. We focus our attention on a specific scenario, shown in Fig. 4.21(a), where the positions of UE_1 (robot 15) and UE_2 (robot 13) are in movement with respect to the AP. The robots are initially positioned on the spots marked with a green cross. From this position, each robot requests access to the 802.11 network and a reliable link is established. The robots move along their given trajectories, and due to the environment (mainly metallic obstacles, marked on the map as yellow rectangles), from a given position



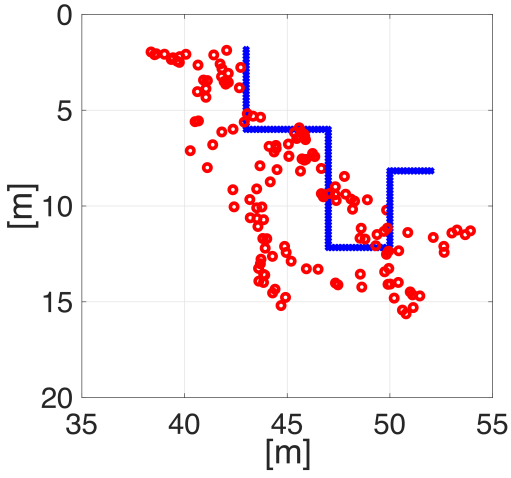
(a) Robot 9 trajectory.



(b) Robot 11 trajectory.



(c) Robot 13 trajectory.



(d) Robot 15 trajectory.

Figure 4.18: Real (blue) and estimated (red) trajectories in 4 different tests.

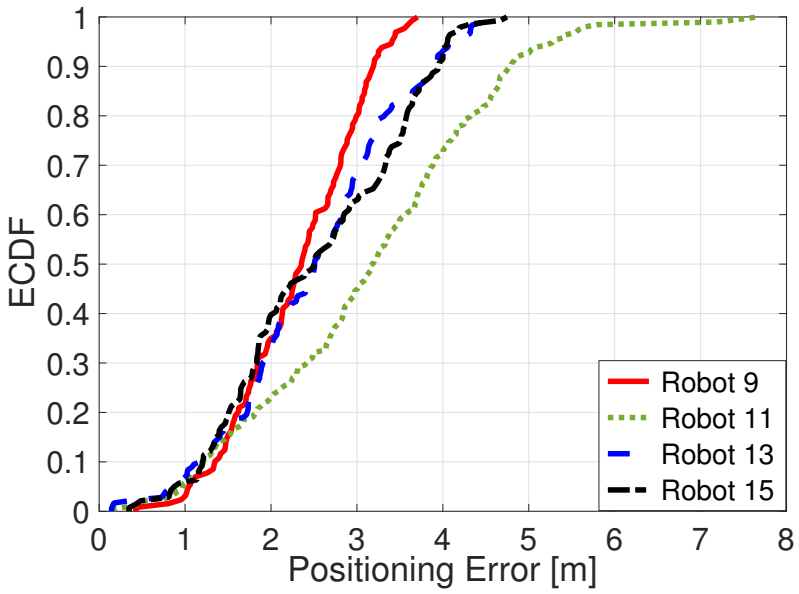


Figure 4.19: ECDF of positioning error, with robots localized in separate tests.

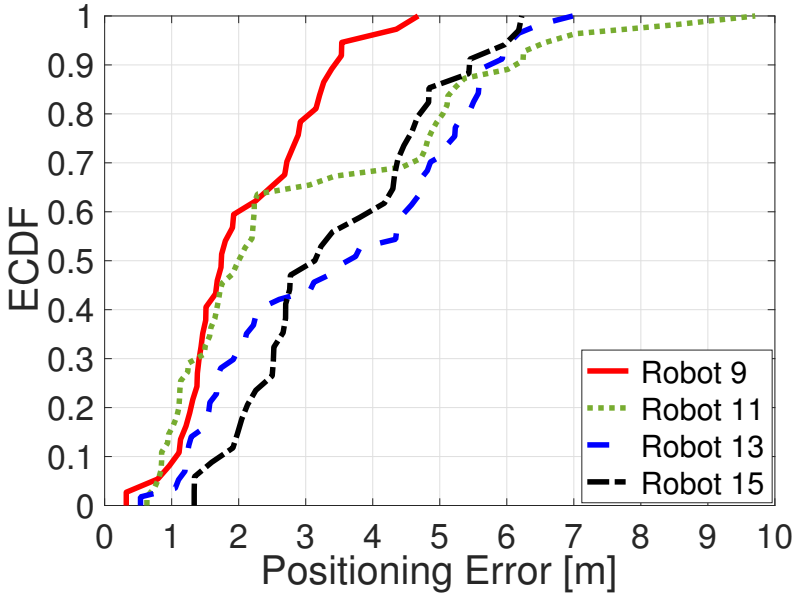
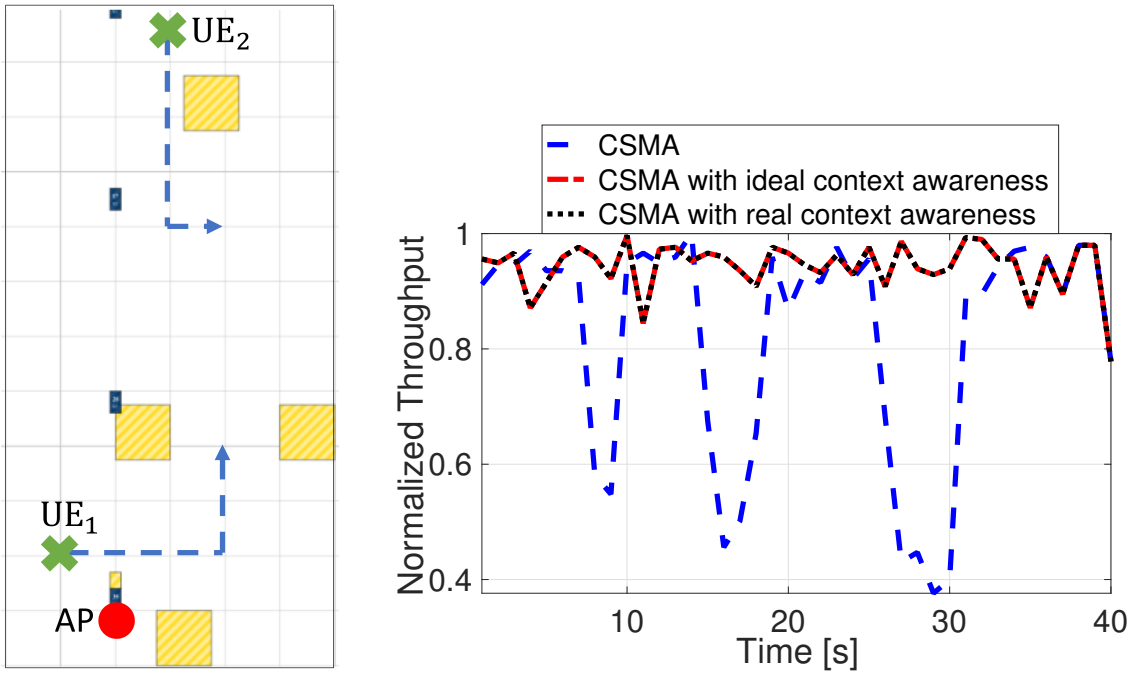


Figure 4.20: ECDF of positioning error, with TWINS localizing together all four robots.



(a) Scenario II.

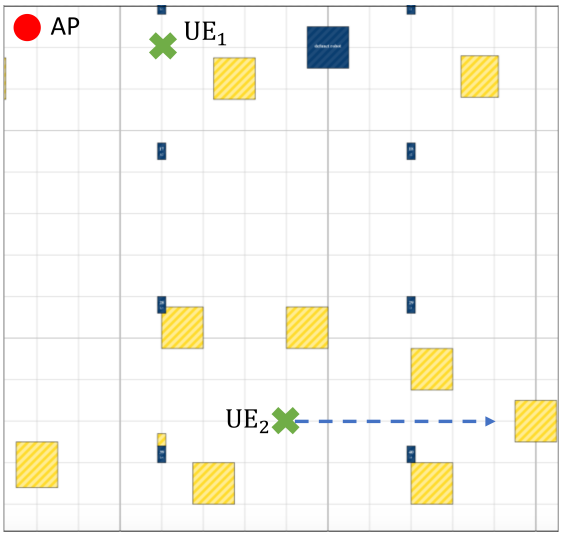
(b) Network throughput.

Figure 4.21: Allocation of MAC CSMA/CA resources in presence of blockage with different resources allocation strategies in Scenario II of Testbed I.

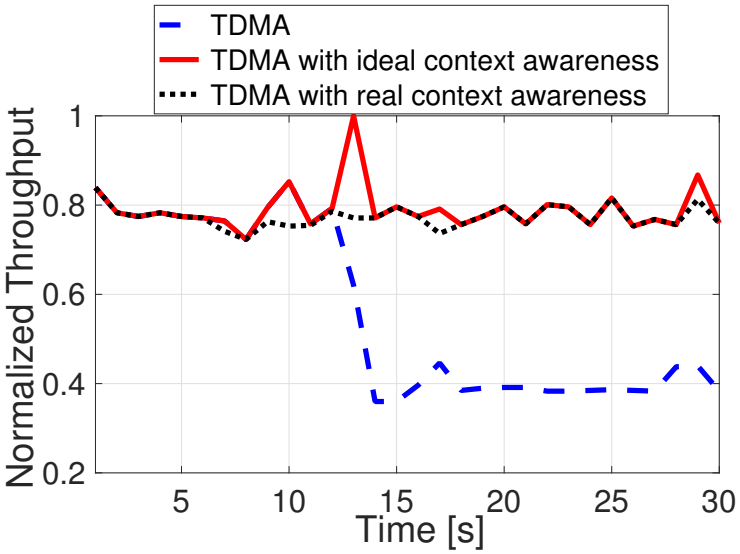
and on wards the link quality decreases, drastically reducing the network throughput. To avoid this degradation we apply the method presented in Sec. 4.3.2 to allocate traffic only for links accepting the hypothesis H_1.

