FINE-GRAINED INDOOR POSITIONING AND TRACKING SYSTEMS

Inauguraldissertation
der Philosophisch-naturwissenschaftlichen Fakultät
der Universität Bern

vorgelegt von

Zan Li

von Zibo, China

Leiter der Arbeit:
Professor Dr. Torsten Braun
Institut für Informatik

Copyright Notice

FINE-GRAINED INDOOR POSITIONING AND TRACKING SYSTEMS

Inauguraldissertation
der Philosophisch-naturwissenschaftlichen Fakultät
der Universität Bern

vorgelegt von

Zan Li

von Zibo, China

Leiter der Arbeit:
Professor Dr. Torsten Braun
Institut für Informatik

Von der Philosophisch-naturwissenschaftlichen Fakultät angenommen.

Der Dekan:
Bern, 15.03.2016 Prof. Dr. Gilberto Colangelo

Abstract

Indoor positioning has attracted considerable attention for decades due to the increasing demands for location based services. In the past years, although numerous methods have been proposed for indoor positioning, it is still challenging to find a convincing solution that combines high positioning accuracy and ease of deployment. Radio-based indoor positioning has emerged as a dominant method due to its ubiquitousness, especially for WiFi. RSSI (Received Signal Strength Indicator) has been investigated in the area of indoor positioning for decades. However, it is prone to multipath propagation and hence fingerprinting has become the most commonly used method for indoor positioning using RSSI. The drawback of fingerprinting is that it requires intensive labour efforts to calibrate the radio map prior to experiments, which makes the deployment of the positioning system very time consuming. Using time information as another way for radio-based indoor positioning is challenged by time synchronization among anchor nodes and timestamp accuracy. Besides radio-based positioning methods, intensive research has been conducted to make use of inertial sensors for indoor tracking due to the fast developments of smartphones. However, these methods are normally prone to accumulative errors and might not be available for some applications, such as passive positioning.

This thesis focuses on network-based indoor positioning and tracking systems, mainly for passive positioning, which does not require the participation of targets in the positioning process. To achieve high positioning accuracy, we work on some information of radio signals from physical-layer processing, such as timestamps and channel information. The contributions in this thesis can be divided into two parts: time-based positioning and channel information based positioning. First, for time-based indoor positioning (especially for narrow-band signals), we address challenges for compensating synchronization offsets among anchor nodes, designing timestamps with high resolution, and developing accurate positioning methods. Second, we work on range-based positioning methods with channel information to passively locate and track WiFi targets. Targeting less efforts for deployment, we work on range-based methods, which require much less calibration efforts than fingerprinting. By designing some novel enhanced methods for both ranging and positioning (including trilateration for stationary targets and particle filter for mobile targets), we are able to locate WiFi targets with high accuracy solely relying on radio signals and our proposed enhanced particle filter significantly outperforms the other commonly used range-based positioning algorithms, e.g., a traditional particle filter, extended Kalman filter and trilateration algorithms. In addition to using radio signals for passive positioning, we propose a second enhanced particle filter for active positioning to fuse inertial sensor and channel information to track indoor targets, which achieves higher tracking accuracy than tracking methods solely relying on either radio signals or inertial sensors.

Contents

List of Figures

List of Tables

Preface

The work presented within this thesis was achieved during my employment as Research Assistant and PhD Student at the Institute of Computer Science (INF) of the University of Bern. During that time, I was mainly funded over the Eurostar project *In3DGuide*, as well as the *SwissSenseSynergy* project financed by the Swiss National Science Foundation (SNF). These two public organizations deserve my honest gratitude for supporting me and giving me the opportunity to pursue my research.

I express my gratitude to Prof. Dr. Torsten Braun, head of the Communication and Distributed Systems group (CDS), for supervising this work and for his insightful advices. Prof. Dr. Torsten Braun encouraged and motivated me to publish my results in various reputable conferences and journals.

I also thank Prof. Dr. Horst Hellbrück for reading this work and providing valuable feedbacks, and Prof. Dr. Oscar Nierstrasz, who was willing to be the co-examinator of this thesis.

