Algorithms for robust indoor localization and sensing using off-the-shelf devices

by

Alejandro Blanco Pizarro

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in

Telematic Engineering

Universidad Carlos III de Madrid

Advisor: Joerg Widmer

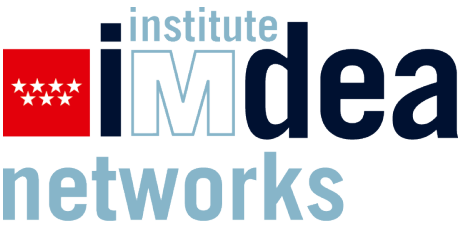
March, 2022

Algorithms for robust indoor localization and sensing using off-the-shelf devices

Prepared by:
Alejandro Blanco Pizarro, IMDEA Networks Institute, Universidad Carlos III de Madrid
Contact: alejandro.blanco@imdea.org

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

This work has been supported by:



Acknowledgements

This thesis is the result of several years of effort and study and it has been one of the toughest endeavors of my life. Although getting a PhD is a single-person academic achievement, my supervisor, Dr. Joerg Widmer, has guided me wisely in every step. Without him I would not have finished it. I want to express my whole gratitude to him for giving me the chance to pursue an academic career and joining his research group.

As in every journey, I have met incredible people that helped me to become a better researcher. I would like to thank Norbert Ludant and Dr. Pablo Jiménez Mateo for their support, specially during the first year. I would like to thank also Dr. Francesco Gringoli and Marco Cominelli for embracing me during my first internship in Brescia. Moreover, I am very grateful to Dr. Michele Rossi and Dr. Francesca Meneghello with whom I have worked in a remote internship at the end of the PhD.

I am thankful to the amazing colleagues of my research group who accompanied me during the stressful days: Dr. Jesús Omar Lacruz, Dr. Claudio Fiandrino, Dr. Hany Assasa, Dolores García, Dr. Guillermo Bielsa, Dr. Joan Palacios, Nina Grosheva, Giulia Attanasio, Serly Moghadas, and Sai Pavan. To them and to the other awesome people of IMDEA Networks, I am grateful for making the institute a pleasant place to work.

Last but not least, I would like to extend my gratitude to my family, who gave me unconditional support, and to my girlfriend Sara, who makes me a better person with her infinite love.

Published Content

This thesis is based on the following published papers:

[1] Blanco, A. and Ludant, N. and Jiménez Mateo, P. and Shi, Z. and Wang, Y. and Widmer J. Performance Evaluation of Single Base Station ToA-AoA Localization in an LTE Testbed. In: *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (IEEE PIMRC 2019).* Istanbul, Turkey, Sep. 2019. https://doi.org/10.1109/PIMRC.2019.8904454

- This paper has been fully integrated in Chapter 3
- The role of the author of the thesis in this work is focused on the design, implementation and experimentation of an LTE localization testbed for a single base station, and the writing of the paper.

[2] Blanco Pizarro, A. and Beltrán, J. P. and Cominelli, M. and Gringoli, F. and Widmer, J. Ubiquitous Localization with Off-the-Shelf IEEE 802.11ac Devices. In: *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services (ACM MobiSys 2021).* Virtual, WI, USA, Jun. 2021. https://doi.org/10.1145/3458864.3468850

- This paper has been fully integrated in Chapter 4
- The author's role in this work is focused on the implementation of the proposed algorithm for decomposing the channel as well as the localization scheme. The author's roles also is focused on the collection and processing of the experiments, implementing state-of-the-art schemes to compared with the proposed system, and the writing of the paper.

[3] Gringoli, F. and Cominelli, M. and **Blanco, A.** and Widmer, J. AX-CSI: Enabling CSI Extraction on Commercial 802.11 ax Wi-Fi Platforms. In: *Proceedings of the 15th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization (WiNTECH 2021).* New Orleans, USA, Feb. 2022. https://doi.org/10.1145/3477086.3480833

- This paper has been fully integrated in Chapter 5

- The author's role in this work is focused on the writing as well as helping with the experiments.

Other publications that have not been used in this thesis are:

[4] Andrés Ramiro, C. and Fiandrino, C. and **Blanco, A.** and Jiménez Mateo, P. and Ludant, N. and Widmer, J. OpenLEON: An End-to-End Emulator from the Edge Data Center to the Mobile User. In: *Proceedings of the 12th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization (WiNTECH 18)*. New Delhi, India, Nov. 2018.. http://doi.org/10.1145/3267204.3267210

[5] Jiménez Mateo, P. and **Blanco, A.** and Ludant, N, and Marugan Borelli, M. and García-García, A. and Loch, A. and Shi, Z. and Wang, Y. and Widmer, J. A Comprehensive Study of Low Frequency and High Frequency Channel Correlation. In: *International Conference on Computing, Networking and Communications (ICNC 2019)*. Honolulu, HI, USA, Feb. 2019. http://doi.org/10.1109/ICCNC.2019.8685565

[6] Fiandrino, C. and **Blanco Pizarro, A.** and Jiménez Mateo, P. and Andrés Ramiro, C. and Ludant, N. and Widmer, J. openLEON: An end-to-end emulator from the edge data center to the mobile user. In: *Computer Communications (COMPUT COMMUN)*. Dec. 2019. http://doi.org/10.1016/j.comcom.2019.08.024

Abstract

Localization has been mainly an optional feature of cellular networks since they have been designed for communication and many location-based services have used GPS to provide more functionalities like navigation, rescue and many more. GPS gets outstanding accuracy in outdoor environments, but its performance drastically degrades in indoor settings since the GPS signals hardly go through walls. However, most human activities are concentrated in indoor environments and location-based services cannot be carried out successfully by GPS. To overcome this limitation, wireless protocols are appealing to fulfill indoor localization requirements while communication is ongoing. Operators, chipset vendors and application developers are paying attention to exploiting location information to provide new applications like augmented reality and indoor navigation. Moreover, localization can be used for network optimization and researchers are actively investigating it. For instance, intelligent handover can exploit location information to guess which Access Point (AP) is the most suitable one before doing the handover. In addition, a range of applications can exploit it as well such as Multiple-Input Multiple-Output beamforming, millimeter-wave beam alignment, etc. For the last decade, sensing has been appealing to not only provide location information but also context awareness. This enables human activity and event recognition, vital sign monitoring, user identification, mapping, imaging, etc. To ensure the good performance of these applications, accurate and ubiquitous positioning is needed. To this end, 5G and the newest Wi-Fi protocols, IEEE 802.11ac and 802.11ax, are becoming the key technologies to provide outstanding indoor localization since they incorporate larger array configurations and wider channel bandwidths than previous wireless protocols.

Researchers have made a great effort to provide indoor localization and decimeter level of accuracy has been achieved. However, this outstanding performance has been evaluated using a great number of APs and assuming that every AP has a clear Line-Of-Sight (LOS) to the device. However, typical indoor wireless deployments tend to have sparse AP densities since they are optimized for coverage and not for localization. For instance, a Wi-Fi infrastructure usually contains one AP per room and a 5G deployment tends to have a limited number of AP as well. Moreover, indoor environments are generally rich in multipath components that interfere with the estimation of the direct path. This

is particularly challenging in Non-Line-Of-Sight (NLOS) settings as obstacles can block the direct path and a system might detect an NLOS path and not the obstructed LOS path. As a result, the performances of state-of-the-art localization schemes drastically degrade their accuracy in realistic deployments.

A localization algorithm that copes well with NLOS settings and wireless deployments with sparse AP densities is needed for precise and pervasive localization. Also, implementing and testing it in cutting edge devices is crucial to exploit the improved hardware features of the newest wireless protocols. Therefore, this thesis aims at providing a framework for accurate localization even in challenging scenarios. Sensing research shares methodologies with localization since sensing applications require extracting location information from NLOS paths as localization does from the direct path. Hence, this thesis also aims at exploring how the proposed localization framework can be used for sensing applications.

We start delving into wireless localization by exploring what an LTE localization system can achieve. This is particularly beneficial since 5G and LTE will coexist for a while until 5G provides ubiquitous coverage. Therefore, LTE needs to fulfill the localization requirements for a range of applications if 5G is not available. To this end, we implement and evaluate an LTE localization system for a single AP using software-defined radios. We observe that LTE achieves a median error of 2 m in LOS cases. However, the LTE performance drastically degrades to 4.6 m of median error in NLOS settings. These results point out that LTE provides a positioning accuracy that complies with a great number of location-based services in LOS. Nevertheless, applications that demand ubiquitous localization may not be correctly carried out in NLOS settings.

To tackle the NLOS issue, we implement UbiLocate, a Wi-Fi location system that copes well with common AP deployment densities and works ubiquitously, i.e., without excessive degradation under NLOS. UbiLocate demonstrates that meter-level median accuracy NLOS localization is possible through (i) an innovative angle estimator based on a Nelder-Mead search, (ii) a fine-grained time of flight ranging system with nanosecond resolution, and (iii) the accuracy improvements brought about by the increase in bandwidth and number of antennas of IEEE 802.11ac. In combination, they provide superior resolvability of multipath components, significantly improving location accuracy over prior work. We implement our location system on off-the-shelf 802.11ac devices. Our experimental evaluation shows an overall improvement of the localization performance by a factor of 2-3.

The latest generation of Wi-Fi standards, IEEE 802.11ax, brings new hardware capabilities that improve the performance of localization and sensing systems. In particular, the 160MHz of channel bandwidth and the four times denser spectrum significantly improve the resolvability of the multipath components compared to its predecessor, IEEE 802.11ac. We present the first tool to collect the most accurate

CSI ever from off-the-shelf devices. To further validate the platform, we carry out a preliminary measurement campaign to compare the localization accuracy of IEEE 802.11ax with 802.11ac. Our results show that, as expected, IEEE 802.11ax provides superior performance improving the accuracy by a factor of 1.75 for LOS and NLOS settings.

Sensing research goes beyond localization since it aims at providing context awareness. We explore the integration of the proposed multipath decomposition algorithm as well as the testbed for sensing applications. In particular, we tackle human respiration rate estimation since it is appealing as it does not require any specialized hardware. Our results show that an accurate respiration rate estimation is possible by decomposing the channel.

In summary, location-based services demand accurate and ubiquitous localization. However, the state-of-the-art localization systems do not cope well with realistic wireless deployments and their positioning performances drastically degrade in these environments. Hence, we provide a localization framework that copes well with realistic wireless deployments and with NLOS settings. We conclude that resolving accurately the multipath components enables pervasive and precise localization. In addition, sensing enables new applications that are helpful in many issues since it provides not only location but also context awareness. Hence, we show that algorithms and testbeds that are designed for localization can be also utilized for sensing applications by tackling respiration rate estimation.

Table of Contents

Acknowledgements v

Published Content vii

Abstract ix

Table of Contents xiii

List of Tables xvii

List of Figures xix

List of Acronyms xxiii

1. Introduction 1

1.1. Introduction . . . 1
1.2. Motivation . . . 2
1.3. Challenges and the main goal of the thesis . . . 3
1.4. Contributions of the thesis . . . 5
1.5. Outline of the thesis . . . 6

2. Background on wireless localization 9

2.1. Location techniques . . . 9
2.1.1. Triangulation . . . 9
2.1.2. Trilateration . . . 11
2.1.3. Hybrid (AoA + distance) . . . 12
2.2. Wireless model . . . 13
2.2.1. Path parameters . . . 14
2.2.2. Overview of the estimation of path parameters . . . 15

3. Performance Evaluation of Single Base Station ToA-AoA Localization in an LTE Testbed 21

3.1. Introduction . . . 21

3.2. LTE background . . . 22
3.2.1. LTE Synchronization . . . 23
3.2.2. LTE Reference Signals . . . 23
3.2.3. Localization in LTE . . . 24
3.3. Localization Process Description . . . 24
3.3.1. AoA Estimation . . . 24
3.3.2. ToA estimation . . . 26
3.4. Implementation . . . 27
3.5. Numerical Results . . . 28
3.5.1. AoA . . . 28
3.5.2. ToA . . . 29
3.5.3. Localization . . . 30
3.5.4. Further observations . . . 30
3.6. Conclusion . . . 32

4. Accurate Ubiquitous Localization with Off-the-Shelf IEEE802.11ac Devices 33
4.1. Introduction . . . 33
4.2. UbiLocate overview . . . 34
4.2.1. Angle estimation . . . 35
4.2.2. Ranging . . . 39
4.2.3. Localization . . . 40
4.3. Implementation . . . 43
4.3.1. Extracting accurate CSI . . . 43
4.3.2. Extracting timestamps . . . 44
4.3.3. Implementation of the FTM procedure . . . 45
4.4. Experimental Evaluation . . . 47
4.4.1. Testbed setups . . . 47
4.4.2. Comparison with other systems . . . 49
4.4.3. High density scenario . . . 50
4.4.4. Medium density scenario . . . 51
4.4.5. Low density scenario . . . 54
4.4.6. Additional Considerations . . . 55
4.5. Related Work . . . 58
4.6. Conclusions . . . 60

5. AX-CSI: Enabling CSI extraction on commercial 802.11ax Wi-Fi platforms 61
5.1. Introduction . . . 61
5.2. The AX-tended CSI extractor . . . 62

5.2.1. CSI data layout 64
5.2.2. Testing the CSI extraction platform beyond 80 MHz with SDRs . 65
5.3. CSI Extraction Performance 67
5.4. Results with HE PHY 69
5.5. IEEE 802.11ax localization performance 71
5.6. Related work 76
5.7. Conclusions 78

6. Respiration Rate Estimation using Commodity Wi-Fi 79
6.1. Introduction 79
6.2. Passive localization 80
6.3. Breathing detection 81
6.3.1. Path parameter estimation 83
6.3.2. Breathing signal extraction and rate estimation 84
6.3.3. CSI cleaning 85
6.4. Respiration evaluation 86
6.5. Conclusions 87

7. Conclusions 93
7.1. Future work 95

References 97

List of Tables

5.1. Mapping of OFDM subcarriers to memory indices for the first PHY table for a generic receiving radio core. For spatial stream **k**, the corresponding PHY table starts at location $\mathbf{k} \cdot \mathbf{2048}$. 64
5.2. Performance comparison between the previous 802.11ac tool (Nexmon CSI) and the 802.11ax extractor (AX-CSI) in terms of CSI extracted per second. 68

List of Figures

2.1. Triangulation example 10
2.2. Trilateration example 12
2.3. Hybrid (AoA + distance) localization example 13
2.4. A signal arriving at the receiver array with a certain angle θ_{rx}. The antenna spacing is d which corresponds to half a wavelength. 16
2.5. Example of an FTM session of 1 burst of for 4 measurements. 19

3.1. LTE location system components 27
3.2. Empirical CDF for AoA and ToA error 29
3.3. CDF of the localization error 30
3.4. Measurement errors of ToA and overall localization (AoA+ToA) in the office. Each point contains the MUSIC spatial spectrum as well as the error values for ToA and localization. Besides, on the right side two MUSIC spectrum and two error values appear to consider whether the door is closed or open 31

4.1. NLOS example with obstructed los path. 36
4.2. Standard FTM (left) sends dedicated messages per pair of nodes and UbiLocate (right) broadcasts a single frame per node for ranging with all other nodes. 39
4.3. ASUS 802.11ac router with custom antenna array. 42
4.4. Phase differences for the antennas pairs. 42
4.5. Enhanced CSI extraction platform with the modifications to collect ToF. 44
4.6. Testbeds. 48
4.7. High density testbed. 50
4.8. Empirical CDF for AoA and ToF error for all APs, and for LOS (solid lines) and NLOS (dashed lines). 51
4.9. Localization performance of UbiLocate compared to state-of-the-art systems. 53
4.10. Empirical CDF for AoA and ToF error of UbiLocate compared to state-of-the-art systems for the low density scenario for all APs. 56

4.11. Localization performance of UbiLocate and state-of-the-art systems for the low density scenario. 56
4.12. UbiLocate location accuracy of different configurations of (number of antennas/bandwidth) 57
4.13. Time complexity for UbiLocate and SpotFi. 58

5.1. Overview of the CSI extraction process in the adopted architecture. Operations are split between the D11 core and the ARM CPU and performed in the order indicated by the arrows. When a target frame is received, the D11 core switches off the radio (1); then, the ARM CPU reads the CSI data (2), sends them to the userspace (3), and finally restarts the radio (4). 62
5.2. Layout in the PHY table of core #0 of the data associated to every OFDM subcarrier for a 160 MHz VHT frame using 2x2 MIMO. Only two spatial streams are reported here; four of them will require up to 8192 values. . . 63
5.3. SDR setup for transmitting 160 MHz frames with 2x2 MIMO using two SDRs with smaller bandwidth. The SDRs need to be externally synchronized. 66
5.4. The latency introduced by the CSI processing chain depends on the number of extracted subcarriers. 68
5.5. Detail of the CSI of 20 MHz frames. The spectral resolution achieved with HE PHY is four times larger than with VHT PHY. Amplitude is measured in arbitrary units as reported by the tool. 69
5.6. CSI extracted from 40 MHz HE frames when the transmitted frames are arbitrarily pre-distorted using filters with a particular frequency response. CSI amplitude is normalized to 1 outside the stop-band region. 70
5.7. Spectrum of a 160 MHz frame overlaid with that of the eight constituting 20 MHz channels. 71
5.8. Amplitude of all the 16 CSI profiles extracted from a single 4x4 160 MHz HE frame. Each row contains the CSI collected at one RX core; columns represent the four spatial streams. 72
5.9. Scenario for the IEEE 802.11ax localization evaluation. The red dot represents the AP position and the blue dots the client positions. 73
5.10. Empirical CDF for AoA and ToF errors. 74
5.11. Empirical CDF for localization errors. 76

6.1. Example of passive localization. 81
6.2. Example of how the path length is affected by the displacement of the chest 82
6.3. Floor plan of the scenario for the respiration evaluation. 87
6.4. Analysis of the raw respiration signals. (a) is the raw CSI phase, (b) is the raw CSI amplitude and (c) is the raw respiration of the proposed scheme. 89

6.5. Respiration analysis for the normal respiration rate. 90
6.6. Respiration analysis for the slow respiration rate. 91

List of Acronyms

AoA Angle of Arrival

AoD Angle of Departure

AP Access Point

bpm breaths per minute

CFO Carrier Frequency Offset

COTS Commercial Off-The-Shelf

CPU Central Processing Unit

CSI Channel State Information

eNB Evolved Node B

FPGA Field-Programmable Gate Array

FTM Fine Time Measurement

GPS Global Positioning System

HE High Efficiency

IoT Internet of Things

LOS Line-Of-Sight

LS Least Square error

LTE Long Term Evolution

MIMO Multiple-Input Multiple-Output

MISO Multiple-Input Single-Output

mmWave Milimiter-wave

MUSIC MUltiple SIgnal Classification

NLOS Non-Line-Of-Sight

OFDM Orthogonal Frequency-Division Multiplexing

RTT Round-Trip Time

SDR Software-Defined Radio

SIMO Single-Input Multiple-Output

SISO Single-Input Single-Output

SLAM Simultaneous Localization and Mapping

SRS Sounding Reference Signal

TdoA Time difference of Arrival

ToA Time of Arrival

ToF Time of Flight

UE User Equipment

ULA Uniform Linear Array

USRP Universal Software Radio Peripheral

VHT Very-High Throughput

WARP Wireless Open-Access Research Platform

1 Introduction

1.1. Introduction

Satellite systems have been widely adopted to provide outstanding position accuracy all over the world and many location-based services have been exploiting such precision. Vehicle navigation, object tracking, rescue and many more applications would have not been possible without them. Nowadays, there exist several satellite systems such as Global Positioning System (GPS), Galileo and GLONASS, that can provide sub-meter level of accuracy [7]. However, these systems become unreliable in indoor environments since the satellite signals suffer from high attenuation due to the building walls. As a result, the decoding of the signal might not be carried out successfully and the positioning fails. If the decoding is possible, the multipath effects will degrade the localization performance. Since most human activities are concentrated in indoor environments and the satellite systems cannot meet the requirements, other wireless protocols have been investigated for localization purposes. In particular, Wi-Fi and cellular networks, such as 3G and LTE, have been extensively used to provide indoor localization.

To do so, any wireless protocol typically deploys Access Points (APs) to provide the clients access to the Internet. To this end, every AP measures location information from the client, typically the Angle of Arrival (AoA) and the distance of the client to the AP. After collecting location data from several APs, which includes distance and/or AoA estimates, a localization algorithm (like trilateration and triangulation) is run to compute the client position. Although Wi-Fi and 3G/LTE are all wireless protocols, their use in indoor localization has been developed in different ways. On the one hand, localization by cellular networks has been motivated by government institutions like the Federated Communication Commission of the United States and the European Commission. In particular, they both requested the cellular network operators to enable a minimum positioning requirement in case of an emergency call. On the other hand, Wi-Fi has not been requested to provide localization requirements, therefore chipset vendors have fully focused on communication. However, researchers have paid much more attention

to Wi-Fi than cellular networks. This was motivated by the fact that researchers can build localization testbeds easily using Commercial Off-The-Shelf (COTS) Wi-Fi devices. The research community has made a great effort over the past two decades to provide accurate positioning while communication is carried out, without disturbing the regular performance of the latter.

In the early 2000s, the pioneering Wi-Fi location systems [8–12] achieved several meters of accuracy in indoor scenarios, which was an outstanding achievement. Despite their efforts, those systems were still not able to provide the same level of accuracy as GPS did in outdoors. The main reason for that poor performance was that they were implemented using the very first Wi-Fi protocols such as IEEE 802.11b/g, and their hardware characteristics were limited. In the early 2010s, the rollout of IEEE 802.11n enabled a great number of localization approaches since it permits estimating the AoA and the Time of Flight (ToF)/Time difference of Arrival (TdoA). In particular, the 3x3 Multiple-Input Multiple-Output (MIMO) enabled AoA as more than one antenna was available, and the broader channels (up to 40MHz) made ToF and TdoA more reliable [13]. As a result, decimeter level indoor localization was achieved [14–17]. In contrast, the accomplished positioning accuracy by cellular networks was more limited during the last decades. For instance, 3G enabled TdoA and received signal strength to derive the distance of the client, but the reported errors were above 50 m. More accurate approaches were enabled by LTE but still, the errors were around 20 m [18].

1.2. Motivation

Nowadays, cellular network operators, chipset vendors and application developers are paying attention to exploiting location information to provide new location-based services, for example, robot navigation, autonomous driving, augmented reality and many more applications. In addition, positioning can be used for network optimization like predictive handover, MIMO beam-forming, beam training overhead reduction in Milimiter-wave (mmWave) communications and many more. The researchers' endeavors go beyond localization and they aim at providing context awareness. To this end, sensing is becoming a potential extension of indoor localization since it inherits the same methodology of extracting features of the radio-frequency signals. In particular, localization requires getting the direct path, i.e., the path that goes from the transmitter to the receiver, since this path conveys the location information of the client. Sensing, instead, aims at exploiting Non-Line-Of-Sight (NLOS) paths that provide location information from reflectors in the environment. Hence, a sensing application can estimate the position of persons and objects. This enables to count the number of people in a room and to detect intruders. Moreover, sensing can detect and understand small variations of reflected paths so that it allows detecting human activities, monitoring vital signs, and many more

applications [19–22].

The applications mentioned above require accurate and ubiquitous positioning. While sub-meter level of localization accuracy is achieved in dense APs deployments and with high channel qualities, i.e., every AP has a clear Line-Of-Sight (LOS) to the client, accurate and pervasive localization is remarkable challenging in sparse APs densities and in NLOS settings. In particular, the majority of localization algorithms require coverage from several APs. However realistic indoor wireless deployments are optimized for coverage, and therefore two APs, at most, will be available per room for positioning. In addition, indoor environments are rich in multipath components that interfere with the estimation of the direct path, then a system needs to minimize the influence of NLOS paths. This becomes more challenging in crowded indoor scenarios where moving obstacles block the direct path making it weaker than reflected paths. Therefore, a system might take an NLOS path since it cannot discriminate the obstructed direct path, which results in an unreliable position estimate. To the best of our knowledge, these two issues, sparse AP densities and NLOS setting, have not been extensively analyzed and the majority of the state-of-the-art systems do not perform well in such conditions.

To deal with that, 5G and the new Wi-Fi standards, IEEE 802.11ac and 802.11ax, are becoming promising candidates since they incorporate hardware features that improve localization accuracy. In particular, the higher number of antennas and the broader channels alleviate multipath issues significantly because the angular and the time resolutions improve by increasing the number of antennas and the bandwidth. As a result, a location system improves the resolvability of the direct path and a more precise localization can be achieved. Recent studies have experimentally analyzed the potential performance of an indoor localization system based on IEEE 802.11ac [23–25]. However, the authors tackled common indoor environments that have been extensively analyzed before. While the majority of the Wi-Fi localization schemes have been validated using COTS, few practical works for 5G can be found [26, 27]. This is because accessing real traces from operator base stations is usually restricted.

1.3. Challenges and the main goal of the thesis

As we discussed previously, outstanding indoor localization accuracy can be achieved but it is usually limited to scenarios where the APs have a clear LOS to the client and with a dense AP density. These two conditions might not be available in realistic indoor wireless deployments. Hence, the localization accuracy degrades drastically. To deal with that, this thesis aims at addressing the following challenges:

Robust multipath discriminator. An obstructed direct path makes the NLOS paths stronger. As a consequence, the estimation of the direct path becomes more challenging as other paths have more influence on the received wireless channel, i.e.,

the direct path might be masked by others. Therefore, a direct path estimator needs to cope well with the interference of reflected paths and resolve all of them accurately. If not, either a system might extract the direct path but it will be influenced by other paths, or a system takes, instead, an NLOS path as a direct path. For both cases, the AoA and the ToF estimates from the direct path are unreliable and the positioning might fail. This is particularly decisive in sensing applications as well. As explained in the motivation, sensing requires the estimation of NLOS paths to get location information from the objects in the environment. The NLOS paths are usually weaker and estimating them is a similar problem to get the obstructed direct path for localization. Hence, extracting low power paths is also critical for sensing applications. Therefore, both localization and sensing demand a robust multipath discriminator.

Sparse AP density. Realistic wireless deployments contain two APs, at most, per room. Therefore, the localization relies on few estimates and not all the APs provide valuable location information. For example, consider that two APs are available for positioning. One has a clear LOS, which provides a precise position, and the other has an obstacle in the middle of the direct path and its localization is unreliable. Hence, combining equally the two estimates results in a non-optimal solution since the location of the client will be shifted to a wrong position due to the influence of the unreliable AP. Therefore, detecting and filtering unreliable estimates is crucial for accurate positioning.

Testbeds. Testbeds are also a key component of the localization systems since they are used to implement and validate the proposed algorithms. In particular, testbeds enable to extract Channel State Information (CSI) data for real traces so that a system estimates the AoA and the ToF. Depending on the technology, testbeds cope with different challenges as we discuss below.

- On the one hand, cellular networks testbeds usually rely on specialized hardware and researchers implement on them custom software. On the other hand, there are open source software implementations [28, 29] that support end-to-end LTE communications. However, they are designed to provide stable communication rather than being specialized software for LTE localization. As a result, modifications are necessary to enable positioning.

- The majority of the Wi-Fi systems have been extensively developed on COTS devices. However, these systems need to extract the CSI data from the COTS firmware, and chipset vendors restrict access to the firmware by default. Hence, firmware hacking and reverse engineering are needed to enable CSI extraction. In addition, the extracted raw CSI data is contaminated by hardware imperfections that do not affect the communication but make localization unreliable. Therefore, these imperfections have to be handled before extracting AoA and/or ToF.

The main goal of the thesis is to provide a framework that aims at achieving

accurate and pervasive indoor localization even in challenging environments, where current localization schemes do not perform well. Moreover, sensing applications also exploit position information from reflected paths as localization does from the direct path. Hence, algorithms as well as testbeds that are designed for localization can be utilized for sensing research. Therefore, we also aim at exploring how sensing can exploit the proposed framework.

1.4. Contributions of the thesis

In particular, this thesis makes the following main contributions:

Accurate path decomposition. As explained before, extracting the obstructed LOS path from the received channel is very challenging as it is likely to be masked by reflected paths. Hence, resolving all the paths to accurately extract the direct path without any interference from NLOS paths enables an accurate AoA and ToF estimation of the client. Therefore, we develop a path decomposition algorithm that iteratively estimates the strongest path and removes it from the received channel, so that weaker paths can be extracted. This path cancellation ends once all the paths have been extracted. This results in a coarse estimation of the channel since the paths are highly correlated and a residual interference still remains. Hence, we apply a refinement by a Nelder-Mead search which aims at minimizing the module of subtraction between the received channel and the coarse estimation of the channel from the previous step. Therefore, the resulting estimated channel is relatively free of interference and all the paths are accurately extracted. We then obtain the AoA and ToF estimates from the direct path to enable precise positioning. In addition, the scope of this algorithm goes beyond localization and any application that requires extracting reflected paths can exploit it. We enumerate the potential applications that can use it: simultaneous localization and mapping, wireless imagining, vital signs monitoring and many more applications.

Localization algorithm. After extracting the AoA and ToF of the client, every AP estimates its coordinates. Thus, a system combines the estimates from several APs to compute the overall localization. However, not all the APs provide the same level of precision since many of them are in NLOS, which make their estimation unreliable. To deal with that, we propose a localization algorithm that first filters inaccurate APs. Second, it computes the overall localization by combing client position estimates in terms of the quality of the channel giving more weight to the APs that provide good estimates. In particular, we measure the channel quality by the mean excess delay. This metric measures the time delay by averaging the time of arrival of the paths according to their power. Hence, if the direct path is much stronger than reflected paths, this metric has a value close to zero. When the NLOS paths are stronger, the mean excess delay has a larger value. This is particularly beneficial in applications that are sensitive to higher outliers

since the larger localization errors are mainly caused by unreliable position estimates. We can enumerate the potential applications that can use it: robot navigation, autonomous driving, medical equipment location and many more applications.

Testbeds. Implementing the localization algorithms using cutting edge platforms results in a more precise localization because the improved hardware features of the latest wireless protocols increase the resolvability of multipath components. To this end, we provide an LTE localization testbed that is fully based on srsLTE [28]. By default, srsLTE is not a specialized software for localization, hence we modify it to support the Sounding Reference Signal (SRS). This signal occupies almost the whole bandwidth and therefore provides high time resolution. We also provide COTS testbeds for the newest Wi-Fi protocols, IEEE 802.11ac and 802.11ax, as well as a methodology to tackle internal hardware imperfections.

Profound validation. We carry out extensive measurement campaigns to validate the performance of the proposed localization algorithms. We do comprehensive comparisons of them with state-of-the-art schemes.

Sensing validation. Sensing also exploits localization information from reflected paths as localization does from the direct path. Hence, we aim at validating that the proposed path decomposition algorithm is capable of extracting accurately reflected paths. To this end, we tackle human respiration rate estimation since it requires getting the path of the signal that bounces off the chest without any interference from other paths. We implement and validate a human respiration rate estimator that achieves a low rate of errors.

1.5. Outline of the thesis

We describe below the outline of the thesis as well as specific contributions and findings of every chapter.

Chapter 2. We give an overview of wireless localization. We provide details about the wireless channel model, the path parameters, i.e., the AoA and ToF, that are contained in the received channel, and how a system can localize a client.

Chapter 3. 5G enables outstanding localization performance but the 5G rollout will take years to provide ubiquitous coverage. Therefore, 5G and Long Term Evolution (LTE) will coexist for a while. Hence, evaluating the LTE localization performance is important to understand which applications can be fulfilled without 5G. In this chapter, we implement an LTE [1] localization system using software-defined radios. The system fuses angular and distance information of the client to localize it with a single AP. The distance is extracted through a Time of Arrival (ToA) estimation using the LTE SRS, and the AoA is measured using the MUltiple SIgnal Classification (MUSIC) algorithm. The evaluation shows a localization median error of 2 m and 4.6 m for LOS and NLOS,

respectively. Despite the low bandwidth of LTE, the LOS precision that LTE provides is sufficient for a range of application, while it may fail in fulfilling the localization requirements in NLOS conditions.

Chapter 4. The previous chapter validated that NLOS degrades localization accuracy significantly. In particular, the NLOS issue arises in many of the regular indoor wireless deployments because walls, pieces of furniture and moving objects can block the direct path. Hence, we delve into robust localization to deal well with realistic deployments without an excessive degradation in NLOS setting. To this end, we propose UbiLocate [2], an indoor localization system that achieves NLOS meter-level median accuracy. Ubilocate provides robust indoor positioning through (i) an innovative angle estimator based on a Nelder-Mead search, (ii) a fine-grained time of flight ranging system with nanosecond resolution, and (iii) the accuracy improvements brought about by the increase in bandwidth and number of antennas of IEEE 802.11ac. In combination, they provide superior resolvability of multipath components, significantly improving location accuracy over prior work by a factor of 2 and 3 for LOS and NLOS settings. We implement our location system on off-the-shelf 802.11ac devices.

Chapter 5. Technology is constantly evolving and it is crucial to make available cutting edge platforms to researchers. In particular, the new IEEE 802.11ax standard enables a much denser spectrum and larger bandwidths than its predecessor, IEEE 802.11ac. These features are essential to achieve a more precise localization in challenging indoor scenarios. We present the first system ever capable of extracting CSI from 802.11ax consumer devices using the Broadcom 43684 Wi-Fi chipset. This platform [3] can extract up to 160 MHz-wide CSI using 4x4 MIMO, and it is compatible with the latest HE PHY. To further validate the usefulness of this platform, we evaluate the localization performance of an 802.11ax implementation compared to an 802.11ac one. We conclude that 802.11ax provides superior accuracy in LOS and NLOS improving the precision by a factor of 1.75. This evaluation is not reported in the paper [3] which this chapter is based on. We then enable a custom implementation of Fine Time Measurement on the IEEE 802.11ax devices to apply the same localization scheme proposed in Chapter 4.

Chapter 6. Going beyond localization enables many useful applications in the field of health care, security and many more. In this chapter, we investigate how sensing research can exploit the proposed localization framework. In particular, we tackle human respiration rate estimation. To this end, we apply the proposed channel decomposition algorithm to extract the path of the signal that bounces off the human chest. Therefore, we can extract this path relatively free of interference from other paths. As a consequence, we can recover the respiration signal that is contained in this path, and estimate the respiration rate. We implement this system using the COTS Wi-Fi device presented in Chapter 4. Our preliminary results show that the system is capable of getting the respiration signal with low breathing rate errors.

Chapter 7. This chapter concludes the thesis by summarizing our findings and discussing possible research directions.

2 Background on wireless localization

In this chapter, we first describe in detail the algorithms for localization assuming that the Angle of Arrival (AoA) and distance of the client are known. We then explain the wireless model where we introduce how the path parameters (AoA, distance) are contained in the received channel and we give an overview on how to extract the path parameters.

2.1. Location techniques

Indoor wireless localization has been deeply investigated using many wireless technologies like Wi-Fi, cellular networks, Bluetooth, ultra wide-band, and visible light communication [30]. A typical indoor wireless deployment consists of several Access Points (APs) that provide Internet access to clients. To do so, a pair of client/AP establishes a communication and exchanges frames by sending radio-frequency signals. While the communication is carried out, the AP can estimate the location components of the client such as the AoA and the distance from the client signal. Finally, a system combines these location components of several APs with the known positions of the APs to compute the client position.

Consider A as the number of APs. We denote the coordinates of the a-th AP as $\boldsymbol{p_a} = [x_a \ y_a]^T$ where $a = 1, \ldots, A$ and $(\cdot)^T$ means the transpose operator. We denote the estimated client AoA and distance as $\hat{\theta_a}$ and $\hat{d_a}$ for the a-th AP. We also denote the client position as $\boldsymbol{p_c} = [x_c \ y_c]^T$.

Several location algorithms estimate the client coordinates, we describe the most relevant ones below.