Figure 4.21(b) shows the results in terms of network throughput, normalized with the respect to the maximum achieved throughput, along the 40-second UE_1 and UE_2 trajectories. Throughput is measured with the tool Iperf. Some time-aggregation effects may appear in the figure as the measurement cycles may not match each other. When using simple CSMA/CA, the performance presents a degradation in accordance with the radio link blockage. As we know the real position of the user along its trajectory (this is possible thanks to the knowledge of the actual position as provided by the high accurate Localization and Positioning Engine (LPE) in WiSHFUL), we can verify the performance of the algorithm in the ideal condition where the direct link for UE_2 and the AP is subject to blockage and without any error in the position estimation (an error is in reality still present, but ignored for simplicity in this ideal case). In case the link is in a NLOS state, we can immediately stop the traffic on that link, therefore maintaining the network throughput constant (red solid line, CSMA/CA with ideal context awareness).

Next, we use the positioning system as a source of context awareness information to compare a real-world scenario against the previous ideal scenario. In this case, we use the angular estimation, selecting a confidence level $p = 0.7$, as input to decide whether



(a) Scenario III.



(b) Network throughput.

Figure 4.22: Allocation of MAC TDMA resources in presence of blockage with different resources allocation strategies in Scenario III of Testbed I.

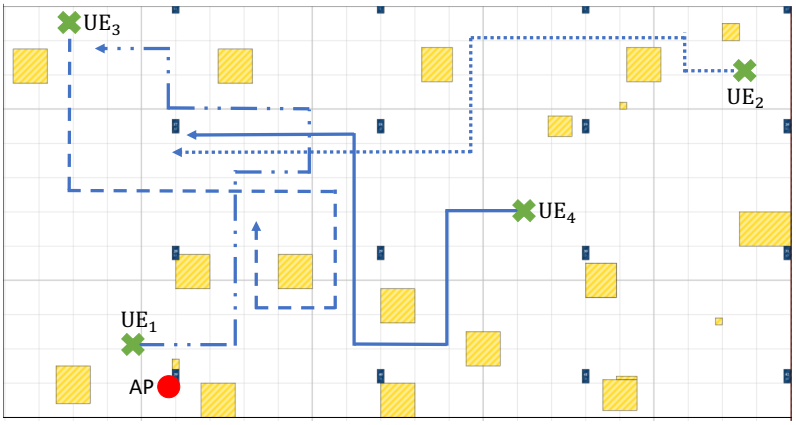
the UE is in a zone of low network coverage. The overall performance (dotted black line, CSMA/CA with real context awareness) in our experiment shows no difference between real and ideal results, suggesting that the positioning error is low enough for the link decision. In fact, pointing at the estimated position of the user and taking its angular error, we have that the real position is inside the angular error (UE_2 in Fig. 4.21(a)), and so we cluster the links as NLOS as the real case.

TDMA allocation.

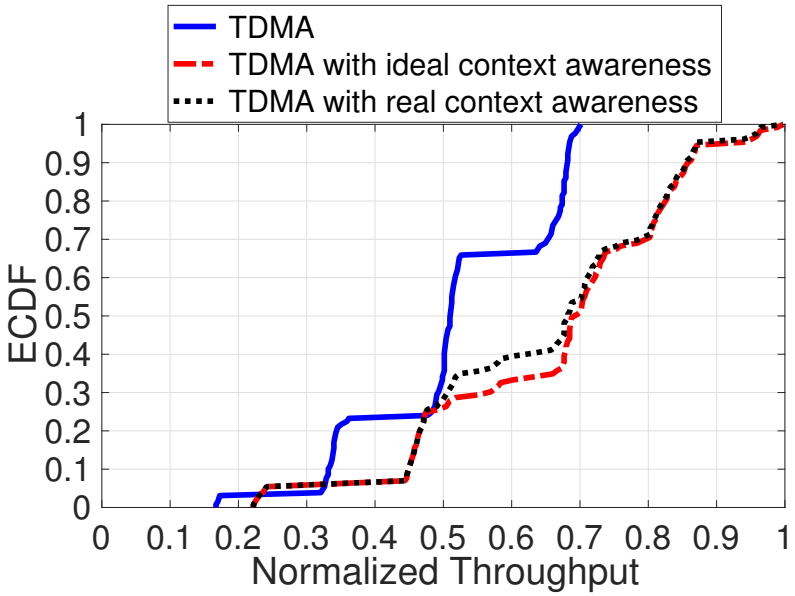
We then study a MAC-based on TDMA allocation in two different scenarios of Testbed I. Implementation-wise, in the latter cases we do not stop the traffic, but we allocate different time slots, using the implementation presented in Sec. 4.4. In particular, the control program configures the TDMA assigned slot based on context awareness extracted from the positioning system itself.

The scenario is illustrated in Fig. 4.22(a) and it considers a MAC-based on TDMA allocation with one static and one mobile node. Specifically, UE_1 (robot 15) is fixed, while UE_2 (robot 13) is in movement with respect to the AP. We analyze the normalized network throughput along a 30-second trajectory. From Fig. 4.22(b), we observe that using simple TDMA with a fair (equality-wise) allocation of the resources (50%-50% with 2 nodes, blue dashed line), the performance presents a degradation in accordance with the radio link blockage. As stated before, if we are aware of the provided context information, we are able to avoid the overall degradation, therefore maintaining an optimal network throughput. The overall performance (dotted black line, TDMA with real context awareness) shows the throughput improvement compared to the simple-TDMA setting, but this time it shows some performance loss compared to the ideal case due to estimation error in the positioning. Yet, the gain with respect to a simple TDMA allocation of 50%-50% is evident from the figure.

The last scenario is illustrated in Fig. 4.23(a) and it considers a MAC-based on TDMA allocation with four mobile nodes. We analyze the normalized network throughput along 130-second trajectories. We plot the ECDF of the normalized network throughput in Fig. 4.23(b). We observe that using simple TDMA with a fair (equality-wise) allocation of the resources (25% for each node, blue line), the ECDF presents a maximum normalized throughput close to 0.7, which corresponds to the median value of TDMA with ideal context awareness. Also this time the dotted black line (TDMA with real context awareness) shows the throughput improvement compared to the simple-TDMA setting, and at the same time it shows some performance loss compared to the ideal case due to estimation error in the positioning.



(a) Scenario IV.



(b) ECDF of the network throughput.

Figure 4.23: ECDF of the normalized network throughput with different MAC TDMA resources allocation strategies in Scenario IV of Testbed I.

4.7. Discussion

The goal of this work was experimenting context-awareness capable MACs in industrial-like scenarios. We have introduced our WiFi positioning system TWINS and we have presented our detection mechanism of blockage conditions at sub 6GHz frequencies. We have integrated them in the open source WiSHFUL testbed, that has been further enhanced with a customized version of the TDMA implementation, and we have elaborated MAC resource allocation strategies based on position estimations that effectively improve the network performance of robots moving in a harsh environment full of metallic objects and walls. In conclusion, positioning mobile targets can be beneficial in harsh environments and context data should be integrated in network protocol stack to optimize the overall network performance.

5 Virtual Sensors

Inertial sensors in smartphones have been exploited extensively in the past years for applications such as dead-reckoning aided localization, fine-grained gesture and posture recognition, and mobile gaming [21, 25–28]. Smartphone sensors such as gyroscopes, compass, and accelerometer have received particular attention, and they can nowadays operate efficiently in terms of energy consumption. However, they can be affected by significant noise and error compared to industry grade sensors. More importantly, the wireless infrastructure does not have direct access to inertial sensors in mobile devices, calling for the installation of dedicated applications in the mobile device responsible to serve as a collector of sensor measurements. Such a mobile app is usually also responsible to process inertial sensor data, as continuously transmitting raw input data to the infrastructure would result in an unacceptably high network traffic load.

Driven by the above observations, in this chapter we aim to answer to the following question: **can the infrastructure**, and in particular the WiFi APs **infer information such as rotation speed** (similar to what a gyroscope would measure) **and acceleration** (similar to what an accelerometer would measure) **of the mobile device without access to its physical inertial sensors?** We call these mechanisms *virtual inertial sensors* of mobile devices, as the infrastructure performs radio measurements to estimate the information a mobile would typically extract from the inertial sensors.

A straightforward idea is to estimate the mobile's acceleration using ranging measurements. We confirm in this work that this is indeed possible, showing how well the acceleration can be estimated with commodity devices. What is more challenging is the question of how to achieve a reliable rotation estimate. Naturally, as the rotation speed is related to angular information, one may consider to exploit Channel State Information (CSI) and map it to the angle. CSI has received high attention in the research community in the past years to address problems such as indoor localization and estimating context information of users. However, with only AP-side CSI it is not possible to reliably estimate the rotation speed of mobile devices with a single antenna. The fundamental problem is that CSI measures angle variation using the AP as reference

system instead of the client. This results in an error that increases with the range between the AP and the mobile device.