Many thanks go to my colleagues at the institute and in our research group for being part of a great team. In particular, I want to thank Desislava Dimitrova, Zhongliang Zhao, Thomas Staub, Markus Anwander, Carlos Anastasiades, Andre Gomes, Jose Luis Carrera, Philipp Hurni, Jonnahtan Saltarin, Marcel Stolz, Denis Rosario, Mikael Gasparyan, Ali Marandi, Almerima Jamakovic-Kapic, and Eryk Schiller for their good cooperation over the past years.

Special thanks go to Daniela Schroth, the secretary of the CDS research group, for her giant support in the administrative tasks during all these years.

I am especially thankful to Dr. Desislava Dimitrova for her kind helps on my doctoral study and Dr. Zhongliang Zhao for his patient proofreading of this thesis and many amendment suggestions.

I would also like to thank all the students, who contributed to this thesis in one way or another. In particular, thanks go to Urs Zysset, Adrian Kurt, Marcel Stolz, Danilo Burbano, and Jose Luis Carrera who performed their Bachelor thesis or Master thesis under my guidance.

Finally, I am deeply grateful to my wife Qiao for her support, understanding, and sharing a wonderful time together. I am grateful to my parents for their support and encouragement.

Chapter 1

Introduction

Positioning humans and objects indoors with high accuracy is a prerequisite for a large range of services in navigation, assistant living, health care and many more areas. The Global Positioning System (GPS) [92] as the dominating solution outdoors falls short to deliver reliable performance in indoor spaces, which triggers investigations on indoor positioning systems in recent years. Despite the importance of location information it is still challenging to find a convincing solution that combines high positioning accuracy and ease of deployment. In this work, positioning accuracy is defined as the error between the estimated location and the ground truth location of the target. Ease of deployment is for reducing calibration efforts to deploy the system. The development of such systems covers several aspects including the choice of system architecture and processed radio parameters to derive location. The inherent merits and weaknesses of each choice, combined with the stringent positioning requirements, make the system design a nontrivial task.

1.1 Overview

In the past decades, numerous methods have been proposed to solve indoor positioning issues like vision [75], magnetic sensor [113], radio [124] and inertial sensor [50] based positioning. Vision-based positioning processes video data for positioning. In this technology, optical devices like cameras are mounted at fixed locations to extract visual features as spatial references and form a map. Then, acquired new visual features are matched to the map to locate the target. The drawback of this technology is its high costs to mount cameras. Features of indoor magnetic fields have been recently proposed to construct the map for indoor positioning because it is ubiquitous and magnetic sensors are available in modern smart phones. This technology has been recently used for indoor navigation at the terminal side. Nowadays, radio signals are also ubiquitous indoor. For example, signals of WiFi and cellular networks have covered most of the indoor areas. Therefore, intensive research has been conducted to adopt radio signals for indoor positioning. Besides indoor navigation, location information estimated by using radio signals is also used for many other applications, such as network planning and user behaviour analysis. In this work, we mainly work on radio-based indoor positioning. Additionally, inertial sensor information is investigated to improve the performance of radio-based indoor positioning. Therefore, radio-based positioning and Pedestrian Dead Reckoning (PDR) based on inertial sensors (also called Inertial Measurement Units (IMUs)) are described in more details in the remainder of this section.

1.1.1 Radio-based Indoor Positioning

Location (or position) information can be conveyed in absolute location and relative location. Absolute location describes the location of a place (or target) based on a fixed point in the map. The most common way is to identify the absolute location using coordinates such as latitude and longitude crisscross the earth or X and Y axes in a local coordinate system. A relative location describes the location of a place (or target) relative to another absolute location. Radio-based indoor positioning is used to estimate the absolute location of a target in a local coordinate system.