2.1.1. Triangulation

It is an angle-based localization algorithm that involves the cooperation of at least two APs. Figure 2.1 illustrates this algorithm. In the figure, three APs have estimated the client AoA, and a system merges all these estimates to get the client coordinates. In

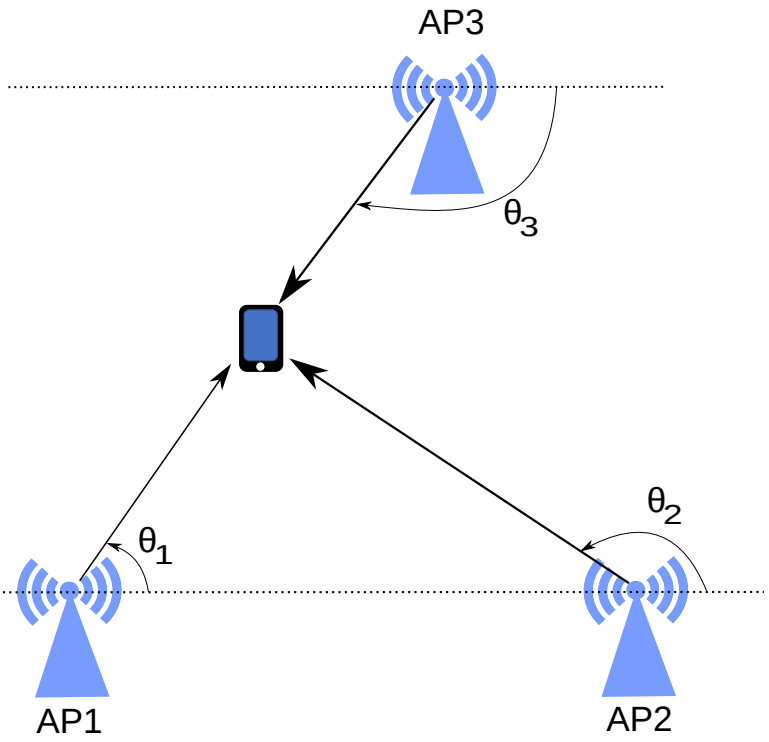


Figure 2.1: Triangulation example

a 2D space, we can express the AoA in terms of the coordinates of the a-th AP and the client as follows:

$$tan(\hat{\theta}_a) = \frac{y_c - y_a}{x_c - x_a} \ . \tag{2.1}$$

If we separate the coordinates of the AP from the client ones, we have the following equation:

$$x_c tan(\hat{\theta}_a) - y_c = x_a tan(\hat{\theta}_a) - y_a \ . \tag{2.2}$$

The expression above can be expressed as a linear system to consider all the APs in a matrix form:

$$\boldsymbol{Q}\boldsymbol{p}_c = \boldsymbol{f} \ , \tag{2.3}$$

with $\boldsymbol{Q}$ is a matrix of size $A \times 2$, $\boldsymbol{p}_c$ is the client position and $\boldsymbol{f}$ is a vector of size $A \times 1$

In particular, $\boldsymbol{Q}$ and $\boldsymbol{f}$ contain the following:

$$\boldsymbol{Q} = \begin{bmatrix} tan(\hat{\theta}_1) & -1 \\ tan(\hat{\theta}_2) & -1 \\ \vdots & \vdots \\ tan(\hat{\theta}_A) & -1 \end{bmatrix}, \boldsymbol{f} = \begin{bmatrix} x_1 tan(\hat{\theta}_1) - y_1 \\ x_2 tan(\hat{\theta}_2) - y_2 \\ \vdots \\ x_A tan(\hat{\theta}_A) - y_A \end{bmatrix} \ .$$

By applying the Least Square error (LS) method, the estimated client position is given

by the LS solution:

$$\hat{\boldsymbol{p}}_c = (\boldsymbol{Q}^T\boldsymbol{Q})^{-1}\boldsymbol{Q}^T\boldsymbol{f} \ . \tag{2.4}$$

2.1.2. Trilateration

Trilateration is a distance-based localization scheme that requires that every AP estimates the distance to the client. In particular, every estimated distance defines a set of possible positions of the user. These positions are inside a circumference where its center is the position of the AP and the radius is the estimated distance. Hence, the client position is the point where all the circumferences intersect. Figure 2.2 illustrates the algorithm.

For the a-th AP, the equation of the circumference is expressed as follows:

$$(x_c - x_a)^2 + (y_c - y_a)^2 = r_a^2 \ , \tag{2.5}$$

with $r_a = \hat{d_a}$.

However, this is a non-linear equation. To linearize it, we use the j-th constraint. Therefore we add and subtract x_j and y_j as follows:

$$(x_c + x_j - x_j - x_a)^2 + (y_c + y_j - y_j - y_a)^2 = r_a^2 \ , \tag{2.6}$$

with $a = 1, 2, \ldots, j-1, j+1, \ldots, A$

Reordering and grouping the above equation results in:

$$(x_c - x_j)(x_a - x_j) + (y_c - y_j)(y_a - y_j) = \frac{1}{2}[r_j^2 - r_a^2 - d_{aj}^2] \ , \tag{2.7}$$

where $d_{aj} = \sqrt{(x_a - x_j)^2 + (y_a - y_j)^2}$.

If we arbitrarily select the first constraint ($j = 1$), we have a system of linear equations:

$$\begin{cases} (x_c - x_1)(x_2 - x_1) + (y_c - y_1)(y_2 - y_1) = 1/2\,[r_1^2 - r_2^2 - d_{21}^2] = b_{21} \\ (x_c - x_1)(x_3 - x_1) + (y_c - y_1)(y_3 - y_1) = 1/2\,[r_1^2 - r_3^2 - d_{31}^2] = b_{31} \\ \vdots \\ (x_c - x_1)(x_A - x_1) + (y_c - y_1)(y_A - y_1) = 1/2\,[r_1^2 - r_A^2 - d_{A1}^2] = b_{A1} \end{cases} \tag{2.8}$$

The above set of linear equations can be expressed in a matrix form. For simplicity, we use the same notation as in Section 2.1.1 for the matrix Q and the vector f. Thus, we express the set of linear equations as follows:

$$\boldsymbol{Q}\boldsymbol{p}_c' = \boldsymbol{f} \ , \tag{2.9}$$

with $\boldsymbol{Q}$ is a matrix of size $(A-1) \times 2$, $\boldsymbol{p}_c'$ is a vector of 2×1 and $\boldsymbol{f}$ is a vector of size $(A-1) \times 1$.

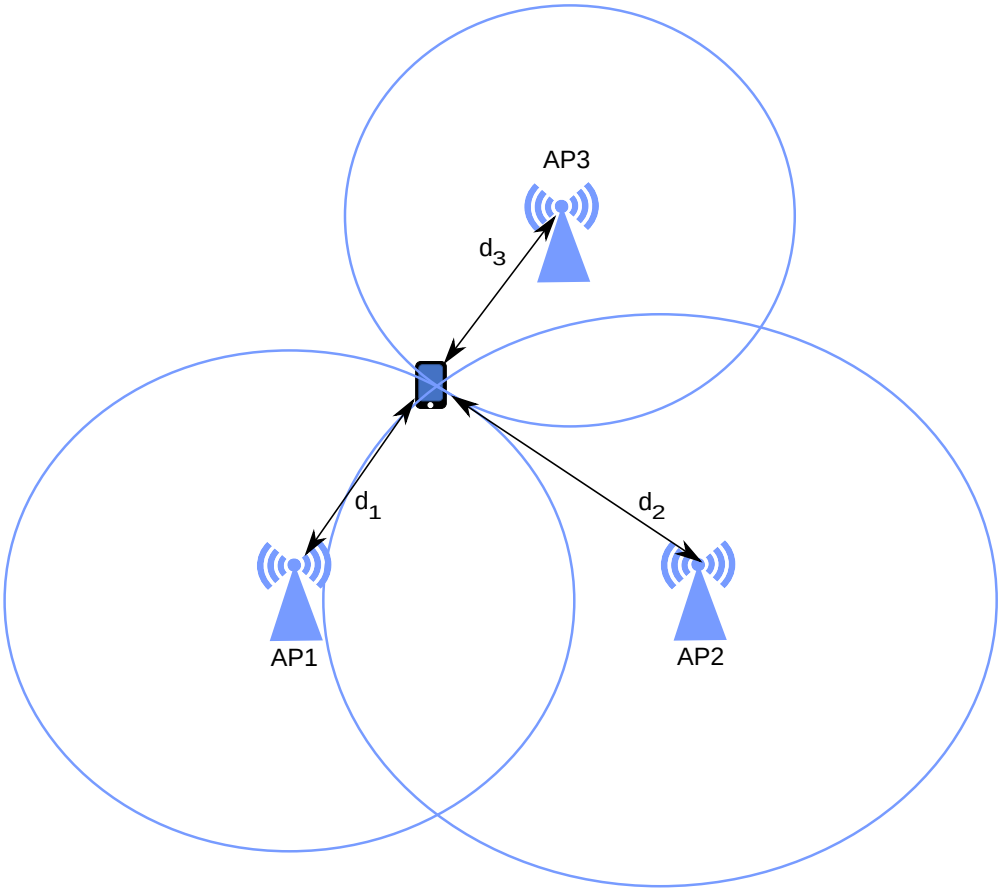


Figure 2.2: Trilateration example

In particular:

$$\boldsymbol{Q} = \begin{bmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \\ \vdots & \vdots \\ x_A - x_1 & y_A - y_1 \end{bmatrix}, \boldsymbol{p}'_c = \begin{bmatrix} x_c - x_1 \\ y_c - y_1 \end{bmatrix}, \boldsymbol{f} = \begin{bmatrix} b_{21} \\ b_{31} \\ \vdots \\ b_{A1} \end{bmatrix}$$

By applying the LS method, we can find the solution as:

$$\hat{\boldsymbol{p}}'_c = (\boldsymbol{Q}^T \boldsymbol{Q})^{-1} \boldsymbol{Q}^T \boldsymbol{f} \ . \tag{2.10}$$

Finally, the estimated client coordinates correspond to the LS solution plus the coordinates of the first AP: $\hat{\boldsymbol{p}}_c = \hat{\boldsymbol{p}}'_c + \boldsymbol{p}_1$.

2.1.3. Hybrid (AoA + distance)

In this approach, every AP estimates the coordinates of the client by combining the AoA and the distance estimates. Figure 2.3 illustrates this localization scheme, where the

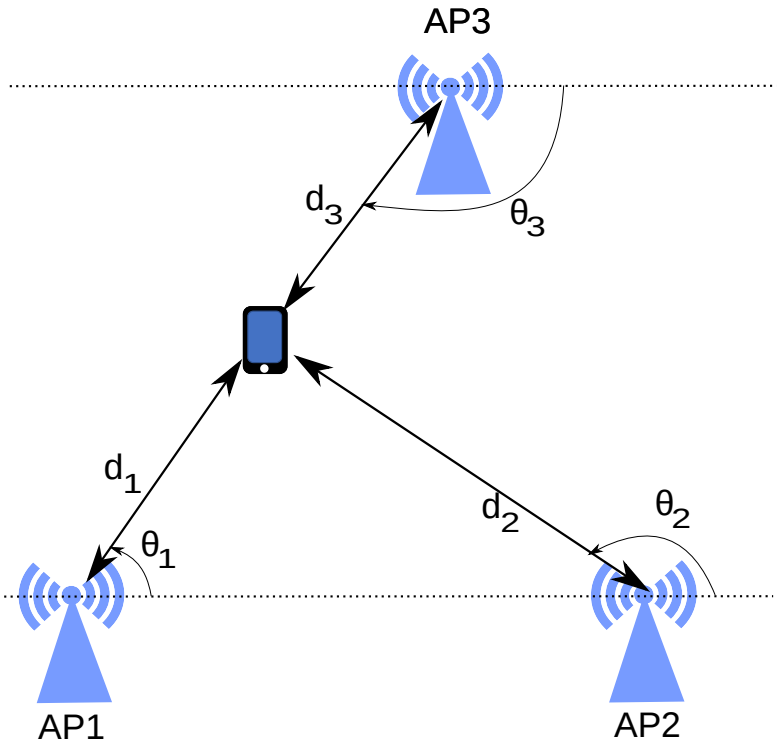


Figure 2.3: Hybrid (AoA + distance) localization example

three AP estimate the position of the user. For the a-th AP, the estimated location of the client is given by:

$$\hat{\mathbf{p}}_{a,c} = \mathbf{p}_a + \hat{d}_a \begin{bmatrix} \cos\hat{\theta}_a \\ \sin\hat{\theta}_a \end{bmatrix} . \tag{2.11}$$

Finally, a system combines the position estimates from all the APs to get a more robust client location. To do so, a weighted average is applied as follows:

$$\hat{\mathbf{p}}_c = \frac{\sum_{a=1}^{A} \hat{\mathbf{p}}_{a,c} \cdot w_a}{\sum_{a=1}^{A} w_a} . \tag{2.12}$$

The weights can be selected according to different strategies. For example, the most simple one could be a regular mean where the weights have the same value. Another strategy might select the weights in terms of the distance or the channel quality which could indicate the reliability of the position estimates.

2.2. Wireless model

In an indoor environment, the transmitted signal propagates through a multipath channel. Hence, the signal traverses through multiple paths and the received signal is superimposed. Every path is defined by the path parameters. These are components

of location information that define how the signal has traveled from the transmitter to the receiver, and they are contained in the wireless channel. Hence, this section aims at explaining the wireless channel and the path parameters. For localization, a system needs to estimate AoA and/or the distance of the client that are path parameters of the direct path. We then give an overview on how to estimate the AoA and the distance.

2.2.1. Path parameters

We start this section by explaining the path parameters for the Multiple-Input Multiple-Output (MIMO) case since it is the most general type of communication. We provide also details about how the received MIMO channel is defined in terms of the path parameters. We explain the channel for the other types of communications, which are Single-Input Multiple-Output (SIMO), Multiple-Input Single-Output (MISO) and Single-Input Single-Output (SISO).

Consider a MIMO system where the transmitter and the receiver have Uniform Linear Arrays (ULAs) of L and M antennas with an antenna spacing of half a wavelength. The transmitter sends a set of Orthogonal Frequency-Division Multiplexing (OFDM) signals $\boldsymbol{s}[k] = [s_0[k], s_1[k], ..., s_{L-1}[k]]$ over K subcarriers and L antennas. The signals propagate to the receiver through a multipath channel with P different paths that are characterized by:

• **Complex attenuation γ_p.** The signal suffers an attenuation of γ_p along path p.

• **Angle of arrival $\theta_{rx,p}$.** The signal arrives at each antenna with a phase delay determined by the antenna spacing. The phase shift $[\phi(\theta_{rx,p})]_m$ at the m-th receive antenna as function of the AoA for the p^{th} path is given by:

$$[\phi(\theta_{rx,p})]_m = e^{-j2\pi m \sin(\theta_{rx,p}) d/\lambda} \ , \tag{2.13}$$

where d is the antenna spacing of the ULA.

Since the antenna spacing is defined as half a wavelength, i.e., $d = \lambda/2$. This simplifies the above equation:

$$[\phi(\theta_{rx,p})]_m = e^{-j\pi m \sin(\theta_{rx,p})} \ . \tag{2.14}$$

We express the steering vector, i.e., the vector of phase shifts for the whole array as:

$$\boldsymbol{\phi}(\theta_{rx,p}) = [[\phi(\theta_{rx,p})]_0, ..., [\phi(\theta_{rx,p})]_{M-1}]^T \ . \tag{2.15}$$

• **Angle of departure $\theta_{tx,p}$.** Similarly, $[\phi(\theta_{tx,p})]_l$ is the phase shift for the l^{th} transmit antenna as a function of the Angle of Departure (AoD):

$$[\phi(\theta_{tx,p})]_l = e^{-j\pi l \sin(\theta_{tx,p})} \ , \tag{2.16}$$

and we denote the vector of phase shifts for the whole array by $\boldsymbol{\phi}(\theta_{tx,p}) = [[\phi(\theta_{tx,p})]_0, ..., [\phi(\theta_{tx,p})]_{L-1}]^T$.

• **Path delay τ_p.** Each path p experiences a different propagation delay determined by its length. In the frequency domain, this delay represents a phase shift $\psi(\tau_p)[k]$ between adjacent subcarriers:

$$\psi(\tau_p)[k] = e^{-j2\pi k\Delta_f \tau_p} \ , \tag{2.17}$$

where Δ_f is the spacing between consecutive subcarriers.

With the parameters above, we can express the MIMO channel as follows:

$$\mathbf{H}[k] = \sum_{p=0}^{P-1} \boldsymbol{\phi}(\theta_{rx,p})\gamma_p \boldsymbol{\phi}^{\mathrm{H}}(\theta_{tx,p})\psi(\tau_p)[k] \ , \tag{2.18}$$

where $(\cdot)^H$ is the Hermitian operator. The received signal is:

$$\mathbf{y}[k] = \mathbf{H}[k]\mathbf{s}[k] + \mathbf{w}[k] \ , \tag{2.19}$$

where $\mathbf{w}[k]$ is L-dimensional white Gaussian noise in the frequency domain, i.e., $\mathbf{w}[k] = [w_0[k], w_1[k], ..., w_{L-1}]$. For a known $\mathbf{s}[k]$, we can then estimate the channel as:

$$\hat{\mathbf{H}}[k] = \mathbf{y}[k]\mathbf{s}^*[k] = \hat{\mathbf{H}}[k] = \mathbf{H}[k] + \hat{\mathbf{w}}[k] \ , \tag{2.20}$$

where $(\cdot)^*$ is the conjugate operator.

The hardware configuration of transmitter and receiver can differ. It might that none of them have an array or only one of them. In particular, MISO does not consider AoA since no array is used at the receiver side, SIMO does not consider AoD and SISO neither AoA nor AoD. The SISO channel is expressed as:

$$\mathbf{H}_{SISO}[k] = \sum_{p=0}^{P-1} \gamma_p \psi(\tau_p)[k] \tag{2.21}$$

2.2.2. Overview of the estimation of path parameters

The direct path contains the distance and the AoA of the client. Therefore, extracting them accurately from the observed received channel is needed for accurate positioning. We start this subsection by explaining the most simple case where the AoA is estimated assuming that a single path arrives at the receiver. We generalize this case to address the AoA estimation in a multipath environment. Finally, we show how the distance can be estimated.

2.2.2.1. AoA estimation in a single path channel

Consider a SIMO communication, where the transmitter has a single antenna and the receiver has a ULA with an antenna spacing of half a wavelength. We also consider that only one signal arrives at the receiver, i.e., the transmitter sends a signal which propagates through a single path channel and arrives at the receiver with a certain incident angle, θ_{rx}. Figure 2.4 illustrates this case. In the example, the signal arrives with a certain delay at antenna 1 compared to antenna 0. In particular, this delay is caused by the distance of $d \cdot sin(\theta_{rx})$. Hence, it introduces a phase shift that we already defined in Equation (2.13).

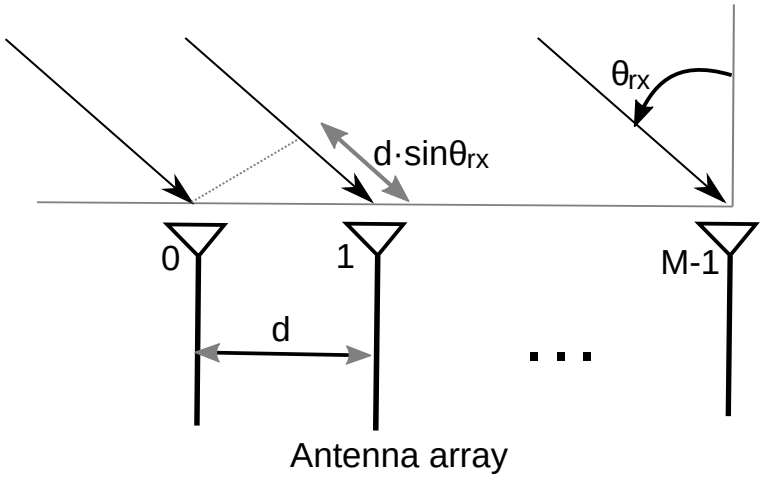


Figure 2.4: A signal arriving at the receiver array with a certain angle θ_{rx}. The antenna spacing is d which corresponds to half a wavelength.

The observed received channel is expressed as follows:

$$\hat{\mathbf{H}}_{SIMO}[k] = \boldsymbol{\phi}(\theta_{rx})\gamma\psi(\tau)[k] + \boldsymbol{w}[k] \ , \tag{2.22}$$

The AoA is extracted by finding the angular component of the received channel over multiple antennas. Selecting one subcarrier, the AoA can be estimated by the observed phase difference between two consecutive antennas as:

$$\hat{\theta} = arcsin(\hat{\Delta}_{m,k}/\pi) \ , \tag{2.23}$$

where $\hat{\Delta}_m$ is the observed phase difference between the antennas m and $m-1$ for the subcarrier k.

To remove the effects of the noise, the above operation can be averaged over several pairs of antennas and subcarriers to get a smoother AoA estimate.

2.2.2.2. AoA estimation in a multipath channel

In an indoor environment, the received signal is usually a superimposed signal, where every signal arrives with an incident angle. We express the multipath channel as follows:

$$\hat{\mathbf{H}}_{SIMO}[k] = \sum_{p=0}^{P-1} \boldsymbol{\phi}(\theta_{rx,p})\gamma_p\psi(\tau_p)[k] + \boldsymbol{w}[k] \ , \tag{2.24}$$

The very first AoA estimator was designed to get the AoA of a superimposed signal based on beamforming techniques [31]. In particular, this algorithm steers the received channel at a certain angle (θ) and measures the output power. We express the output power for a certain angle as follows:

$$P(\theta) = \boldsymbol{\phi}^H(\theta)\boldsymbol{R}\boldsymbol{\phi}(\theta) \ , \tag{2.25}$$

with R is the correlation matrix of the array which we define it as:

$$\boldsymbol{R} = \frac{1}{K}\sum_{k=0}^{K-1} \hat{\boldsymbol{H}}^H_{SIMO}[k]\hat{\boldsymbol{H}}_{SIMO}[k] \ . \tag{2.26}$$

Therefore, the estimated AoA is the one that maximizes the output power:

$$\hat{\theta} = \arg\max_{\theta} P(\theta) \quad \frac{-\pi}{2} \leq \theta \leq \frac{\pi}{2} \ . \tag{2.27}$$

The resolution of this approach depends on the number of antennas and might not cope well with closer incident signals. To deal with that, researchers proposed super-resolution algorithms by applying an eigendecomposition. The most famous algorithm is the MUltiple SIgnal Classification (MUSIC) algorithm [32] which separates the signal and the noise sub-spaces. The signal sub-space contains the eigenvectors associated with the incident signals and the remaining eigenvectors, i.e., the eigenvectors that are not in the signal sub-space, are associated with the noise sub-space. In particular, MUSIC exploits the eigenvector property of orthogonality. This property defines that the eigenvectors are orthogonal to each other, i.e., the output of the scalar product of two eigenvectors is zero. Hence, the signal sub-space is orthogonal to the noise sub-space. MUSIC finds the AoA by multiplying a set of steering vectors, each one associated with a possible AoA, by the noise sub-space. The AoA related to the steering vector which is orthogonal to the noise sub-space is the AoA of the incoming signal.

However, MUSIC was designed to work with uncorrelated signals and signals are highly correlated in a multipath environment since they come from the same source. This is common in indoor environments where walls, pieces of furniture and people moving around create reflected signals. In such conditions, the eigendecomposition may fail and the signal subspace is not correctly computed. To overcome this limitation, compressed

sensing [33] is appealing since it copes well with highly correlated signals. Nevertheless, its time complexity is quite demanding since it requires complex optimization solvers. Therefore, its usability can be prohibitive in real-time applications. In Chapter 4, we provide the details of our proposed algorithm that estimates all the path parameters and achieves better performance than state-of-the-art estimators while having lower time complexity.

2.2.2.3. Distance estimation

The path delay represents the propagation time of the signal from the transmitter to the receiver. Hence, the delay can be multiplied by the speed of light to get the distance. However, in practice, the transmitter and the receiver are not typically synchronized. As a consequence, the receiver does not know when the transmitted signal was sent and therefore, the observed delay at the receiver has a time uncertainty. We summarize the main two methods to obtain the distance below.

Attenuation-based: The signal path loss depends on the distance [34,35]. Therefore, the estimated attenuation is used to derive the distance as follows:

$$P_{rx} = P_0 - 10\mu log_{10}(d), \tag{2.28}$$

where P_{rx} is the received power in dB, P_0 is the received power at a distance of 1 m in dB, and μ is the path loss exponent.

Consequently, the distance is expressed as follows:

$$\hat{d} = 10^{(P_0 - P_{rx})/(10\mu)} \tag{2.29}$$

ToF-based: As explained before, the transmitter and the receiver are not typically synchronized, hence the observed path delay at the receiver is unreliable for distance estimation. To overcome this limitation, ToF-based approaches aim at synchronizing the communication by the cooperation of multiple devices. Either a system gets the absolute value of the path delay by having synchronized receivers and using one of them as a reference [36], or it uses a timestamp-based technique that allows getting the Round-Trip Time (RTT). This timestamping approach has been standardized by the IEEE in the IEEE 802.11 protocol called Fine Time Measurement (FTM). FTM measures the distance between the AP and the client which are the Responder and the Initiator, respectively. Figure 2.5 shows an example of an FTM session where packets and their associated acknowledgments are exchanged. In particular, four timestamps are needed per measurement, which are t_1, t_2, t_3 and t_4. The Responder starts the measurement by sending a packet and it records the time of departure of the packet in t_1. This packet is received by the Initiator and the Initiator records the time of arrival of the packet in t_2. The Initiator sends back an ACK and it records the time of departure of the ACK in t_3.

Finally, the ACK is received by the Responder and it records the time of arrival of the ACK in t_4. Then, RTT is computed as follows:

$$RTT = \frac{1}{n}\sum_{x=1}^{n}((t_4(x) - t_1(x)) - (t_3(x) - t_2(x))) \ , \tag{2.30}$$

where n is the number of measurements.

We are able to determine the distance in meters using the speed of light c as $\hat{d} = c \cdot RTT/2$.

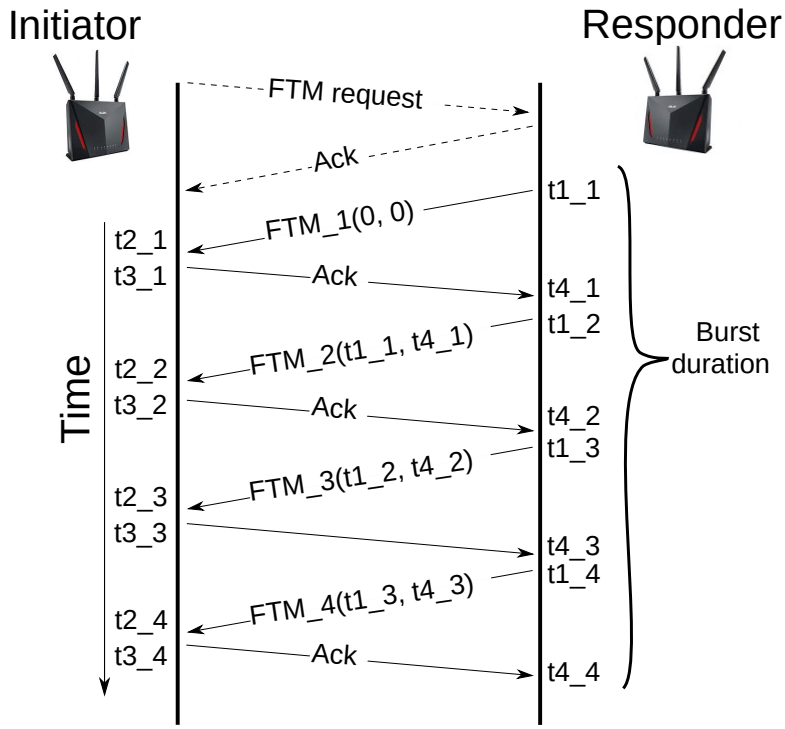


Figure 2.5: Example of an FTM session of 1 burst of for 4 measurements.

The performance of FTM has been analyzed thoroughly in [37] and the authors concluded that FTM is very accurate in outdoor environments whereas it gets a detrimental performance in indoor settings due to the multipath effects. This is caused by the arrival of several paths that are close in time. Hence FTM cannot separate them properly and the resulting RTT estimate is biased. We address this issue in Chapter 4 where we implement an FTM-like protocol much more robust to multipath than plain FTM.

From a practical perspective, distance-based localization systems are easier to deploy than angle-based ones since they do not require antennas arrays. They can be also implemented on the client as they do not require high computational costs. In contrast, angle-based localization systems need antenna arrays for AoA estimation and the computational cost of the AoA estimators are usually excessive for a client Central

Processing Unit (CPU). Hence, they are usually run in the AP or servers.

3 Performance Evaluation of Single Base Station ToA-AoA Localization in an LTE Testbed

3.1. Introduction

Location systems for cellular networks are rapidly gaining in importance, both due to the proliferation of location based services, as well as the promise of much more powerful mobile network management and control mechanisms that use location as context information. This is especially important in light of increasing device densities and wireless data rates. In fact, strict requirements for localization have been already specified in the future 5G standard, requiring an accuracy of 1m or less in 99% of the cases for indoor and outdoor [38,39]. However, the transition from Long Term Evolution (LTE) to 5G will be progressive. 5G rollout will begin in major urban areas as replacing most of the mobile infrastructure in one go is unaffordable for operators. This fact is reflected in the Cisco forecast and trend 2017-2022 [40] which predicts that the 5G devices and connection constitute over 3% of global mobile devices and connections by 2022. This leaves many areas and devices with only partial or even without 5G coverage. Ubiquitous location systems therefore have to incorporate legacy mobile technology, most importantly LTE.

While the current localization techniques enabled by the LTE standard cannot achieve the high level of 5G location accuracy, they can provide a performance that is sufficient for a range of important applications. Several techniques for localization are included in the 3GPP standard, such as fingerprinting or triangulation and trilateration approaches, based on Angle of Arrival (AoA), Time of Arrival (ToA) and reference signal received power. Such systems face many challenges: either the localization error reaches up to dozens of meters, they have high response times, or require coverage by several Evolved Node B (eNB), which, in particular indoors, is not always given. In addition, synchronization and/or clock errors are likely to degrade the accuracy due to the combination of several eNB measurements.

Hence, single eNB localization techniques using ToA and AoA measurements to estimate the direction and distance between the User Equipment (UE) and the eNB are

highly important for next generation networks. Furthermore, such techniques will greatly benefit from upcoming technologies such as Massive Multiple Input Multiple Output (MIMO) systems [41], and from higher bandwidth allocation. Thus, evaluating and understanding the achievable accuracy and behavior of single eNB localization in current LTE networks is necessary to assess which location-based services can be supported. Moreover, it serves as an useful starting point for next generation localization techniques.

To the best of our knowledge, there are no works in the literature evaluating single eNB localization performance of ToA plus AoA in a practical LTE testbed. Some works study other technologies and approaches for localization in LTE [42,43], where the authors assess various LTE localization systems, but unfortunately the errors are above 20m which severely limit their usefulness. Besides, most of them are only theoretical studies and thus, they do not include the effects of real-world hardware, the channel environment, as well as implementation complexity. While some practical studies exist, they do not target single eNB localization. For example, the authors in [44] have evaluated a coordinated localization system of several eNBs. Other works go beyond the LTE standard [45] and use sampling rates that are higher than those of current LTE systems.

In this chapter, we evaluate the achievable positioning accuracy for an LTE-compliant system. We consider a 2D space where the system exploits angular and distance information for the localization. The distance is extracted through ToA estimation using the LTE Sounding Reference Signal (SRS), and the AoA is measured using the MUltiple SIgnal Classification (MUSIC) algorithm. Our system is implemented on the eNB side to benefit from its improved RF hardware, antenna characteristics and computational power.

To do so, we carry out a measurement campaign in two different locations. The results of our performance evaluation indicate a median location error around 2m. While this performance does not meet the localization requirements of 5G and some Internet of Things (IoT) applications, such as robotics and augmented reality. But, it suffices for a great range of IoT applications which do not require a strict sub-meter accuracy, such as transportation and moving, sensing things, healthcare and many more [46, 47].

The rest of the chapter is organized as follows. Section 3.2 reviews LTE characteristics and functionality related for the positioning system. In Section 3.3, we describe the location system itself. Section 3.4 provides the details of our measurement campaign and Section 3.5 presents the evaluation results. Finally, Section 3.6 concludes the chapter.

3.2. LTE background

This section gives an overview of the LTE concepts [48] that are related to localization: synchronization, reference signals and how localization can be implemented within the LTE standard.

3.2.1. LTE Synchronization

Time and frequency synchronization between eNB and UE are critical to ensure reliable communication as well as precise positioning. The synchronization process includes two main steps:

- Timing symbol synchronization: It is the first timing synchronization in LTE. The purpose is to set the boundaries between symbols.
- Timing subframe and frame synchronization: In order to know exactly when to send the data, the UE needs to know the frame and subframe boundaries.

The first step is done exploiting the redundancy in each symbol to estimate the boundary, given that the Cyclic PRE-fix (CP) is a copy of the end of the symbol added at the beginning of it.

The second step uses the Primary and Secondary Synchronization Signals (PSS and SSS). The UE estimates the cross-correlation of the PSS to synchronize every half frame since the PSS is sent two times per frame. Afterwards, the same process is applied to the SSS to achieve frame synchronization.

3.2.2. LTE Reference Signals

The reference signals are known sequences which are used for multiple purposes like channel estimation or equalization. The signals of interest for localization are:

- Sounding Reference Signal (SRS): The SRS is transmitted in the uplink direction occupying up to 18MHz in its maximum configuration. It is used by the eNB to measure the uplink channel and the timing advance over a wider bandwidth. The signal is based on Zadoff-Chu sequences [49]. The main property of interest is the ideal cross-correlation: the cross-correlation of this sequence with a cyclic shifted version of itself is zero, except in the position of the lag.
- Positioning Reference Signal (PRS): The PRS is transmitted in the downlink direction. Its main purpose is to measure the ToA at the UE side and, consequently, it can be used for positioning. Similar to the SRS, this signal occupies a wider bandwidth up to the maximum available.

While, in contrast to the PRS, the main purpose of the SRS is not to localize. It is very useful for this purpose since it occupies almost the whole bandwidth and therefore provides reasonably high time resolution. Besides, the Zadoff-Chu properties make the SRS more reliable in case of multi-path effects, thanks to the ideal cross-correlation property.

3.2.3. Localization in LTE

While LTE design naturally focuses on communications first and foremost, it does support localization services through two main mechanisms [50]:

- Observed time difference of arrival: Using a similar concept as GPS, the UE measures the downlink signals from several synchronized eNBs, and the localization is done measuring the time difference of arrival among them. Starting with Release 9, the PRS was included to enhance the accuracy of this mechanism. Furthermore, the same concept for the uplink direction was added in Release 11, where several eNBs measure the same signal sent by the UE.

- Enhanced cell ID: Measurements like timing advance, angle of arrival and reference signal received power, together with the ID of the serving cell are used to improve the estimation of the position of the UE.

3.3. Localization Process Description

Especially for indoor positioning, single-eNB localization techniques are appealing. They only require local processing at the eNB or UE without any further LTE enhancements that may not be present in all LTE deployments.

For this reason, we consider a 2D location system based on LTE, where the UE is synchronized by the eNB over the air. The eNB has to measure two variables to determine the UE location, the first is the AoA, giving the direction at which the UE is located, and the second is the ToA, which provides an estimate of the distance between both.

3.3.1. AoA Estimation

There are several algorithms to estimate the AoA of an incoming signal, but none are included in the LTE standard. For our system, we have selected MUSIC algorithm [51]. It is a classic algorithm for AoA estimation which is gaining relevance thanks to MIMO [52]. MUSIC is based on decomposing through an eigendecomposition the correlation matrix of the signal into two subspaces: the signal and the noise. The signal subspace contains the information regarding the AoA of the incoming signal. To extract the AoA, instead of using the signal subspace, MUSIC exploits the eigenvector property of orthogonality to multiply a set of steering vectors, each one associated with a possible AoA, by the noise subspace. The AoA related to steering vector which is orthogonal to the noise subspace is the AoA of the incoming signal.