We then present a solution to estimate mobile device rotation that relies on Fine Time Measurements (FTM), a protocol that has been standardized in the IEEE 802.11-2016 standard, with corresponding devices now entering the market. FTM allows to estimate the distance between the AP and the mobile, even without associating the mobile to the AP. We retrieve FTM measurements from a commodity AP working on a 80 MHz channel communicating to a Google Pixel 3 smartphone, and show that FTM can serve purposes beyond the original scope of estimating the distance. This is feasible even if FTM does not take advantage of the multiple antennas of the AP and the multiple subcarriers of the spectrum, as CSI does. Furthermore, in comparison to physical inertial sensors, it is not affected by dead reckoning, since it relies on absolute distance estimates to avoid error accumulation over time.

The rest of the chapter is organized as follows. We first study the simple case of estimating the acceleration using FTM samples (Section 5.1) and then present a geometric model that maps FTM samples to angular rotation, applying signal processing theory in order to provide an estimator of the rotation speed (Section 5.2). We then perform an experimental comparison of performance of an FTM-based estimator versus a CSI-based estimator using commodity 802.11ac chipsets, demonstrating the potential of the proposed approach (Section 5.3). We finally draw conclusion and discuss the main findings of the experiments conducted in this work (Section 5.4).

5.0.1. Applications

Using radio measurements as virtual inertial sensors could give the opportunity to explore several applications:

Secure localization. Several methods for positioning rely nowadays on the inputs from inertial sensors provided by the mobile device. However, these inputs could be corrupted or hacked. An independent method to verify the correctness of these inputs can be derived using virtual inertial sensors, just using radio measurements. Virtual inertial sensors could be also used in absence of inputs from inertial sensors, for instance in case these data cannot be trusted for positioning or are not accessible.

User behavior. The infrastructure can infer the movements of the mobile device as proposed in this work and exploit this information for areas such as crowd profiling and gaming without requesting the installation of any app to get access to the inertial sensors of the mobile device.

Robust communication at mm-wave. After beam training, environment variations as well as mobility and rotation of the mobile device (UE, user equipment) can cause swift changes of the link quality, and continuous beam adaptation is needed to maintain a high data rate. UE rotations must be compensated for link maintenance. For instance, [65]

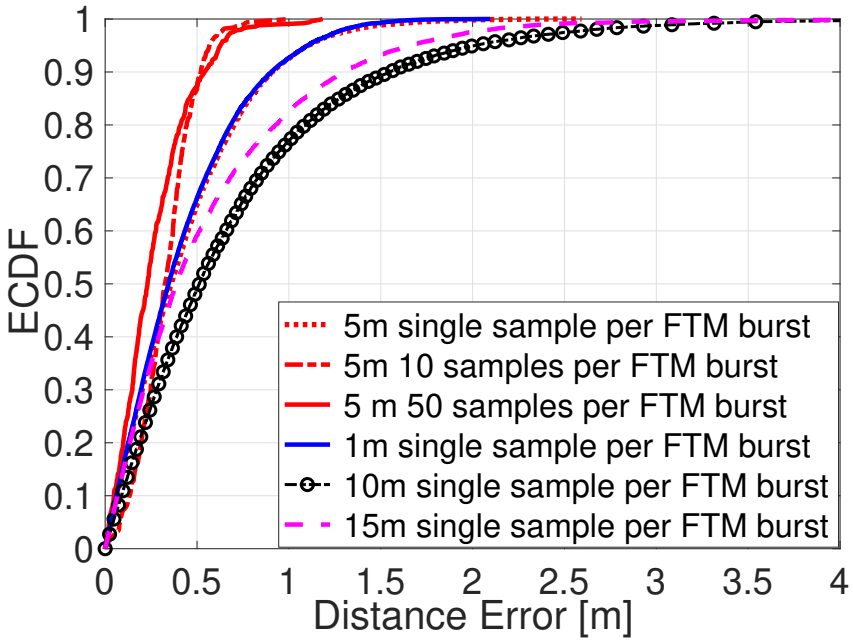


Figure 5.1: Distance estimation using FTM measurements with 10 and 50 samples per burst, at 5 m.

relies on gyroscope sensors to reduce the beam search under mobility. Any variation of the angle must be estimated and the chosen antenna sector adjusted accordingly, otherwise the mm-wave link quality would suffer dramatically. Inputs from relative rotation information as provided in this work could facilitate beam maintenance, or provide insights on why the link was disrupted.

5.1. Background and Acceleration

The background on the FTM protocol of the IEEE 802.11mc amendment has been already provided in Section 3.2 of the chapter 3.

The FTM protocol allows to increase the number of samples per burst. For this to occur, the AP sends several FTM frames in a sequence and the STAtion (STA) estimates the Round Trip Time (RTT) for each pair of FTM message/ACK. In order to understand the accuracy of FTM for distance estimation, we evaluate the distance accuracy at difference distances and with different bursts. The results are plotted in Fig. 5.1. In case of only one sample per burst, the results show that for all cases we achieve a median ranging error of 0.5 m. The figure also highlights the benefit of a higher number of samples per burst, confirming the results presented in [52].

However, the RTT for each pair of FTM message/ACK is not available in the report, but only the average RTT over the burst with the corresponding distance. This is a main issue for our study. In fact, experimental results in Sec. 5.3 show that the rotation

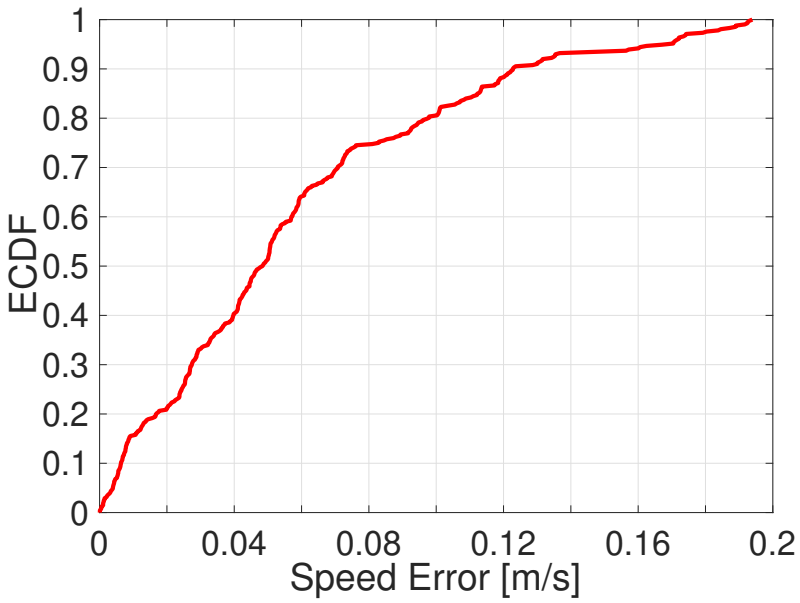


Figure 5.2: We exploit FTM ranging measurements d from the AP to the mobile device to estimate the walking speed.

accuracy decreases with increasing burst size. The reason is that averaging several samples for moving targets results in information loss. In conclusion, as we consider moving devices (accelerating or rotating), we focus only on RTT measurements with burst size equal to one packet.

5.1.1. Acceleration

We first evaluate the simple case of estimating the user walking speed using only FTM measurements. The experiment focuses on understanding how reliable this estimation is. The drawback of CSI is that it cannot be used for this purpose. We instead exploit FTM measurements to estimate the walking speed of the UE. Our confidence is that the mobility component of the UE can be identified at a low frequency using the Fast Fourier Transform (FFT) of the distance samples sequence:

$$d(t) \xrightarrow{\mathcal{F}} F(f) \tag{5.1}$$

where $\mathcal{F}$ indicates the Fourier transform.

We denote the frequency transform as $F(f)$ and calculate the frequency at which $F(f)$ presents the main peak. We estimate the walking speed as:

$$\hat{\omega} = f_{PEAK} \cdot d_{norm}. \tag{5.2}$$

where d_{norm} is the difference between the maximum and the minimum of estimated

distances in a time window of 1 s.

We conduct a test to evaluate this methodology. In this test a human carries the mobile phone moving toward from the AP at an average speed of 0.5 m/s. Fig. 5.2 shows the ECDF of the speed error in m/s, reporting a median error of around 0.047 m/s.

5.2. Mobile device rotation

For the estimation of the mobile device rotation, we first define the scenario under study, then explain why CSI cannot provide reliable rotation speed and then present our solution based on FTM measurements.

5.2.1. Scenario

For a human carrying the mobile, let us assume that the elbow is close to the torso, with the right upper arm (which extends from the shoulder to the elbow) in parallel to the torso. The right shoulder of the human serves as the origin of the Z axis and the radius δ_z is the length of the forearm, which extends from the elbow to the hand carrying the mobile and it represents the distance between the mobile and its rotation axis. An illustration is presented in Fig. 5.3. Nevertheless, the system works also without the assumption that the elbow is close to the torso, as long as the changes in the radius δ_z can be considered negligible during the measurement.