Some radio signals are particularly fit to indoor positioning, such as Ultra Wide Band (UWB) signals, which achieve sub-meter positioning accuracy. However, such signals require additional devices to support indoor positioning and are only used in some particular scenarios or applications such as mobile robot navigation [70], assistant living [38] or health care [134]. For example, positioning with UWB signals used in mobile robot navigation [70] is achieved by equipping robots with UWB sensors. In assistant living [38] or health care [134] applications, the target people including patients or elderly people need to carry the UWB nodes for positioning. However, in some scenarios, it is impractical to use these specific devices. For example, passengers in an airport need their location information to be navigated to their boarding gates. In a shopping mall, the shop owners are interested in their customers' location information and investigate their buying behaviours. To provide such indoor positioning services, large amount of research has been conducted to adopt more commonly used radio signals for indoor positioning. Particularly, IEEE 802.11 (WiFi) is currently the dominant local wireless network standard for short-range communication in indoor environments and is the leading technology for indoor positioning [124, 123, 24, 129]. IEEE 802.15.4, e.g., Zigbee, is another wireless standard for short-range communication. It is widely used in home automation, wireless sensor network and industry applications. It has attracted interests of researchers as an alternative to WiFi positioning [106, 57]. Besides short-range communication techniques, indoor positioning based on cellular networks such as GSM [119], UMTS [27] and LTE [43], has also been investigated for decades, due to their wide coverage and large number of end users.

By analyzing some radio parameters of the received packets such as Received Signal Strength Indicator (RSSI), timestamps, and channel information, numerous radio-based positioning algorithms have been proposed to locate the targets. Radio-based indoor positioning is classified as range-based and range-free techniques. Range is defined as the distance between the target and an Anchor Node (AN). ANs are deployed at known locations to estimate the location of the target.

Fingerprinting [124, 24, 129] is the most commonly used range-free method, which consists of two steps: offline training and online positioning. The purpose of offline training is to build up a radio map, which contains features of radio signals like RSSI or channel information in different training locations, i.e., fingerprints. Online positioning is the procedure to locate the target based on the radio map. In online positioning, pattern matching algorithms are adopted to match the measured radio signal features to the fingerprints in the radio map. This method achieves high positioning accuracy but is very labour intensive to calibrate (train) the radio map, which makes the deployment of the system very time consuming. Although fingerprinting has been developed for decades, nowadays intensive research is still conducted in this area, especially for automatic offline calibration [91], which can dramatically reduce calibration and deployment efforts.

In contrast to fingerprinting, range-based indoor positioning methods do not rely on the radio map but need to convert the measured radio signal parameters (features) to the propagation distances between the target and different ANs. These methods consist of two steps: ranging and positioning. In the ranging phase, different radio parameters can be converted to the range information based on different propagation models. For example, time information is converted to range considering the speed of radio signals [87]. Received signal strength can be converted to range using a commonly used Log-Distance Path Loss (LDPL) model [49]. After ranging, positioning algorithms are adopted to convert the range information to the location of the target. Trilateration or multilateration algorithms, which model the positioning problem as a convex optimization problem, are often used to locate stationary targets. Bayesian estimation methods, e.g., Kalman filters and particle filters, which consider the tracking problem as a Hidden Markov Model (HMM), are more suitable to track mobile targets. Compared to fingerprinting, range-based indoor positioning requires less labour efforts for calibration and deployment prior to positioning, although it normally needs to calibrate the ranging models.

1.1.2 Pedestrian Dead Reckoning

In contrast to radio-based indoor positioning, PDR [50, 121] leverages inertial sensors in smart phones e.g., accelerometer, magnetometer, and gyroscope, to estimate the relative location of the target by detecting steps, estimating stride length and heading orientation. By integrating the estimated relative locations at sequential time intervals, PDR systems can track the trajectories of targets. In contrast to inertial sensors used in mobile robot navigation, inertial sensors in smart phones are normally cheap and suffer from measurement noise. Because PDR integrates the relative locations at sequential time intervals, small positioning errors resulting from the noise in low cost inertial sensors are magnified and therefore PDR is normally prone to accumulated errors. Since radio-based indoor positioning estimates