Consider a Uniform Linear Array (ULA) of antennas where the number of antennas is M and the distance between antennas is half of the wavelength. Due to multipath effects, there are P signals which arrive at the ULA. Each signal has an associated AoA, θ_{rx}

where $p \in \{0, 1, \cdots, P-1\}$. The ULA output is defined as:

$$\mathbf{x}(t) = \sum_{p=0}^{P-1} \mathbf{a_p}(\alpha_p) s_p(t) + \mathbf{w}(t), \tag{3.1}$$

where $s_p(t)$ is the signal source, $\mathbf{a_p}$ is the steering vector associated to s_p and $\mathbf{w}(t)$ is a vector of white Gaussian noise. The steering vector is given by:

$$\mathbf{a}(\alpha_\mathbf{p}) = [1, e^{-j\frac{2\pi d}{\lambda} sin(\alpha_p)}, \cdots, e^{-j\frac{2\pi d(K-1)}{\lambda} sin(\alpha_p)}] \tag{3.2}$$

We can write the correlation matrix of the input stream as follows:

$$R_x(t) = E\{x(t)x^H(t)\} = \mathbf{\Phi S \Phi^H} + \sigma^2 \mathbf{I}, \tag{3.3}$$

where $\mathbf{S}$ is the signal covariance matrix, the operator σ^2 is the noise power, $\mathbf{I}$ is the identity matrix and $\mathbf{\Phi}$ is the steering matrix. $\mathbf{S}$ is assumed to be definite positive (rank P), which gives the following representation:

$$\mathbf{R} = \mathbf{U \Lambda U^H} \tag{3.4}$$

where $\mathbf{U}$ is a unitary matrix which contains the eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix which contains M positive and real eigenvalues.

MUSIC splits the eigenvalue/vector pairs into two subspaces, the signal and noise. The signal eigenvectors are P eigenvectors where each one has an associated eigenvalue larger than the noise power such that $(\lambda_0, \cdots, \lambda_{P-1} > \sigma)$, whereas the remaining ones are equal to the noise power $(\lambda_P = \cdots = \lambda_{M-1} = \theta)$. This results in the following representation:

$$\mathbf{R} = \mathbf{U_s \Lambda_s U_s^H} + \mathbf{U_n \Lambda_n U_n^H} \tag{3.5}$$

where the columns of the signal subspace matrix, $\mathbf{U_s}$, are the P principal eigenvectors and $\mathbf{U_n}$ contains the remaining $M - P$ eigenvectors.

The orthogonality property of the eigenvectors ensures that $\mathbf{U_n}$ is orthogonal to $\mathbf{U_s}$ and, consequently, to the steering matrix. Therefore, MUSIC exploits this fact to extract the AoA. To do so, MUSIC defines the spatial spectrum as:

$$\mathbf{p_M}(\theta) = \frac{1}{\|\mathbf{a^H}(\theta)\mathbf{U_n}\|} \quad 0 \leq \theta \leq \pi \tag{3.6}$$

In a scenario where the main path is the Line-Of-Sight (LOS), the associated AoA is

the angle which maximizes the MUSIC spatial spectrum:

$$\hat{\theta} = \arg\max_{\theta}(\mathbf{p_M}(\theta)) \tag{3.7}$$

3.3.2. ToA estimation

Since the UE is synchronized with the eNB, the ToA can simply be measured by cross-correlation between the SRS sent by the UE and the received one by the eNB. The argument of the maximum value of the cross-correlation indicates the communication lag. In fact, this lag does not represent the propagation delay introduced by the channel, it represents twice that value because the same delay is added in the synchronization process of the UE. That is, the UE time reference, which is used to determine the start of the transmission of a frame, is the one of the eNB plus the propagation delay of the channel, since the synchronization is done over the air. LTE defines the concept of timing advance to compensate for it and avoid that the UE sends the information outside the slots given by the eNB. eNB measures the timing advance in the same way as the ToA, by the cross-correlation of the random access preamble or with the SRS, and sends it to the UE. The timing advance resolution, as is defined in LTE, corresponds to a distance of 78.12m. In case the UE is closer to the eNB than this resolution (which is the case in our experiments), the timing advance is 0 and does not have any effect.

The SRS received by the eNB in the frequency domain is:

$$\mathbf{Y}[k] = \mathbf{H}[k]\mathbf{X}[k] + \mathbf{W}[k], \tag{3.8}$$

where $\mathbf{H}[k]$, $\mathbf{X}[k]$ and $\mathbf{W}[k]$ represent the k^{th} sample of the the channel, the SRS sent by the UE and white Gaussian noise, respectively, in the frequency domain.

We denote the observed cross-correlation between the received and sent SRS as $r_{SRS}[n]$, given by:

$$\mathbf{r_{SRS}}[n] = IDFT\{\mathbf{Y}[k]\mathbf{X}^*[l]\}, \tag{3.9}$$

where n is the discrete time sample, the symbol $(\cdot)^*$ denotes the conjugate operator and $IDFT\{\cdot\}$ denotes the Inverse Discrete Fourier Transform.

The eNB can estimate the lag by finding the argument which maximizes the absolute value of the cross-correlation:

$$\hat{i} = \arg\max_{n} |\mathbf{r_{SRS}}[n]| \tag{3.10}$$

This argument corresponds to a given sample and because the SRS is a time discrete sequence, this sample corresponds to a given moment in the discrete time. Let us define the moment when the UE sends the SRS as $t_0 + \delta$, where δ is the propagation delay introduced by the channel. In addition, $t_{\hat{i}}$ is the moment that the cross-correlation peaks

at the eNB, which corresponds to $t_0 + 2\delta$. Consequently, we can express the ToA as:

$$\delta = \frac{t_{\hat{i}} - t_0}{2} \tag{3.11}$$

Finally, we can express the distance given by the ToA d_{ToA} as:

$$d_{ToA} = c\delta, \tag{3.12}$$

where c is the speed of light in the space. Note that the d_{ToA} values are also discrete and its resolution directly depends on the bandwidth of the SRS.

3.4. Implementation

Our setup consists of one eNB and one UE. The UE side runs on a laptop with 16GB of RAM and 4 cores at 2.8GHz and we use an inexpensive bladeRF Software-Defined Radio (SDR) for the radio communications. The eNB is a desktop PC with 16 GB RAM and 8 cores at 3.4 GHz. We also run the Evolved Packet Core (EPC) in a virtual machine on the same desktop PC, with 1 core and 6GB RAM. The eNB is connected to an Universal Software Radio Peripheral (USRP) X310, a type of SDR, which serves as radio interface. We measure AoA using an additional Location Measurement Unit (LMU) composed of a USRP X310 equipped with two TwinRX daughterboards to ensure phase synchronization among the four available channels. The four antennas are LTE compliant with a gain of 7dBi and arranged as an uniform linear array. This LMU is driven by another PC with the same characteristic as the eNB one. The complete setup is depicted in Figure 3.1.

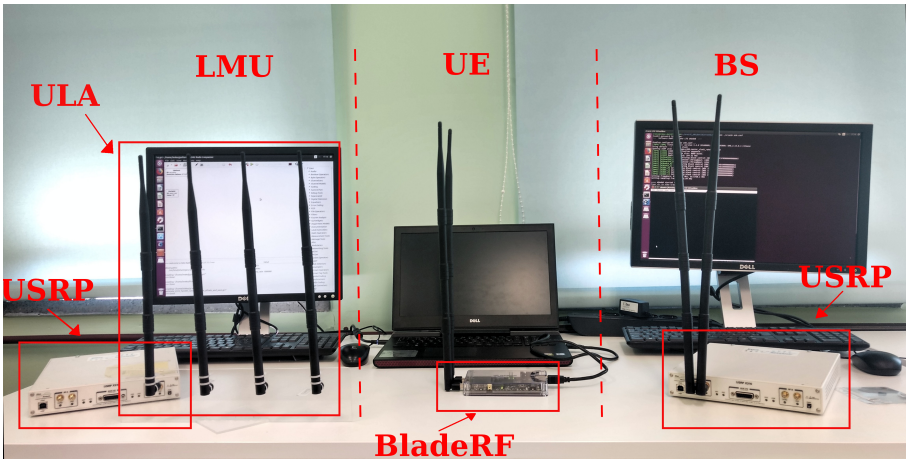


Figure 3.1: LTE location system components

The software for the LTE functionality (eNB, UE and EPC) is based on srsLTE [28]. srsLTE is a fully operational open source software implementation of the LTE cellular systems. We chose this setup since it performed better and was more stable than the other

most commonly used LTE software implementation, OpenAirInterface [29]. The LMU is based on GNU Radio [53], an open source software which provides signal processing blocks for SDRs. Out of the box, srsLTE does not fully support the SRS on the eNB side, whereas the UE can correctly send it. We thus modify the srsLTE eNB implementation to set the SRS configuration in the Radio Resource Control (RRC) messages as determined by the standard, specifically, the system information block 2 and RRC connection setup.

Regarding the LTE configuration, the system uses the maximum available bandwidth of 20 MHz in frequency division duplexing mode with a normal CP. The center frequency is 1.8GHz. We configure the SRS to be sent every 50 ms with a bandwidth of 18MHz, the maximum that LTE supports. This gives a ToA resolution of 8.625m.

We run our measurements in two indoor locations at IMDEA Networks. The first one is an empty room without furniture with a size of 21x9m where perfect coverage and visibility are available over the whole scenario. In contrast, the second scenario is a 19x15m office space, where the furniture and dividing walls create Non-Line-Of-Sight (NLOS) areas and increase the number of multipath components. Figure 3.4 shows the floor plan of the office (as well as the measurement results). It has a central open area with desks, chairs and screens, and individual offices on the right and left hand side of it. The obstacles that create NLOS areas are the pillars and the dividing wall indicated in thick black, as well as the glass walls which separate the offices from the open area. For the rest of the chapter, we will refer to the first scenario as the auditorium and the second one as the office.

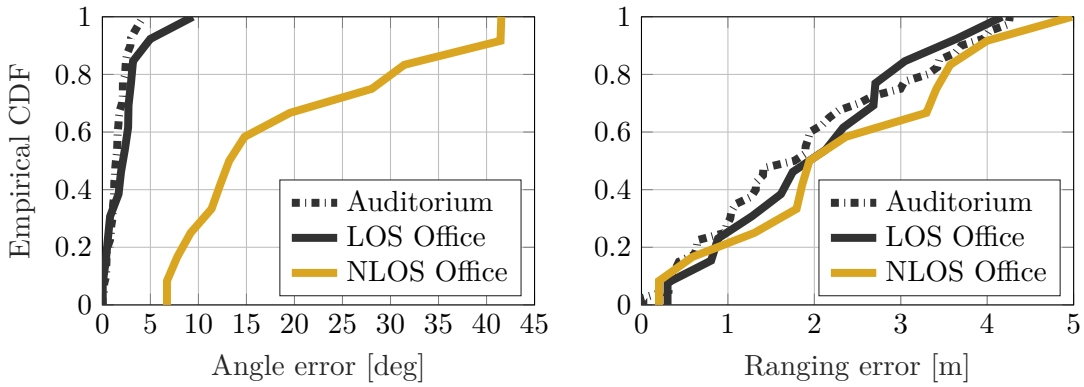
There are 40 and 25 measurement points in the auditorium and office scenarios, respectively. For 5 of the points in the office scenario that are located inside closed offices, we took measurements twice, once with the office door open and once with it closed. We carry out experiments in each point measuring the AoA and ToA to determine the overall localization accuracy.

3.5. Numerical Results

This section illustrates the results of the LTE location system. The results of the AoA and ToA estimation are first discussed separately, and then the performance of joint estimation for localization is presented. We note that the localization accuracy is degraded in case of NLOS. For this reason, we separately show the cases of LOS and NLOS in the office scenario. Finally, we provide a visual representation of the measurements in the office scenario to assess the performance in detail.

3.5.1. AoA

Figure 3.2(a) shows the Cumulative Distribution Function (CDF) of the AoA error. For the case of LOS, the errors in the auditorium and the offices are very similar, in



(a) AoA performance

(b) ToF ranging performance

Figure 3.2: Empirical CDF for AoA and ToA error

contrast to the case of NLOS. The fiftieth percentile of the cases have an AoA error below 1.4° and 2.2° in the auditorium and LOS office, whereas in the NLOS office case, they are below 13.4°. The effects of NLOS are extremely noticeable in the office and degrade considerably the AoA performance. Also, the maximum error in LOS office is considerably larger compared to the auditorium due to the richer multipath environment. Moreover, the highest errors in the auditorium and LOS office case do not come from the furthest points. Thus, this indicates that they are caused by multipath instead of the distance.

3.5.2. ToA

As explained in Section 3.3.2, the measured ToA depends on the symbols of the SRS and is measured with the granularity of a symbol length. In addition, for each measurement point, we see a certain variability in the ToA discrete values. Such variations can be caused by systematic delays because of several factors, such as hardware processes of the internal FPGA of the SDRs, software process executed on the PC and minor synchronization errors of the system. While this effect degrades the performance relying on individual ToA measurements, the system can achieve higher accuracy when averaging over the set of measurements due to these fluctuations. In other words, as these fluctuations produce that the ToA varies between two or more discrete values, the system estimates one depending on the underlying ToA. Hence, averaging over the set of measurements results in an more accurate ToA. At the same time, this mitigates the synchronization errors.

The ToA results are illustrated in Figure 3.2(b). They behave very similar for all scenarios. First, it indicates that the lack of LOS does not significantly degrade the accuracy. Second, the office environment has more multipath components compared to auditorium, but the performance in both is very similar, indicating that the multipath

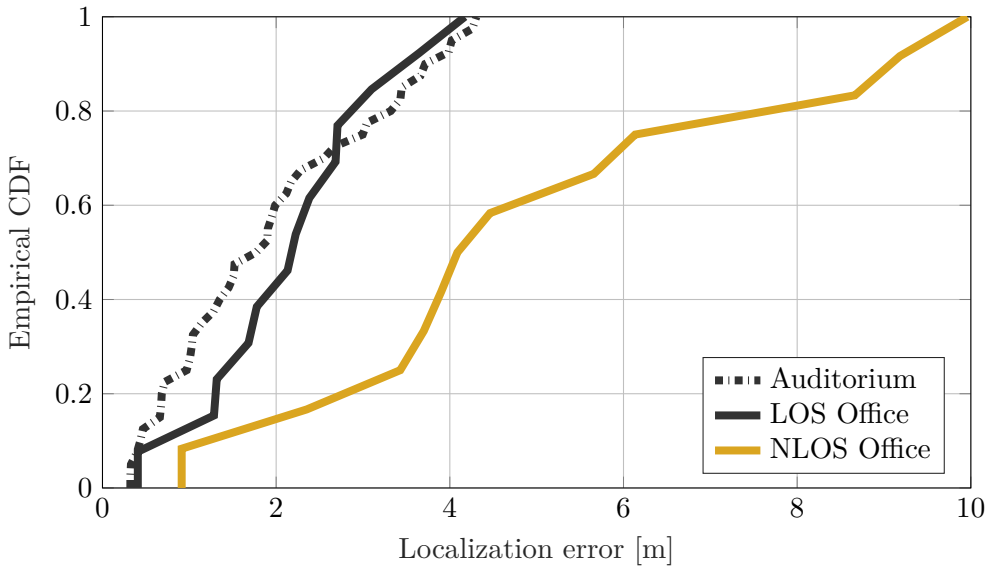


Figure 3.3: CDF of the localization error

does not significantly deteriorate performance. The median distance error is below or equal to 1.76, 2.05 and 1.92m for auditorium, LOS and NLOS office. Although, the ToA resolution of an individual ToA measurement is 8.625m, the variability in the measurements plays a key role to improve the performance when averaging measurements over time.

3.5.3. Localization

Figure 3.3 shows the CDF for the localization error. The localization performance mainly depends on the AoA accuracy in the NLOS cases and not on the ToA. The results for auditorium and LOS office show a similar behavior with a median error of 1.76 and 2.12m, respectively, whereas in the NLOS office it is 4.67m. In extreme cases, the location can have an error of up to 9.97m in NLOS due to very low AoA accuracy. However, the remaining scenarios have a maximum error of 4.3 and 4.48m.

3.5.4. Further observations

Figure 3.4 shows the floor plan of the office scenario where we illustrate the MUSIC spatial spectrum for each measurement point. The text boxes below these profiles represent the error values in meters of the ToA (top) and the overall localization with AoA and ToA (bottom). The black and yellow color of the MUSIC spatial spectrum represents whether a point is LOS or NLOS. In addition, for the cases within the offices, the dashed line represents measurements taken with the door open and the solid one with the door closed.

In the third row and for the cases of NLOS in the open area, the spatial spectrum indicates that the AoAs come from a reflection instead of the direct path. Note also that for the case behind the wall, where there are no strong reflections, the AoA is slightly shifted from 90° to 80°. In the two furthest points from the eNB within the offices, the AoAs come from the door, whereas with the door closed, the AoAs come from the direct path. For the remaining points, the performance of the AoA behaves similar for both cases.

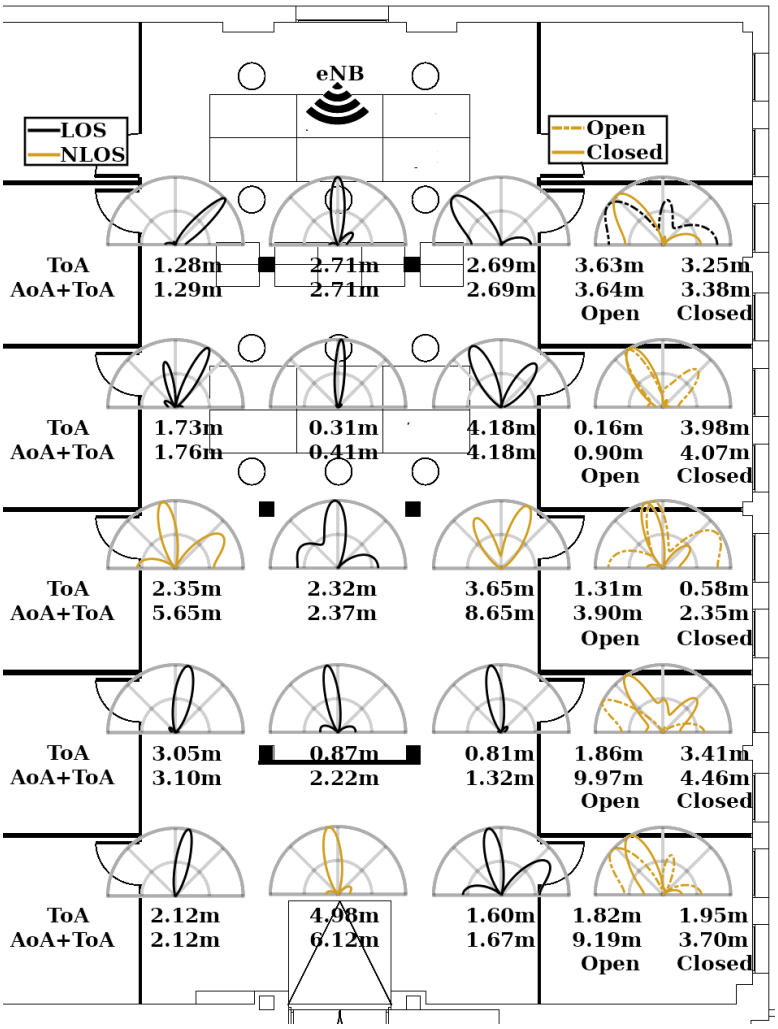


Figure 3.4: Measurement errors of ToA and overall localization (AoA+ToA) in the office. Each point contains the MUSIC spatial spectrum as well as the error values for ToA and localization. Besides, on the right side two MUSIC spectrum and two error values appear to consider whether the door is closed or open

3.6. Conclusion

While 5G network standards come with very strict localization requirements and improved localization mechanisms, LTE networks will continue to be used along with 5G for quite some time to come. This implies that ubiquitous location-based services will also have to make use of LTE localization schemes whenever 5G is not available. Evaluating the potential of the current LTE standard for localization is thus important to understand overall performance of localization in future mobile systems. We have tested the performance of a single eNB localization system in an LTE testbed based on software-defined radios. The measurements show that in LOS conditions the system has a median localization error around 2m. However, localization accuracy is degraded in cases of NLOS with a median error of 4.67m. Finally, we observe that an LTE location system can achieve an accuracy that covers a wide range of location-based service requirements, even including some of the future IoT scenarios.

4 Accurate Ubiquitous Localization with Off-the-Shelf IEEE802.11ac Devices

4.1. Introduction

Wireless localization has become an important application of wireless communications, and the positioning accuracy has improved substantially over the past two decades of research. However, Non-Line-Of-Sight (NLOS) settings drastically reduce the localization performance as seen in Chapter 3. In particular, the insufficient accuracy of the previous evaluation is mainly due to the limited hardware characteristics of Long Term Evolution (LTE) and the use of rudimentary signal processing algorithms. In this chapter, we present UbiLocate, a ubiquitous Wi-Fi location system that works both under Line-Of-Sight (LOS) and NLOS conditions. We achieve good NLOS localization through the improvements brought about by IEEE 802.11ac in terms of bandwidth and number of antennas, in combination with novel signal processing for multipath decomposition, that jointly help to resolve multipath effects much more accurately. In particular, this chapter makes the following main contributions:

• **Optimized AoA extraction.** Classic algorithms such as MUSIC [32] and ESPRIT [54] have been widely used to analyze RF signals for path parameter estimation, especially Angle of Arrival (AoA). [34, 55, 56]. Recently, compressed sensing techniques have been demonstrated to provide better accuracy [33, 57, 58]. However, their application can be computationally prohibitive in common scenarios. In order to reduce the computational complexity, UbiLocate iteratively determines a first estimate of the path parameters and then refines it through a Nelder-Mead search [59]. This minimization results in a more accurate multipath decomposition, and UbiLocate achieves an AoA accuracy improvement of a factor of 2 for LOS and 1.5 for NLOS settings compared to state-of-the-art algorithms [14].

• **Controlled Ranging.** Estimating the absolute Time of Flight (ToF) and thus the distance between client and Access Point (AP) requires timestamped packet exchanges, as standardized in the 802.11 Fine Time Measurement (FTM) protocol [37,60]. However,

FTM is inaccurate in multipath-rich environments [61]. UbiLocate uses a custom protocol similar to FTM that has lower overhead and is more robust by decomposing the multipath channel to accurately determine the ToF of the first path. Again, UbiLocate improves the ToF estimation accuracy by a factor of 2 for LOS and and 1.5 for NLOS compared to plain FTM.

• **Filtering reliable APs.** Depending on the specific scenario, the estimates from different APs have different fidelity. When averaging the location information provided by all APs, low quality estimates may contaminate the overall location accuracy. UbiLocate therefore includes a mechanism to evaluate the quality of different estimates, giving more weight to the APs that provide good estimates.

• **Implementation on off-the-shelf devices.** We implement the UbiLocate system on off-the-shelf Asus AC2900 RT86U routers that support IEEE 802.11ac with 4x4 Multiple-Input Multiple-Output (MIMO) and 80 MHz of bandwidth. The improved hardware capabilities increase localization accuracy since the larger bandwidth and number of antennas allow for better time and space resolution. We can thus extract the path parameters more accurately than with the older IEEE 802.11n standard. We modify the router firmware to access Channel State Information (CSI) in order to estimate AoA, Angle of Departure (AoD), and ToF. (i) UbiLocate is the first location system implemented on off-the-shelf devices that works with 80 MHz Wi-Fi channels and does not require a non-disclosure agreement (the existing 80 MHz location systems [24, 25] use Quantenna devices that require such a non-disclosure agreement). (ii) It is also the first IEEE 802.11ac-based location system that can simultaneously derive both angle and absolute distance to a target device, whereas prior work uses two separate co-located devices for this purpose [25, 61].

We deploy UbiLocate in a large office environment and test it with different AP densities and with both LOS to NLOS measurement points. Our performance evaluation shows that UbiLocate achieves meter-level median accuracy even for pure NLOS and low AP density scenarios. It outperforms current state-of-the-art systems by a factor of 2-3. Finally, we release our tool to extract CSI and perform FTM-like ranging to the research community to foster wireless systems research with 802.11ac. We believe that it will prove similarly useful as the widely used CSI tool for 802.11n [62], given the hardware improvements offered by 802.11ac. The CSI extractor tool with the modified firmware and documentation are available in a github repository [63].

4.2. UbiLocate overview

UbiLocate locates a wireless device using AoA, AoD, and ToF information. This is relatively straightforward when several APs with direct LOS are within range. However, typical indoor Wi-Fi deployments do not provide ubiquitous LOS coverage since NLOS

links can provide sufficiently high data rates.

In such complex environments with NLOS, the multipath channel and the resulting superposition of different signals at the receiver significantly affects the quality of the location estimate. Even under pure NLOS, good location accuracy is feasible as long as the location system can discriminate between obstructed LOS paths and the NLOS paths coming from reflections. The latter must be discarded, since they lead to erroneous angle and ToF estimates. By definition, obstructed LOS paths pass through an obstacle, and thus their signal power may be severely attenuated compared to other NLOS paths. Accurately detecting them requires a fine-grained multipath decomposition of the channel.

As is common for wireless location systems, we assume that the positions of the APs are fixed and known. To discriminate the multipath components, UbiLocate minimizes the norm of the difference between the observed received signal and estimated superimposed signals and their path parameters. The number of possible combinations of path parameters makes brute force minimization computationally prohibitive, but if an approximate estimate is known, the minimization can be sped up significantly. To this end, we first compute rough estimates of the path parameters and then refine them through a Nelder-Mead search [64].

This provides better accuracy than the widely used MUSIC and similar approaches [14] which resolve the paths in one round. UbiLocate iteratively estimates the parameters of the strongest path and then subtracts them from the received signal. This allows UbiLocate to estimate the parameters of weak paths that would otherwise be masked by stronger ones and is especially critical in NLOS environments. In contrast to prior iterative approaches [65–67], we further refine the estimation to remove imperfections which leads to improved angle accuracy in the challenging cases we target in this chapter.

Figure 4.1 shows a typical NLOS scenario in which reflected paths may be stronger than the obstructed LOS one. In addition, as shown on the right in this example figure, path $p1$ and path $p2$ are close in time and angle, and the uncertainty around path $p2$ makes it hard to discriminate the two. In such a scenario, accurate multipath decomposition is key for good location system performance.

4.2.1. Angle estimation

For device localization, UbiLocate requires extracting the angles for the direct or obstructed LOS path, provided such a path exists. This path is contained in the observed received channel in the frequency domain, which we already defined in Section 2.2.1. This channel is expressed as a function of the attenuation, the path delay, the AoA and the AoD as follows:

$$\hat{\mathbf{H}}[k] = \sum_{p=0}^{P-1} \boldsymbol{\phi}(\theta_{rx,p})\gamma_p \boldsymbol{\phi}^{\mathrm{H}}(\theta_{tx,p})\psi(\tau_p)[k] + \mathbf{w}[k] = \sum_{p=0}^{P-1} \mathbf{H}_p[k] + \mathbf{w}[k] \ , \tag{4.1}$$

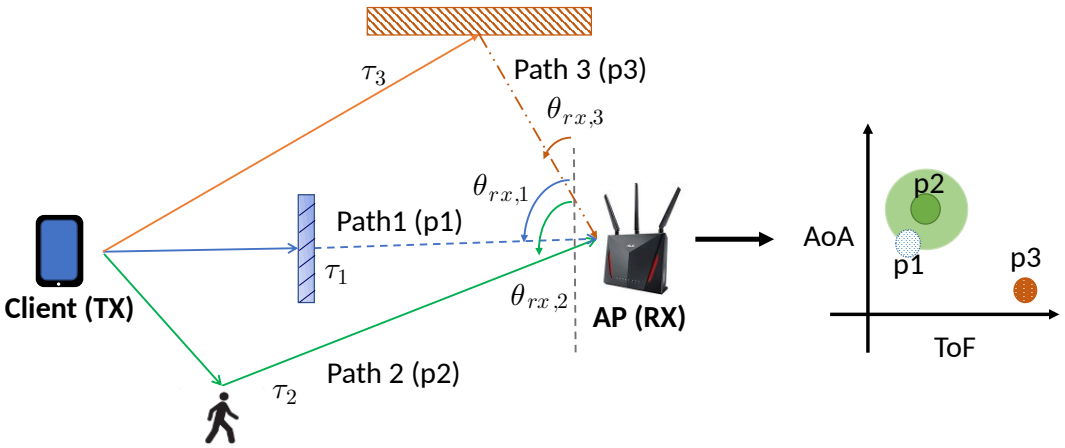


Figure 4.1: NLOS example with obstructed los path.

with k is the k-th subcarrier, P is the number of paths, $\phi(\theta_{rx,p})$ is the vector of phase shifts at the receiver Uniform Linear Array (ULA) introduced by the AoA, $\psi(\tau_p)[k]$ is the path delay introduced by the propagation of the signal, γ_p is the complex attenuation, $\phi(\theta_{tx,p})$ is the vector of phase shifts at the transmitter ULA introduced by the AoD and $\mathbf{w}[k]$ is an L-dimensional white Gaussian noise in the frequency domain, where L means the number of transmitter antennas

The direct path is the one that typically arrives earliest in time before any of the NLOS paths coming from reflections, i.e., the one with the smallest τ_p. Note that the ToF τ_p is not an absolute value but reflects relative delay differences among paths. (For ranging, UbiLocate uses a customized FTM implementation.) While directly using AoD information is not useful due to potential rotation of the device to be located, estimating it jointly with the other path parameters considerably improves the path resolvability [68].

To extract parameters of all paths, our objective is to find an expression for $\mathbf{H}[k]$ that minimizes $\|\hat{\mathbf{H}}[k]-\mathbf{H}[k]\|$. $\hat{\mathbf{H}}[k]$ is the observed channel and $\mathbf{H}[k]$ contains the contribution of each path according to the estimated path parameters. However, minimization by brute force is computationally prohibitive due to the large number of combinations of path parameter. Hence, we split the minimization into two steps. We first perform a greedy matching projection to iteratively estimate the path parameters. We then perform a minimization through Nelder-Mead search based on the extracted path parameters from the first step to refine them.

4.2.1.1. Greedy estimation

Through greedy matching projection we iteratively compute the contribution of the strongest path, estimate its parameters, reconstruct it, and then subtract it from the overall measured channel. The output of the subtraction is the channel residual and using

the residual we can then estimate the second strongest path's contribution, and so on, until the parameters of all significant paths are estimated. This allows estimating accurately even the weak paths often found in NLOS scenarios, since we first remove the contribution of the stronger ones. As is illustrated in Figure 4.1, depending on the properties of the reflectors, paths $p2$ and $p3$ may be significantly stronger than the obstructed LOS path $p1$.

We apply a matching projection to the observed channel and the path parameters that maximize it are the ones from the strongest path $p = 0$. We then remove this path from the observed channel and apply matching projection to the residual to obtain the second strongest path $p = 1$, and so on. In general, in iteration p we extract path p as the strongest path of the residual as:

$$(\tau_p, \theta_{rx,p}, \theta_{tx,p}) = \\ \underset{\tau_p, \theta_{rx,p}, \theta_{tx,p}}{\arg\max} \sum_k \boldsymbol{\phi}^H(\theta_{rx,p}) \hat{\mathbf{H}}_p^{\mathrm{r}}[k] \boldsymbol{\phi}(\theta_{tx,p}) \psi^*(\tau_p)[k]\,, \tag{4.2}$$

The path parameters produce phase shifts, where $\boldsymbol{\phi}(\theta_{rx,p})$ and $\boldsymbol{\phi}^H(\theta_{tx,p})$ are the phase shifts introduced by the AoA and AoD at receiver and transmitter antennas, and $\psi(\tau_p)[k]$ that of the path length for subcarrier k. We multiply these phase shifts by their conjugates in the projection, so that only the correct path parameters maximize it. The residual in iteration p is given by

$$\hat{\mathbf{H}}_p^{\mathrm{r}}[k] = \hat{\mathbf{H}}[k] - \sum_{p'=0}^{p-1} \boldsymbol{\phi}(\theta_{rx,p'}) \gamma_{p'} \boldsymbol{\phi}^{\mathrm{H}}(\theta_{tx,p'}) \psi(\tau_{p'})[k]\,, \tag{4.3}$$

where the residual for $p = 0$ is the original channel $\hat{\mathbf{H}}_0^{\mathrm{r}}[k] = \hat{\mathbf{H}}[k]$.

To solve the optimization problem in Equation (4.2), we first determine τ_p. To do so, we convert the channel from the frequency domain to the time domain $\mathbf{H}[t]$, by applying an over-sampled inverse discrete Fourier transform to the channel. In the time domain, the path delay τ_p of the strongest path is directly the time t value that maximizes $\|\mathbf{H}[t]\|$. This channel is given by a combination of *sinc* functions with maxima in the different delays. Now, given τ_p we have

$$\begin{aligned} \mathbf{H}[\tau_p] &= \textstyle\sum_{p'=0}^{p-1} \boldsymbol{\phi}(\theta_{rx,p'}) \gamma_{p'} \boldsymbol{\phi}^H(\theta_{tx,p'}) (\psi^{\mathrm{H}}(\tau_p) \psi(\tau_{p'})) \\ &\simeq \boldsymbol{\phi}(\theta_{rx,p}) \gamma_p \boldsymbol{\phi}^H(\theta_{tx,p}) \end{aligned}\,, \tag{4.4}$$

and $\hat{\mathbf{H}}[\tau_p] = \mathbf{H}[\tau_p] + \bar{\mathbf{w}}[\tau_p]$ with noise at the instant τ_p computed as $\bar{\mathbf{w}}[\tau_p] = \sum_{k=0}^{K} \hat{\mathbf{w}}[k] \psi(\tau_p)^*[k]$. With this, we can estimate the angle information. Instead of jointly estimating the $\theta_{rx,p}$ and $\theta_{tx,p}$, we first estimate $\theta_{tx,p}$ by a grid search assuming that $\theta_{rx,p}$

is unknown. This results in the following formulation:

$$\max_{\theta_{rx,p},\theta_{tx,p}} [1,0,0,\ldots]\hat{\mathbf{H}}[\tau_l]\boldsymbol{\phi}(\theta_{tx,p}). \tag{4.5}$$

Having estimated $\theta_{tx,p}$, we can iteratively refine either angle by a grid-search assuming that the other is known which increases the estimation accuracy. This individual estimation of two parameters is much faster than a joint estimation of two parameters. We refine the angle estimation by maximizing the following expression:

$$\max_{\theta_{rx,p},\theta_{tx,p}} \boldsymbol{\phi}(\theta_{rx,p})^H\hat{\mathbf{H}}[\tau_l]\boldsymbol{\phi}(\theta_{tx,p}) \ . \tag{4.6}$$

Once all parameters for one path are estimated, we recompute γ_p as a linear MMSE solution to minimize the error between the measured channel and the reconstructed one.

4.2.1.2. Refinement search

The previous estimation of the path parameters may contain imperfections since the paths are highly correlated. This may leak information of the parameters from weaker paths to stronger ones and vice versa. To refine the estimates, we carry out a Nelder-Mead search to minimize $\|\hat{\mathbf{H}}[k] - \mathbf{H}[k]\|$. This optimization method iteratively generates sets of points that compose a simplex polytope. The *gradient expression* for the refinement problem is very complex, whereas below we show how to obtain an *objective function* that is simple to evaluate. This makes Nelder-Mead search a much better fit for the specific problem of multi-path refinement than gradient descent. While Nelder-Mead search itself is well studied, to the best of our knowledge it has never been applied to the problem of path parameter estimation.