5.2.2. Exploring CSI for rotation speed estimation

A possible solution could be to leverage CSI at the AP, largely exploited in the literature for positioning and context information [20, 23, 66–69]. The reported CSI is a matrix containing one complex number per sub-carrier and per received antenna at the AP. Even in this work we use the WiFi chipset introduced in chapter 3, with four antennas and 234 subcarriers. From CSI, we can infer the Angle Of Arrival (Angle-Of-Arrival (AOA)) of WiFi signals from the UE to the AP. But a fundamental aspect of the rotation is about the rotation axis. In fact, detecting the UE rotation using AOA information causes failures in the mobile rotation estimation as *CSI estimates the rotation using the AP as reference system instead of the client.* In other terms, it estimates the angle rotation of the client as seen from the AP. More in details, let us call the mobile as UE and the origin of the reference system as UE_0. Applying CSI works well only when the UE rotates around the AP, with the AP at the origin of the Z axis UE_o. However, this does not happen in practice. The worst case occurs for mobile device faraway from the AP: rotations with radius δ_z can be hardly sensed from the AP relying on the AOA, and the estimator would declare the device as static or quasi-static. Fig. 5.4 illustrates the estimated angle AOA and how sensitivity changes with the ratio of the distance d to

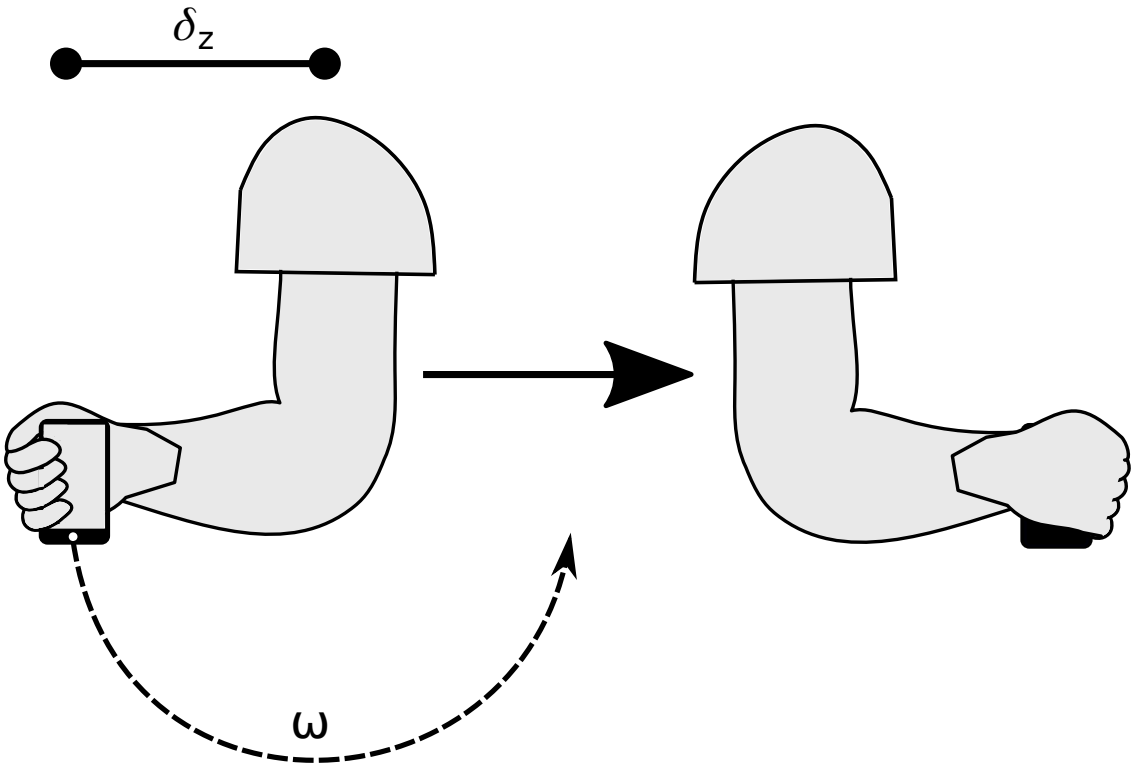


Figure 5.3: Illustration of scenario with human carrying the mobile, and rotation movement of the UE along a circular trajectory (rotation asis: shoulder/upper arm).

the radius δ_z.

Based on the above explanation, in the next subsection we introduce our method to infer the rotation speed of the UE performing measurements from the AP.

5.2.3. From FTM ranging to rotation

We present a method for estimating the rotation of the UE using only distance measurements performed from the AP. Our insight is that a short series of FTM ranging measurements can provide not only the distance to the target UE, but also information about the relative rotation of the UE. We will show that this can done even if FTM does not take advantage of the multiple antennas and subcarriers of the hardware (in contrast to CSI).

A schematic representation of the problem is presented in Fig. 5.5. We define the relationship between the FTM ranging measurement d and the rotation speed ω as:

$$d^2(t) = d_0^2 + \delta_z^2 + 2d_0\delta_z \cos(\omega t)\,, \tag{5.3}$$

where $d_0 = \|\mathbf{p}^{\mathrm{AP}} - \mathbf{p}^{UE_0}\|$. In the ideal case where the ranging measurements are not affected by measurement noise and multipath, the result of Eq. 5.5 is an harmonic function

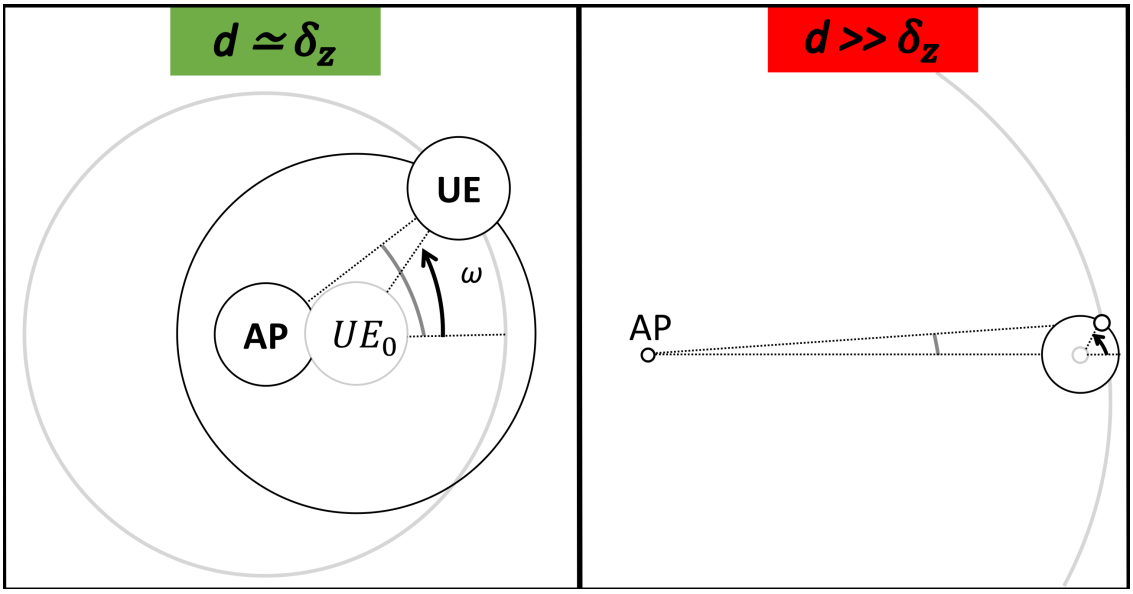


Figure 5.4: AOA estimates using CSI measurements collected at the AP. As the distance d increases, large variations of the angle at the mobile device are seen as small variations in the angle estimated at the AP. In other terms the sensitivity of the AOA variation estimated at the AP decreases as the distance increases, resulting in larger errors.

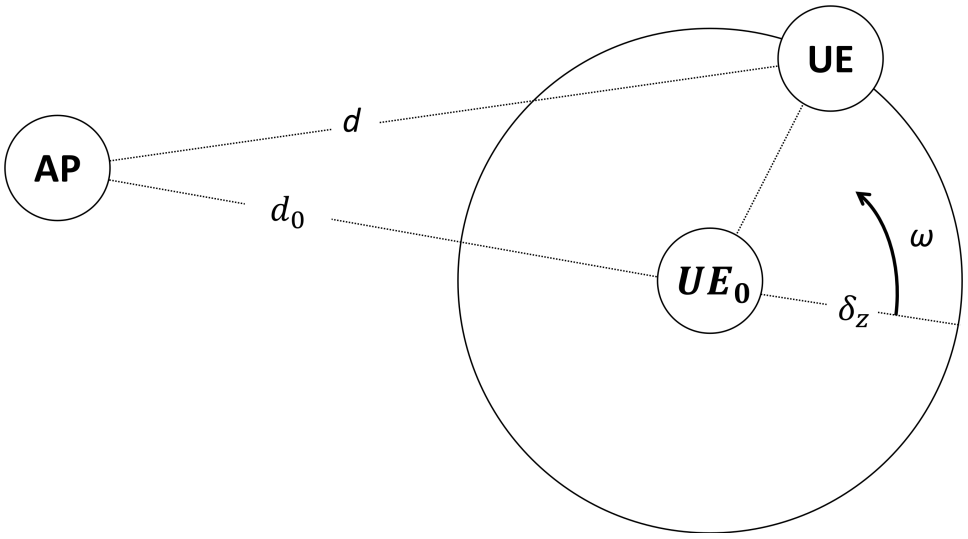


Figure 5.5: We exploit the FTM ranging measurements d from the AP to UE to estimate the rotation speed ω.

of period $T = 1/\omega$ and amplitude $2d_0\delta_z$. However, in a real case of rotation, the result of Eq. 5.5 is the sum of different harmonics at different frequencies.