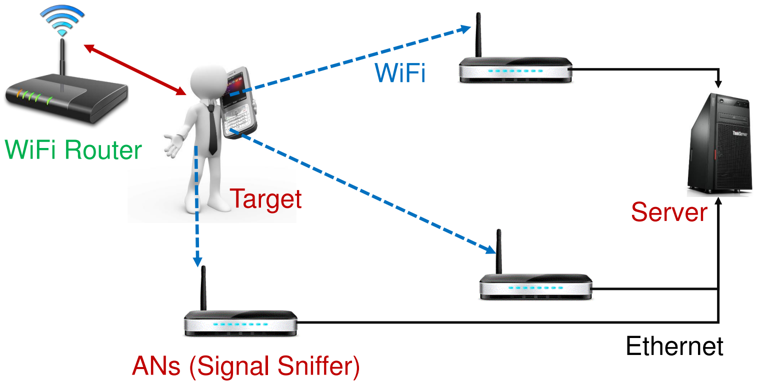


Figure 1.1: A Passive Indoor Positioning System for WiFi Users

the absolute locations and inertial sensor information used in PDR provides relative locations between sequential time intervals, these two methods are complementary. Therefore, numerous hybrid approaches [64, 104, 50] have been proposed to fuse the inertial sensor information and radio information to improve the tracking/positioning accuracy.

1.1.3 Active and Passive Positioning Systems

Irrespective of the positioning algorithms, indoor positioning systems are classified as active and passive positioning systems based on the target's participation. In an active positioning system, target devices need to actively participate in the positioning process. In a passive positioning system, the target devices are oblivious to the positioning process. These two types of positioning systems fit different demands of different applications. For example, as the aforementioned application, where the passengers want to be navigated to their boarding gates, it normally requires the users' devices, i.e., smartphone, to actively participate in the positioning process. Passive positioning systems for WiFi users are attractive for third-party providers of positioning and monitoring services. For example, shop owners can passively capture the radio signals from their customers' devices and locate the users in a remote server. Figure 1.1 shows an example of a passive positioning system for WiFi users. Compared to active positioning, which mainly relies on the users' devices, passive positioning systems require additional equipments (signal sniffers) to passively capture the radio signals and locate the target users in a powerful remote server. Passive positioning systems can support positioning multiple targets and run positioning algorithms with high computation requirements.

1.2 Problem Statement

Numerous algorithms have been proposed for indoor positioning to adopt different radio parameters and inertial sensors. However, different positioning methods

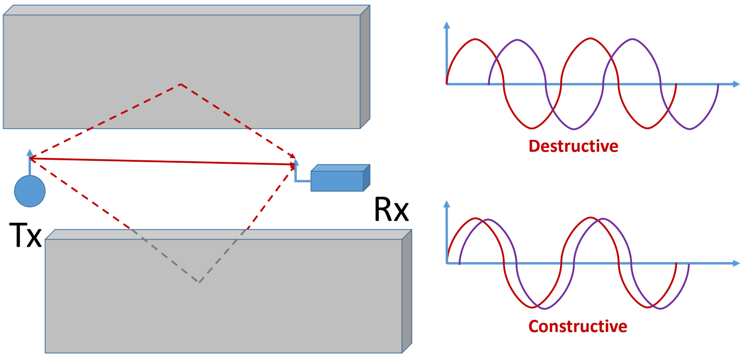


Figure 1.2: Destructive and Constructive Signals

pose different challenges. In this section, we review the challenges and state the problems that we target to solve in this thesis.

1.2.1 RSSI-based Indoor Positioning

RSSI is one of the most commonly used radio parameters for indoor positioning, because it is available in most of the commercial devices. As described in [126], parameters extracted from the received radio signals for positioning are classified as coarse-grained and fine-grained. Coarse-grained parameters are obtained from MAC (Media Access Control) layer on the packet level, which is after physical-layer processing. Fine-grained parameters are extracted during physical-layer processing, which provide more information for positioning than coarse-grained parameters. RSSI is a coarse-grained parameter obtained from MAC layer in ZigBee, WiFi and other wireless technologies. The main drawback of RSSI is its temporal fluctuations in complex indoor environments because of channel fading. Channel fading is classified as multipath induced fading and shadow fading.