We use a vectorized version of the problem $\hat{\mathbf{h}}_\mathrm{v} = \Phi\gamma$, with $\hat{\mathbf{h}}_\mathrm{v} = [\mathrm{v}(\hat{\mathbf{H}}[0])^\mathrm{T},\ldots,\mathrm{v}(\hat{\mathbf{H}}[K-1])^\mathrm{T}]^\mathrm{T}$, $[\Phi]_{:,p} = \psi(\tau_p) \otimes (\phi^*(\theta_{tx,p}) \otimes \phi(\theta_{rx,p}))$ and $[\gamma]_p = \gamma_p$. This way, we have $\hat{\mathbf{h}}_\mathrm{v}$ as the vector containing all measurement information, Φ as all path contributions and γ as their complex gains. Note that only Φ has a dependency on the path parameters and each column depends only on one path, while γ behaves as a weight vector for the different path contributions. Converting the formulation from $\min\|\hat{\mathbf{H}}[k] - \mathbf{H}[k]\|$ to the vectorized version of the problem $\min\|\hat{\mathbf{h}}_\mathrm{v} - \Phi\gamma\|^2$ makes it easy to evaluate the minimization. Now let $\Phi^\perp$ be the orthonormalization by Gram-Schmidt of Φ and $\mathbf{A}$ the invertible square matrix such that $\Phi = \Phi^\perp\mathbf{A}$ to simplify the incoming equations. Then

$$\begin{aligned}
&\min\|\hat{\mathbf{h}}_\mathrm{v} - \Phi\gamma\|^2 = \min\|\hat{\mathbf{h}}_\mathrm{v}\|^2 + \|\mathbf{A}\gamma\|^2 - \mathcal{R}(\hat{\mathbf{h}}_\mathrm{v}^\mathrm{H}\Phi^\perp\mathbf{A}\gamma)\\
&= \min\|\hat{\mathbf{h}}_\mathrm{v}\|^2 - \|(\Phi^\perp)^\mathrm{H}\hat{\mathbf{h}}_\mathrm{v}\|^2 + \|(\Phi^\perp)^\mathrm{H}\hat{\mathbf{h}}_\mathrm{v} - \mathbf{A}\gamma\|^2\\
&= \min\|\hat{\mathbf{h}}_\mathrm{v}\|^2 - \|(\Phi^\perp)^\mathrm{H}\hat{\mathbf{h}}_\mathrm{v}\|^2
\end{aligned} \tag{4.7}$$

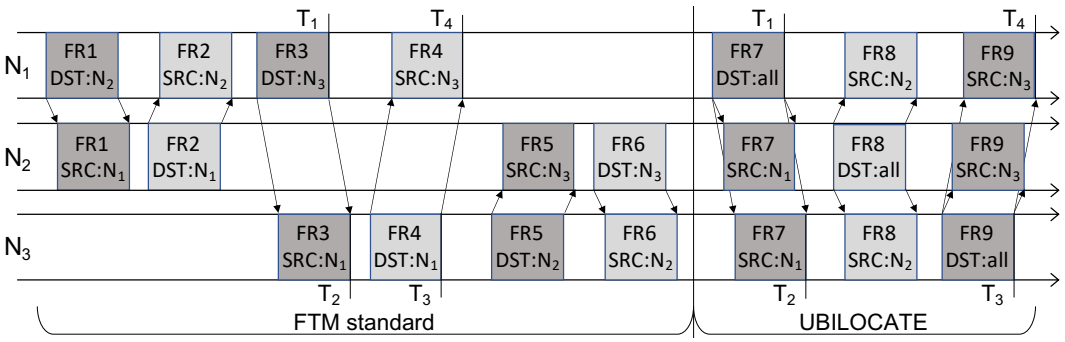


Figure 4.2: Standard FTM (left) sends dedicated messages per pair of nodes and UbiLocate (right) broadcasts a single frame per node for ranging with all other nodes.

This formula is very fast to evaluate, making it amenable to a Nelder-Mead search over the path parameters $\theta_{rx,p}, \theta_{tx,p}, \tau_p$ in expression (4.7). Then, γ is recomputed as the linear MMSE solution using the refined parameters.

The direct path p_{dp} corresponds to the index p with the smallest τ_p. To avoid spurious results, we add a power regularization term

$$p_{\text{dp}} = \min_p \tau_p - 0.0001 \frac{\gamma_p}{\max_{p'} \gamma_{p'}} \ . \tag{4.8}$$

Finally, the estimated AoA at the AP is given by

$$\hat{\theta} = \theta_{rx,p_{dp}} \ . \tag{4.9}$$

4.2.2. Ranging

Accurate ToF information is crucial for ranging and thus for localization. Unfortunately, locating a target node with multiple APs leads to several problems that must be addressed to achieve good performance. Each AP is running its own clock source, and since the different clocks are not synchronized, it is not possible to correct ranging estimates simply by post-processing the collected CSI data. Obtaining accurate ToF estimates between each AP and the client requires multiple packet exchanges with timestamps, as in the FTM protocol. While this protocol was standardized several years ago [37], the majority of current Wi-Fi devices do not support it (including the ones we instrument for this work). At the same time, FTM measurements of devices that do support it show suboptimal performance in multipath-rich environments. We thus introduce in our framework the first implementation of an FTM-like protocol that obtains accurate ranging information on off-the-shelf 802.11ac devices that support CSI extraction.

In Figure 4.2, we highlight the differences between the standard FTM and our implementation by showing how ranging is performed with three nodes $N_n, n \in \{1, 2, 3\}$

with time on the x-axis. Standard FTM uses unicast frame-ack exchanges, whereas UbiLocate broadcasts frames asynchronously to all other nodes. This significantly reduces the number of frames for ranging with multiple nodes.

For the FTM frames 1-6, we indicate the destination (at transmitter) and the source (at receiver) and the corresponding times. For instance, frame 3 ($FR3$) is transmitted at time T_1 by node N_1 (the initiator) to node N_3 (the responder) that receives it at time T_2. Afterwards, N_3 responds by transmitting frame $FR4$ to node N_1 at time T_3, which is received at node N_1 at time T_4. In addition to specifying the frames carrying the timestamps, FTM defines a mechanism to collect timestamps measured by the responder at the initiator. Then, FTM uses T_1, T_2, T_3 and T_4 to evaluate the Round-Trip Time (RTT) and thus the distance $\hat{d}$:

$$\begin{aligned} \text{RTT} &= (T_4 - T_1) - (T_3 - T_2) \\ \hat{d} &= (RTT/2) \cdot c \end{aligned} , \tag{4.10}$$

where c is the speed of light. By using both transmit and receive times it is possible to remove the reaction time uncertainty, i.e., the delay between frames $FR3$ and $FR4$. The procedure can be repeated multiple times to average results and obtain a more accurate estimate [37]. For FTM, $N(N-1) = 6$ frames are required to compute the $N = 3$ distances, resulting in a quadratic overhead.

Instead, UbiLocate requires only $N = 3$ broadcast frames as shown in the right part of the figure. A frame includes the $N - 1$ timestamps when the last frame from each of the other nodes was received, as well as the transmit timestamp for the frame itself. These frames are transmitted asynchronously by each node, and are opportunistically reused by other nodes, resulting in a linear overhead. This also removes the need for a dedicated collection mechanism. The three frames $FR7$-$FR9$ are used to compute the three distances. We first use $FR7$ in place of $FR3$, and we call T_1 the time when $FR7$ is transmitted by node N_1, and T_2 the time when it is received at node N_3. We then use $FR9$ in place of $FR4$, sent and received at T_3 and T_4, respectively. As $FR9$ embeds T_2 and T_3 (among other timestamps), upon receiving it, node N_1 can use the same equation above to determine the distance. We can reuse $FR7$ together with $FR8$ to evaluate the distance between nodes N_1 and N_2. Similarly, we can reuse $FR9$ with $FR8$ to estimate the distance between N_2 and N_3.

4.2.3. Localization

With the information discussed previously, AP a can estimate the location $\hat{\mathbf{y}}_a$ of the target device in Cartesian coordinates using

$$\hat{\mathbf{y}}_a = \mathbf{x}_a + \hat{d}_a \begin{bmatrix} \cos\hat{\theta}_a \\ \sin\hat{\theta}_a \end{bmatrix} , \tag{4.11}$$

where $\mathbf{x}_a$ is the (known) position of AP a, $\hat{d}_a$ is the estimated distance of the target device from the AP, and $\hat{\theta}_a$ is the AoA estimated at the AP.

Since Equation (4.11) holds for any AP, we have a system of A such equations, where A is the number of APs. However, not all APs provide equally useful location information and a simple strategy that averages all estimated positions $\hat{\mathbf{y}}_a$ with equal weights is suboptimal. To identify and filter out unreliable estimates, UbiLocate uses a metric that measures the dominance of multipath components with respect to the direct path in the received signal. The specific metric used by our system is the *mean excess delay* [69], given by the weighted average of the delays of every single multipath component with respect to the direct path, with relative path power as the weight. More precisely, assuming that we can discriminate $P > 1$ different paths, the mean excess delay $\tau_{m,a}$ for AP a is:

$$\tau_{m,a} = \frac{\sum_{p=0}^{P-1} \|\gamma_p\|^2 (\tau_p - \tau_0)}{\sum_{p=0}^{P-1} \|\gamma_p\|^2} \,, \tag{4.12}$$

where γ_p and τ_p are the complex attenuation and ToF of path p, respectively, and τ_0 is the ToF of the first received path.

If the contribution of the multipath components is small compared to the direct path, $\tau_{m,a}$ will tend to 0, whereas larger values of $\tau_{m,a}$ indicate stronger multipath. Hence, a large mean excess delay is an indication that the position estimate $\hat{\mathbf{y}}_a$ of AP a might be less reliable. UbiLocate uses a threshold τ_{th} and discards the estimates whose mean excess delay exceeds τ_{th}. Since this metric largely depends on the geometry of the scenario, obstacles and many other factors, fixing an absolute threshold for this metric to remove unreliable APs could lead to also removing useful APs. To address this, for each measurement point UbiLocate applies a dynamic threshold relative to the AP with the lowest mean excess delay, τ_{lw}. Specifically, τ_{th} is equal to two times τ_{lw}.

We denote by A' the set of APs for which the mean excess delay $\tau_{m,a}$ is below τ_{th}. Then, given $|A'|$ estimates along with their corresponding mean excess delay $\tau_{m,a}$, UbiLocate computes the final position of the target node with a weighted centroid approach:

$$\hat{\mathbf{y}} = \frac{\sum_{a=0}^{|A'|-1} \hat{\mathbf{y}}_a \cdot (\tau_{m,a})^{-1}}{\sum_{a=0}^{|A'|-1} (\tau_{m,a})^{-1}} \,. \tag{4.13}$$

This way, estimates with a small mean excess delay receive a higher weight. UbiLocate thus discards very unreliable estimates and gives higher importance to estimates from the most reliable APs. Furthermore, UbiLocate addresses the following issues:

Extreme angles. Extreme angles are defined as angles below -75° and above +75°. For these cases, UbiLocate's AoA estimator may takes the opposite solution (i.e., UbiLocate estimates -75° when the correct AoA is +75°), due to the fact that the relative phase differences become close when the angles approach ±90° and the system is

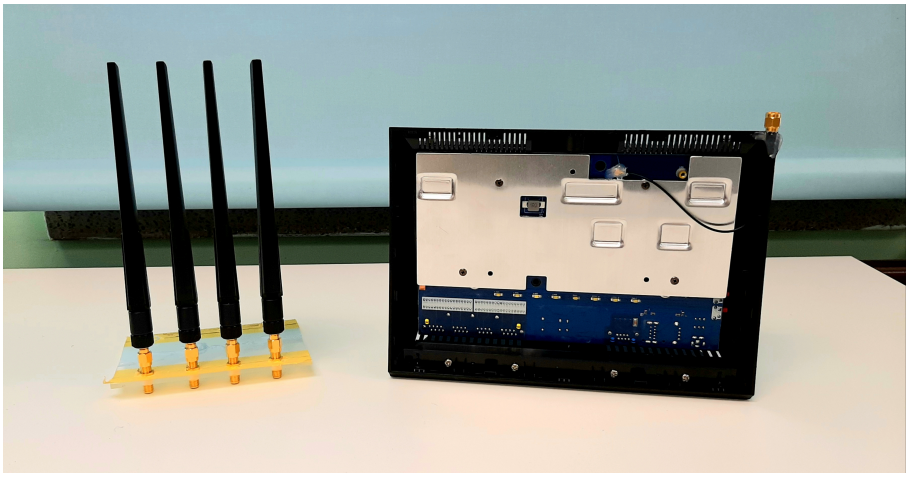
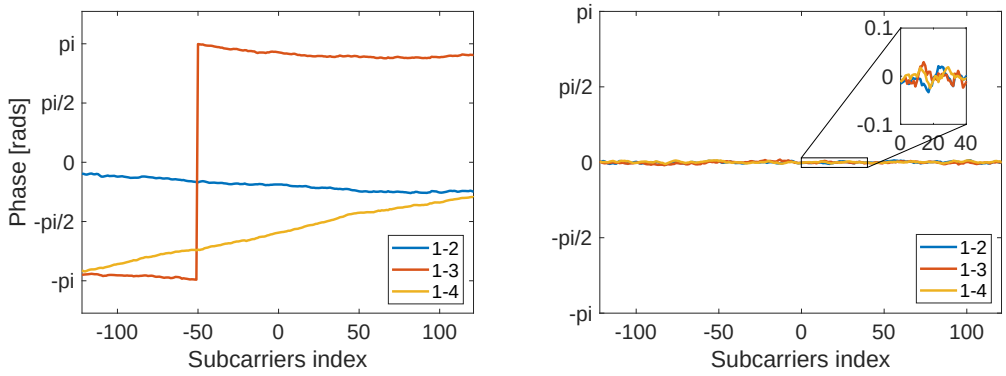


Figure 4.3: ASUS 802.11ac router with custom antenna array.



(a) Before calibration

(b) After calibration

Figure 4.4: Phase differences for the antennas pairs.

affected by noise. To overcome this issue, UbiLocate considers both possible AoA values and computes the two resulting positions. UbiLocate then chooses the one that has the minimum distance to the position estimates from other APs.

Disagreement between position estimates. When UbiLocate combines estimates from very few APs, a single outlier may lead to large location errors. UbiLocate handles the specific case when only two APs are available for the localization. If the distance between location estimates is high, this indicates that the estimate of one of the AP is an outlier and thus combining the two estimates may degrade the location accuracy. UbiLocate then takes the AP with the lowest ToF estimate for the localization when the distance between the two position estimates exceeds 4 m.

4.3. Implementation

We build UbiLocate on the Nexmon project that provides a first step towards CSI extraction from several chipsets developed by Broadcom [70]. We largely improve over this prior work to consistently extract *accurate and reliable* CSI, implement features that make CSI extraction more flexible, and add support for timestamping both received and transmitted frames with very high accuracy.

For our implementation we select the Asus AC2900 RT-AC86U router since it supports 80 MHz 802.11ac with up to four spatial streams in a 4x4 MIMO configuration. The firmware that we developed replaces the standard one by Broadcom and can capture the CSI matrix for frames with configurable MAC addresses. In addition, it recognizes the type of frame, including the spectral width and the spatial configuration, and collects the CSI matrix accordingly. Since the router exposes only three antenna SMA plugs externally, we remove the front panel to access the fourth internal UFL connector and attach a custom antenna array handler to the four antennas of the router as shown in Figure 4.3.

We now discuss how we (1) validate the collected CSI and process it to estimate AoA and AoD, (2) provide the timestamping features, and (3) finally implement the enhanced FTM procedure.

4.3.1. Extracting accurate CSI

A range of preliminary measurements with our hardware platform reveal that calibration is needed to remove hardware imperfections that would otherwise affect the CSI and render it too unreliable for localization tasks. Specifically, we address the following problems:

Phase offset between antennas. While all the RF chains share the same sampling clock and reference signal (to tune to a given frequency), an unpredictable phase offset between each pair of antennas appears every time the system is tuned to a new Wi-Fi channel. As a result, the measured phase delay may not correspond to the one measured by the AoA or AoD algorithms. This unpredictable phase offset then remains flat over time.

Echos. We observe that the router generates echos from a received signal, i.e., the signal is repeated in the time domain. The time distribution of such echos is fixed and they never change.

We devise a procedure to remove these two imperfections which consists of a calibration experiment that has to be repeated every time we configure the equipment. During the setup, we capture a full CSI matrix with the four antennas connected to the same single-chain transmitter. This can be easily achieved by connecting the output ports of a 4-way splitter to four short cables that are also used to connect the four

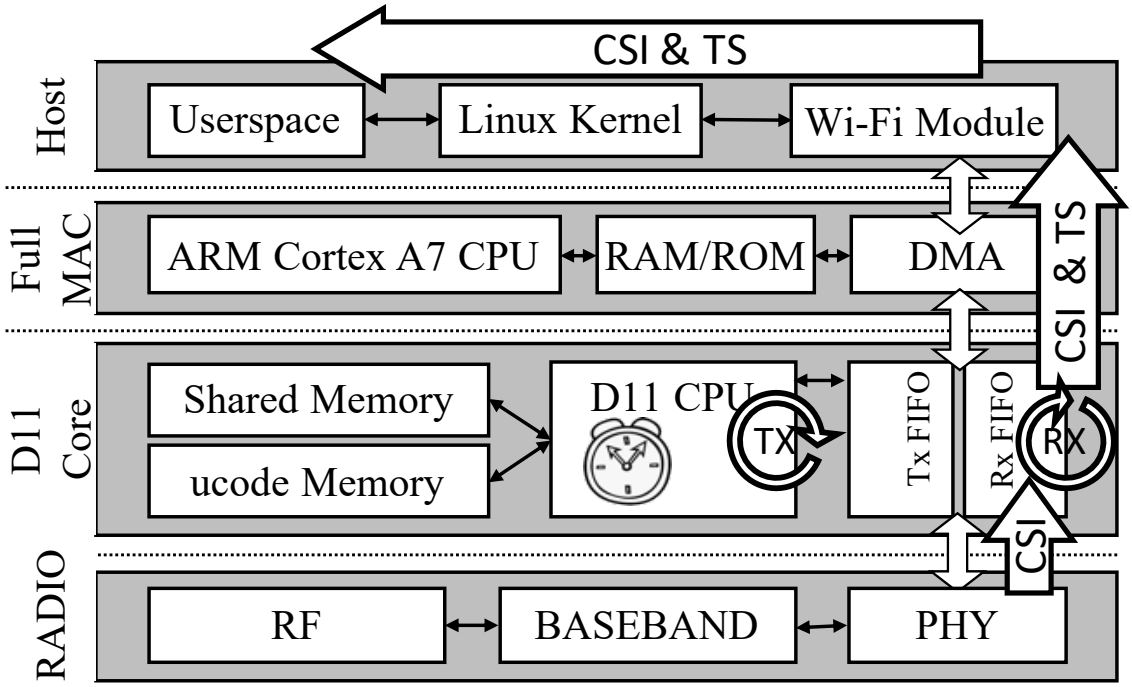


Figure 4.5: Enhanced CSI extraction platform with the modifications to collect ToF.

external antennas during the localization experiment later. The splitter ensures that all the signals arrive in phase, and hence the AoA of the transmitted frame is at 0 degrees. Thus, all phase offsets measured between the receive chains depend only on the (random) configuration of the local oscillator. The rationale behind this experiment is that the full CSI matrix captured during the calibration phase represents a reference signal that can be used for *correcting* the CSI vectors captured afterwards. To this end, we perform the (element-wise) Hadamard division $\oslash$ between new CSI vectors and the reference one:

$$calibrated\,CSI = CSI \oslash reference\,CSI$$

Figure 4.4 shows the phase offset between different pairs of antennas before (Figure 4.4(a)) and after applying the calibration (Figure 4.4(b)). The residual phase offset appears to be minor Gaussian noise, confirming that this procedure reliably removes the phase imperfections due to the hardware configuration.

4.3.2. Extracting timestamps

Implementing a ToF measurement procedure similar to standard FTM requires accurate time-stamping capabilities in both the transmit and receive directions, and the majority of the Wi-Fi chipsets, including the one in the chosen platform, simply lack them. We hence used the Nexmon firmware patching framework [71] to add these capabilities to the platform, following a similar approach to the one in [36]. The main modifications

involve the software that runs in the D11 CPU, a microcontroller that manages all time-critical operations such as channel access, beaconing, generation of reply frames, etc. This software consists of a single main loop that i) can neither be interrupted by internal IRQs nor by the upper layer ARM Wi-Fi core; and ii) branches into secondary functions when the hardware reports conditions that require additional work. In particular, we modify two functions that belong to the reception and transmit paths, respectively.

The first function is invoked when a preamble is detected and performs multiple checks on the first bytes of the incoming frame to decide how to process it. The CSI extraction patch already adds a single instruction loop that spins until the frame is completely received and then it pushes both the CSI data and the frame to the host. We further customize this loop by adding instructions to sample the value of the high-frequency clock of the system whenever the frame ends. We represent this modification in Figure 4.5 with the spinning wheel on the right data pipe (reception path). As shown, after the CSI data is retrieved from the PHY at the bottom, the RX timestamp travels up to the application that runs in user space and collects all the data.

The second function is invoked when the hardware verifies that all the conditions required for transmitting a frame are satisfied (i.e., the channel was idle long enough for the backoff counter to reach zero, no more energy is detected in the channel, no other operations are pending, etc). When this happens, the hardware is already transmitting the frame preamble. The function can then customize the transmission and monitor it until it terminates. We add two modifications here. The first writes the timestamps overheard during the previous transmissions as well as the device's own timestamp into the frame. The second consists of a new loop that waits until the end of the transmission and is represented in the figure with the spinning wheel on the left data pipe (transmit path). With this code we capture the transmit timestamp.

We obtain both timestamps by sampling the high-frequency clock that runs at the speed of the D11 CPU. For the chosen platform this corresponds to 192.6 MHz, and thus the receive timestamp has an uncertainty of 1.56 m when in perfect LOS without any multipath. In the presence of multipath, the timestamps are affected by the arrival of all the paths which leads to a bias. However, UbiLocate can extract an accurate time of arrival of the first path using the CSI data of the timestamp packet. Specifically, every path that arrives at the receiver introduces a phase rotation in the CSI of the subcarriers. By decomposing the channel in time domain, UbiLocate can eliminate the multipath bias and thus estimate a much more accurate time of arrival for the direct path.

4.3.3. Implementation of the FTM procedure

To evaluate the ToF between two nodes, we use the same set of equations as in Equation (4.10) for standard FTM. We consider two frames traveling in opposite directions—relatively close in time—and we combine the four corresponding timestamps

of two transmissions and two receptions. However, different from FTM, the frames do not belong to a specific *frame-ack* exchange. Instead, they are transmitted by the nodes asynchronously. In our experiments we transmit such frames frequently, so that frames from each AP are close in time to that of the client, but other strategies are possible: i.e., the client might initiate the procedure by transmitting a train of frames and all APs can schedule the same number of transmissions as soon as they receive the first frame from the client. We leave such modifications for future work.

To generate the ToF-related frames we use the injection capabilities available in the Nexmon CSI framework. We implement a user-space application that uses a PID controller to generate frames at a configurable rate, e.g., one frame every 4 ms. We also modify the D11 code to keep the same pacing at the access layer. This solution is key to avoid any DMA-related delay and ensure that at any moment in time there are enough "close" frames transmitted by all nodes, so that ToF estimation and thus ranging can be done. We further implement a back-pressure mechanism to avoid saturating the DMA memory when the queue holding injected frames starts to build up.

Another deviation from standard FTM is the fact that our implementation has no initiator and responder. For this reason, we cannot store timestamps at the responder and collect them later from the initiator. Hence, we modify the D11 code to store transmission timestamps directly inside the frame. To this end, we additionally modify one of the two functions described in the previous section. With this modification, we obtain all the necessary information for running the ranging procedure by capturing traffic traces at all nodes. In these traces we have the frames, corresponding reception timestamps and CSI data, and the transmission timestamps generated by the sender.

We finally describe the overall procedure for evaluating ToF between a pair of nodes N_1 and N_2. We start by processing the traces captured at each node, containing the frames received from the other one. We then align the clock of node N_1 to that of N_2. We extract from all frames collected by N_1 the reception timestamp (at N_1) and the transmission timestamp (generated by N_2). We then apply linear regression to remove the clock skew between the two nodes, adjusting both reception and transmission timestamps. For each frame transmitted by N_1 we associate the closest frame in time received from N_2 and we apply Equation (4.10) to the four-tuple of timestamps, yielding a ToF estimate. Since each AP transmits these broadcast ToF packets asynchronously, collisions are avoided by means of the standard DCF channel access mechanism of IEEE 802.11. However, we observe a variability in the ToF estimates due to systematic delays introduced by Wi-Fi packet processing similar to plain FTM [37]. These delay differences follow a Gaussian distribution which is centered approximately at the correct ToF value. We can thus remove this uncertainty by averaging over a certain number of estimates to compute a smoother ToF. We observe that 20 around estimates for good accuracy. As UbiLocate sends broadcast ToF packets every 4 ms, on average 80 ms are required to

compute a smoothed ToF estimate. We also tested UbiLocate's ToF with different levels of background traffic and do not observe any degradation in raw ToF estimation accuracy. With fully backlogged background traffic, which corresponds background traffic rate of 500 Mbps, UbiLocate gets around 40 ToF estimates per second which results in smoothing ToF estimates over 500 ms.

4.4. Experimental Evaluation

We now evaluate the location accuracy of UbiLocate in a realistic setup and compare it to several state-of-the-art location systems.

4.4.1. Testbed setups

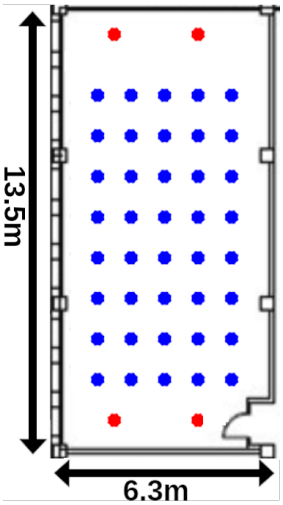
To provide a comprehensive performance comparison of UbiLocate and state-of-the-art location schemes, we test three different deployments. The first is a simple scenario with high AP density, where all APs have a LOS path to the station. This corresponds to the benign conditions under which location systems are usually tested. Second, we evaluate a medium density scenario where the station usually sees several APs with a mix of LOS and NLOS conditions, which tests the systems under adverse conditions. Finally, we move to a much larger and more sparse environment where usually only two or three APs are available at a time. This corresponds to the most common real-world deployments that are optimized for Wi-Fi coverage, rather than localization performance.

High density testbed The high density environment comprises four APs, each one placed in the corner of a room of size 85 m^2, as shown in Figure 4.6(a). The deployment has an AP density of 1/21.25 m^2. We further ensure that each AP has a clear direct path to the station.

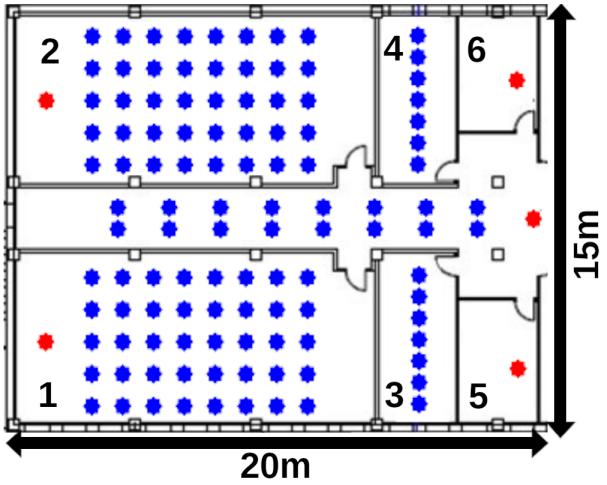
Medium density testbed The map of this testbed is depicted in Figure 4.6(b). The area is approximately 300 m^2, contains 5 APs, and has an AP density of 1/60m^2. It has seven distinct areas: six rooms, not all of which contain an AP, and one central corridor.

We consider 110 measurement points located in rooms 1, 2, 3, 4, and in the corridor, shown as blue dots in Figure 4.6(b). The five APs used to localize the target are shown as red dots and they are placed in rooms 1, 2, 5 and 6, and in the central corridor. With this deployment we ensure that: 1) the majority of target locations are in LOS with exactly one of the APs; 2) some of the target locations—the ones in rooms 3 and 4—are not in LOS with any of the APs; and 3) two APs—namely the ones in rooms 5 and 6—do not have a clear LOS to any of the target locations. Finally, for this deployment we also test different pure NLOS scenarios, where for each measurement point *we specifically remove the only AP that does provide LOS*, if any.

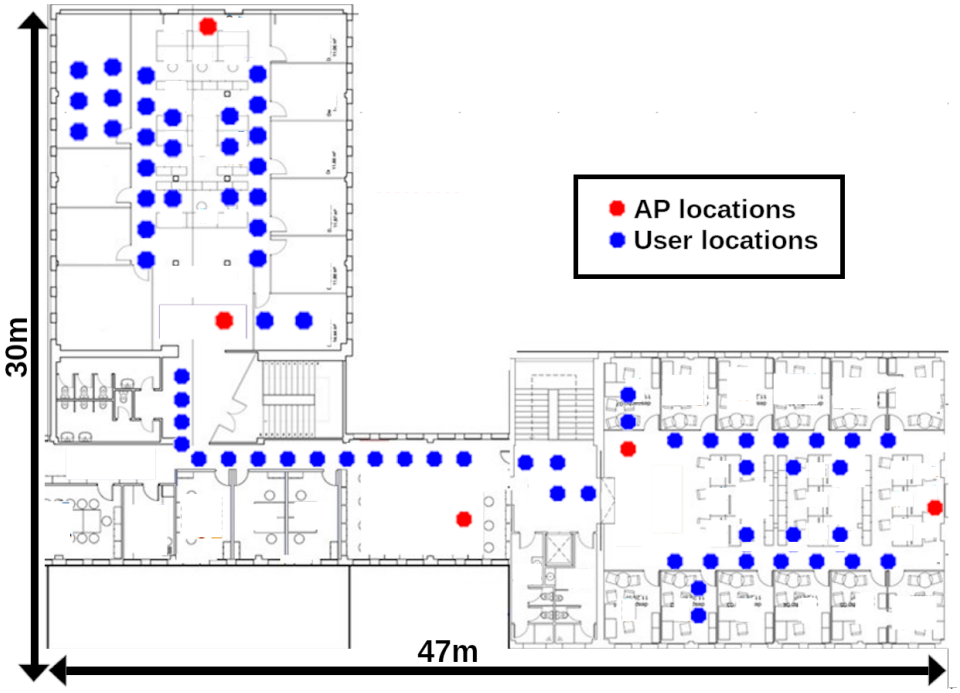
Low density testbed. This testbed pushes the location systems to their limit with a much more sparse deployment. This is in fact the most realistic scenario, with an AP



(a) High density testbed.



(b) Medium density testbed.



(c) Low density testbed.

Figure 4.6: Testbeds.

density close to that of the actual production Wi-Fi deployment in this office building. It comprises two wings of a building and one central area that connects them as shown in Figure 4.6(c), with a total area of 578 m^2 and an AP density of 1/115 m^2. Each wing contains an open plan area with desks, as well as closed offices on either side. The dividing walls, furniture and the people moving around (measurements were taken during daytime) create a rich multipath environment and many areas without LOS. The scenario comprises 70 measurement points, and each point is usually covered by only two (in the best case by three) APs, whereas in the other scenarios most measurement points are covered by all APs. As a result, in this setting it is crucial to properly merge the location information from the few APs within range.

In all the considered scenarios, the APs are working in monitor mode, extracting one CSI matrix for every received frame. For ranging, each AP exchanges 802.11 frames with the target device following the procedure described in Section 4.2.2. A central controller connected to the APs via Ethernet gathers all the data to compute the AoA and the distance for every AP as described in the previous sections. Finally, the algorithm presented in Section 4.2.3 is executed on the controller to estimate the position of the device.

4.4.2. Comparison with other systems

We benchmark the performance of UbiLocate against the following three state-of-the-art indoor location systems.

Spotfi [14] is a Wi-Fi location system which combines angle measurements from several APs to determine the device position. Spotfi computes AoA and path delay using a 2-dimensional MUSIC algorithm with spatial smoothing for accurate AoA estimates.

FUSIC [61] is based on ToF measurements to determine the device position. It relies on FTM ranging and uses the 1-dimensional MUSIC algorithm to reduce multipath effects.

SPRING [25] combines both AoA and ranging information to provide single AP localization. It uses the MUSIC algorithm for AoA and FTM for the distance. While SPRING was originally designed to work with only one single AP, in our experiments we average the estimates of all of the APs to provide a better position estimate.

We compare these systems against two different versions of UbiLocate, one that estimates the position using AoA, AoD and ToF, and a more basic version which only takes into account AoA and ToF. To distinguish different versions of UbiLocate and indicate the main features used by each system, we apply the following labeling scheme: letters A, D, and T identify a system using AoA, AoD and ToF information, respectively. For example, UbiLocate [AT] is used to refer to the basic implementation of UbiLocate that only uses AoA and ToF, whereas UbiLocate [ADT] refers to the full version that uses all information.

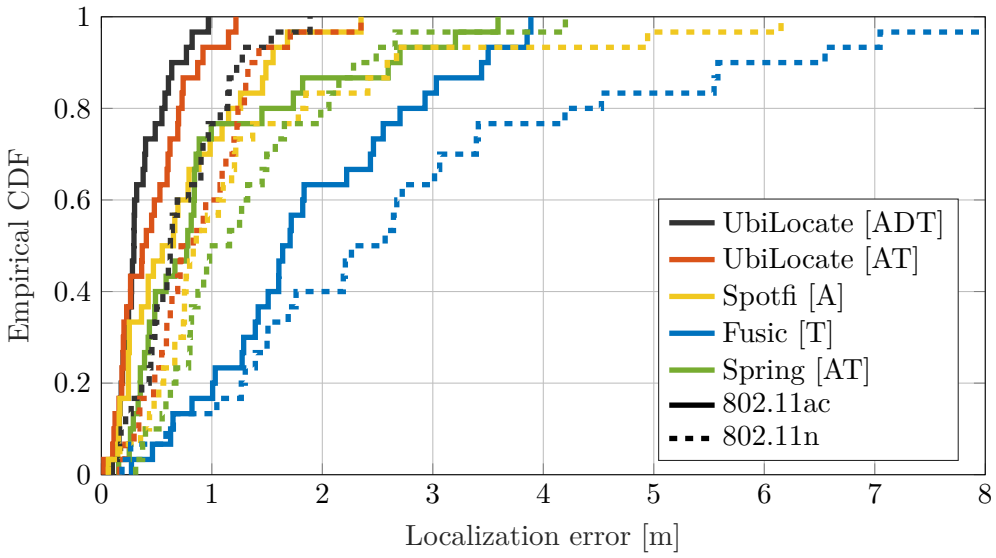


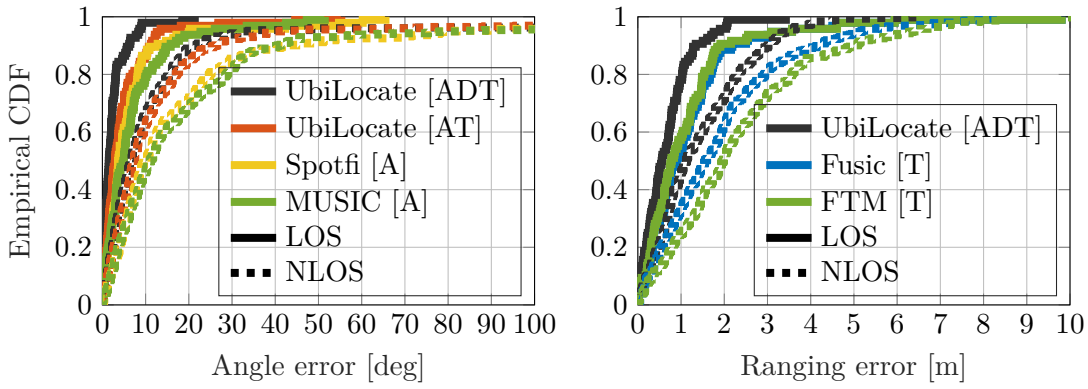
Figure 4.7: High density testbed.

Before we delve into the overall performance of our location system, we first study the performance of the individual components of UbiLocate, i.e., the angle and ranging estimates, in isolation. In particular, the UbiLocate AoA estimator is compared against the 1D MUSIC AoA estimator and against the one used in Spotfi. We also compare UbiLocate's ranging subsystem to vanilla FTM and to the improved FTM-based ranging proposed in FUSIC.

While the majority of the systems described above were designed for and evaluated with the older IEEE 802.11n Wi-Fi standard, we compare them against UbiLocate both for IEEE 802.11n and IEEE 802.11ac. SPRING also originally uses 80 MHz frames but is based on a proprietary Quantenna platform. In addition to the improved hardware capabilities of recent devices, the new 802.11ac standard supports 4x4 MIMO and up to 80 MHz of bandwidth. These features help significantly to resolve multipath effects. For reference, 802.11n systems are limited to 3x3 MIMO and up to 40 MHz of bandwidth.

4.4.3. High density scenario

The aim of this experiment is to localization performance of UbiLocate and the rest of the system in a benign multipath environment and to compare it against the state-of-the-art approaches in an environment similar to the one they have been designed for. This also allows validating that the performance of the state-of-the-art algorithms matches the results reported in the respective papers. To this end, we first consider a simple LOS environment, as indicated in Figure 4.6(a). It comprises four APs, with one AP placed in each of the corners of room 1. We use 40 location measurement points in an area of 85 m^2. In addition, to show how the improved capabilities offered by the 802.11ac



(a) AoA performance

(b) ToF ranging performance

Figure 4.8: Empirical CDF for AoA and ToF error for all APs, and for LOS (solid lines) and NLOS (dashed lines).

standard impacts location accuracy, we evaluate all methods both for IEEE 802.11ac as well as IEEE 802.11n configurations. As shown in Figure 4.7, when using 802.11ac frames, UbiLocate achieves sub-meter localization accuracy for all the target points and a median error of 30 cm for [ADT] and 40 cm for [AT]. On the other hand, Spotfi and SPRING have a median error of 60 cm and 70 cm and a maximum error of 2.3 m and 3.6 m. FUSIC has the worst performance with a median error of 1.7 m. As expected, we observe that moving from 802.11ac to 802.11n leads to a performance degradation for all of the systems. For example, UbiLocate's median error increases from 30 cm to 60 cm for [ADT] and from 40 cm to 85 cm for [AT], respectively. Since the relative performance of the approaches does not differ substantially between 802.11n and 802.11ac, for the remaining experiments we only compare the performance of all systems with an 802.11ac configuration. It is worth highlighting that UbiLocate [ADT] is the only system that achieves sub-meter location accuracy for all measurement points in the high density testbed, making it an excellent fit for location-based services that are sensitive to errors.