Our confidence is that the rotation component becomes evident in a transformed domain, namely the frequency domain. For this reason, we propose the following method. We gather FTM ranging measurements $d(t)$ in a time window of one second. This sequence is upsampled by a factor N and transformed into sequence $d'(t)$ (for a review of upsampling

process see e.g. [70]). Upsampling allows us to smooth the sequence. We then compute the FFT of the upsampled sequence:

$$d'^2(t) \xrightarrow{\mathcal{F}} F(f) \tag{5.4}$$

where $\mathcal{F}$ indicates the Fourier transform. We then remove the Direct Current (DC) component (caused by d_0^2), and process the resulting sequence in the frequency domain. We investigated several options to process the sequence, and we present here the one best performing.

5.2.4. Criterion to infer the rotation with noisy data

We introduce a criterion to estimate the rotation frequency in presence of noisy data. The idea of the criterion is to integrate the area of the $F(f)$ after DC removal. For small rotation speeds, the spike related to the real rotation speed occurs at small frequency $F(f)$. For this reason, the smaller is the rotation speed, the faster we expect that the integrated area increases as a function of the frequency f. This intuition will be confirmed in the experimental evaluation in Section 5.3 and Fig. 5.7.

More formally, we integrate $F(f)$ after DC removal and calculate $f_{1/2}$, the frequency at which the area of the normalized FFT is equal to 1/2: $\sum_{f_0}^{f_{1/2}} \mathrm{F}(f) = \frac{F_{tot}}{2}$, where f_0 is the first frequency component after the DC component and F_{tot} is the total area of the FFT after DC removal. We compute the Pearson correlation coefficient of the experimental set of $f_{1/2}$ with respect to the real rotation speeds. The result is a correlation of around 0.6, for different distances. We compute the linear regression of the experimental $f_{1/2}$ with the respect to the real rotation speed, and we can then estimate the rotation speed, as

$$\hat{\omega} = m \cdot f_{1/2} + q, \tag{5.5}$$

where m and q are the slope and the bias of the linear regression, respectively. It follows that, since we analyze Eq. 5.3 in the frequency domain, we are able to estimate the angular speed without any knowledge of the radius δ_z, as shown in Eq. 5.5. We note that the methodology works as long as $\delta_z > 0$, which holds in the scenario defined in Section 5.2.1. We also note that, as the proposed method does not look only at the dominant peak in the frequency domain but the whole frequency spectrum, the proposed method will provide an average $\hat{\omega}$ in case of noise in the estimate.

We finally evaluate the accuracy of the proposed algorithm for rotation speed estimation in Sec. 5.3.

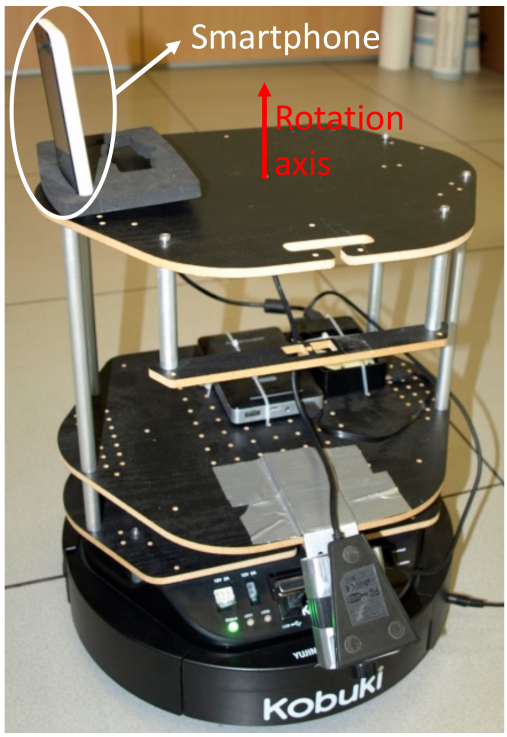


Figure 5.6: The experimental setup for controlled rotation measurements.

5.3. Evaluation

In this section, we compare the proposed rotation estimator with another solution that exploits CSI. For this purpose, we analyze the performance of both estimators, the proposed one obtained from FTM and the one from CSI and, in both cases, we use a Google Pixel 3 as mobile device. In order to collect FTM and CSI measurements, we choose the same platforms used in chapter 3 (Sec. 3.4.1).

5.3.1. Rotation

We perform experiment in a corridor of an office area, where multipath can occur. As shown in Fig. 5.6, we mount the UE on top of Kobuki Turtlebot II robot running ROS (Robotic Operative System), in order to produce a fixed rotation speed. More specifically, the rotation mechanism of the robot is connected to a RaspberryPi where we run a Python code in order to control its rotation speed. AP and UE are positioned at distance of 1, 5, 10 and 15 m from each other. In case of rotation, the robot produces different rotation speeds, namely 7, 33 and 70 °/sec. For each experiment, we gather measurements in 1 second window.

From FTM. After collecting a set of around 100 FTM ranging samples in a window of 1 second, we apply the methodology presented in Sec. 5.2.3. We upsample by a factor of $N = 10$. Based on our experience, $N = 10$ provides a good trade-off between performance

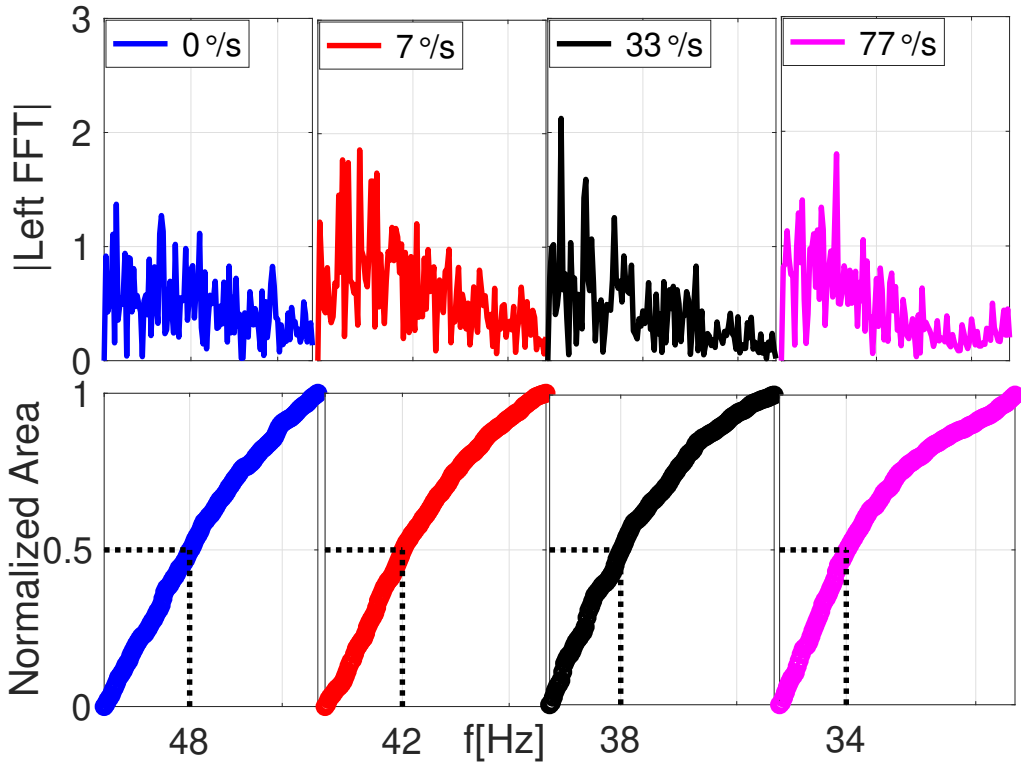


Figure 5.7: Normalized FFT of FTM samples at 5 m and normalized area. The figure shows that the smaller is the rotation speed, the faster the integrated area increases as a function of the frequency f and reaches the target $f_{1/2}$.

and higher processing time. Fig. 5.7 shows the normalized $F(f)$ as a function of the rotation speed ω, from a single snapshot at 5 m. The FFT analysis shows that the frequency of the dominant component increases with the speed of rotation. We first evaluate the accuracy of the proposed method for rotation speed estimation, increasing the FTM burst size at 5m only. The Fig. 5.8 shows that the RMSE increases using 1, 3, 10 and 50 samples per burst. We then evaluate the accuracy of the proposed method for rotation speed estimation. Fig. 5.9 shows the ECDF of the rotation speed error of all cases, at 1 m (5.9(a)), 5 m (5.9(b)), 10 m (5.9(c)) and 15 m (5.9(d)). In all cases, for high ω (70°/s) the method loses in accuracy. Yet, as shown in the figure, we are able to estimate the rotation rate with reasonable error in one second, specially for low speeds. In particular, we show that we achieve a median rotation error of less than 3 degrees for low rotation speeds for all distances under tests.