In indoor environments, a radio signal propagates to a receiver through multiple propagation paths, in which the signals have different delays, attenuations and phase shifts. Then, RSSI is simply calculated as the power of the received signal, which combines multiple versions of the original signal. Therefore, a small change in the multipath propagation may lead to significant changes of the relative phases in constructive or destructive signals as shown in Figure 1.2. This will cause significant fluctuations in RSSI. This fading is introduced by multipath propagation and hence it is called multipath induced fading. The operating frequency of a radio signal plays an important role in the multipath effects. For a higher operating frequency, the wavelength of the signal, which is the distance that the radio signal travels during the period of one radio cycle, is shorter and correspondingly the phase varies more over the same propagation distance. Therefore, operating at a higher frequency, the reflected signals face more dynamic phase shifts from the original signal and the multipath induced fading is more severe. Nowadays, most of the commonly used radio signals operate at very high frequency. For example, WiFi operates on 2.4GHz and 5GHz, which faces significant multipath effects.

Shadow fading is caused by obstacles between the transmitter and receiver, such as walls, doors and people. The radio signals penetrating obstacles are attenuated. Because of the complex indoor environments, this kind of attenuation is very difficult to measure or model. Some path loss models have been proposed such as the LDPL model [49], which is more suitable for a wide range of environments but is inaccurate for indoor environments. Because the layouts in different indoor environments are normally different, a generic model introduces large errors.

Because of multipath induced fading and shadow fading, range-based indoor positioning relying on RSSI is challenging. Therefore, fingerprinting is currently the most effective method for indoor positioning based on RSSI. However, as mentioned before, fingerprinting is very labour intensive and costs large amount of calibration efforts. In this thesis, **we aim to provide a positioning solution, which provides high positioning accuracy but also has low calibration efforts**. To achieve this goal, some fine-grained radio parameters at the physical layer instead of RSSI are adopted for range-based indoor positioning.

1.2.2 Channel Information for Indoor Positioning

Channel information is an important characteristic in OFDM (Orthogonal Frequency Division Multiplexing) systems, which have multiple subcarriers operating on different frequencies. Channel information can be classified as CSI (Channel State Information) in frequency domain and CIR (Channel Impulse Response) in time domain. CSI provides channel information with amplitude and phase over multiple subcarriers. CIR characterizes amplitude and phase information over multiple propagation paths in time domain. CIR can be converted to CSI by Fast Fourier Transform (FFT) and CSI to CIR by Inverse FFT (IFFT).

In contrast to RSSI as a coarse-grained MAC layer radio parameter, CSI and CIR are fine-grained physical layer radio parameters, which can be considered as fine-grained power information but are able to characterize the signals from multipath propagation. However, as mentioned in several works [132, 126], CIR can just distinguish clusters of propagation paths instead of each individual paths. Therefore, CIR can only mitigate multipath propagation but not completely eliminate the effects. Additionally, it has very limited improvement on shadow fading in Non Line Of Sight (NLOS) condition. Therefore, even if CIR as a fine-grained information is used for indoor positioning, it still has remaining ranging errors. Therefore, in this thesis, **we work on an accurate model for ranging by using channel information and robust positioning algorithms to mitigate the remaining ranging errors**.

Additionally, **how to extract CSI information is another problem for indoor positioning with this fine-grained information**. Most of the work using CSI or CIR for indoor positioning relies on a special WiFi card (IWL5300), which extracts CSI based on an improved firmware provided by the authors of [60]. There are sev-

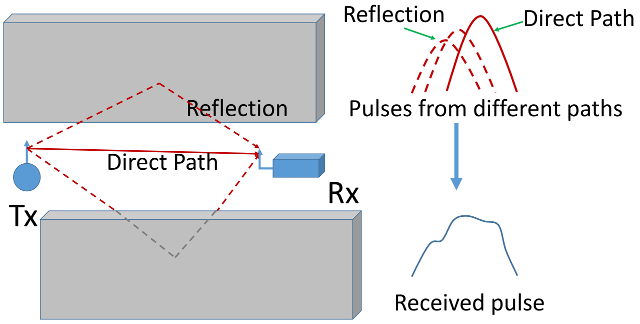


Figure 1.3: Influence of Multipath on Timestamps

eral limitations to use the firmware such as the resolution of channel information, working mode and no CSI from overheard packets. Therefore, to design a passive indoor positioning system by using CSI information from overheard packets, we can not rely on this WiFi card and other commercial WiFi cards.