4.4.4. Medium density scenario

After evaluating UbiLocate in a simple LOS and dense environment, we now study how the individual features AoA and ToF behave in more complex settings with LOS and NLOS. Afterwards, we will show how these features translate into localization performance.

4.4.4.1. Analysis of individual features

We test the performance of the different angle and ranging algorithms in the large deployment scenario shown in Figure 4.6(b).

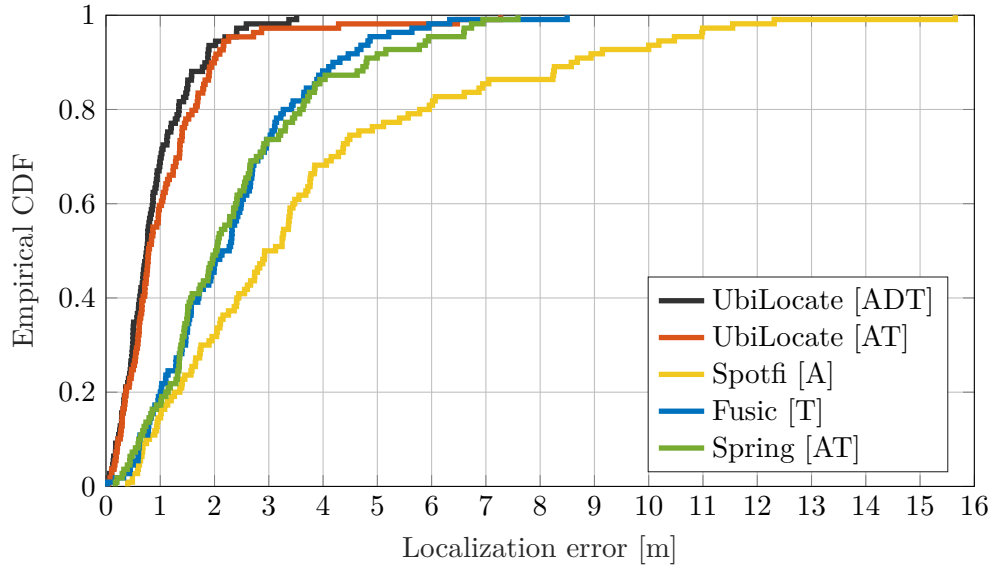
AoA. As shown in Figure 4.8(a), under LOS conditions UbiLocate achieves an excellent median error of 1 degree and 3 degrees for the [ADT] and [AT] versions respectively and a maximum error of 20 and 50 degrees, while Spotfi and MUSIC both have a significantly higher median error of 3.7 and 4.4 degrees and a maximum error of 65 and 55 degrees, respectively. For the measurement points that have NLOS, UbiLocate achieves a median error of 6 degrees while that of the other two approaches is above 10 degrees. A striking difference can be seen for the maximum error achieved 90% of the times: while UbiLocate has an error of at most 20 and 25 degrees for [ADT] and [AT] which is still partly usable, both Spotfi and MUSIC errors reach 40 degrees, which is indicative of significant outliers. We attribute our improvements to the Nelder-Mead search algorithm described in Section 4.2.1, which iteratively refines our estimates of the AoA by removing the effects of undesired multipath components. Note that the graph includes the raw estimates for all APs within range, whereas for the actual localization the AP estimates are weighted and filtered (i.e, not all estimates are used).

Ranging. Results for the ranging subsystem are shown in Figure 4.8(b). We verify that UbiLocate can perform ranging more accurately than FTM and FUSIC. Specifically, we measure a median error of 43 cm for UbiLocate (90% of the times below 1.3 m in LOS conditions), while FUSIC and FTM both achieve similar performance, with 0.8 m and 2 m for 50% and 90% of the cases, respectively. Also for NLOS conditions, UbiLocate ranging accuracy outperforms the other methods, with a median error of 1.1 m, while FUSIC and FTM have errors of 1.6 and 1.9 m. The key features of our system that enable this good performance are the accurate timestamping capabilities we added to the firmware of the devices (see Section 4.3).

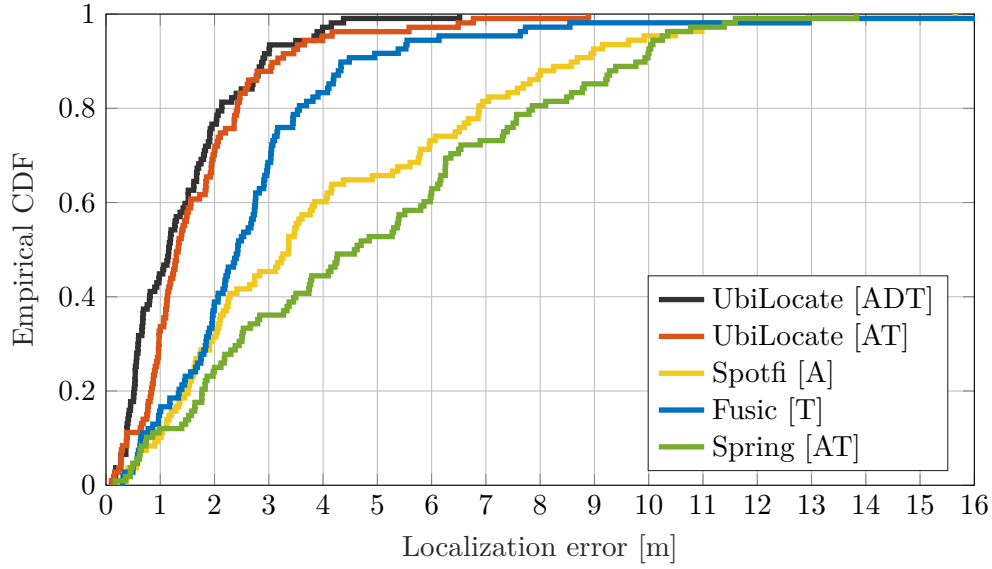
4.4.4.2. General localization

We now evaluate the overall localization accuracy of the different approaches in the medium density scenario. Specifically, we demonstrate the robustness of UbiLocate against NLOS and how UbiLocate deals with potentially contradictory location information from different APs in two spatial contexts: the LOS + NLOS deployment and a special case of only NLOS.

LOS + NLOS. This deployment scenario is shown in Figure 4.6(b) and comprises five APs and 110 measurement points. In the best case, there is only one AP in LOS while the other APs are in NLOS. Thus, it is critical to exploit primarily the information extracted from this AP as it provides the most accurate location information, while minimizing the contribution of unreliable information from some of the NLOS APs. The results are shown in Figure 4.9(a). Clearly, UbiLocate achieves a significant median accuracy improvement of around a factor of 2 compared to state-of-the-art algorithms for both UbiLocate versions. Specifically, UbiLocate's median error is 0.75 m while for SPRING, FUSIC and Spotfi it is 2 m 2.1 m and 3 m, respectively. Furthermore, the maximum



(a) Medium density LOS+NLOS.



(b) Medium density NLOS *only*

Figure 4.9: Localization performance of UbiLocate compared to state-of-the-art systems.

error of UbiLocate is 3.5 m and 6 m for the [ADT] and [AT] versions, whereas the maximum errors of SPRING, FUSIC, and Spotfi are much higher at 7.5 m, 9 m and 15.5 m, rendering them unsuitable for many indoor location based services. While the median errors of [ADT] and [AT] are similar, the additional AoD information used in [ADT] significantly reduces the maximum error compared to [AT]. This superior performance is not only related to the more accurate AoA and ToF subsystems of UbiLocate, but also the particular localization strategy that identifies the most reliable APs and weighs their contributions based on their estimated quality. For completeness, we also tested UbiLocate [A], i.e., a pure AoA system which runs only on the APs without any station-side modifications. It achieves a median error of 1.2 m.

NLOS-only Finally, we evaluate the systems in a setting, where we force all measurement points to be in full NLOS. To this end, we remove the respective APs that does provide LOS information, if any, i.e., for the measurement points in room 1 we remove the AP in room 1 and test the localization performance in that room with the remaining APs. This process is repeated for all other rooms as well. While this scenario is extreme and LOS will be available for at least some of the locations in a regular deployment, it gives a good indication of the expected performance when additional moving obstacles (such as persons) in the rooms obstruct and distort the only available LOS path.

The results are shown in Figure 4.9(b). There is a small performance degradation in localization accuracy, but UbiLocate still provides meter-level median accuracy with an error of 1.1 m and 1.2 m for the [ADT] and [AT], respectively. In contrast, the median errors for SPRING, FUSIC, and Spotfi are 4 m, 2.6 m and 3.5 m, respectively, around a factor of 2 to 3 worse than UbiLocate.[1] Finally, UbiLocate [A] has a median error of 2.2 m. This good overall performance of UbiLocate in NLOS indicates that path information from APs under obstructed LOS is valuable, if the paths can be resolved accurately.

4.4.5. Low density scenario

Compared to the previous scenarios, the low density scenario shown in Figure 4.6(c) is much more sparse and for half of the points the client only sees two APs. This AP density is realistic for real-world deployments, where coverage depends very much on the geometry of the deployment. For the points with only two APs, Fusic and Spotfi cannot determine a location since they both need at least 3 APs within range to locate the user. In addition, the office furniture and the people moving around produce a rich multipath environment and dynamic channel conditions. We again first show the performance of the individual features and then the general localization performance.

[1] Note that in [14] a higher NLOS accuracy for Spotfi was reported. However, their NLOS deployment typically has around two APs with LOS per measurement point, whereas we consider as true NLOS only points for which *none* of the APs are in LOS.

4.4.5.1. Individual features

We compare the AoA and ToF estimation in this challenging case to the previous scenarios. Again, the graphs include the raw estimates for all APs, whereas for the actual localization, UbiLocate filters out some of the outliers.

AoA. The AoA results are shown in Figure 4.10(a) with UbiLocate obtaining a median error of 2.7 and 8.5 degrees for LOS and NLOS settings for the [ADT] version and 4.3 and 12 degrees for [AT]. Spotfi and MUSIC have similar performance and achieve a median error of 5.6 and 6.2 degrees for LOS and 12 and 13 degrees for NLOS, respectively. This degradation in the LOS and NLOS performance is caused by the larger distances and the rich multipath compared to the medium and high density scenarios.

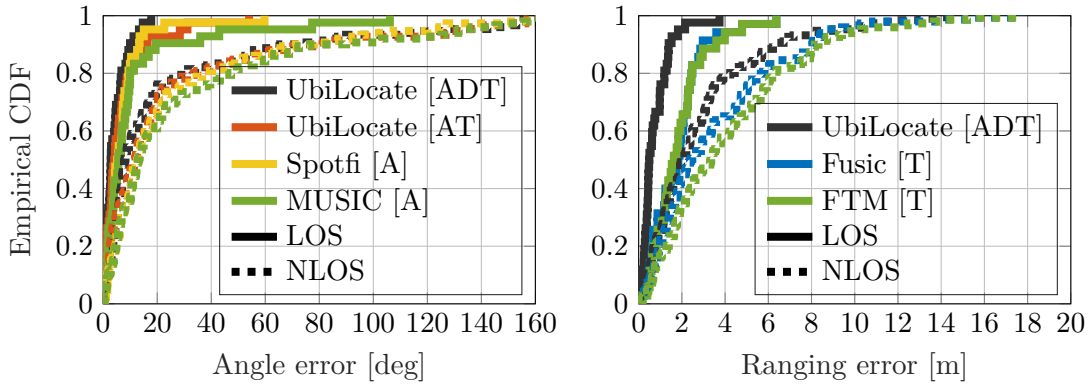
Ranging. As shown in Figure 4.10(b), UbiLocate has an excellent median error of 0.5 m in LOS while for NLOS it achieves 2 m. FUSIC and FTM have the same median error of 1.8 m for LOS and 2.8 and 3.4 m in NLOS, respectively. UbiLocate has the lowest maximum error of 12 m, while Fusic and FTM errors reach 18 m.

4.4.5.2. General localization

The localization errors can be found in Figure 4.11. Since Fusic cannot be applied for all of the measurement points, its CDF curve do not reach 1, whereas SPRING and Spotfi do since they work with just a single AP or two APs respectively. As in the other evaluation, the [ADT] and [AD] versions of UbiLocate have similar performance with a median error of 1 m, while Spring achieves a 4 m error and Spotfi a 4.1 m. Regarding the highest errors, UbiLocate [ADT] and [AD] reach 10 m while SPRING and Spotfi have up to 24 m and 29 m. As expected, the low AP density and the rich multipath environment produce larger outliers compared to the medium density NLOS case. Similar to Spotfi, UbiLocate [A] achieves a median error of 2.8 m. The few large outliers with UbiLocate come from extreme points in far corners of the building that have large angles to the one or two APs within range under NLOS. In those cases, achieving better accuracy is only possible by deploying another AP. It is worth highlighting that UbiLocate generally deals very well even with such a sparse scenario with a complex channel environment, a low error in most cases. While the performance of UbiLocate is similar for the low density and NLOS-only scenarios, the reasons are different. The NLOS-only scenario is more dense with four APs in coverage, but none of them have a clear LOS, whereas the low density scenario has only 2 or 3 APs to localize, but there are cases with a clear LOS. These different effects happen to compensate each other in the specific scenarios under study.

4.4.6. Additional Considerations

Impact of MIMO and bandwidth. All the experiments described up to this point are performed with the same 4x4 MIMO configuration with 80 MHz bandwidth. However,



(a) AoA performance

(b) ToF ranging performance

Figure 4.10: Empirical CDF for AoA and ToF error of UbiLocate compared to state-of-the-art systems for the low density scenario for all APs.

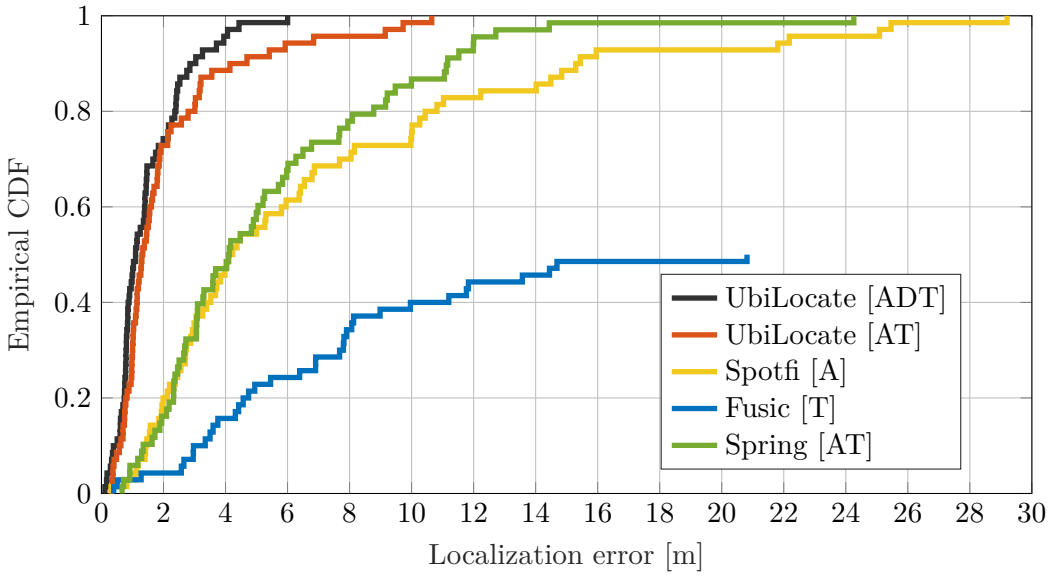


Figure 4.11: Localization performance of UbiLocate and state-of-the-art systems for the low density scenario.

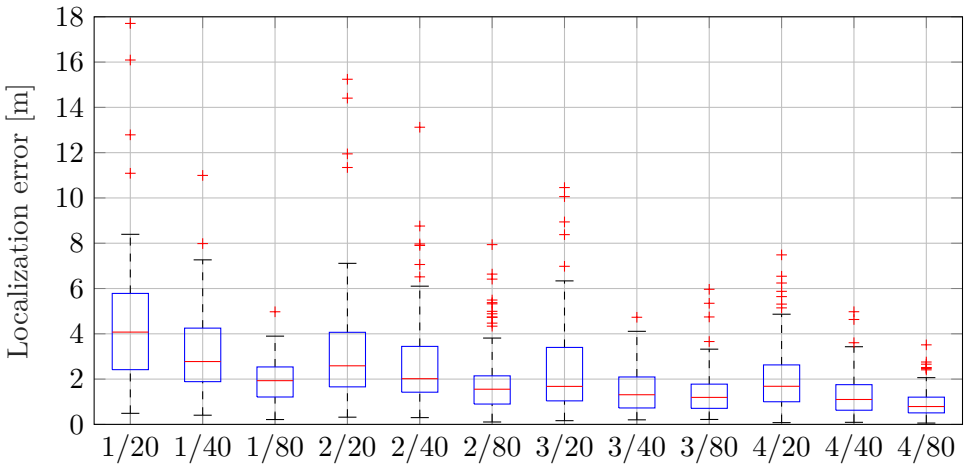


Figure 4.12: UbiLocate location accuracy of different configurations of (number of antennas/bandwidth)

in principle UbiLocate can work with any hardware configuration. To characterize the localization performance for devices ranging from low-end to high-end hardware complexity, we evaluate UbiLocate for the following bandwidth and MIMO configurations in the medium density testbed (LOS + NLOS). We consider three bandwidth combinations (20/40/80 MHz) and four antenna configurations (1x1/2x2/3x3/4x4), resulting in 12 configurations overall.

The results of this evaluation are illustrated in the box plot in Figure 4.12. Let us first consider the 4x4 MIMO configuration. As expected, the median error rises from 0.7 m when working with a bandwidth of 80 MHz to 1 m and 1.6 m when reducing the bandwidth to 40 MHz and 20 MHz, respectively. The worst performance is obtained with the single-antenna system and 20 MHz of bandwidth, with a median error of 4 m and a maximum error of 18 m. For comparison, the median error with one antenna and 80 MHz is only 1.8 m. Finally, the importance of AoA information can be seen from the sudden drop in the median localization error when moving from the 1x1 to the 2x2 MIMO configurations. However, decimeter-level median accuracy can only be achieved with 3x3 and 4x4 MIMO and 80 MHz, or with 4x4 MIMO and 40 MHz bandwidth, indicating that 802.11n hardware capabilities with 3x3 MIMO and 40 MHz are insufficient to achieve this very high accuracy.

Time complexity. Time complexity plays a crucial role especially in real-time processing. The dimensionality of the parameters and their granularity considerably affect the time complexity of the optimization algorithm. To evaluate it, we run the two versions of UbiLocate and Spotfi using the traces collected during the localization evaluation. UbiLocate applies Nelder-Mead search for the five most significant paths. The server used for this evaluation is an Intel(R) Core(TM) i7-6800K CPU with 3.40GHz and

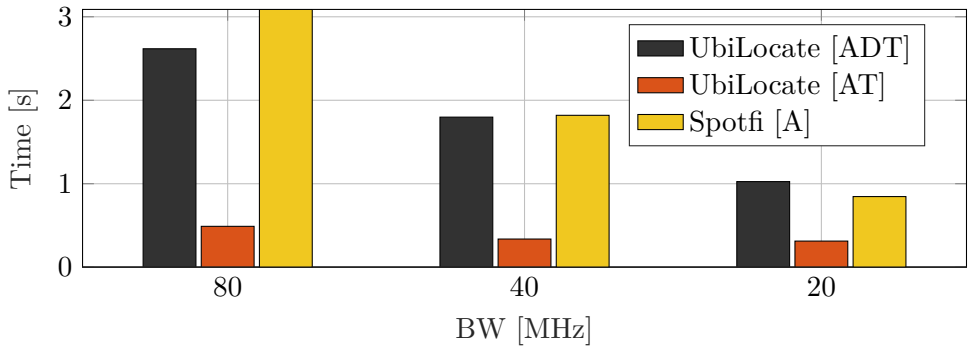


Figure 4.13: Time complexity for UbiLocate and SpotFi.

16 GB of RAM running MATLAB 2019a. The results are illustrated in Figure 4.13. We observe that for 80 MHz, UbiLocate [ADT], which estimates the three path parameters, is faster than Spotfi which only estimates two. UbiLocate [AT], which estimates two parameters, has an execution time of half second and reduces the time complexity by 85% compared to Spotfi. The significant difference in time complexity between [AT] and [ADT] comes from the combinatorial complexity with respect to the number of path parameters to be estimated. This is exacerbated by the high number of spatial streams of the MIMO system, since the channel complexity increases with the possible transmit/receiver antenna pairs.[2] While the time complexity of UbiLocate [AT] is significantly lower than that of [ADT], adding AoD information does reduce the maximum localization error, as discussed in Section 4.4.4. In addition, we observe that when we reduce the bandwidth by half, the time required to run the system is also approximately reduced by half. Our implementation uses unoptimized Matlab code using the predefined Matlab functions. We expect an improvement of the time complexity by a factor of 5-10 with an optimized implementation in native C.

4.5. Related Work

Wireless localization is a very hot topic and has been widely studied both from theoretical and practical perspectives [72]. Below, we survey the most important approaches in the research area.

Path parameter estimators. Extracting the path parameters of the radio-frequency signal has been largely analyzed for positioning purposes, especially in the field of AoA estimation. Many classical algorithms such as MUSIC [32] and ESPRIT [54] are currently used but they do not well resolve AoAs of highly correlated signals [73]. Spatial smoothing techniques allow decorrelating these signals [74, 75] and provide better

[2]Note that if Spotfi were to consider AoD together with AoA and the path delays, its time complexity would increase over-proportionally, since the Nelder-Mead search we use deals with the complexity increase more efficiently than 3D MUSIC.

performance. Compressed sensing further improves over these algorithms [33, 57, 58]. These schemes rely on a search that minimizes the difference between the overall signal and the superimposed signals in terms of the path parameters. However, the complexity of the algorithms is computationally prohibitive when multiple path parameters are estimated jointly, due to the extremely high number of possible combinations. To deal with that, UbiLocate firstly estimates the path parameters iteratively, and then refines them using the Nelder-Mead search algorithm.

Active localization. Here, the goal is to estimate the position of the device which sends the radio frequency signal. We can distinguish the following main approaches:

RSSI-based: The propagation losses of the radio frequency signal is modeled to estimate the distance between transmitter and receiver. Many well-known models can be found in the literature [17,35,76–79]. Unfortunately, this approach has been demonstrated to provide limited accuracy compared to other approaches, as the received power depends on many environmental factors.

ToF-based: Timestamps are used in the MAC layer together with echoing techniques to measure round trip time [80–83], and consequently the distance between AP and the target device. This can be extended using dead reckoning [84] to provide the user location with only a single AP. This concept was later standardized as the FTM protocol. It was tested in [37] and in [60], where the claimed sub-meter accuracy was validated. However, this accuracy can usually not be achieved in rich multipath environments, whereas our approach is better able to deal with multipath.

AoA-based: Estimating the angle of arrival from an incoming signal is a well-known topic in the field of array processing [85]. Combining angle of arrival measurements from several APs can provide very good localization accuracy. This was validated in [86] where sub-meter accuracy was achieved with large antenna arrays that are not yet feasible in Commercial Off-The-Shelf (COTS) devices. This work was extended to COTS devices in [14] and in [55], where a two-dimensional (2D) MUSIC implementation (AoA+ToF) is carried out, improving the performance of 1D MUSIC at the cost of increasing the computational complexity. It has been also extended to 3D (AoA+ AoD + ToF) in [87].

Hybrid (RSSI/ToF + AoA)-based: Several systems combine angle and distance measurements to localize a device from a single AP. SPRING [25] combines AoA and ToF data derived from two separates hardware devices. Also CUPID [34] extracts angle information from CSI but uses only coarse RSSI to estimate distance. UbiLocate exploits both angle and ToF information in the same device.

NLOS. The NLOS case was rarely tackled in the past because it is an extremely challenging problem. Having the main path partially or completely obstructed by an object significantly complicates accurate path parameter estimation. The majority of prior works dealt with NLOS using the high bandwidth available in ultra-wideband systems [88–91]. For Wi-Fi there are several proposals for imaging and mapping through walls [22,56,

90], but they need flexible high-performance hardware such as software-defined radios and custom antenna arrays. In addition, active anchors [92,93] and reconfigurable intelligence surfaces [94,95] help dealing with NLOS and improve positioning accuracy in NLOS cases, but such special purpose hardware is not available in regular Wi-Fi deployments. While several localization systems claim to tackle NLOS issues, many of them evaluated the localization accuracy in mixed LOS/NLOS environments with a very high fraction of available LOS paths [14, 96]. None of them were evaluated under pure NLOS conditions. To the best of our knowledge, UbiLocate is the first Wi-Fi location system that not only works in pure NLOS scenarios, but even achieves sub-meter accuracy.

Wi-Fi testbeds. The most widely used CSI extraction tool for localization is [62] and a lot of works build upon this platform. However, it uses the outdated IEEE 802.11n standard, which limits the potential performance and foregoes the hardware capabilities of new Wi-Fi standards such as 802.11ac. Designs based on software-defined radios are appealing due to their high-quality RF hardware, flexibility, and powerful processing capabilities of FPGAs. There are even full stack Wi-Fi implementations for 802.11a/g/n and 802.11a/g/p available through openWi-Fi [97] and GNU Radio [98]. With such software-defined radio systems, the clock can be sampled more accurately and with the reduced dispersion the ToF measurements would need little or no averaging compared to UbiLocate, whereas the CSI and thus angle estimation accuracy would be largely the same. However, for practical real-world deployments, it is of critical importance that location systems can be implemented on COTS devices without modification to the underlying hardware.

4.6. Conclusions

In this chapter, we tackle the challenges of accurate wireless localization in *realistic* indoor Wi-Fi deployments. While many works in the recent literature achieve excellent performance under ideal conditions with high AP densities, we target two critical assumptions that are key to realistic environments: i) the prevalence of NLOS paths when estimating a device position, and ii) the scarcity of APs, i.e., "anchor" nodes with known location. Based on these assumptions we developed UbiLocate, an IEEE 802.11ac-based Wi-Fi location system that works with realistic AP deployment densities. UbiLocate exploits both a refined AoA extractor and a fine-grained ToF ranging system to achieve sub-meter accuracy even in tough NLOS conditions. Our experimental evaluation in a number of common scenarios shows an overall improvement of the localization performance by a factor of 2-3 compared to state-of-the-art systems, both under LOS and NLOS conditions.

5 AX-CSI: Enabling CSI extraction on commercial 802.11ax Wi-Fi platforms

5.1. Introduction

Technology is always evolving and wireless protocols are providing higher data rate as well as lower latency. These improvements enable a range of new applications from virtual reality to remote surgery, and they are possible by the improved hardware characteristics of the latest wireless protocols. In particular, the newest Wi-Fi protocol, IEEE 802.11ax, provides a 4x4 Multiple-Input Multiple-Output (MIMO) setting with a channel bandwidth of 160 MHz and four times denser spectrum than its predecessor, IEEE 802.11ac, which only supports 4x4 MIMO with 80 MHz of bandwidth. These two hardware features allow superior accuracy of wireless localization and sensing systems.

However, all state-of-the-art works [99] were limited by the physical characteristics of the Very-High Throughput (VHT) PHY of IEEE 802.11ac—or even older PHYs—and do not exploit the new features from the new 802.11ax standard, like the new structure of High Efficiency (HE) frames. This work presents the first publicly available system[1] that can extract Channel State Information (CSI) data from off-the-shelf devices supporting the HE PHY introduced in the latest 802.11ax Wi-Fi standard, with bandwidth up to 160 MHz and 4x4 MIMO. These new features likely enable unprecedented improvements for any CSI-based applications. To assess it, we implement a UbiLocate (the proposed IEEE 802.11ac-based localization system in Chapter 4) version that works for the new standard. Our preliminary results show that IEEE 802.11ax enables, as expected, a more accurate localization performance since it improves positioning by a factor of 1.75 in Line-Of-Sight (LOS) and Non-Line-Of-Sight (NLOS) settings, compared to IEEE 802.1ac.

The rest of the chapter is organized as follows. We first present the different solutions available today for CSI extraction in Section 5.6. We then introduce our system in Section 5.2, and we compare its performance with its predecessor in Section 5.3. In Section 5.4, we show for the first time some impressive results achieved with the new HE

[1] We release our CSI tool at https://ans.unibs.it/projects/ax-csi

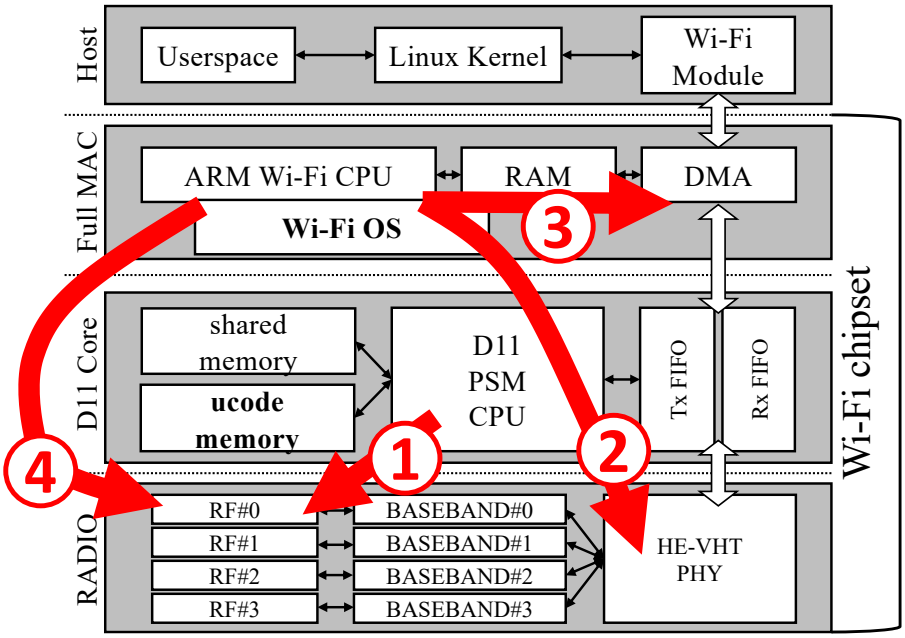


Figure 5.1: Overview of the CSI extraction process in the adopted architecture. Operations are split between the D11 core and the ARM CPU and performed in the order indicated by the arrows. When a target frame is received, the D11 core switches off the radio (1); then, the ARM CPU reads the CSI data (2), sends them to the userspace (3), and finally restarts the radio (4).

PHY in 802.11ax. Section 5.5 shows the details of the localization evaluation as well as the results. Finally, we draw the conclusions in Section 5.7.

5.2. The AX-tended CSI extractor

The implementation of this new CSI extraction tool has been inspired by our previous work on 802.11ac [100]. More specifically, we ported the tool we developed for the Broadcom 4365 chipset and adapted it for the new Broadcom 43684 802.11ax chipset. The reference platform is the Access Point (AP) RT-AX86U, developed by Asus, which incidentally is the successor of the RT-AC86U, the previous reference for our 802.11ac CSI extractor. However, adapting the tool to 802.11ax required significant modifications.

Figure 5.1 shows the classic FullMAC architecture adopted by Broadcom. All the 802.11-specific functions are managed internally by the chipset; on the Host, the main Linux operating system only configures the radio and exchanges data through the top DMA interface in the form of Ethernet-like traffic. Inside the wireless card, operations are split between the ARM CPU and the D11 microcontroller. The ARM CPU runs the "Wi-Fi OS" that controls all the functions that are not time-critical while the D11 microcontroller manages time-sensitive operations, like channel access and generation of reply frames. In our previous 802.11ac tool, CSI extraction is entirely managed by the

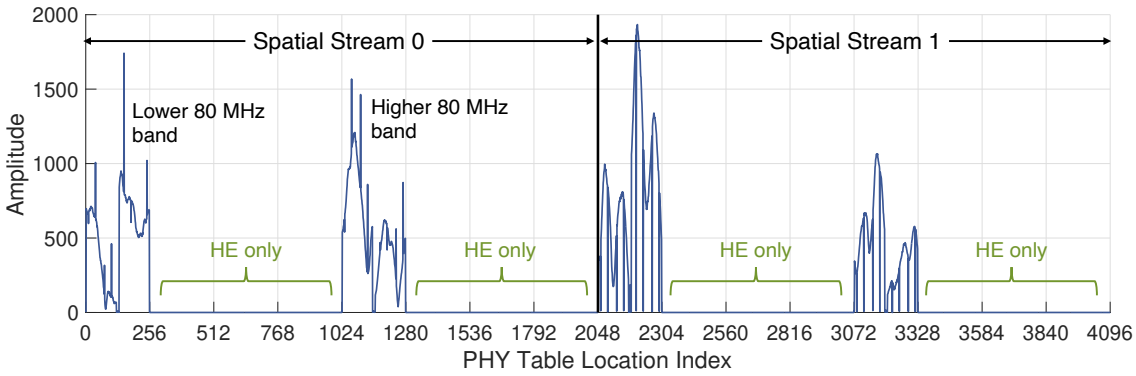


Figure 5.2: Layout in the PHY table of core #0 of the data associated to every OFDM subcarrier for a 160 MHz VHT frame using 2x2 MIMO. Only two spatial streams are reported here; four of them will require up to 8192 values.

D11 core: when a new target frame is decoded by the underlying hardware, i) it switches off the receiving circuitry to freeze the CSI data; ii) it extracts the CSI data from the PHY and pushes it to the ARM CPU in the form of additional frames following the original payload; and iii) it finally restarts the radio receiver. No delays occur with this approach, and CSI data flow from the bottom to the top through the vertical double arrows in Figure 5.1 on the right. Unfortunately, we could not directly port this approach from the old 4365 to the new 43864 chipset. In the latter, in fact, the ucode memory is already almost full because of the complexity of 802.11ax operations. There is room only for a few instructions inside the main loop, and the ARM CPU must now perform most of the CSI-related operations.

We will now describe in detail how the original 802.11ax architecture has been modified to extract the CSI, with reference to Figure 5.1. First, we patched the D11 core to react to frames with specific content only, e.g., to a specific MAC address or frame control type. When a new target frame is received, the D11 core stops the radio (1) and no other frames can be received from now on. Then, the frame is pushed through the Rx FIFO towards the memory of the ARM CPU as it would usually happen in the standard architecture. We modified the main function in the Wi-Fi OS that processes incoming frames. When the frame coming from the Rx FIFO is detected, a new function is called to extract the CSI data from the PHY (2), embed it into one or more crafted UDP datagrams, and deliver it to the Host through the DMA interface (3). An application running on the main CPU of the Host receives the UDP datagrams containing the CSI and stores them into a packet trace. Finally, once all the CSI data are sent to the Host, the modified function in the Wi-Fi OS re-enables the radio to receive new Wi-Fi frames (4). As we will see later, performing all these operations in the ARM CPU rather than in the D11 core is slower, and in general it reduces the CSI throughput of the platform.

Table 5.1: Mapping of OFDM subcarriers to memory indices for the first PHY table for a generic receiving radio core. For spatial stream **k**, the corresponding PHY table starts at location $\mathbf{k} \cdot \mathbf{2048}$.

PHY	Bandwidth	Subcarriers	Memory indices
VHT, HT and Legacy	20 MHz	64	[0, 64)
	40 MHz	128	[0, 128)
	80 MHz	256	[0, 256)
	160 MHz	512	[0, 256) and [1024,1280)
HE	20 MHz	256	[0, 256)
	40 MHz	512	[0, 512)
	80 MHz	1024	[0, 1024)
	160 MHz	2048	[0, 2048)

5.2.1. CSI data layout

Like in the previous 802.11ac chipset, CSI data are organized into four PHY tables, one for each of the four radio cores. However, the structure of the tables is different, as it accounts for the higher number of Orthogonal Frequency-Division Multiplexing (OFDM) subcarriers available in the HE PHY introduced with IEEE 802.11ax.