From CSI. Also for this evaluation, we collect around 100 AOA samples in a window of 1 second. We leverage MUltiple SIgnal Classification (MUSIC) algorithm to identify the strongest AOA, which is a technique widely used in the literature [19, 23, 51]. This technique performs the subspace decomposition of the autocorrelation matrix. For the estimation of the strongest path we consider the highest peak in the MUSIC spectrum. We then apply the same methodology presented in Sec. 5.2.3, as for FTM samples. Fig. 5.9

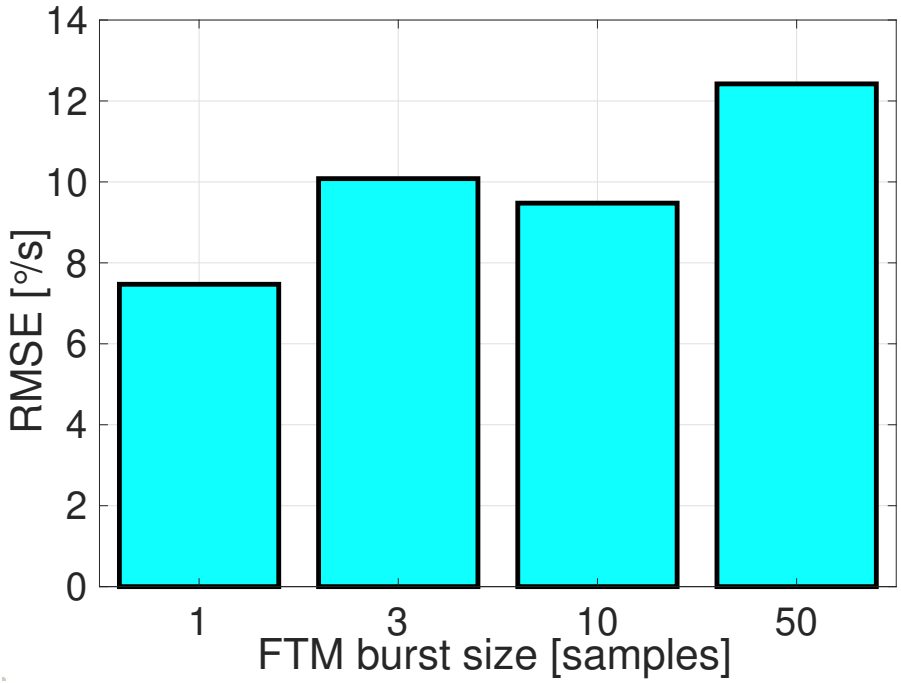


Figure 5.8: Rotation estimation at 5m using 1, 3, 10 and 50 samples per burst.

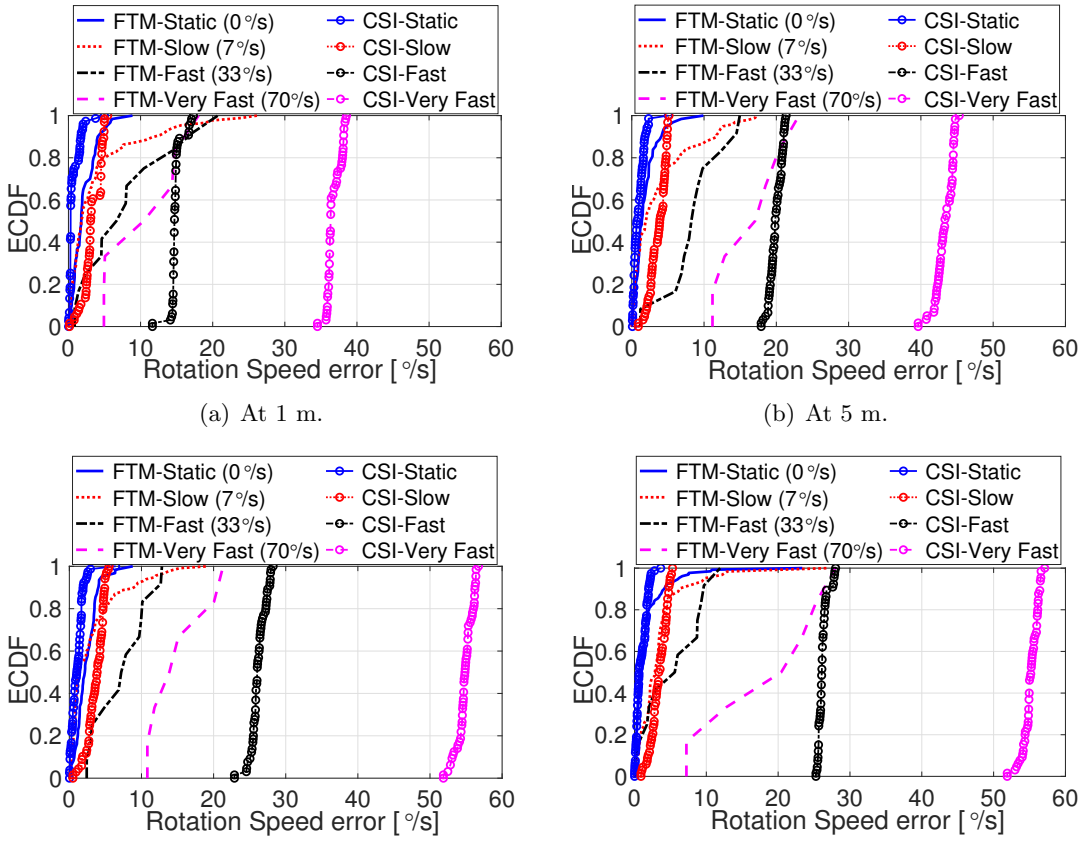
shows the ECDFs of the obtained rotation speed error for all four cases, at 1 m (5.9(a)), 5 m (5.9(b)), 10 m (5.9(c)) and 15 m (5.9(d)). Overall, CSI performs worst that the proposed FTM-based method.

We analyze the results more in details in the bar plot in Fig. 5.10, where we show a comparison between the median estimates of FTM and CSI approaches. The "ground truth" is further plotted, which indicates the rotation speed as set on the Kobuki Turtlebot II robot.

Fig. 5.10 shows that the performance of CSI degrades more significantly as the rotation speed increases. CSI shows dependence on the distance between the AP and the UE, confirming our finding that performance become poor at larger distances (cf. Section 5.2.2). Instead our proposed method based on FTM provides robust performance as the speed and distance increase. Given that FTM can also be used to estimate the acceleration of the mobile device (cf. Section 5.1.1), our FTM-based solution can be used standalone as virtual inertial sensor for estimating both acceleration and rotation speed.

5.4. Conclusion and Discussion

In this chapter we have presented the concept of virtual inertial sensors, where the infrastructure estimates parameters that characterize the movements of a mobile device without any access to physical inertial sensors but just using radio measurements. We have introduced a model that uses only WiFi FTM to infer the rotation speed and the walking



(a) At 1 m.

(b) At 5 m.

(c) At 10 m.

(d) At 15 m.

Figure 5.9: Exploitation of UE ranging information and CSI to define the rotation speed ω, at 1, 5, 10 and 15 meters.

speed of a user. We have shown that CSI is not suitable to estimate the rotation of mobile terminals as it operates on a wrong reference system to estimate the rotation (as confirmed by our experiments). It also cannot be used to estimate the mobile device acceleration. We have evaluated the proposed approaches with experiments, demonstrating a median rotation error of less than 3 degrees for low rotation speeds, and a median walking speed error of around 0.047m/s. As discussed, the main limitation is that the proposed method requires that the device is held and rotates around an axis such as human upper arm. Rotations around the same axis of the mobile device cannot be detected. There are several aspects that must be addressed to clarify the practicality of the proposed methodology. First of all, experiments have been conducted using robots, and the question that must be addressed is the validation of the results using humans. A second open question is to perform studies of rotations under mobility to study how well we can separate the rotation component from the accelerometer one. A third aspect it to understand how to estimate more complex movements of the mobile. Finally, we envision that by fusing FTM and CSI we can leverage some of the benefits of both metrics to design more robust

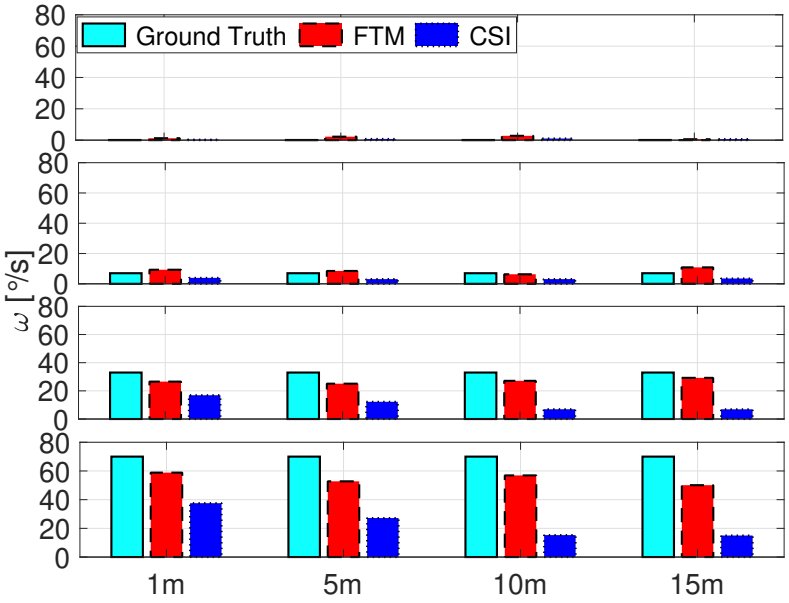


Figure 5.10: Comparison between FTM (red bars) and CSI (blue bars) approaches for detecting the UE rotation.

virtual inertial sensors.