1.2.3 Time-based Indoor Positioning

Physical-layer time information is another fine-grained information, which can be used for ranging. Time-based indoor positioning is challenging mainly due to two aspects: synchronization between ANs and timestamp accuracy. Due to the high speed of radio signals, i.e., $3 \cdot 10^8 m/s$, a small synchronization offset between ANs generates large positioning errors. It is nontrival to achieve synchronization accuracy between ANs in the range of nanoseconds, especially for large scale deployments where shared cables between ANs for synchronization are normally infeasible. Therefore, it is important for time-based indoor positioning to find **a reliable synchronization solution between ANs, quantify the synchronization offset, and compensate the offset**.

The accuracy of timestamps is influenced by two factors: bandwidth of signals and multipath propagation. At physical layer, narrower bandwidth leads to longer duration of a signal's symbol, which is defined as a pulse in digital baseband transmission. A timestamp is given at the time when a reference symbol in a received packet is received. Benefiting from its extremely short symbol duration, UWB signals achieve timestamps with nanosecond accuracy. However, for signals with narrower bandwidth, **how to distinguish the time delay within one symbol duration** is a challenge. For example, the bandwidth of a GSM signal is 200KHz and the symbol duration is $5 \mu s$. Therefore, a solution should be designed to estimate the time delay within $5 \mu s$. Additionally, replicas of the original signal from different propagation paths arrive at the receiver with different delays. Due to the long symbol duration, the waveforms of original signal from the direct path and replicas from reflection paths overlap with each other within one symbol duration. The timestamp for the signal from the direct path is required to calculate the range based on the propagation delay. However, as shown in Figure 1.3, because of the overlap between the signals from the direct path and reflection paths, it is very challenging to distinguish them and generate a timestamp on the direct path. Therefore, multipath propagation severely influences the ranging accuracy using time information, especially for narrow-band signals. Hence, **a positioning algorithm needs to be designed to use this time information for narrow-band signals to locate the user**.

1.2.4 Challenges for Passive and Active Indoor Positioning

Passive indoor positioning systems are attractive for third-party providers of positioning and monitoring services. In contrast to active positioning, which can leverage all the available information such as inertial sensors in smart phones, one of the critical challenges for passive positioning is the limited amount of available information. Since the targets or users do not actively participate in the positioning process, inertial sensor information measured at the terminal side (users) can not be transmitted to the AN sides. It can only utilize the measured radio parameters, such as RSSI, time information and channel information, and hence passive positioning systems rely on the radio-based indoor positioning. Therefore, **how to efficiently use these limited radio parameters to accurately locate the users** is a problem in passive positioning systems.

Active indoor positioning systems can leverage more information besides radio parameters, such as inertial sensors. As mentioned before, radio based indoor positioning provides absolute locations in a local coordinate system and PDR algorithms give relative locations between sequential time intervals. These two methods are complementary and it is intuitive to fuse these information for indoor positioning and tracking. Therefore, **how to fuse these available information including inertial sensors and radio parameters to accurately track the users** is a problem that we aim to address in this thesis for active positioning systems.

1.3 Thesis Contributions

The main goal of this thesis is to provide indoor positioning and tracking solutions with high accuracy and low calibration efforts (ease of deployment) by using fine-grained physical layer information of radio signals. First, this thesis mainly contributes to passive indoor positioning with two fine-grained radio parameters, i.e., time information and channel information. Second, we also contribute to active indoor positioning by fusing the inertial sensor information and channel information. The contributions in this thesis are supported by theoretical analysis and evaluated in real-world prototype systems. All our positioning algorithms and systems are designed and implemented at the network side. Therefore, our work has potential to be used by third-part providers for location based services or network operators for network planning. For example, parts of our proposed algorithms have been transferred to our Swiss industry partner (DFRC AG) in the Eurostar project (In3DGuide). They adopt our proposed physical layer timestamps (in Section 1.3.1) in their prototype system to analyze the activities of GSM users.