First of all, our preliminary analysis revealed that each PHY table now holds up to 8192 complex values: trying to read more leads to crashing the system. This number corresponds exactly to the maximum number of subcarriers expected in a 160 MHz HE frame with 4x4 MIMO, since every single stream can have up to 2048 subcarriers. Further investigation allowed us to map the data in the PHY tables to specific OFDM subcarriers, as shown in Table 5.1, where we report the offsets for a generic spatial stream; in general, CSI data for the k-th stream start at offset $k\cdot$ 2048. We have determined the CSI data layout by dumping the content of the PHY table under different conditions: i) by tuning the radio on different bandwidths; ii) by receiving frames with different bandwidths; iii) with different types of encodings (i.e., VHT or HE); iv) and MIMO configurations. Since UDP datagrams are limited to 1500 B, we can embed no more than 256 CSI values in each datagram reporting the CSI to the user space. While one single UDP datagram is sufficient to report one spatial stream of a 80 MHz VHT frame, two are required for each spatial stream of a 160 MHz VHT frame. This implies that 32 datagrams are generated when extracting 160 MHz VHT CSI with 4x4 MIMO, which increases to 128 for HE frames of identical configuration.

Interestingly, the layout of the CSI data in the PHY tables is rather peculiar, especially for 160 MHz-wide VHT transmissions, as shown in Figure 5.2. The 160 MHz spectrum is not reported in contiguous memory locations, but it is split into two halves corresponding to the lower and higher 80 MHz parts of the spectrum. This gives us a useful insight into the implementation of the radio subsystem on the chipset: rather than a single radio working at 160 MHz, the system likely employs two radios at 80 MHz, each one

providing on half of the entire band. A similar mechanism has been already observed on the old 802.11ac 4365 chipset. Even though this chipset officially supports only 80 MHz bandwidth with 4x4 MIMO, we experimentally extracted 160 MHz CSI with 2x2 MIMO by assigning different radios to different parts of the spectrum. The new chipset apparently uses the same principle but doubled the number of radios in order to achieve 4x4 MIMO at 160 MHz. We believe that the rationale behind this implementation is that the same circuitry should be able to manage not only full 160 MHz transmissions, but also 80P80 configurations in which two 80 MHz signals can be located everywhere in the 5 GHz band. In addition, using two radios with smaller bandwidth might be cheaper than implementing a single radio with very large bandwidth.

5.2.2. Testing the CSI extraction platform beyond 80 MHz with SDRs

Our experiments require crafting Wi-Fi frames with precise features in order to showcase the performance of the novel CSI extraction tool. For this reason, we resort to Software-Defined Radio (SDR) platforms for some of the tests presented in the next sections. While we also added functions to the CSI extraction tool to inject HE-encoded frames through the Asus AP, we cannot finely control their timings. SDRs radios, instead, can transmit frames with very accurate timings: still, transmitting 160 MHz frames requires quite expensive hardware. For instance, the USRP N300 SDRs manufactured by Ettus Research, which is already an expensive solution, can manage the transmission of Wi-Fi frames up to 80 MHz. We present here a workaround that allows us to use a couple of these devices to transmit up to 2x2 160 MHz HE-encoded frames, or alternatively four much cheaper USRP N210 SDRs to transmit similarly encoded frames but limited to a 1x1 spatial configuration.

In Figure 5.3 we focus on the first solution and we show how we can jointly use two N300 SDRs radios to transmit 160 MHz Wi-Fi frames. Each board is responsible for the transmission of half of the signal, i.e., either the lower or higher 80 MHz portion of the complete frame spectrum. When properly synchronized, the two halves sum up and generate a valid 160 MHz OFDM frame.

In Algorithm 1, we summarize the steps needed to generate the two 80 MHz portions of the frames in MATLAB using the WLAN Toolbox. By default, MATLAB creates a wide-band Wi-Fi signal by generating the corresponding I/Q samples that should be transmitted at 160 MS/s. Our script parses the MATLAB vector with the I/Q samples and recovers the complete sequence of OFDM symbols. Knowing the specific format of each OFDM symbol, the script applies to it an FFT and obtains the original OFDM constellation spectrum that is composed of 512 subcarriers in a VHT frame, or 2048 in an HE frame. Splitting each OFDM symbol into two *semi-symbols* in the frequency domain is straightforward as we only have to consider half of the total subcarriers, either in the lower or in the higher part of the spectrum. At this point, our script simply applies an

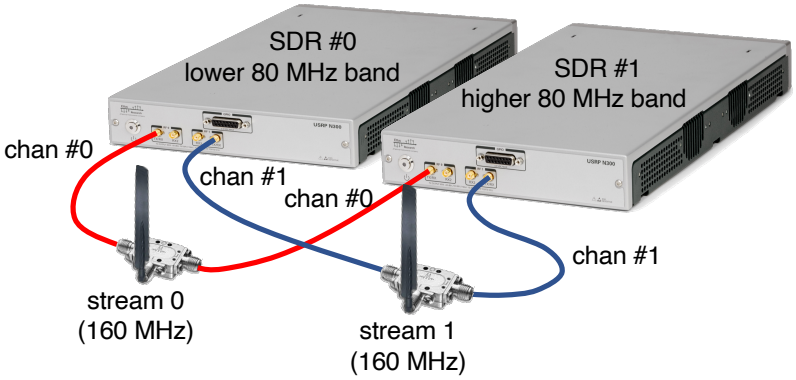


Figure 5.3: SDR setup for transmitting 160 MHz frames with 2x2 MIMO using two SDRs with smaller bandwidth. The SDRs need to be externally synchronized.

Algorithm 1: Split a 160 MHz Wi-Fi frame generated with the MATLAB WLAN Toolbox into two 80 MHz semi-signals. The spectrum of each semi-signal corresponds to the lower/higher part of the spectrum of the original signal.

Require: 160 MHz Wi-Fi frame at 160 MS/s
Ensure: Two 80 MHz signals at 80 MS/s whose joint spectrum is equivalent to the one of the input signal
1: Separate OFDM symbols & remove guard interval
2: Apply FFT to each OFDM symbol
3: Split left/right (low/high) band
4: **for** each semi-signal **do**
5: Apply IFFT on each half-symbol
6: Add back guard interval
7: **end for**

IFFT to each semi-symbol and, after adding back the guard interval as dictated by the standard, it ends up with two *semi-signals*, each one having half of the spectral content of the original 160 MHz signal.

In order to transmit a decodable Wi-Fi frame, the two semi-signals must be transmitted by two separate USRP N300, one tuned to the central frequency of the lower 80 MHz portion of the spectrum and the other tuned to the center of the upper 80 MHz one. We cannot, in fact, use the two TX chains of a single USRP N300 as they cannot be tuned to different frequencies by design. While Figure 5.3 also shows that two separate SDRs—each with two TX chains—allow transmitting 2 spatial streams, we only use a single stream configuration in this work. Needless to say, the two semi-signals must be synchronized: in our experiments, we achieve this by using the clock distribution module named Octoclock, also manufactured by Ettus Research.

5.3. CSI Extraction Performance

To cope with the limited amount of space available in the ucode memory of the new D11 core, we had to move a large part of the CSI processing code to the memory of the ARM processor. Even though the ARM CPU is quite fast, the concurrence with the D11 core and the increased latency have a detrimental effect on the system's performance.

In the old CSI extraction system [100], the D11 core had complete control over the data transfer operations. Once a target frame was detected, the D11 core itself configured the *deaf mode* on the radio hardware and immediately pushed the CSI data to the upper layers of the processing chain. However, due to the space constraints in this implementation, now the D11 core can only be configured to set the *deaf mode* on the radio, and we have to wait for a trigger that indicates that the data of a frame are available in the ARM CPU's memory before proceeding with the CSI extraction. While waiting for this trigger, the Wi-Fi chipset remains idle.

In order to estimate the impact of waiting for the complete transfer to the ARM memory on the system's performance, we set up an experiment to measure this latency. In this experiment, the ARM CPU does not extract any CSI data nor sends UDP datagrams to the user space, but just restarts the receiver. We use the SDR to transmit identical frames multiple times with a fixed time delay between successive frames. The fixed delay is reduced every time we repeat the experiment until we find that not all the transmitted frames have been received. This implies that the delay between two frames is too small and the receiving radio has not yet been restarted. The minimum delay for which all the frames are received is 50 μs, i.e., this is the transfer latency we are trying to determine. This is a considerable amount of time, given that some frames with a single data symbol (encoded with short guard interval at 80 MHz) can last as little as 43.6 μs. In Figure 5.4, we indicate this result with the label `NO-OP`.

Then we repeat the same experiment to measure the latency introduced by the process of pushing the CSI to the user space. We do not extract CSI data yet, but we just craft a single UDP datagram that it is sent to upper layers through the DMA interface as seen in Figure 5.1. We repeated the experiment for different sizes of the UDP datagram corresponding to the size it would have when receiving 20 MHz, 40 MHz or 80 MHz VHT frames respectively. In all cases, we measured a latency of 68 μs that is independent of the datagram size. This is reported in Figure 5.4 with the label `NO CSI`.

Finally, we run the experiment with the CSI extraction mechanism in place. We measure the latency for different number of extracted subcarriers, from a minimum of 64 (VHT frame at 20 MHz, 1x1) up to a maximum of 32768 (HE frame at 160 MHz, 4x4). We notice that when the CSI data fit into a single UDP datagram (256 subcarriers or less) the delay is equal to 95 μs plus a quantity that is proportional to the size of the CSI (625 ns per subcarrier). The additional delay of 27 μs with respect to the one measured in the `NO`

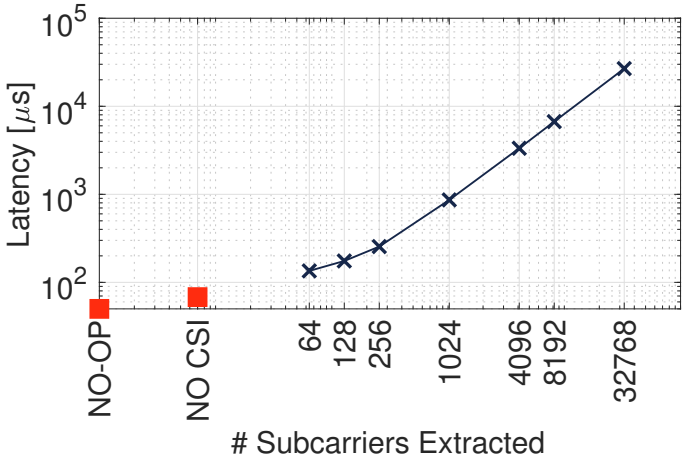


Figure 5.4: The latency introduced by the CSI processing chain depends on the number of extracted subcarriers.

Table 5.2: Performance comparison between the previous 802.11ac tool (Nexmon CSI) and the 802.11ax extractor (AX-CSI) in terms of CSI extracted per second.

encoding	VHT	VHT	VHT	VHT	VHT	HE
BW [MHz]	80	80	80	80	160	160
MIMO	1x1	4x1	1x4	4x4	4x4	4x4
# subcarriers	256	1024	1024	4096	8192	32768
NexmonCSI	8223	3034	2927	168	–	–
AX-CSI	3348	1101	1087	295	148	37

`CSI` case is due to the necessity of stopping/restarting the D11 core before/after reading the PHY tables. When more UDP datagrams are generated, we can see in Figure 5.4 that the latency increases almost linearly up to the maximum 32768 subcarriers.

We then run another experiment to have a more convenient comparison of the performance of AX-CSI with respect to Nexmon CSI, which is the previous tool developed for 802.11ac. In Table 5.2 we report the number of CSI extracted each second for different configurations using the VHT PHY. Since the old tool cannot extract CSI from the latest HE PHY (and from 160 MHz VHT frames neither, by default), for this configuration we only report results for the new tool. Overall, the new tool performs slightly worse in terms of CSI captured per second than the previous one for 802.11ac. For instance, the old tool can extract almost three times more CSI than the new one from VHT frames transmitted at 80 MHz with different configurations. However, the situation is different when more data have to be processed, like in the 80 MHz 4x4 case where AX-CSI extracts almost twice the data extracted by Nexmon CSI in the same time span. This is due to a hardware bottleneck in the previous implementation in which D11 core was directly pushing CSI data to the DMA interface. Although the new system appears to be slower than the old one for several configurations, we believe that it can extract CSI measurements at a sufficiently high rate to enable many interesting wireless applications.

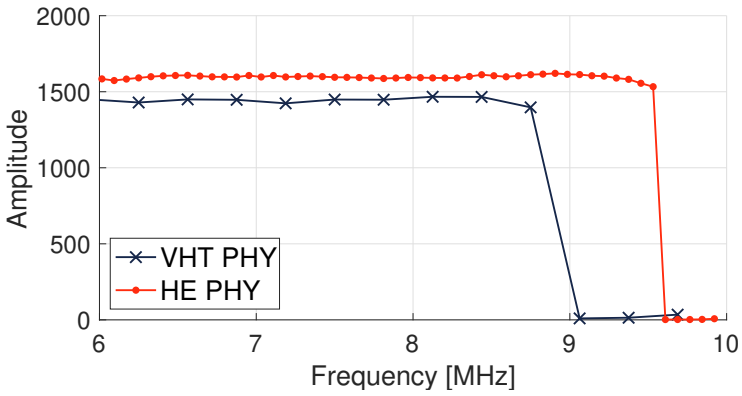
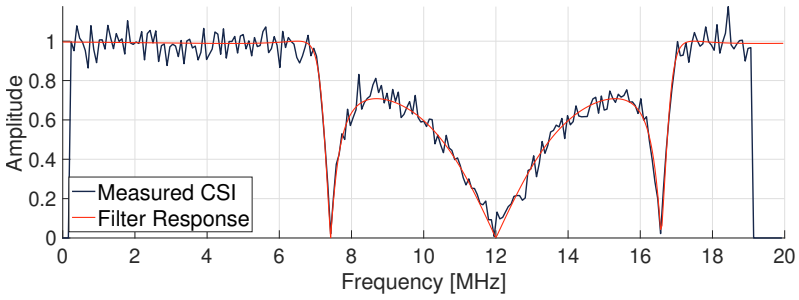


Figure 5.5: Detail of the CSI of 20 MHz frames. The spectral resolution achieved with HE PHY is four times larger than with VHT PHY. Amplitude is measured in arbitrary units as reported by the tool.

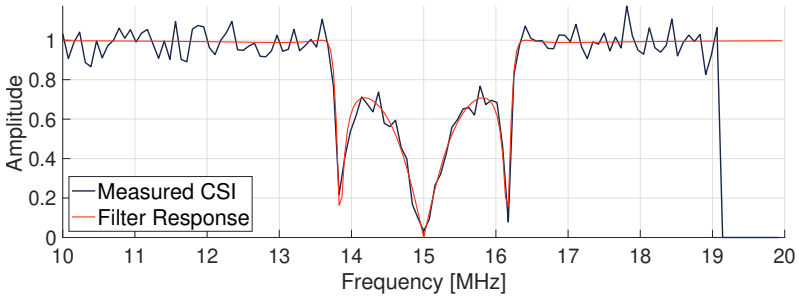
5.4. Results with HE PHY

In this section, we investigate some interesting features of the CSI extracted using our tool from 802.11ax frames based on the new HE PHY. We start by showing in Figure 5.5 a comparison between the CSI extracted from 20 MHz frames using the VHT PHY and HE PHY, respectively. Both frames are transmitted from the same SDR connected by cable to one receiving radio of the Asus AP. Here, we zoom into a small portion of the band to better appreciate the increased spectral resolution obtained with HE encoding (red dots) with respect to the VHT encoding (blue crosses). The former, in fact, adopts a four-times smaller subcarrier spacing, i.e., 78.125 kHz instead of the usual 312.5 kHz. Higher spectral resolution is key to accurately model the frequency response of channels with steep variations between adjacent subcarriers. We also notice that the HE spectrum has a much smaller guard band (see Figure 5.5, on the right) which actually increases the amount of "useful" bandwidth. With respect to sensing applications, this slightly wider spectrum available in HE PHY may provide better time-of-flight or angle-of-arrival estimates even in the 20 MHz band.

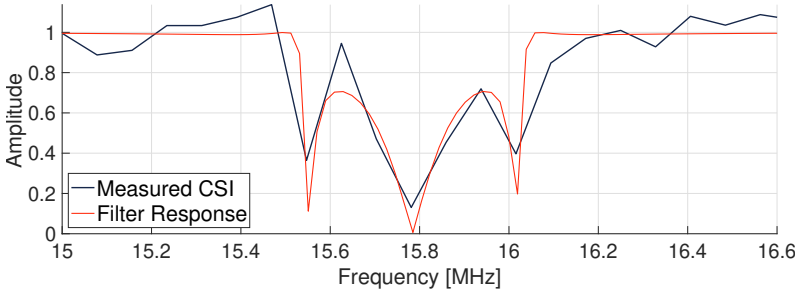
We then ran some experiments to showcase the advantages mentioned above introduced by HE encoding. To this end, (i) we generated with MATLAB a 40 MHz HE frame; (ii) we filtered the corresponding sequence of I/Q samples with a stop-band complex filter; and (iii) we transmitted the signal using cables from the SDR to the target Asus AP from which we collected the CSI as usual. We repeated this experiment decreasing the filter width each time as we show in Figures 5.6(a) to 5.6(d). For Figures 5.6(a) and 5.6(b) we chose quite large stop-bands, respectively 8 MHz and 2 MHz. The specific frequency response of the filter can be determined with high fidelity in the collected CSIs: this means that even high frequency components in the CSI profile can still be discriminated at the receiver. In the last two figures at the bottom, instead, we chose very thin stop-bands,



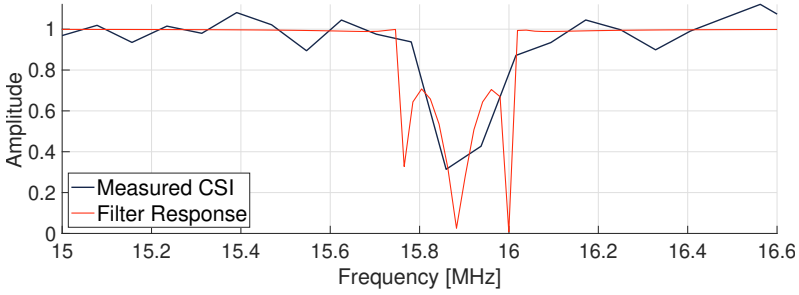
(a) Wide filter.



(b) Thin filter.



(c) Micro filter.



(d) Nano filter.

Figure 5.6: CSI extracted from 40 MHz HE frames when the transmitted frames are arbitrarily pre-distorted using filters with a particular frequency response. CSI amplitude is normalized to 1 outside the stop-band region.

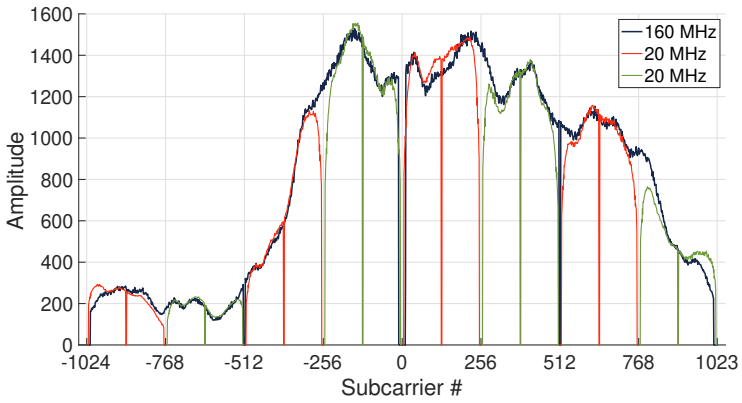


Figure 5.7: Spectrum of a 160 MHz frame overlaid with that of the eight constituting 20 MHz channels.

respectively 400 kHz and 200 kHz, which are respectively slightly larger and smaller than the subcarrier spacing in the case of VHT-encoded frames, to make the experiment more challenging. In the first case the shape of the CSI is still good enough to roughly represent the filter response. In the second case, instead, the width of the filter response becomes too small to be captured accurately. However, there are still two subcarriers that provide some hints about the central frequency of the filter.

In Figure 5.7 we show the CSI from a 160 MHz HE frame transmitted from the SDR over the air. We immediately notice that the wireless channel between transmitter and receiver appears to be extremely selective in frequency. The level of details that is available with such a large bandwidth makes the extraction tool extremely interesting for sensing experiments where minor variations in the environment can be observed only in some portions of the spectrum; with our tool, we can capture all of them at once. This is highlighted in the figure where we also overlay the CSIs of the 8 individual 20 MHz HE subchannels, properly normalized, each with its 256 subcarriers. Apart from the obviously different behavior at each channel boundary (caused by the guard bands), the 20 MHz subchannels closely match the eight times larger 160 MHz spectrum.

Finally, we report in Figure 5.8 a visualization of the full CSIs extracted for one 160 MHz HE frame with 4x4 MIMO, transmitted by another Asus AP. With our tool we can capture all the sixteen resulting CSI profiles for an astonishing number of 32768 subcarriers in total. To the best of our knowledge, this is the first CSI extraction tool that is capable of extracting such a huge amount of data from a single Wi-Fi frame.

5.5. IEEE 802.11ax localization performance

The new IEEE 802.11ax hardware features such as channel bandwidths up to 160MHz (before 80MHz for IEEE 802.11ac) and a much more dense spectrum, i.e., more available subcarriers, enable a more precise localization performance. In particular, doubling the

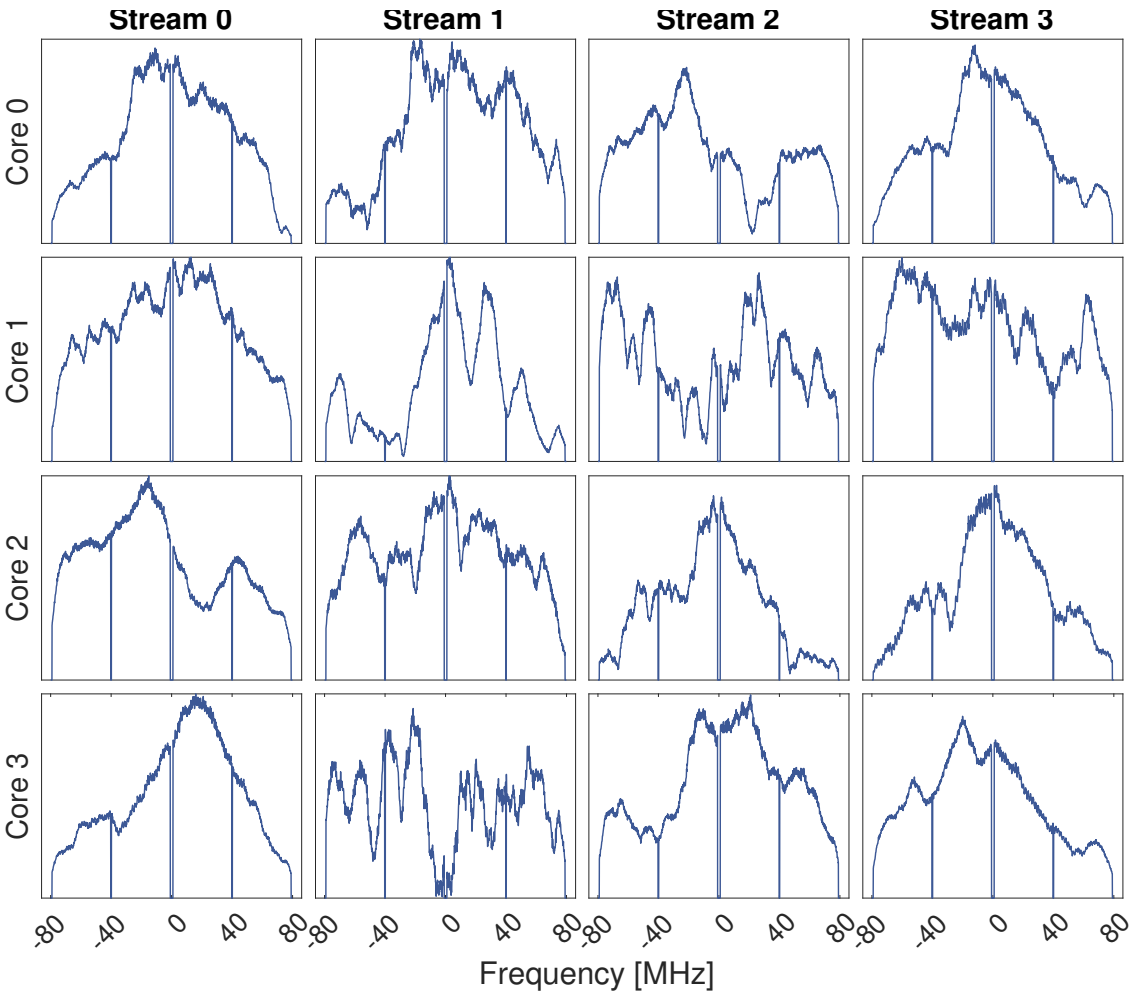


Figure 5.8: Amplitude of all the 16 CSI profiles extracted from a single 4x4 160 MHz HE frame. Each row contains the CSI collected at one RX core; columns represent the four spatial streams.

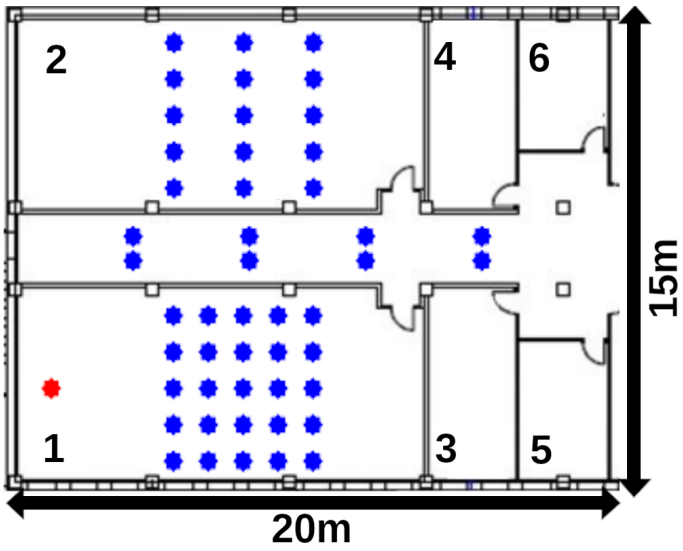
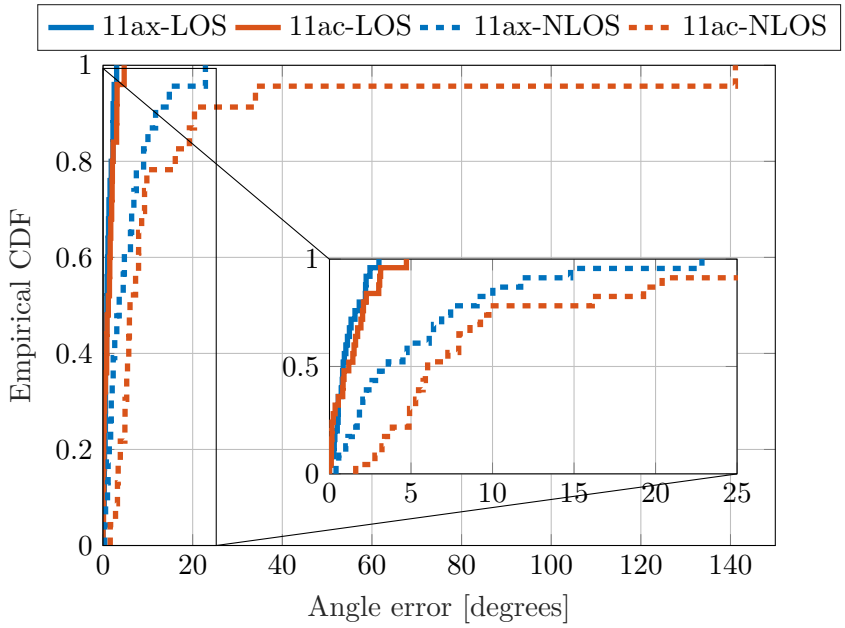


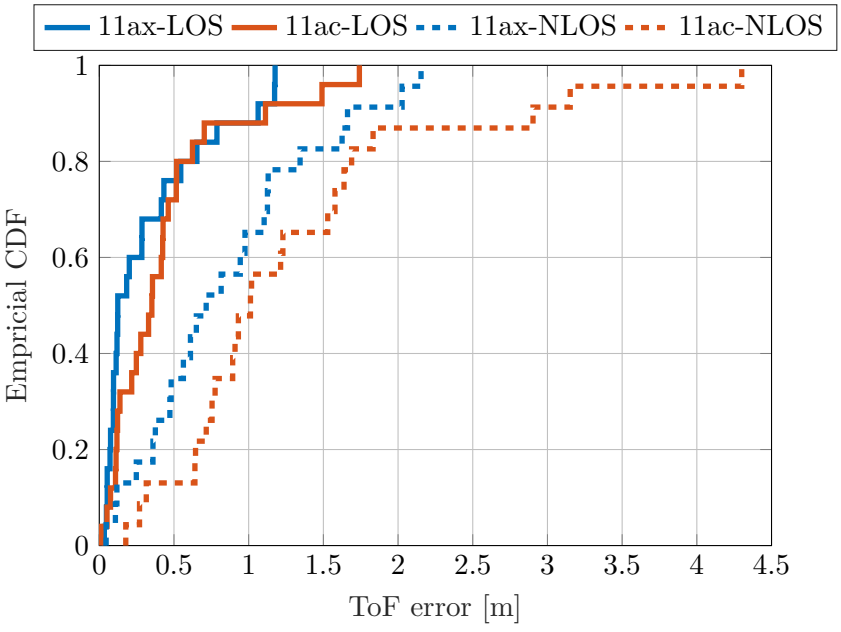
Figure 5.9: Scenario for the IEEE 802.11ax localization evaluation. The red dot represents the AP position and the blue dots the client positions.

bandwidth results in doubling the time resolution to discriminate two incident signals that arrive close in time. This is highly beneficial in multipath environments since the NLOS paths may arrive close in time to the direct path, and the estimation of the latter one is typically contaminated by the influence of NLOS paths. This contamination can be mitigated by increasing the time resolution. Thus, we expect to have an improvement by a factor of two in the Time of Flight (ToF) accuracy using an IEEE 802.11ax implementation compared to IEEE 802.11ac. For the Angle of Arrival (AoA) case, the number of antennas does not increase compared to 802.11ac, but there are more subcarriers which results in a more robust AoA estimation, as seen in Equation (4.2). Hence, we expect an improvement in the AoA accuracy too.

To evaluate the potential improvement in the position accuracy that 802.11ax enables compared to 802.11ac, we implement a UbiLocate (the proposed localization system in Chapter 4) version that works with the new standard. In particular, we enable the Fine Time Measurement (FTM) like protocol to work with the 802.11ax devices and we modify the UbiLocate AoA estimator to support the new hardware characteristics. We carry out preliminary measurements in the scenario depicted in Figure 5.9 which is one of the scenarios that we used to evaluate UbiLocate in Chapter 4. This scenario contains 50 points (blue points), 25 for LOS and another 25 for NLOS. The CSI measurements of those points are measured by the AP in room 1 (red point). This scenario contains fewer points and APs than in the last chapter since this evaluation is in an early stage and we show a preliminary localization evaluation. To provide a deep assessment, we first show the performances of AoA and ToF separately. We then show the overall localization performance.



(a) AoA performance



(b) ToF ranging performance

Figure 5.10: Empirical CDF for AoA and ToF errors.

AoA: The AoA results are shown in Figure 5.10(a). The performance of both, 11ax and 11ac, are extremely similar and accurate for the LOS case. They both get a median error of 1.1 degrees and maximum errors of 3 degrees and 4.7 degrees for 11ax and

11ac, respectively. Regarding NLOS, 11ax and 11ac achieve median errors of 3.6 degrees and 6.1 degrees and maximum errors of 22 degrees and 140 degrees, respectively. The maximum error of 11ac is caused by selecting an NLOS path as the direct path instead of selecting the obstructed direct path. This might happen in challenging settings where the NLOS paths are much stronger than the direct path and the estimator cannot resolve the latter one. But, exploiting the new hardware features of 11ax enables a superior multipath resolvability since the 11ax AoA estimator is able to extract the obstructed direct path in the hardest cases, which reduces outliers considerably. This is beneficial for applications that demand precise localization everywhere and are susceptible to larger errors.

ToF: The ToF results are shown in Figure 5.10(b). 11ax provides two times more accurate performance than 11ac because 11ax gets a median error of 17 cm while 11ac achieves 35 cm in LOS settings. This validates the ToF theoretical improvement by a factor of two that comes from doubling the bandwidth. Regarding NLOS, 11ax also provides a superior performance since it achieves a median error of 0.7 m while 11ac achieves 1 m. It is worth highlighting that 11ax minimizes the largest errors as the maximum error of 11ax is 2.1 m while the one of 11ac is 4.2 m. Unfortunately, the NLOS ToF improvement does not correspond to a factor of two and we report an improvement of 1.5. Theoretically, doubling the bandwidth results in doubling the resolution in discriminating two paths that arrive close in time. However, in NLOS cases, there might be more than two paths that arrive close which makes the resolvability of the direct path more challenging. In addition, the NLOS paths are likely to be stronger than the direct path which also complicates its discrimination. All of these in combination make that the improvement in NLOS does not correspond to the theoretical improvement.

General localization: The localization performance is reported in Figure 5.11. The LOS median errors are 37 cm and 21 cm for 11ac and 11ax, respectively. This corresponds to a localization improvement by a factor of 1.75. Regarding NLOS, 11ax provides a median improvement by a factor of 1.75 as well since 11ax gets a median error of 1.2m whereas 11ac gets 2.1 m. This indicates that doubling the bandwidth does not completely result in two times more accurate overall localization performance but it is close to the theoretical factor. We would like to emphasize that a LOS median error of 21 cm is an impressive achievement. This accuracy is close to the precision that a system using massive channel bandwidths, in order of gigahertz, and much larger antenna arrays can get. These hardware configurations are utilized in Milimiter-wave (mmWave) communications and practical mmWave localization works [101–103] reported median errors in terms of 10-20 cm. Finally, the reported NLOS evaluation of the previous chapter, in particular in Figure 4.9(b), shows that 11ac achieves an NLOS median error around 1 m while this evaluation shows 2.1 m. This degradation in the performance is caused by having only one AP for positioning in this preliminary evaluation, while in the

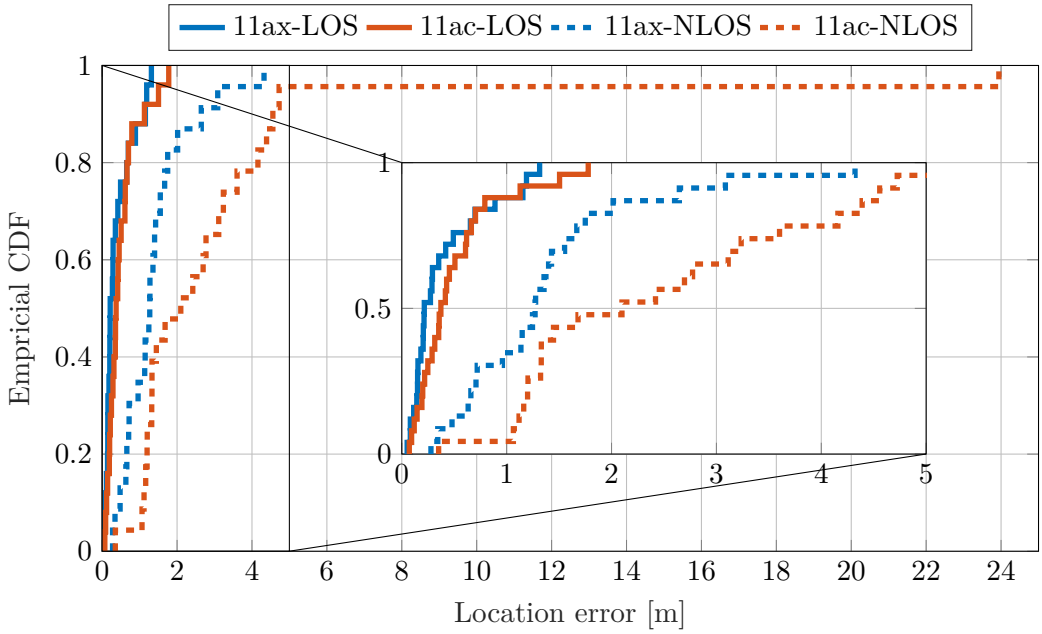


Figure 5.11: Empirical CDF for localization errors.

other evaluation the number of available APs was four. Hence, we expect to get two times more precise 11ax NLOS performance by adding up to four APs.