6 Conclusions

In this thesis, we have investigated the ToF technique for bringing localization solutions in indoor environments, just using commodity WiFi hardware and without requiring any calibration or user intervention. We have presented different ideas on how extracted location data may be exploited not only for navigation, but also for the benefit of the network itself or for the investigation of physical behaviours.

In the Part II of the thesis, we have identified the main research problems to obtain reasonable positioning accuracy using only measurements performed by commodity APs. We have designed solutions that fit to different scenarios such as offices, shopping malls and apartments and combining different techniques for reducing the number of the deployed APs in the system. More specifically, in Chapter 2 we have introduced a multi-APs localization system, where the ToF echo technique has been proposed to estimate the distance between the WiFi AP and the STA. In this chapter, in order to mitigate the device-related Gaussian noise and the multipath, we have developed a filter based on a combination of statistical learning techniques to train an environmental-specific linear regression model. These filtered timing signals have been then used as ranging inputs of the multi-lateration problem for positioning. We have evaluated our system in multipath environments such as office areas, across multiple testbeds which cover different surfaces. Instead, Chapter 3 has proposed a solution to localize a COTS smartphone device using radio measurements performed by a single WiFi AP. The proposed solution is based on the combination of the FTM and CSI for ranging and AOA estimates, respectively. We have exploited PHY information to detect the number of paths and their directions and we have used this information to derive a new method for filtering ranging measurements obtained with the FTM protocol. We have then evaluated the system in multipath rich environments, able to achieve reasonable positioning accuracy in areas comparable to typical flat sizes.

In the Part III of the thesis, we have demonstrated that location information can be used not only for customer services but also for wireless resources allocation strategy and for the estimation of information that a mobile device would typically extract from

the inertial sensors. More in detail, Chapter 4 has proposed to exploit the knowledge of location to derive context information to dynamically allocate wireless resources in time and space to target devices. We have exploited the spatial geometry of the APs and a statistical model that maps the user position's spatial distribution to an angle error distribution to derive a hypothesis test to declare if the link is in LOS or NLOS. We experimentally showed that context information applied to wireless resources protocol help increasing the network throughput in an industrial-like scenario. Finally, Chapter 5 has presented a new concept of virtual inertial sensors. In this chapter we have proposed a method for estimating the user walking speed and a novel method for the rotation of a mobile device without accessing the inertial sensors of the mobile itself. We have evaluated and demonstrated the proposed approaches with experiments, and we have compared the rotation method with a solution that explores CSI measurements. Using FTM that works with only one single antenna, we have achieved better performance than a CSI-based estimator that exploits four antennas and multiple sub-carriers at the AP. In conclusion, this new concept of virtual inertial sensors can be leveraged by location systems and sensing mechanisms to improve localization and design better and more secure communication.

Appendices

References

[1] M. Rea, A. Fakhreddine, D. Giustiniano, and V. Lenders, "Filtering noisy 802.11 time-of-flight ranging measurements from commoditized wifi radios," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2514–2527, Aug 2017.

[2] M. Rea, D. Garlisi, H. Cordobés de la Calle, and D. Giustiniano, "Location-aware mac scheduling in industrial-like environment," in *Broadband Communications, Networks, and Systems*, V. Sucasas, G. Mantas, and S. Althunibat, Eds. Cham: Springer International Publishing, 2019, pp. 201–211.

[3] M. Rea, H. Cordobés de la Calle, and D. Giustiniano, "Time-of-flight wireless indoor navigation system for industrial environment," in *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation Characterization*, ser. WiNTECH '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 37–44. [Online]. Available: https://doi.org/10.1145/3349623.3355476

[4] M. Rea, T. E. Abrudan, D. Giustiniano, H. Claussen, and V.-M. Kolmonen, "Smartphone positioning with radio measurements from a single wifi access point," in *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, ser. CoNEXT '19. New York, NY, USA: Association for Computing Machinery, 2019, pp. 200–206. [Online]. Available: https://doi.org/10.1145/3359989.3365427

[5] M. Rea, D. Giustiniano, and J. Widmer, "Virtual Inertial Sensors with Fine Time Measurements."

[6] M. Rea, H. Cordobés de la Calle, D. Giustiniano, D. Garlisi, P. Gallo, S. Giannoulis, and I. Moerman, "Poster: Integration of wifi tof positioning system in the open, flexible and adaptive wishful architecture," in *Proceedings of the 11th Workshop on Wireless Network Testbeds, Experimental Evaluation CHaracterization*, ser. WiNTECH '17. New York, NY, USA: Association for Computing Machinery, 2017, pp. 103–104. [Online]. Available: https://doi.org/10.1145/3131473.3133334

[7] M. Rea, H. Cordobés de la Calle, D. Giustiniano, and V. Lenders, "Demonstration abstract: Robust wifi time-of-flight positioning system," in *Proceedings of the 15th International Conference on Information Processing in Sensor Networks*, ser. IPSN '16. Vienna, Austria: ACM, 2016. [Online]. Available: http://eprints.networks.imdea.org/1192/

[8] M. Rea, H. Cordobés de la Calle, and D. Giustiniano, "Demonstration abstract: Twins: Time-of-flight based wireless indoor navigation system," in *Proceedings of the 17th International Conference on Information Processing in Sensor Networks*, ser. IPSN '18. Porto, Portugal: ACM, 2018. [Online]. Available: http://eprints.networks.imdea.org/1869/

[9] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location-support system," in *Proc. ACM MobiCom'00*, Aug. 2000.

[10] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system," *ACM Trans. Inf. Syst.*, vol. 10, no. 1, pp. 91–102, Jan. 1992. [Online]. Available: http://doi.acm.org/10.1145/128756.128759

[11] D. Haehnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose, "Mapping and localization with RFID technology," in *Proc. IEEE ICRA'05*, Apr. 2004.

[12] S. Gezici, Z. Tian, G. B. Giannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 70–84, July 2005.

[13] P. Bahl and V. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM'00*, Mar. 2000.

[14] M. Youssef and A. Agrawala, "The horus wlan location determination system," ser. MobiSys '05. New York, NY, USA: ACM, 2005, pp. 205–218. [Online]. Available: http://doi.acm.org/10.1145/1067170.1067193

[15] H. Lim, L.-C. Kung, J. Hou, and H. Luo, "Zero-configuration, robust indoor localization: Theory and experimentation," in *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, April 2006, pp. 1–12.

[16] A. Goswami, L. E. Ortiz, and S. R. Das, "Wigem: A learning-based approach for indoor localization," ser. CoNEXT '11. New York, NY, USA: ACM, 2011, pp. 3:1–3:12. [Online]. Available: http://doi.acm.org/10.1145/2079296.2079299

[17] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *In Proc. of ACM MobiCom '10*, 2010, pp. 173–184. [Online]. Available: http://doi.acm.org/10.1145/1859995.1860016

[18] J. Xiong and K. Jamieson, "Arraytrack: A fine-grained indoor location system," in *Presented as part of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*. Lombard, IL: USENIX, 2013, pp. 71–84. [Online]. Available: https://www.usenix.org/conference/nsdi13/technical-sessions/presentation/xiong

[19] S. Sen, J. Lee, K.-H. Kim, and P. Congdon, "Avoiding multipath to revive inbuilding wifi localization," in *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '13. New York, NY, USA: ACM, 2013, pp. 249–262. [Online]. Available: http://doi.acm.org/10.1145/2462456.2464463

[20] J. Gjengset, J. Xiong, G. McPhillips, and K. Jamieson, "Enabling phased array signal processing on commodity wifi access points," ser. Mobicom '14. New York, NY, USA: ACM, 2014.

[21] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," ser. Mobicom '12. New York, NY, USA: ACM, 2012, pp. 293–304. [Online]. Available: http://doi.acm.org/10.1145/2348543.2348580

[22] Z. Chen, Z. Li, X. Zhang, G. Zhu, Y. Xu, J. Xiong, and X. Wang, "Awl: Turning spatial aliasing from foe to friend for accurate wifi localization," in *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*, ser. CoNEXT '17. New York, NY, USA: ACM, 2017, pp. 238–250. [Online]. Available: http://doi.acm.org/10.1145/3143361.3143377

[23] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: Decimeter level localization using wifi," in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, ser. SIGCOMM '15. New York, NY, USA: ACM, 2015, pp. 269–282. [Online]. Available: http://doi.acm.org/10.1145/2785956.2787487

[24] A. T. Mariakakis, S. Sen, J. Lee, and K.-H. Kim, "Sail: Single access point-based indoor localization," in *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '14. New York, NY, USA: ACM, 2014, pp. 315–328. [Online]. Available: http://doi.acm.org/10.1145/2594368.2594393

[25] J. Paek, J. Kim, and R. Govindan, "Energy-efficient rate-adaptive gps-based positioning for smartphones," in *MobiSys*, 2010.

[26] P. Zhou, M. Li, and G. Shen, "Use it free: Instantly knowing your phone attitude," in *Proceedings of the 20th annual international conference on Mobile computing and networking*. ACM, 2014, pp. 605–616.