1.3.1 Time-based Indoor Positioning

The first contribution in this thesis is in the area of time-based passive indoor positioning with narrow-band signals [79, 80, 82, 81]. We propose several novel methods to address the aforementioned challenges in time-based positioning: evaluating and compensating synchronization offsets, designing high-resolution timestamps, and developing positioning algorithms. Details are listed below:

- **GPS synchronization** between ANs is evaluated based on the timing error in the signal processing procedure. First, we calculate the clock offsets between two GPS synchronized ANs using a signal processing method called time recovery. Second, to estimate the clock skews from the noisy clock offset measurements, we propose to adopt Savitzky-Golay (S-G) filter to smooth the noisy measured clock offsets and then accurately calculate the clock skews by differentiating the filtered clock offsets. The main findings in this work are that the maximum clock offset between two GPS synchronized devices reaches $171ns$ but the clock skew is smaller than $1.37 \cdot 10^{-9}$. The large synchronization offsets lead to large positioning errors with a traditional Time Difference Of Arrival (TDOA) method.

- **Synchronization compensation** is achieved in this work by combining GPS synchronization and an enhanced TDOA method called Differential TDOA (DTDOA). DTDOA introduces a Reference Node (RN) to the TDOA technique to compensate the momentary synchronization offsets and eventually improves the positioning accuracy. We offer an analytical model to formally describe the time components of TDOA and the proposed DTDOA method.

- **High-resolution physical-layer timestamps** are designed for two kinds of narrow-band signals, i.e., GSM and IEEE 802.15.4 signals, inspired by information from signal processing at the physical layer. The timestamp is taken with a resolution of nanoseconds.

- **Two novel positioning algorithms** based on DTDOA for narrow-band signals are proposed in this work. The first one adds to range-based positioning algorithms and employs a Linear Least Square (LLS) algorithm for DTDOA by introducing a new intermediate parameter. The second one (DTDOA-based fingerprinting) adds to range-free positioning and it is the first to propose and demonstrate the feasibility of DTDOA-based fingerprinting in indoor environments.

- **Two systems for GSM and IEEE 802.15.4 signals** have been designed using Software Defined Radio (SDR) techniques to extract the high-resolution physical-layer timestamps and locate the target with time information. Based on these two systems, we have evaluated the proposed algorithms including DTDOA-based LLS and fingerprinting. We are able to achieve around $10m$ ranging accuracy for GSM signals in an outdoor environment with weak multipath effects. Although limited by its narrow bandwidth with IEEE 802.15.4 signals, DTDOA-based LLS is still challenging to achieve accurate indoor positioning, the ranging and positioning accuracy is clearly improved by using DTDOA compared to a traditional TDOA. Furthermore, we compare the performance of DTDOA-based fingerprinting and RSS-based fingerprinting in different scenarios. The measurement results show that DTDOA-based fingerprinting achieves quite similar performance as RSS-based fingerprinting but in NLOS conditions DTDOA-based fingerprinting performs better than RSS-based fingerprinting. DTDOA-based fingerprinting achieves a mean accuracy of $3m$.

1.3.2 Indoor Positioning for Stationary Targets using Fine-Grained Power

The second contribution [78] is in the area of passive indoor positioning for IEEE 802.11n signals by using fine-grained power (channel information). Details are listed below:

- **A passive positioning system for WiFi targets**, which can extract channel information from the overheard packets to design some novel methods for ranging and positioning, is the first contribution in this work. As mentioned before, the currently used WiFi card IWL5300 for CSI extraction can not be used for passive positioning. We propose to use SDR techniques to design the WiFi signal sniffers, which act as ANs in the positioning procedures. By decoding the WiFi packets from the physical layer, we estimate the CSI and CIR by using the long preambles in WiFi packets.

- **Ranging by using CIR** is achieved by two steps. First, we mitigate the multipath propagation by extracting the strongest power in CIR as the power from direct path. Second, a novel Nonlinear Regression (NLR) method for ranging is proposed to more accurately map the power information to the propagation distance in complex indoor environments.

- **A new trilateration approach**, joining the Weighted Centroid (WC) and Constrained Weighted Least Square (CWLS) algorithms, is proposed to mitigate the impact of ranging errors.