5.6. Related work

The CSI is a key element in Wi-Fi communications. However, most off-the-shelf devices just use the CSI internally in the PHY section of the Wi-Fi system, and only a few of them can report them to the user by default.

This work stems from Nexmon csi, a popular csi extraction platform for 802.11ac Broadcom chipsets [100]. As of today, this is one of the most comprehensive tools so far for CSI analysis, enabling the CSI extraction from 802.11ac frames (supporting both VHT PHY and 4x4 MIMO) on a wide range of Broadcom chipsets. This platform was extended into a Wi-Fi 802.11ac-based localization system [2] that deals with NLOS settings. The authors added more features such as obtaining the timestamps of received Wi-Fi frames and a methodology to tackle hardware imperfections that make the device unreliable for localization purposes.

The literature suggests that the most popular chipset used for CSI-based sensing research is the Intel Wi-Fi Link 5300. On this chipset, the Linux 802.11n csi Tool [104], based on custom firmware and open-source Linux drivers, is used to access the CSI. However, the spectral resolution of the CSI is limited to 30 values corresponding to

"subcarrier groups" rather than one value for each OFDM subcarrier. A complex value with signed 8-bit real and imaginary parts is reported for each group, which means that each group roughly corresponds to two subcarriers for 20 MHz channels and four subcarriers for 40 MHz channels. Recently, a newer tool was released for an 802.11ac Intel platform, specifically for the Intel 9260 card [105]. However, this card has only two antennas; thus, the best MIMO setting is 2x2, limiting research based on the angle of arrival and departure. By contrast, our csi extractor tool enables 4x4 MIMO, which allows for discriminating more paths in the space domain.

CSI data of individual subcarriers can be extracted for Qualcomm Atheros chipsets using the Atheros csi Tool [106]. This platform is entirely implemented in software and builds on top of the open-source Linux kernel driver `ath9k`, supporting many different chipsets like the AR9580, AR9590, AR9344, and QCA9558. Unlike the Intel Tool, the Atheros one offers finer quantization of the data extracted for each subcarrier, as both the real and the imaginary part take value in the range of integers $[-512, 512]$. However, also this platform is limited to 802.11n. To the best of our knowledge, newer Qualcomm Atheros QCA988x chipsets, supporting 802.11ac and powered by the `ath10k` wireless driver, still cannot be used for CSI collection as no open-source nor proprietary CSI extraction platforms were released.

Quantenna Communications, a chipset manufacturer for high-end APs, offers another solution for CSI analysis. Some recent papers [107, 108] revealed that some Quantenna chipsets could report the CSI for 802.11ac frames. However, there is little knowledge about the performance of this platform since it is provided only to customers and system developers.

Lastly, it is worth mentioning alternative solutions based on SDRs. Well-known SDRs like the Wireless Open-Access Research Platform (WARP) [109] and the Universal Software Radio Peripheral (USRP) [110] could be used in principle for collecting CSI. Such platforms combine high flexibility with large bandwidths, as some radio front-ends can achieve 160 MHz and even more. Implementing a real-time Wi-Fi stack in software is, however, not straightforward. Recently, a project named *openwifi* developed on Field-Programmable Gate Array (FPGA) an open-source implementation of the Wi-Fi stack [111] that allows CSI analysis and manipulation. However, *openwifi* supports only 802.11n at 20 MHz with no multiplexing (there is actually an ongoing effort to add support for 2x2 MIMO). Moreover, the major drawback of using SDR platforms is that they are usually orders of magnitude more expensive than consumer Wi-Fi devices; hence, their usage is often confined to specialized research labs.

5.7. Conclusions

The capability to collect and analyze CSI data from real wireless network deployments is currently driving many research activities. In particular, the interest of many research groups is focused on opportunistic Wi-Fi sensing where it is important to collect CSI data at a very high rate, that cover as large a portion of the spectrum and as many subcarriers as possible. In this chapter, we presented the first tool to collect the most accurate CSI ever, thanks to its compatibility with the latest generation Wi-Fi standard IEEE 802.11ax. Our tool supports CSI collection from transmissions with up to four spatial streams and up to 160 MHz of spectral bandwidth per stream, extracting up to 32768 subcarriers per incoming frame. To further validate the usefulness of the platform, we carry out a preliminary evaluation to measure the localization accuracy that 802.11ax can enable. Our results show that IEEE 802.11ax provides superior performance compared to its predecessor, IEEE 802.11ac. In particular, the new standard improves the localization accuracy by a factor of 1.75 in LOS and NLOS settings. We believe that our system is an important contribution to the research community and has the potential to become a widely adopted tool.

6 Respiration Rate Estimation using Commodity Wi-Fi

6.1. Introduction

As seen in previous chapters, localization needs to extract location information of the user from the direct path to provide new services like indoor navigation, tracking and many more applications. Sensing applications, instead, require extracting the Non-Line-Of-Sight (NLOS) paths as they convey location information of persons and/or objects that are present in the environment. This is beneficial for security purposes as intruders can be detected [112,113]. In addition, sensing also aims at understanding small variations of the path parameters of NLOS paths which result in detecting small displacements of reflectors. Hence, a system can detect abnormal behavior [114] of persons. Moreover, it is also helpful in-home care applications since sensing allows to detect falling [115,116], recognize human activities [117–119] and monitor vital signs [120–122]. Specifically, respiration rate estimation using wireless signals is appealing as it does not require any intrusive equipment whereas specialized respiration equipment needs physical contact with the patient that may affect negatively the estimation. This is particularly beneficial for patients that suffer from sleep apnea, where patients stop breathing regularly while they are sleeping. We thus focus on respiration estimation using wireless signals.

Respiration can be represented as a periodic signal. In every cycle, the chest enlarges in the inhalation phase followed by a pause and then the chest compresses in the exhalation phase followed by another pause. Hence, the path of the signal that bounces off the human chest changes its length according to respiration. In particular, the displacements make the length of the path a bit larger or smaller. These small variations of the length of the path result in phase shifts in the complex attenuation which lead to obtain the respiration signal. State-of-the-art schemes exploit the interaction between static paths, i.e., paths whose complex attenuation values do not change over time, and the path of the chest, which is a dynamic path. This interaction produces changes in the amplitude of the Channel State Information (CSI) measurements so that the respiration signal can be recovered. However, the path of the chest is considerably weaker compared to

the static paths. Therefore, the changes in the amplitude might be almost negligible and the extraction of the respiration signal may fail. We propose, instead, to resolve all the multipath components so that we can remove the contamination of static paths and extract the path of the chest relatively free of interference. Hence, we obtain the respiration signal as it is contained in the phase of the complex attenuation of the path that comes from the chest. We implement this approach in a Commercial Off-The-Shelf (COTS) Wi-Fi testbed. Our preliminary results show that accurate respiration rate estimation is possible with a low rate of error by decomposing the multipath components.

This chapter is organized as follows. We explain in Section 6.2 how a sensing system can exploit NLOS paths to estimate the position of reflectors. Section 6.3 demonstrates how the small variations in the length of the path of the chest caused by the respiration can be used to extract the respiration signal. In the same section, we describe the proposed respiration signal extractor and rate estimator. We provide the details of the preliminary measurements as well as the results of the respiration evaluation in Section 6.4. Finally, Section 6.5 concludes this chapter by summarizing our findings.

6.2. Passive localization

The first step forward from active localization to sensing is to passively localize objects such as persons in the environment. During the previous chapters, we assume active localization, i.e., the client sends a radio-frequency signal so that a system extracts location information from the direct path of the signal to estimate the client position. In contrast, passive localization does not require the person to carry a wireless device. The person can be passively localized by estimating position information from the reflection that the person creates when the signal bounces off the human body. Active localization requires extracting the direct path which is the path that goes from the client to the Access Point (AP), passive localization, instead, requires extracting the reflected paths.

To this end, a transmitter and a receiver establish a link and the transmitter sends a signal which propagates through a multipath channel. The received CSI contains not only the location information from the direct path but also from paths that reflected off objects or persons in the environment. Figure 6.1 shows an example of passive localization. In this example, there are three paths. The first one is the direct path, p1, and two NLOS paths which are p2 and p3. As seen in the figure, all the paths have the following parameters. We denote Angle of Arrival (AoA) as θ_{rx}, Angle of Departure (AoD) as θ_{tx} and the path delay as τ. We can passively localize the person by triangulation. In particular, a system needs to compute the position of the vertex that corresponds to the point where the signal bounces off the human body. To do so, the system can solve the triangle and get the person position by applying simple algebra theorems knowing the position of the transmitter and the receiver, the AoA or the AoD and the τ of the direct path (p1) as

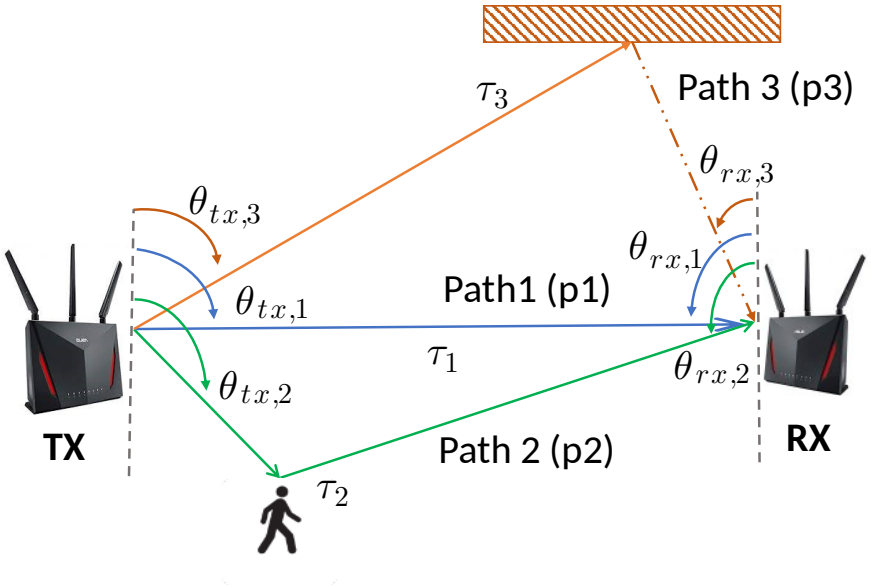


Figure 6.1: Example of passive localization.

well as p2.

6.3. Breathing detection

Sensing goes beyond localization and it aims at understanding small variations in reflected paths that can be used for respiration rate estimation. Breathing can be represented as a periodic signal. In every cycle, the chest enlarges in the inhalation phase followed by a pause and then the chest compresses in the exhalation phase followed by another pause. These displacements change the length of the path of the signal that bounces off the human chest, i.e., the path becomes a bit larger or smaller. These small variations of the length of the path result in phase shifts in its complex attenuation. Figure 6.2 illustrates this fact. In this example, the transmitter sends a Wi-Fi signal which propagates through a multipath channel. One of the paths reflects off the human chest and the length of this path gets shorter for the inhalation and longer for the exhalation. Since breathing is a periodic signal, we can model the path difference as well denoted $d(t)$. Hence, we express the phase shift introduced by $d(t)$ to the complex attenuation of the path that comes from the chest as follows:

$$e^{-2j\pi\frac{d(t)}{\lambda}}, \tag{6.1}$$

with λ is the wavelength of the radio-frequency signal.

To the best of our knowledge, the very first attempt in respiration rate estimation

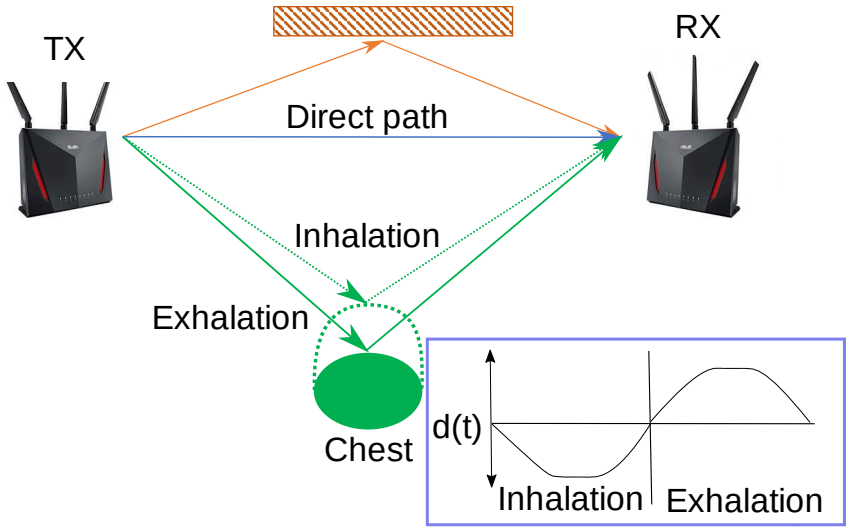


Figure 6.2: Example of how the path length is affected by the displacement of the chest

was the system [118] which is based on the received signal strength. However, the signal strength is usually affected by environmental factors that degrade the performance. Therefore, this system requires the patient to hold the device close to the chest to achieve a reliable respiration rate estimation. More accurate approaches have been developed using CSI data. In particular, [120–122] have successfully exploited differences of the amplitude of the CSI measurements over time to extract the respiration signal. In addition, the authors in [123, 124] have introduced the Fresnel zone model as a theoretical model to explain how static paths and the path that comes from the chest interact with each other. This interaction results in changes in the amplitude according to the path difference, $d(t)$. Hence, these amplitude changes can be used to recover the respiration signal. This interaction is expressed as:

$$|H|^2 = |H_s|^2 + |H_d|^2 + 2|H_s||H_d|cos(\alpha) \ , \tag{6.2}$$

where $|\cdot|$ represents the absolute value, H is the overall received channel, H_s is the sum of channels from statics paths, H_d is the channel introduced by path of the chest and α is the phase difference between H_s and H_d.

The works mentioned above recover the respiration signal using the difference in the amplitude over time which is caused by the interaction between static paths and the path that comes from the chest. However, this interaction might not provide a respiration signal clean enough since the power of the dynamic path could be considerably weaker and the respiration signal would not be recovered. For instance, consider the example previously presented in Figure 6.2 which illustrates this fact. All the paths that arrive at the receiver interact with each other, but only the path that reflects off the chest

provides the respiration signal. Nevertheless, this path is likely to be masked by the strongest paths, in particular by the direct path. To avoid it, MultiSense [125] gets a CSI reference measurement that contains the information from static paths to further remove them. However, this limits its usability because this approach requires getting a CSI measurement per room in advance. Another approach tries to isolate the path of the chest by getting a larger bandwidth synthesizing several Wi-Fi channels [126], which is not supported by COTS devices out of the box.

State-of-the-art schemes have dealt with the interference of static paths, but they might not be suitable in practical scenarios. Hence, we propose to remove the interference by resolving all the paths so that the path of the chest is accurately extracted. Therefore, the respiration signal can be recovered since it is contained in the phase of the complex attenuation of the path of the chest. To do so, we apply the algorithm proposed in Section 4.2.1 to resolve all the multipath components. This algorithm decomposes the channel in several paths by applying a minimization by a Nelder-Mead search. Hence, the paths are estimated relatively free of interference. Since respiration is a temporal signal, we sample the channel and at every sample, we decompose the channel to get all the paths. We then evaluate which path conveys the breathing signal and we estimate the respiration rate.

6.3.1. Path parameter estimation

Using the wireless model in Section 2.2.1, the observed received channel in the frequency domain is expressed as a function of the attenuation, the path delay, the AoA and the AoD as follows:

$$\hat{\mathbf{H}}[k] = \sum_{p=0}^{P-1} \boldsymbol{\phi}(\theta_{rx,p})\gamma_p \boldsymbol{\phi}^{\mathrm{H}}(\theta_{tx,p})\psi(\tau_p)[k] + \mathbf{w}[k] = \sum_{p=0}^{P-1} \mathbf{H}_p[k] + \mathbf{w}[k] \ , \tag{6.3}$$

with k is the k-th subcarrier, P is the number of paths, $\boldsymbol{\phi}(\theta_{rx,p})$ is the vector of phase shifts at the receiver Uniform Linear Array (ULA) introduced by the AoA, $\psi(\tau_p)[k]$ is the path delay introduced by the propagation of the signal, γ_p is the complex attenuation, $\boldsymbol{\phi}(\theta_{tx,p})$ is the vector of phase shifts at the transmitter ULA introduced by the AoD and $\mathbf{w}[k]$ is an L-dimensional white Gaussian noise in the frequency domain, where L means the number of transmitter antennas. For convenience, we denote $\hat{\mathbf{H}}[k]$ as $\hat{\mathbf{H}}$, $\mathbf{H}_p[k]$ as $\mathbf{H}_p$, and $\mathbf{w}[k]$ as $\mathbf{w}$ for the subsequent equations of this chapter.

Since respiration is modeled as a periodic signal, sampling the channel in time is needed to extract the respiration signal. The observed channel at a time t is expressed as follows:

$$\hat{\mathbf{H}}(t) = \sum_{p=0}^{P-1} \mathbf{H}_p e^{-2j\pi \frac{d_p(t)}{\lambda}} + \mathbf{w} \ , \tag{6.4}$$

where $d_p(t)$ is the displacement of the p-th path. For the received paths that are static, we assume that $d_p(t)$ is always 0.

For respiration rate estimation it is crucial to accurately extract the path of the chest by removing the interference from static paths. To this end, we apply the path parameter estimator proposed in Section 4.2.1 to resolve all the path parameters at every channel sample. In previous chapters, we use this estimator to obtain the direct path even if it is obstructed, as it also estimates weaker paths. Therefore, this estimator is capable of extracting the path of the chest as well. Estimating the path parameters at every channel sample results in obtaining the estimated path parameters as temporal signals:

$$(\hat{\gamma}_p(t), \hat{\tau}_p(t), \hat{\theta}_{rx,p}(t), \hat{\theta}_{tx,p}(t)), \tag{6.5}$$

In practice, the path estimator might not sort all paths in the same way for all the time samples. The interference cancellation of the strongest paths might introduce some energy leakage to the estimation of the weakest paths, hence sorting the paths by the power could make that they are not arranged in the same way between different time samples. To deal with that, we sort the paths at every sample in terms of the path delay, i.e., the paths are arranged according to the time of arrival. This is a valid assumption in static environments where the path delays remain the same over time.

6.3.2. Breathing signal extraction and rate estimation

The breathing signal is contained in the phase of the complex attenuation of the path that reflects off the chest. Therefore, we need to identify this path among all the other ones. The phase of the complex attenuation of a static path is only influenced by white Gaussian noise while the path of the chest changes its phase according to the respiration. Therefore, we can analyze the frequency spectrum of the phase of the complex attenuation of the paths to detect which one conveys the respiration signal. In particular, the frequency spectrum of a white Gaussian noise temporal signal is assumed to be flat while the respiration signal has a clear frequency component that is stronger than the other components. To do so, we compute a ratio between the absolute value of the maximum peak of the frequency spectrum and the average of the absolute values of the frequency components. If the signal is noisy, the ratio is close to 1 whereas if the signal contains the respiration signal, the ratio will be larger. We denote the phase of the estimated complex attenuation of the p-th path as $\sigma_p(t)$. We apply a discrete Fourier transform to $\sigma_p(t)$ to estimate its frequency spectrum as follows:

$$\rho_p[w] = DFT\{\sigma_p(t)\} \ , \tag{6.6}$$

where $DFT\{\cdot\}$ denotes the discrete Fourier transform.

The ratio is given by:

$$r_p = \frac{max(|\rho_p[w]|)}{avg(|\rho_p[w]|)} \ , \tag{6.7}$$

where $max(\cdot)$ and $avg(\cdot)$ represent the maximum value and the average value of a signal, respectively.

The path that comes from the chest is the path that maximizes the ratio:

$$p_{ch} = \arg\max_p r_p \tag{6.8}$$

Then, we denote the respiration signal as:

$$\sigma_{resp}(t) = \sigma_{p_{ch}}(t) \tag{6.9}$$

However, the respiration signal contains noise since the multipath decomposition algorithm might not completely remove the interference from other paths and some noise residual might persist. The noise can be mitigated by filtering the signal. In particular, we apply a moving average to remove high-frequency components so that we get a smooth version of the respiration signal. We then apply an over-sampled discrete Fourier transform to the signal to obtain its frequency spectrum. We apply an over-sampled DFT to get more resolution in the frequency domain so that the respiration rate is estimated with higher precision. We denote the over-sampled frequency signal as $\rho_{resp}[w]$. Finally, the estimated respiration rate is the frequency component that maximizes the absolute value of $\rho_{resp}[w]$ as follows:

$$\hat{b}' = \arg\max_w |\rho_{resp}[w]|, \tag{6.10}$$

Since the respiration rate is measured in Hz, we convert it to breaths per minute (bpm) by multiplying it by 60. The estimated rate in bpm is:

$$\hat{b} = 60\hat{b}' \tag{6.11}$$

6.3.3. CSI cleaning

The hardware adds an artifact called Carrier Frequency Offset (CFO) that makes the phase of CSI measurements to not be coherent between consecutive samples. The CFO is caused by a mismatch between the frequencies of the sampling clocks of the transmitter and the receiver. In particular, the CFO introduces a phase distortion for every received CSI measurement. We express the channel that is affected by the CFO as:

$$\mathbf{H}_{CFO}(t) = \mathbf{H}(t)e^{-2j\pi\xi_{CFO}}, \tag{6.12}$$

where ξ_{CFO} is the phase distortion of the CFO.

CFO affects to the overall phase of each CSI measurement, i.e., all the paths are affected equally. Since a static path does not change its phase over time, its phase is only affected by the CFO. Thus, we can remove the CFO distortion by estimating the phase of the direct path and subtracting it to the channel. Hence, this results in the following equation:

$$\mathbf{H}_{clean}(t) = \mathbf{H}_{CFO}(t)/e^{-2j\pi\sigma_{dp}(t)}, \tag{6.13}$$

with $\sigma_{dp}(t)$ is the phase of the estimated complex attenuation of the direct path.

This removal does not affect the extraction of the breathing signal since the initial phase of the complex attenuation of the path that comes from the chest does not have any impact on the estimation.

6.4. Respiration evaluation

In this section, we conduct preliminary measurements to assess the proposed respiration rate estimator. To this end, we deploy one transmitter and one receiver in a room of size 10x6 m. The scenario is depicted in Figure 6.3. As Wi-Fi transceivers, we use the Asus AC2900 RT-AC86U COTS device as in Chapter 4. We send Wi-Fi packets periodically using a packet rate of 100Hz. At every packet, we extract the full CSI matrix and store it for a later respiration estimation.

We ask a participant to stand in the room. In particular, the participant was placed in the middle between the transmitter and the receiver with a separation of 1.2 m from them. The position of the participant is represented as the yellow dot in Figure 6.3. We also ask the participant to breathe for one minute and to count the number of breaths so that we get the ground truth. We repeat this process two times, one time for a regular breathing rate and another for a slower rate. The participant reported 13 and 10 bpm for the cases of normal and slow rates, respectively.

We start this evaluation by analyzing the raw CSI measurements. Figure 6.4(a) shows the evolution of the phase of the raw CSI. It can be clearly seen that the signal is quite noisy and the respiration signal cannot be recognized. Figure 6.4(b) shows the evolution of the amplitude of the raw CSI. The amplitude provides a coarse respiration signal but it is quite noisy at some points in time and the respiration shape is missing. We would like to highlight that the amplitude of the CSI data is the input of the state-of-the-art schemes for respiration rate estimation. However, our analysis shows that it cannot provide a clear respiration signal, while our approach of decomposing the channel to extract the path from the chest leads to a much cleaner respiration signal as shown in Figure 6.4(c).

The results of the respiration evaluation are shown in Figure 6.5 and Figure 6.6 for the normal and slow respiration rates, respectively. In particular, Figure 6.5(a) and Figure 6.5(b) show the respiration signal before and after the filtering for the normal

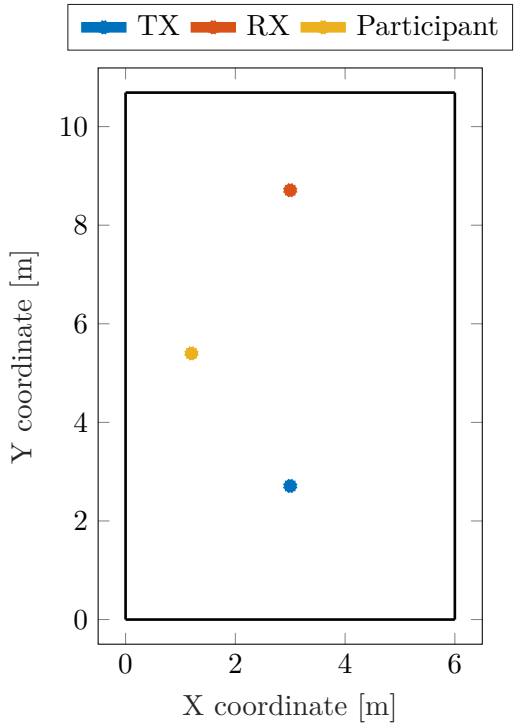


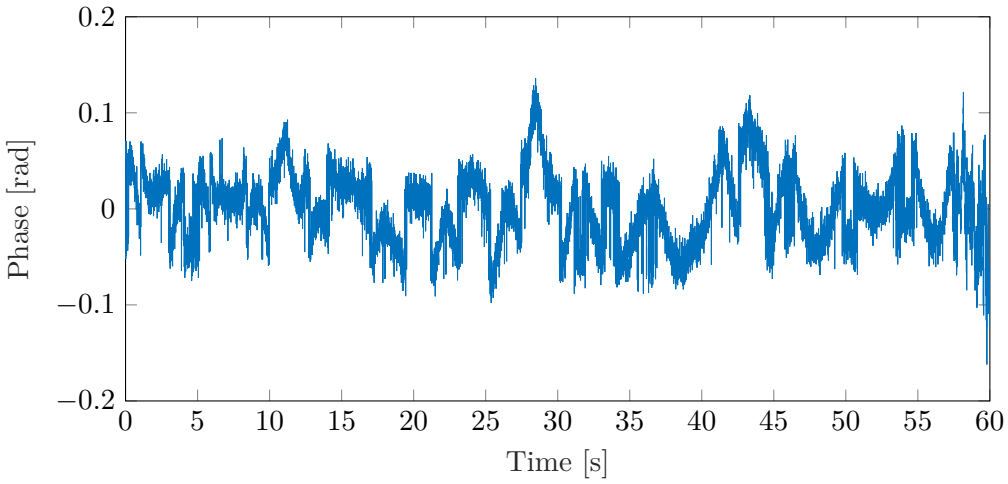
Figure 6.3: Floor plan of the scenario for the respiration evaluation.

rate. The frequency spectrum of the respiration signal can be seen in Figure 6.5(c). The peak in the frequency spectrum corresponds to the estimated respiration rate. In this case, the estimated rate is 13.0693 bpm. Regarding the slow rate, Figure 6.6(a), Figure 6.6(b) and Figure 6.6(c) show the raw respiration signal, the clean respiration signal and the frequency spectrum of the latter one, respectively. In this case, the estimated rate is 10.0493 bpm. According to the ground truth, the normal and the slow rates are 13 and 10 bpm, respectively. Therefore, the errors are below 0.1bpm which validates the precision of the proposed scheme.

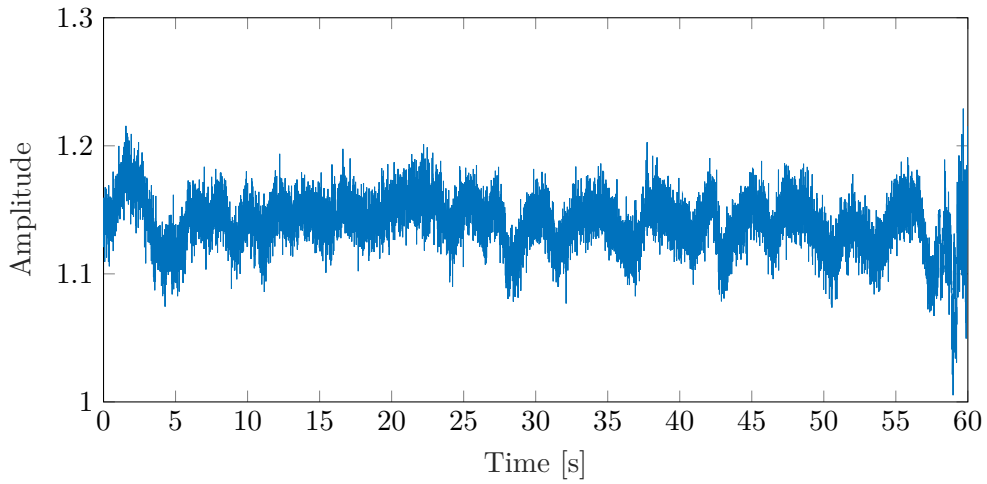
6.5. Conclusions

Going beyond localization enables applications that are useful for example, for network management, security and health care. In particular, respiration rate estimation using wireless signals is appealing since it does not require any specialized equipment. During the last years, respiration rate estimation has been mainly carried out using the overall received channel without any interference cancellation from static paths. We propose, instead, to recover the respiration signal and to estimate the respiration rate by resolving all the multipath components so that the path that comes from the chest is accurately extracted. We implement the proposed scheme using COTS Wi-Fi devices. Our preliminary results show that a precise respiration rate estimation is possible by

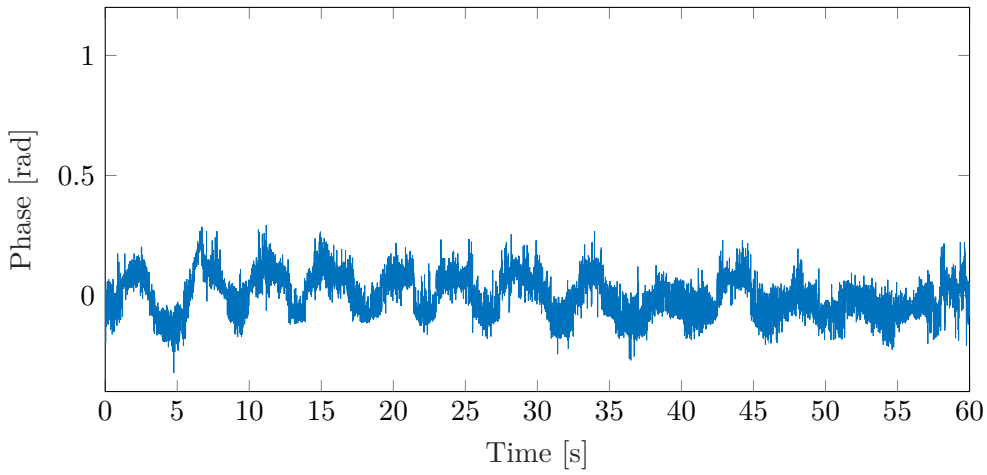
decomposing the channel. For future work, we will analyze how different factors like distances and the rotation of the participant affect the estimation. We also plan to explore the multi-person case.



(a) Raw CSI phase.

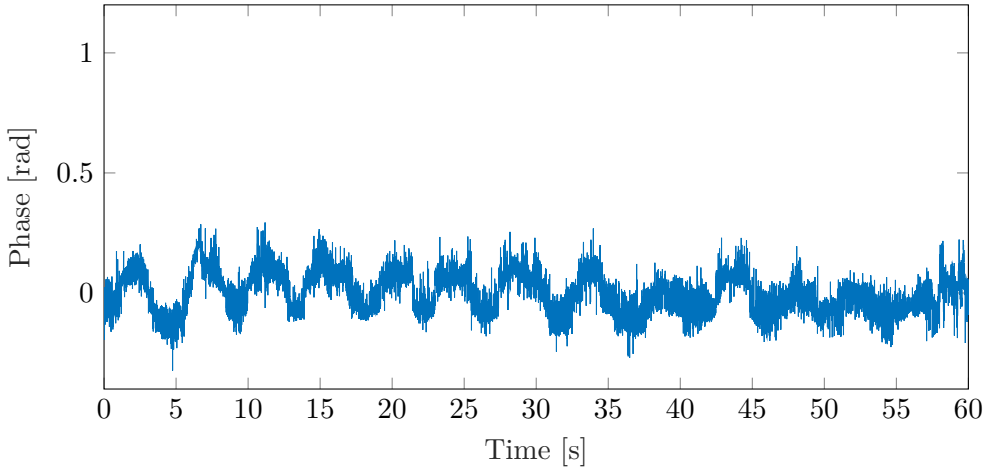


(b) Raw CSI amplitude.

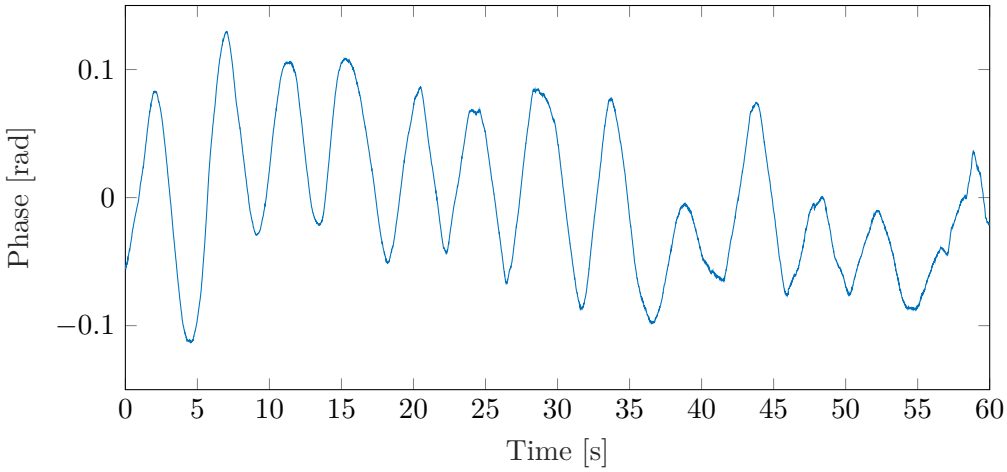


(c) Raw respiration signal

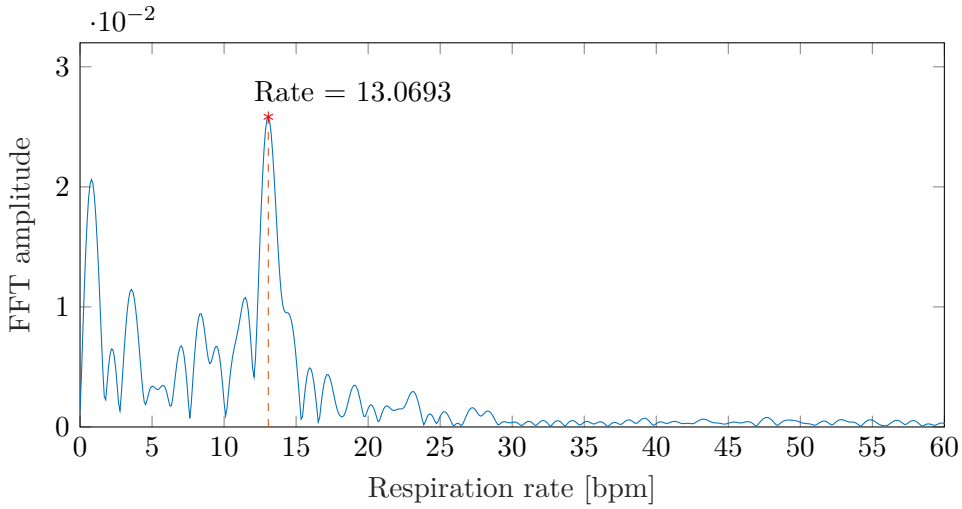
Figure 6.4: Analysis of the raw respiration signals. (a) is the raw CSI phase, (b) is the raw CSI amplitude and (c) is the raw respiration of the proposed scheme.