[27] A. Parate, M.-C. Chiu, C. Chadowitz, D. Ganesan, and E. Kalogerakis, "Risq: Recognizing smoking gestures with inertial sensors on a wristband," in *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '14. New York, NY, USA: ACM, 2014, pp. 149–161. [Online]. Available: http://doi.acm.org/10.1145/2594368.2594379

[28] S. Shen, H. Wang, and R. Roy Choudhury, "I am a smartwatch and i can track my user's arm," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '16. New York, NY, USA: ACM, 2016, pp. 85–96. [Online]. Available: http://doi.acm.org/10.1145/2906388.2906407

[29] Xinrong Li, K. Pahlavan, M. Latva-aho, and M. Ylianttila, "Comparison of indoor geolocation methods in dsss and ofdm wireless lan systems," in *Vehicular Technology Conference Fall 2000. IEEE VTS Fall VTC2000. 52nd Vehicular Technology Conference (Cat. No.00CH37152)*, vol. 6, Sep. 2000, pp. 3015–3020 vol.6.

[30] D. D. McCrady, L. Doyle, H. Forstrom, T. Dempsey, and M. Martorana, "Mobile ranging using low-accuracy clocks," *IEEE Transactions on Microwave Theory and Techniques*, vol. 48, no. 6, pp. 951–958, June 2000.

[31] A. Günther and C. Hoene, "Measuring round trip times to determine the distance between wlan nodes," ser. NETWORKING'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 768–779. [Online]. Available: http://dx.doi.org/10.1007/11422778_62

[32] M. Ciurana, F. Barcelo-Arroyo, and F. Izquierdo, "A ranging system with IEEE 802.11 data frames," in *Radio and Wireless Symposium, 2007 IEEE*, 2007, pp. 133–136.

[33] S. A. Golden and S. S. Bateman, "Sensor measurements for Wi-Fi location with emphasis on time-of-arrival ranging," *IEEE Trans. Mobile Comput.*, vol. 6, no. 10, pp. 1185–1198, Oct. 2007. [Online]. Available: http://portal.acm.org/citation.cfm?id=1313052.1313251

[34] D. Giustiniano and S. Mangold, "Caesar: Carrier sense-based ranging in off-the-shelf 802.11 wireless lan," in *Proceedings of the Seventh COnference on Emerging Networking EXperiments and Technologies*, ser. CoNEXT '11. New York, NY, USA: ACM, 2011, pp. 10:1–10:12. [Online]. Available: http://doi.acm.org/10.1145/2079296.2079306

[35] X. Li and K. Pahlavan, "Super-resolution toa estimation with diversity for indoor geolocation," *Wireless Communications, IEEE Transactions on*, vol. 3, no. 1, pp. 224–234, Jan 2004.

[36] J. Xiong, K. Sundaresan, and K. Jamieson, "Tonetrack: Leveraging frequency-agile radios for time-based indoor wireless localization," in *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '15. New York, NY, USA: ACM, 2015, pp. 537–549. [Online]. Available: http://doi.acm.org/10.1145/2789168.2790125

[37] "Microsoft indoor localization competition, http://research.microsoft.com/en-us/events/ipsn2014indoorlocalizatinocompetition/."

[38] "IEEE std 802.11-2012 (revision of IEEE std 802.11-2007), IEEE standard for information technology," pp. 1–2793, 2012.

[39] T. Bourchas, M. Bednarek, D. Giustiniano, and V. Lenders, "Poster abstract: Practical limits of wifi time-of-flight echo techniques," in *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks*, ser. IPSN '14. Piscataway, NJ, USA: IEEE Press, 2014, pp. 273–274. [Online]. Available: http://dl.acm.org/citation.cfm?id=2602339.2602370

[40] I. Guvenc and C.-C. Chong, "A survey on toa based wireless localization and nlos mitigation techniques," *Communications Surveys Tutorials, IEEE*, vol. 11, no. 3, pp. 107–124, rd 2009.

[41] J. Bilmes, "A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models," Tech. Rep., 1998.

[42] A. J. Wheeler, A. R. Ganji, V. V. Krishnan, and B. S. Thurow, *Introduction to engineering experimentation*. Prentice Hall New Jersey, 1996.

[43] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise Reduction in Speech Processing*, ser. Springer Topics in Signal Processing. Springer Berlin Heidelberg, 2009, vol. 2, pp. 1–4. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-00296-0_5

[44] N. R. Draper, H. Smith, and E. Pownell, *Applied regression analysis*. Wiley New York, 1966, vol. 3.

[45] I. Tinnirello, D. Giustiniano, L. Scalia, and G. Bianchi, "On the side-effects of proprietary solutions for fading and interference mitigation in IEEE 802.11b/g outdoor links," *Comput. Netw.*, vol. 53, no. 2, pp. 141–152, Feb. 2009. [Online]. Available: http://portal.acm.org/citation.cfm?id=1482177.1482269

[46] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 3, pp. 535–547, 2000.

[47] M. R. Gholami, "Positioning algorithms for wireless sensor networks," *Licentiate thesis, Chalmers University of Technology*, 2011.

[48] A. Marcaletti, M. Rea, D. Giustiniano, V. Lenders, and A. Fakhreddine, "Filtering noisy 802.11 time-of-flight ranging measurements," in *Proceedings of the 10th ACM International on Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '14. New York, NY, USA: ACM, 2014, pp. 13–20. [Online]. Available: http://doi.acm.org/10.1145/2674005.2674998

[49] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single wifi access point," in *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. Santa Clara, CA: USENIX Association, 2016, pp. 165–178. [Online]. Available: https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/vasisht

[50] IEEE, "Ieee standard for information technology- telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements-part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications," *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)*, pp. 1–3543, 2016.

[51] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, March 1986.

[52] M. Ibrahim, H. Liu, M. Jawahar, V. Nguyen, M. Gruteser, R. Howard, B. Yu, and F. Bai, "Verification: Accuracy evaluation of wifi fine time measurements on an open platform," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '18. New York, NY, USA: ACM, 2018, pp. 417–427. [Online]. Available: http://doi.acm.org/10.1145/3241539.3241555

[53] Y. Hua and T. K. Sarkar, "Matrix pencil method and its performance," in *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, April 1988, pp. 2476–2479 vol.4.

[54] S. Konishi and G. Kitagawa, *Information criteria and statistical modeling*. Springer Science & Business Media, 2008.

[55] J. Lee, H.-A. Kao, and S. Yang, "Service innovation and smart analytics for industry 4.0 and big data environment," *Procedia Cirp*, vol. 16, pp. 3–8, 2014.

[56] M. Rea, A. Fakhreddine, D. Giustiniano, and V. Lenders, "Filtering noisy 802.11 time-of-flight ranging measurements from commoditized wifi radios," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2514–2527, Aug 2017.

[57] "H2020 WiSHFUL project," http://www.wishful-project.eu.

[58] D. Giustiniano, T. Bourchas, M. Bednarek, and V. Lenders, "Deep inspection of the noise in wifi time-of-flight echo techniques," in *MSWiM '15*, 2015, pp. 5–12.

[59] A. Fakhreddine, D. Giustiniano, and V. Lenders, "Data fusion for hybrid and autonomous time-of-flight positioning," in *IPSN*. ACM, 2018.

[60] E. Kaplan and C. Hegarty, *Understanding GPS: principles and applications*. Artech house, 2005.

[61] R. Langley, "Dilution of precision," GPS World, 1999.

[62] C. A. Ogaja, *Geomatics Engineering: A Practical Guide to Project Design*. CRC Press, 2016.

[63] N. J. Kaminski, I. Moerman, S. Giannoulis, A. Zubow, I. Seskar, and S. Choi, "Unified radio and network control across heterogeneous hardware platforms," 2016.

[64] I. Tinnirello, G. Bianchi, P. Gallo, D. Garlisi, F. Giuliano, and F. Gringoli, "Wireless mac processors: Programming mac protocols on commodity hardware," in *2012 Proceedings IEEE INFOCOM*, March 2012, pp. 1269–1277.

[65] A. Patra, L. Simić, and M. Petrova, "Experimental evaluation of a novel fast beamsteering algorithm for link re-establishment in mm-wave indoor wlans," in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2016, pp. 1–7.

[66] W. Jiang, C. Miao, F. Ma, S. Yao, Y. Wang, Y. Yuan, H. Xue, C. Song, X. Ma, D. Koutsonikolas, W. Xu, and L. Su, "Towards environment independent device free human activity recognition," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '18. New York, NY, USA: ACM, 2018, pp. 289–304. [Online]. Available: http://doi.acm.org/10.1145/3241539.3241548

[67] F. Adib, H. Mao, Z. Kabelac, D. Katabi, and R. C. Miller, "Smart homes that monitor breathing and heart rate," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, ser. CHI '15. New York, NY, USA: ACM, 2015, pp. 837–846. [Online]. Available: http://doi.acm.org/10.1145/2702123.2702200

[68] J. Wang, J. Xiong, H. Jiang, K. Jamieson, X. Chen, D. Fang, and C. Wang, "Low human-effort, device-free localization with fine-grained subcarrier information," *IEEE Transactions on Mobile Computing*, vol. 17, no. 11, pp. 2550–2563, Nov 2018.

[69] G. Wang, Y. Zou, Z. Zhou, K. Wu, and L. M. Ni, "We can hear you with wi-fi!" in *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '14. New York, NY, USA: ACM, 2014, pp. 593–604. [Online]. Available: http://doi.acm.org/10.1145/2639108.2639112

[70] F. J. Harris, *Multirate Signal Processing for Communication Systems*, ser. Mobility '08. Prentice Hall, 2004.