(a) Raw respiration signal

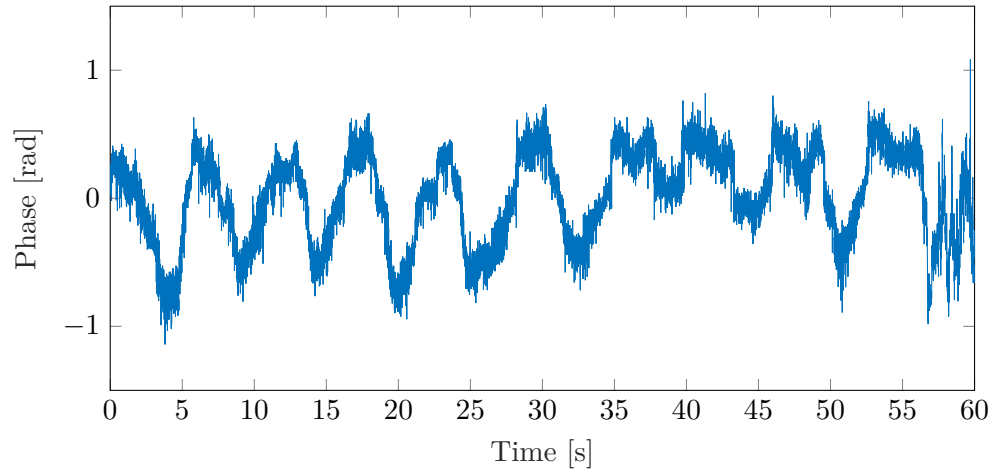


(b) Clean respiration signal (after filtering)

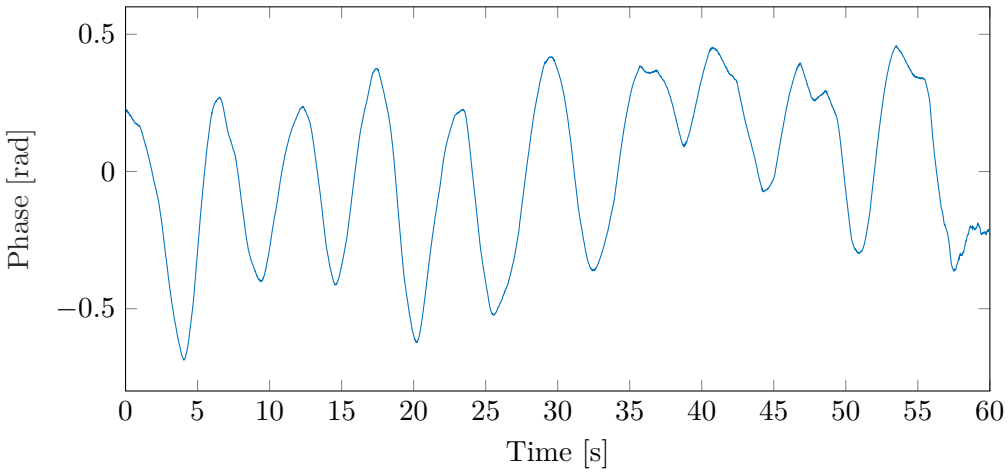


(c) Frequency spectrum of the clean respiration signal

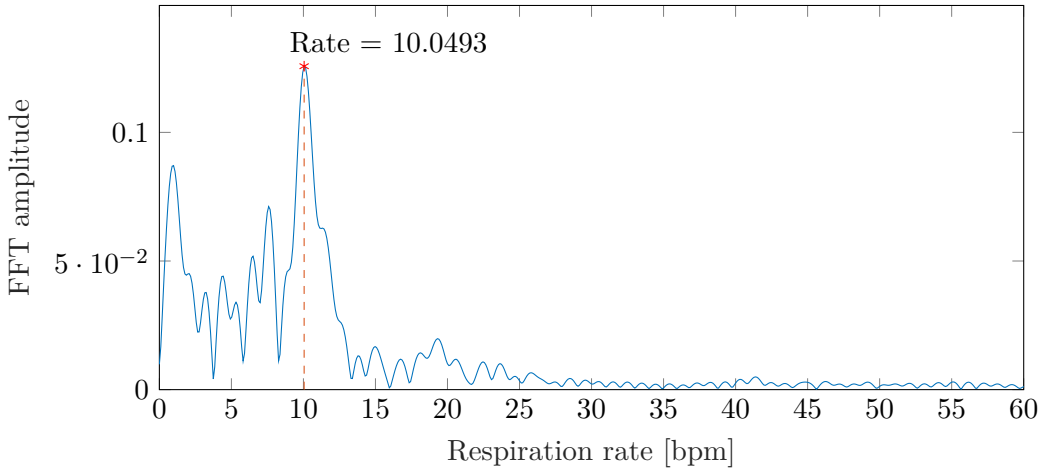
Figure 6.5: Respiration analysis for the normal respiration rate.



(a) Raw respiration signal



(b) Clean respiration signal (after filtering)



(c) Frequency spectrum of the clean respiration signal

Figure 6.6: Respiration analysis for the slow respiration rate.

7 Conclusions

Location-based services are appealing and they demand accurate and ubiquitous positioning. Researchers, cellular networks operators and chipset vendors are making a lot of efforts to standardize pervasive and precise localization by the latest wireless protocols. While recent works conclude that a sub-meter level of localization accuracy is possible, their performances drastically degrade in Non-Line-Of-Sight (NLOS) conditions due to the poor estimation of the obstructed direct path. Therefore, this thesis aims at improving the resolvability of multipath components to not only enable robust positioning by accurately extracting the obstructed direct path but also improve sensing application performances using the superior estimation of NLOS paths.

5G aims at providing outstanding localization accuracy to meet the new location-based services. However, 5G and Long Term Evolution (LTE) will coexist until 5G provides ubiquitous coverage. This implies that several location-based services need to be carried out by LTE. To assess which location-based services can be carried out by current cellular networks, we implement an LTE localization system in Chapter 3. Our localization evaluation shows that LTE provides median error of 2 m for Line-Of-Sight (LOS) settings while the performance degrades drastically in NLOS getting 4.76 m of median error. This chapter concludes that 2 m of median error complies with the positioning requirements of several location-based services but LTE would fail in fulfilling the majority of them in NLOS settings.

The previous LTE performance is limited by the maximum available bandwidth of 20 MHz as the time resolution might not be sufficient to extract the direct path when is obstructed. In Chapter 4, we then delve into robust indoor localization to cope well with NLOS issues by increasing the resolvability of the multipath components. We develop UbiLocate, an IEEE 802.11ac-based Wi-Fi location system that copes well with realistic AP deployments and works ubiquitously, i.e., without excessive degradation under NLOS. UbiLocate achieves one meter NLOS median error through (i) a refined Angle of Arrival (AoA) extractor which accurately decomposes all the multipath components, (ii) a fine-grained Time of Flight (ToF) ranging system that achieves sub-meter accuracy even in

tough NLOS conditions and (iii) using the improved IEEE 802.11ac hardware features of 80MHz of channel bandwidth and 4x4 Multiple-Input Multiple-Output (MIMO). Our experimental evaluation in a number of common scenarios shows an overall improvement of the localization performance by a factor of 2-3 compared to state-of-the-art systems, both under LOS and NLOS conditions.

Technology is always evolving and wireless protocols provide higher data rate by increasing channel bandwidths and the number of antennas. These improved hardware features increase the performance of localization systems and sensing applications. In particular, the latest Wi-Fi protocol, IEEE 802.11ax, enables a channel bandwidth of 160 MHz and a four times denser spectrum than its predecessor, IEEE 802.11ac which only supports 80MHz of channel bandwidth. In Chapter 5, we present the first tool to collect the most accurate Channel State Information (CSI) ever, thanks to its compatibility with the latest generation Wi-Fi standard IEEE 802.11ax. Our tool supports CSI collection from transmissions with up to four spatial streams and up to 160 MHz of spectral bandwidth per stream, extracting up to 32768 subcarriers per incoming frame. To further validate the usefulness of the platform, we carry out a preliminary evaluation to measure the localization accuracy that IEEE 802.11ax enables. Our results, as expected, show a superior performance of IEEE 802.11ax compared to IEEE 802.11ac. In particular, the localization accuracy is improved by a factor of 1.75 in LOS and NLOS settings. This implies that the reported NLOS median error of a meter level in Chapter 4 can be improved to sub-meter level by using the hardware features of IEEE 802.11ax. We would like to emphasize that NLOS sub-meter level of accuracy has not been possible before without using specialized hardware.

Going beyond localization enables applications that are useful for network management, security and health care. Hence, Chapter 6 provides insights into sensing research by exploiting the proposed localization framework. In particular, we tackle respiration rate estimation using Wi-Fi signals. We address this issue by resolving the multipath components and extracting the path that comes from the human chest relatively free of interference from static paths. This enables a much cleaner recovery of the respiration signal than state-of-the-art works since the latter ones extract it without any interference cancellation from static paths. Our preliminary results show an accurate respiration rate estimation which validates that the proposed localization framework enables accurate sensing applications.

In summary, location-based services demand pervasive and precise localization. Recent localization systems achieve a sub-meter level of accuracy in LOS settings and dense Access Point (AP) densities. However, the performances of these systems drastically degrade in realistic wireless deployments. This thesis presents a localization framework to provide accurate and ubiquitous indoor positioning in challenging scenarios. In particular, this thesis presents a series of works that aim at increasing the resolvability of the

multipath components by developing more accurate path decomposition algorithms and exploiting the improved hardware features of the latest wireless protocols. Hence, the direct path is accurately extracted which results in precise positioning. Moreover, sensing applications also exploit position information from reflected paths as localization does from the direct path. Hence, we extend this framework to also provide insight into sensing research by tackling human respiration rate estimation.

7.1. Future work

Wireless protocols are evolving to provide new services in the areas of Industry 4.0, health care, security and many more. In this context, indoor localization will be a component of a more general framework that aims at becoming sensors intelligent by understating the environment. For instance, the incoming Wi-Fi protocol, IEEE 802.11 bf, will incorporate sensing applications [127]. In addition, wireless protocols are integrating traditional bands, sub-6GHz bands, with Milimiter-wave (mmWave) bands in the same device. Hence, researchers are also paying attention to maximizing the overall performance by combining information from both bands. In particular, we identify a number of open research directions that this thesis can contribute.

Chapter 6 provides the details of a human respiration rate estimator using Wi-Fi signals. Its evaluation is at an early stage because we have only analyzed the single-person case and the multi-person one is still ongoing. Our endeavors will focus on determining how many respiration signals can be discriminated since the paths that come from the chests are highly correlated and the respiration signals might be mixed among each other. As a result, a single path might contain information from several signals. We will be able to separate them by a frequency analysis if the rates are not the same.

Simultaneous Localization and Mapping (SLAM) is a potential extension of this framework as the position of obstacles can be estimated by extracting location information from NLOS paths while the system localizes the client. This is highly beneficial for tracking and navigation of mobile machines since obstacles are avoided from their trajectory. However, we need to deal with two issues:

1. Angle ambiguity: The geometry of a Uniform Linear Array (ULA) only allows discriminating angles from -90 to 90 degrees, therefore, the system cannot know if the signal is impinging from the front or the back of the ULA. As a result, there are two possibles positions of the objects. This can be addressed assuming mobility and observing the difference between the two possible positions over time. The true position of the object remains stable while the false one changes over time since the client is moving.

2. Second order reflections: The direct path provides location information from the

client and first order reflections from objects in the environment. However, second order reflections result in wrong position estimates since a system cannot determine the point where the signal reflected the first time. But, a system can filter out second order reflections since every time a signal bounces off, it attenuates ten dBs. Therefore, a threshold can be used according to the power of the direct path to remove second order reflections.

5G and Wi-Fi enable the coexistence of mmWave and sub-6GHz bands, therefore both bands can cooperate to maximize the overall performance. For instance, it has been extensively evaluated that mmWave can exploit AoA information from sub-6GHz bands to reduce the overhead of mmWave beam-training. However, to the best of our knowledge, a multiband localization system that deals with the drawbacks of both bands has not been thoroughly investigated yet. mmWave localization is remarkably accurate if the communication link propagates through the direct path. But if the link is misaligned, i.e., the link propagates through an NLOS path, the mmWave location estimate points toward a reflector and the positioning fails. This is due to the high directional links of mmWave that steer most of the transmitted power into an NLOS path. In contrast, sub-6GHz is not affected by this issue since it uses onmidirectional antennas, but sub-6GHz localization is likely to be less precise than the mmWave one. Hence, a multiband localization system can exploit the outstanding accuracy of mmWave and the robustness of sub-6GHz. To this end, sub-6GHz positioning can be used to detect if the mmWave estimated position is reliable that is when both bands agree on the estimated position of the client. However, if the mmWave link is misaligned, both bands highly disagree on the position of the client, therefore the multiband system assumes that the mmWave localization is pointing toward a reflector. Hence, the system filters out the mmWave localization and uses the sub-6GHz position.

References

[1] A. Blanco, N. Ludant, P. J. Mateo, Z. Shi, Y. Wang, and J. Widmer, "Performance evaluation of single base station toa-aoa localization in an lte testbed," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2019, pp. 1–6.

[2] A. B. Pizarro, J. P. Beltrán, M. Cominelli, F. Gringoli, and J. Widmer, "Accurate ubiquitous localization with off-the-shelf ieee 802.11ac devices," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 241–254. [Online]. Available: https://doi.org/10.1145/3458864.3468850

[3] F. Gringoli, M. Cominelli, A. Blanco, and J. Widmer, *AX-CSI: Enabling CSI Extraction on Commercial 802.11ax Wi-Fi Platforms.* New York, NY, USA: Association for Computing Machinery, 2022, p. 46–53. [Online]. Available: https://doi.org/10.1145/3477086.3480833

[4] C. Andrés Ramiro, C. Fiandrino, A. Blanco Pizarro, P. Jiménez Mateo, N. Ludant, and J. Widmer, "OpenLEON: An End-to-End Emulator from the Edge Data Center to the Mobile User," *Proceedings of the 12th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, 2018.

[5] P. J. Mateo, A. B. Pizarro, N. Ludant, M. M. Borelli, A. García-García, A. Loch, Z. Shi, Y. Wang, and J. Widmer, "A comprehensive study of low frequency and high frequency channel correlation," in *2019 International Conference on Computing, Networking and Communications (ICNC)*, 2019, pp. 876–882.

[6] C. Fiandrino, A. Blanco Pizarro, P. Jiménez Mateo, C. Andrés Ramiro, N. Ludant, and J. Widmer, "OpenLEON: An End-to-End Emulator from the Edge Data Center to the Mobile User," *Computer Communications*, 2019.

[7] X. Li, M. Ge, X. Dai, X. Ren, M. Fritsche, J. Wickert, and H. Schuh, "Accuracy and reliability of multi-gnss real-time precise positioning: Gps, glonass, beidou,

and galileo," *Journal of Geodesy*, vol. 89, no. 6, pp. 607–635, Jun 2015. [Online]. Available: https://doi.org/10.1007/s00190-015-0802-8

[8] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen, "A probabilistic approach to wlan user location estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, Jul 2002. [Online]. Available: https://doi.org/10.1023/A:1016003126882

[9] P. Castro, P. Chiu, T. Kremenek, and R. Muntz, "A probabilistic room location service for wireless networked environments," in *International conference on ubiquitous computing*. Springer, 2001, pp. 18–34.

[10] P. Bahl and V. Padmanabhan, "Radar: an in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, vol. 2, 2000, pp. 775–784 vol.2.

[11] M. Youssef, A. Agrawala, and A. Udaya Shankar, "Wlan location determination via clustering and probability distributions," in *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003).*, 2003, pp. 143–150.

[12] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, 2005, pp. 205–218.

[13] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019.

[14] 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*, 2015, pp. 269–282.

[15] 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*, 2015, pp. 537–549.

[16] J. Xiong and K. Jamieson, "Towards fine-grained radio-based indoor location," in *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, 2012, pp. 1–6.

[17] K. Chintalapudi, A. Padmanabha Iyer, and V. N. Padmanabhan, "Indoor localization without the pain," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*. ACM, 2010, pp. 173–184.

[18] J. Medbo, I. Siomina, A. Kangas, and J. Furuskog, "Propagation channel impact on lte positioning accuracy: A study based on real measurements of observed time difference of arrival," in *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2009, pp. 2213–2217.

[19] W. Wang, A. X. Liu, M. Shahzad, K. Ling, and S. Lu, "Device-free human activity recognition using commercial wifi devices," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1118–1131, 2017.

[20] Y. Gu, L. Quan, and F. Ren, "Wifi-assisted human activity recognition," in *2014 IEEE Asia Pacific Conference on Wireless and Mobile*, 2014, pp. 60–65.

[21] B. Korany and Y. Mostofi, "Counting a stationary crowd using off-the-shelf wifi," in *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 202–214. [Online]. Available: https://doi.org/10.1145/3458864.3468012

[22] B. Tan, K. Chetty, and K. Jamieson, "Thrumapper: Through-wall building tomography with a single mapping robot," in *Proceedings of the 18th International Workshop on Mobile Computing Systems and Applications*, ser. HotMobile '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1–6. [Online]. Available: https://doi.org/10.1145/3032970.3032973

[23] R. Yang, X. Yang, J. Wang, M. Zhou, Z. Tian, and L. Li, "Decimeter level indoor localization using wifi channel state information," *IEEE Sensors Journal*, 2021.

[24] R. Ayyalasomayajula, A. Arun, C. Wu, S. Sharma, A. R. Sethi, D. Vasisht, and D. Bharadia, "Deep learning based wireless localization for indoor navigation," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020, pp. 1–14.

[25] 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*, 2019, pp. 200–206.

[26] A. Decurninge, L. G. Ordóñez, P. Ferrand, H. Gaoning, L. Bojie, Z. Wei, and M. Guillaud, "Csi-based outdoor localization for massive mimo: Experiments with a learning approach," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, 2018, pp. 1–6.

[27] E. Y. Menta, N. Malm, R. Jäntti, K. Ruttik, M. Costa, and K. Leppänen, "On the performance of aoa–based localization in 5g ultra–dense networks," *IEEE Access*, vol. 7, pp. 33 870–33 880, 2019.

[28] I. Gomez-Miguelez, A. Garcia-Saavedra, P. D. Sutton, P. Serrano, C. Cano, and D. J. Leith, "srsLTE: An open-source platform for lte evolution and experimentation," in *Proc. of ACM WiNTECH*, 2016, pp. 25–32.

[29] H. Anouar, C. Bonnet, D. Câmara, F. Filali, and R. Knopp, "An overview of OpenAirInterface wireless network emulation methodology," *ACM SIGMETRICS Perform. Eval. Rev.*, vol. 36, no. 2, pp. 90–94, 2008.

[30] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni, "A survey on wireless indoor localization from the device perspective," *ACM Comput. Surv.*, vol. 49, no. 2, jun 2016. [Online]. Available: https://doi.org/10.1145/2933232

[31] J. P. Burg, *Maximum entropy spectral analysis.* Stanford University, 1975.

[32] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE transactions on antennas and propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[33] D. Malioutov, M. Cetin, and A. S. Willsky, "A sparse signal reconstruction perspective for source localization with sensor arrays," *IEEE transactions on signal processing*, vol. 53, no. 8, pp. 3010–3022, 2005.

[34] 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*, 2013, pp. 249–262.

[35] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, vol. 2. Ieee, 2000, pp. 775–784.

[36] F. Ricciato, S. Sciancalepore, F. Gringoli, N. Facchi, and G. Boggia, "Position and velocity estimation of a non-cooperative source from asynchronous packet arrival time measurements," *IEEE Transactions on Mobile Computing*, vol. 17, no. 9, pp. 2166–2179, 2018.

[37] 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*, 2018, pp. 417–427.

[38] *3GPP TR 38.913 v14.3.0 "5G;Study on Scenarios and Requirements for Next Generation Access Technologies".*

[39] *3GPP TR 22.872 V2.0.0 Technical Specification Group Services and System Aspects, "Study on positioning use cases"*, May 2018.

[40] V. Cisco, "Cisco visual networking index: Forecast and trends, 2017–2022," *White Paper*, 2018.

[41] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive mimo: Benefits and challenges," *IEEE journal of selected topics in signal processing*, vol. 8, no. 5, pp. 742–758, 2014.

[42] J. A. del Peral-Rosado, J. A. López-Salcedo, G. Seco-Granados, F. Zanier, and M. Crisci, "Achievable localization accuracy of the positioning reference signal of 3gpp lte," in *2012 International Conference on Localization and GNSS*, June 2012, pp. 1–6.

[43] W. Xu, M. Huang, C. Zhu, and A. Dammann, "Maximum likelihood toa and otdoa estimation with first arriving path detection for 3gpp lte system," *Transactions on Emerging Telecommunications Technologies*, vol. 27, no. 3, pp. 339–356, 2016.

[44] J. A. del Peral-Rosado, J. A. Lopez-Salcedo, G. Seco-Granados, F. Zaniera, P. Crosta, R. Ioannides, and M. Crisci, "Software-defined radio lte positioning receiver towards future hybrid localization systems," in *31st AIAA International Communications Satellite Systems Conference, International Communications Satellite Systems Conferences (ICSSC)*, 2017, pp. 337–340.

[45] Q. Liu, R. Hu, and S. Liu, "A wireless location system in lte networks," *Mobile Information Systems*, vol. 2017, 2017.

[46] S. M. Razavi, F. Gunnarsson, H. Rydén, Å. Busin, X. Lin, X. Zhang, S. Dwivedi, I. Siomina, and R. Shreevastav, "Positioning in cellular networks: Past, present, future," in *2018 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2018, pp. 1–6.

[47] D. Stojanović and N. Stojanović, "Indoor localization and tracking: Methods, technologies and research challenges," *Facta Universitatis, Series: Automatic Control and Robotics*, vol. 13, no. 1, pp. 57–72, 2014.

[48] *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation.*

[49] M. Gul, X. Ma, and S. Lee, "Timing and frequency synchronization for ofdm downlink transmissions using zadoff-chu sequences," *IEEE Transactions on Wireless Communications*, vol. 14, no. 3, pp. 1716–1729, 2015.

[50] J. A. del Peral-Rosado, R. Raulefs, J. A. López-Salcedo, and G. Seco-Granados, "Survey of cellular mobile radio localization methods: from 1g to 5g," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 2, pp. 1124–1148, 2017.

[51] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[52] X. Zhang, L. Xu, L. Xu, and D. Xu, "Direction of departure (dod) and direction of arrival (doa) estimation in mimo radar with reduced-dimension music," *IEEE communications letters*, vol. 14, no. 12, pp. 1161–1163, 2010.

[53] E. Blossom, "Gnu radio: tools for exploring the radio frequency spectrum," *Linux journal*, vol. 2004, no. 122, p. 4, 2004.

[54] R. Roy and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.

[55] X. Li, S. Li, D. Zhang, J. Xiong, Y. Wang, and H. Mei, "Dynamic-music: accurate device-free indoor localization," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 196–207.

[56] F. Adib and D. Katabi, "See through walls with wifi!" in *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*, 2013, pp. 75–86.

[57] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm," *IEEE Transactions on signal processing*, vol. 45, no. 3, pp. 600–616, 1997.

[58] J. Yin and T. Chen, "Direction-of-arrival estimation using a sparse representation of array covariance vectors," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4489–4493, 2011.

[59] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the nelder–mead simplex method in low dimensions," *SIAM Journal on optimization*, vol. 9, no. 1, pp. 112–147, 1998.

[60] M. Ibrahim, A. Rostami, B. Yu, H. Liu, M. Jawahar, V. Nguyen, M. Gruteser, F. Bai, and R. Howard, "Wi-go: accurate and scalable vehicle positioning using wifi

fine timing measurement," in *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 2020, pp. 312–324.

[61] K. Jiokeng, G. Jakllari, A. Tchana, and A.-L. Beylot, "When FTM Discovered MUSIC: Accurate WiFi-based Ranging in the Presence of Multipath," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 1857–1866.

[62] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11 n traces with channel state information," *ACM SIGCOMM Computer Communication Review*, pp. 53–53, 2011.

[63] alejandroBlancoPizarro, "Ubilocate," https://github.com/IMDEANetworksWNG/UbiLocate, 2021.

[64] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[65] S. Sen, N. Santhapuri, R. R. Choudhury, and S. Nelakuditi, "Successive interference cancellation: A back-of-the-envelope perspective," in *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, 2010, pp. 1–6.

[66] D. Halperin, T. Anderson, and D. Wetherall, "Taking the sting out of carrier sense: interference cancellation for wireless lans," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 339–350.

[67] B. H. Fleury, M. Tschudin, R. Heddergott, D. Dahlhaus, and K. I. Pedersen, "Channel parameter estimation in mobile radio environments using the sage algorithm," *IEEE Journal on selected areas in communications*, vol. 17, no. 3, pp. 434–450, 1999.

[68] Y. Xie, J. Xiong, M. Li, and K. Jamieson, "md-track: Leveraging multi-dimensionality for passive indoor wi-fi tracking," in *The 25th Annual International Conference on Mobile Computing and Networking*, 2019, pp. 1–16.

[69] T. S. Rappaport *et al.*, *Wireless communications: principles and practice*. prentice hall PTR New Jersey, 1996, vol. 2.

[70] F. Gringoli, M. Schulz, J. Link, and M. Hollick, "Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets," in *Proceedings of the 13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, 2019, pp. 21–28.

[71] "NexMon Project," https://github.com/seemoo-lab/nexmon/.

[72] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. Anderson, and P. N. Belhumeur, "A theory of network localization," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, 2006.

[73] T.-J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 4, pp. 806–811, 1985.

[74] S. U. Pillai and B. H. Kwon, "Forward/backward spatial smoothing techniques for coherent signal identification," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 1, pp. 8–15, 1989.

[75] R. T. Williams, S. Prasad, A. K. Mahalanabis, and L. H. Sibul, "An improved spatial smoothing technique for bearing estimation in a multipath environment," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 4, pp. 425–432, 1988.

[76] P. Kumar, L. Reddy, and S. Varma, "Distance measurement and error estimation scheme for rssi based localization in wireless sensor networks," in *2009 Fifth international conference on wireless communication and sensor networks (WCSN)*. IEEE, 2009, pp. 1–4.

[77] İ. Güvenc, "Enhancements to rss based indoor tracking systems using kalman filters," Ph.D. dissertation, University of New Mexico, 2003.

[78] A. Goswami, L. E. Ortiz, and S. R. Das, "Wigem: A learning-based approach for indoor localization," in *Proceedings of the Seventh COnference on emerging Networking EXperiments and Technologies*, 2011, pp. 1–12.

[79] G. V. Zàruba, M. Huber, F. Kamangar, and I. Chlamtac, "Indoor location tracking using rssi readings from a single wi-fi access point," *Wireless networks*, vol. 13, no. 2, pp. 221–235, 2007.

[80] 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*, 2011, pp. 1–12.

[81] M. Youssef, A. Youssef, C. Rieger, U. Shankar, and A. Agrawala, "Pinpoint: An asynchronous time-based location determination system," in *Proceedings of the 4th international conference on Mobile systems, applications and services*, 2006, pp. 165–176.

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

[83] 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, 2017.

[84] 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*. ACM, 2014, pp. 315–328.

[85] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE signal processing magazine*, vol. 13, no. 4, pp. 67–94, 1996.

[86] J. Xiong and K. Jamieson, "Arraytrack: a fine-grained indoor location system." Usenix, 2013.

[87] E. Soltanaghaei, A. Kalyanaraman, and K. Whitehouse, "Multipath triangulation: Decimeter-level wifi localization and orientation with a single unaided receiver," in *Proceedings of the 16th annual international conference on mobile systems, applications, and services*, 2018, pp. 376–388.

[88] I. Guvenc, C.-C. Chong, and F. Watanabe, "Nlos identification and mitigation for uwb localization systems," in *2007 IEEE Wireless Communications and Networking Conference*. IEEE, 2007, pp. 1571–1576.

[89] P. Pannuto, B. Kempke, L.-X. Chuo, D. Blaauw, and P. Dutta, "Harmonium: Ultra wideband pulse generation with bandstitched recovery for fast, accurate, and robust indoor localization," *ACM Transactions on Sensor Networks (TOSN)*, vol. 14, no. 2, pp. 1–29, 2018.

[90] F. Adib, Z. Kabelac, and D. Katabi, "Multi-person localization via RF body reflections," in *12th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 15)*, 2015, pp. 279–292.

[91] A. R. J. Ruiz and F. S. Granja, "Comparing ubisense, bespoon, and decawave uwb location systems: Indoor performance analysis," *IEEE Transactions on instrumentation and Measurement*, vol. 66, no. 8, pp. 2106–2117, 2017.

[92] M. Yang, L.-X. Chuo, K. Suri, L. Liu, H. Zheng, and H.-S. Kim, "ilps: Local positioning system with simultaneous localization and wireless communication," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 379–387.

[93] L.-X. Chuo, Z. Luo, D. Sylvester, D. Blaauw, and H.-S. Kim, "Rf-echo: A non-line-of-sight indoor localization system using a low-power active rf reflector asic tag," in

Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking, 2017, pp. 222–234.

[94] T. Ma, Y. Xiao, X. Lei, W. Xiong, and Y. Ding, "Indoor localization with reconfigurable intelligent surface," *IEEE Communications Letters*, 2020.

[95] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces," in *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, 2019, pp. 1–5.

[96] T. Sathyan, D. Humphrey, and M. Hedley, "Wasp: A system and algorithms for accurate radio localization using low-cost hardware," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 2, pp. 211–222, 2010.

[97] J. Xianjun, L. Wei, and M. Michael, "open-source ieee802.11/wi-fi baseband chip/fpga design," 2019. [Online]. Available: https://github.com/open-sdr/openwifi

[98] B. Bloessl, M. Segata, C. Sommer, and F. Dressler, "An ieee 802.11 a/g/p ofdm receiver for gnu radio," in *Proceedings of the second workshop on Software radio implementation forum*, 2013, pp. 9–16.

[99] Y. Ma, G. Zhou, and S. Wang, "Wifi sensing with channel state information: A survey," *ACM Computing Surveys*, vol. 52, no. 3, pp. 1–36, Jun. 2019. [Online]. Available: http://doi.acm.org/10.1145/3310194

[100] F. Gringoli, M. Schulz, J. Link, and M. Hollick, "Free your csi: A channel state information extraction platform for modern wi-fi chipsets," in *13th International Workshop on Wireless Network Testbeds, Experimental Evaluation & Characterization*, ser. WiNTECH '19, 2019, p. 21–28. [Online]. Available: https://doi.org/10.1145/3349623.3355477

[101] J. O. Lacruz, D. Garcia, P. J. Mateo, J. Palacios, and J. Widmer, "mm-FLEX: An Open Platform for Millimeter-Wave Mobile Full-Bandwidth Experimentation," *Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services*, 2020.

[102] C. Wu, F. Zhang, B. Wang, and K. J. Ray Liu, "mmTrack: Passive Multi-Person Localization Using Commodity Millimeter Wave Radio," *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, 2020.

[103] J. Palacios, P. Casari, and J. Widmer, "JADE: Zero-knowledge device localization and environment mapping for millimeter wave systems," *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017.

[104] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, "Tool release: Gathering 802.11n traces with channel state information," *SIGCOMM Computer Communication Review*, vol. 41, no. 1, p. 53, Jan. 2011. [Online]. Available: https://doi.org/10.1145/1925861.1925870

[105] A. Zubow, P. Gawłowicz, and F. Dressler, "On phase offsets of 802.11 ac commodity wifi," in *2021 16th Annual Conference on Wireless On-demand Network Systems and Services Conference (WONS '21)*, 2021, pp. 1–4.

[106] Y. Xie, Z. Li, and M. Li, "Precise power delay profiling with commodity wifi," in *21st Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '15, 2015, p. 53–64. [Online]. Available: https://doi.org/10.1145/2789168.2790124

[107] M. N. Mahfoudi, "Unlocking wireless sensing potential in wi-fi and iot networks," Theses, Université Côte d'Azur, Oct. 2019. [Online]. Available: https://hal.archives-ouvertes.fr/tel-02431424

[108] R. Ayyalasomayajula, A. Arun, C. Wu, S. Rajagopalan, S. Ganesaraman, A. Seetharaman, I. K. Jain, and D. Bharadia, "Locap: Autonomous millimeter accurate mapping of wifi infrastructure," in *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*, Feb. 2020, pp. 1115–1129. [Online]. Available: https://www.usenix.org/conference/nsdi20/presentation/ayyalasomayajula

[109] R. University, "Warp: Wireless open access research platform," 2020. [Online]. Available: https://warpproject.org/

[110] E. Research, "The universal software radio peripheral usrp software defined radio device," 2021. [Online]. Available: https://www.ettus.com/

[111] X. Jiao, W. Liu, M. Mehari, M. Aslam, and I. Moerman, "openwifi: a free and open-source ieee802.11 sdr implementation on soc," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–2.

[112] S. Li, X. Li, K. Niu, H. Wang, Y. Zhang, and D. Zhang, "Ar-alarm: An adaptive and robust intrusion detection system leveraging csi from commodity wi-fi," in *Enhanced Quality of Life and Smart Living*, M. Mokhtari, B. Abdulrazak, and H. Aloulou, Eds. Cham: Springer International Publishing, 2017, pp. 211–223.

[113] J. Lv, D. Man, W. Yang, X. Du, and M. Yu, "Robust wlan-based indoor intrusion detection using phy layer information," *IEEE Access*, vol. 6, pp. 30 117–30 127, 2018.

[114] D. Zhu, N. Pang, G. Li, and S. Liu, "Notifi: A ubiquitous wifi-based abnormal activity detection system," in *2017 International Joint Conference on Neural Networks (IJCNN)*, 2017, pp. 1766–1773.

[115] Y. Wang, K. Wu, and L. M. Ni, "Wifall: Device-free fall detection by wireless networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 2, pp. 581–594, 2017.

[116] S. Palipana, D. Rojas, P. Agrawal, and D. Pesch, "Falldefi: Ubiquitous fall detection using commodity wi-fi devices," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 1, no. 4, jan 2018. [Online]. Available: https://doi.org/10.1145/3161183

[117] S. Arshad, C. Feng, Y. Liu, Y. Hu, R. Yu, S. Zhou, and H. Li, "Wi-chase: A wifi based human activity recognition system for sensorless environments," in *2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2017, pp. 1–6.

[118] H. Abdelnasser, K. A. Harras, and M. Youssef, "Ubibreathe: A ubiquitous non-invasive wifi-based breathing estimator," in *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 277–286. [Online]. Available: https://doi.org/10.1145/2746285.2755969

[119] B. Fang, N. D. Lane, M. Zhang, A. Boran, and F. Kawsar, "Bodyscan: Enabling radio-based sensing on wearable devices for contactless activity and vital sign monitoring," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 97–110. [Online]. Available: https://doi.org/10.1145/2906388.2906411

[120] J. Liu, Y. Wang, Y. Chen, J. Yang, X. Chen, and J. Cheng, "Tracking vital signs during sleep leveraging off-the-shelf wifi," ser. MobiHoc '15. New York, NY, USA: Association for Computing Machinery, 2015. [Online]. Available: https://doi.org/10.1145/2746285.2746303

[121] X. Liu, J. Cao, S. Tang, and J. Wen, "Wi-sleep: Contactless sleep monitoring via wifi signals," in *2014 IEEE Real-Time Systems Symposium*, 2014, pp. 346–355.

[122] X. Liu, J. Cao, S. Tang, J. Wen, and P. Guo, "Contactless respiration monitoring via off-the-shelf wifi devices," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2466–2479, 2016.

[123] P. Wang, B. Guo, T. Xin, Z. Wang, and Z. Yu, "Tinysense: Multi-user respiration detection using wi-fi csi signals," in *2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2017, pp. 1–6.

[124] H. Wang, D. Zhang, J. Ma, Y. Wang, Y. Wang, D. Wu, T. Gu, and B. Xie, "Human respiration detection with commodity wifi devices: Do user location and body orientation matter?" ser. UbiComp '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 25–36. [Online]. Available: https://doi.org/10.1145/2971648.2971744

[125] Y. Zeng, D. Wu, J. Xiong, J. Liu, Z. Liu, and D. Zhang, "Multisense: Enabling multi-person respiration sensing with commodity wifi," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 4, no. 3, sep 2020. [Online]. Available: https://doi.org/10.1145/3411816

[126] S. Shi, Y. Xie, M. Li, A. X. Liu, and J. Zhao, "Synthesizing wider wifi bandwidth for respiration rate monitoring in dynamic environments," in *IEEE INFOCOM 2019 - IEEE Conference on Computer Communications*, 2019, pp. 181–189.

[127] F. Restuccia, "Ieee 802.11 bf: Toward ubiquitous wi-fi sensing," *arXiv preprint arXiv:2103.14918*, 2021.