We evaluate our system in a comprehensive set of measurements including stationary and mobile targets under complex indoor propagation conditions. Experimental results demonstrate that our proposed NLR model achieves higher ranging accuracy than the LDPL model. Furthermore, the WC-CWLS positioning algorithm achieves higher accuracy and is more robust to ranging errors than the LLS and WC algorithms. By combining the NLR model for ranging and the WC-CWLS algorithm for trilateration, the mean positioning accuracy of the system achieves $2.4m$.

1.3.3 Indoor Tracking for Mobile Targets using Fine-Grained Power

The third contribution [77] is to extend the aforementioned passive positioning system (IEEE 802.11n) to support tracking a mobile target by using the enhanced ranging method and designing an enhanced particle filter. Details are listed below:

- **An enhanced particle filter** for indoor tracking by improving the likelihood function on the observation parameters and introducing a single coordinated turn model is proposed in this work. For particle filters, we have three main scientific contributions. First, instead of using a constant velocity moving model as in most indoor tracking works, we propose to adopt a single coordinated turn model, which considers the angle variation of the moving direction in the movement state and provides higher tracking accuracy for a passive tracking system. Second, we investigate the impact of ranging errors on the likelihood function in the particle filter and the relation between ranging outputs and ranging errors. By weighting the likelihoods of ranging outputs from different ANs, our particle filter mitigates the influence of ranging errors. Third, in a passive positioning system, speed information is normally unavailable to the tracking process because the system can not get the inertial sensor information from the target. In our system, we consider the moving speed limitation on the likelihood by filtering out the uncommonly large moving speed for people in indoor environments.
- **The enhanced ranging methods** based on CIR for multipath mitigation and the NLR model for mapping the power to propagation distance (in Section 1.3.2) are also adopted in this work. Considering the movement of the target, we smooth the sequentially measured fine-grained power by a Savitzky-Golay (S-G) filter, which considers the trend of power changes in the moving window, to further mitigate the multipath effects.

We evaluate our proposed enhanced particle filter in complex indoor environments and compare it to some other commonly used positioning and tracking algorithms. Our proposed enhanced particle filter outperforms the others, such as Kalman filter and trilateration algorithms. Our system passively tracks the WiFi target with an accuracy of $1.5m$ for 50% and $2.3m$ for 90%.

1.3.4 Fusing Inertial Sensor Information and Fine-Grained Power Information for Indoor Tracking

Finally, we contribute to active indoor positioning by fusing inertial sensor information and fine-grained power information with a second enhanced particle filter [76].

- **An enhanced particle filter** to fuse the CIR-based ranging and velocity information is proposed in this work. The two input parameters, i.e., ranges and velocity, are fused in the likelihood function of observations in particle filters. These two parameters are adopted to filter out the particles with low probabilities to observe the measured ranges and velocity.

- **MIMO (Multiple Input and Multiple Output) techniques**, which introduce space diversity into the communication systems by using more than one antennas at the transmitter and receiver, are adopted in this work to mitigate the multipath effects in the ranging step. The aforementioned enhanced ranging methods with CIR and the NLR model (in Section 1.3.2) are adopted to estimate the range information from each receiving antenna. To utilize the spatial diversity between different antennas, we average the estimated ranges from multiple antennas to mitigate the influence of multipath effects.

- **An efficient method to estimate the velocity of the mobile target** is proposed in this work to use the timestamped values from the accelerometer and compass sensors in a smart phone.

- **A network-based positioning system**, which runs our proposed tracking algorithms in a central server, is implemented. Compared to terminal-based positioning system, a network-based positioning system is able to run algorithms with high complexity. In our system, all ANs are implemented on cheap commercial devices and are able to collect inertial sensor and CSI information from the received WiFi packets.

We evaluate our system in a complex environment along three different moving paths. Our proposed tracking method achieves 1.3m for mean accuracy and 2.2m for 90% accuracy, which is more accurate and stable than PDR and range-based positioning methods.

1.4 Thesis Outline

This thesis is structured as follows. In Chapter 2, we describe the background knowledge of this thesis. We review some basic knowledge for GSM, IEEE 802.15.4, IEEE 802.11, and their decoding methods in software defined radio techniques. We introduce the background and related work in the area of indoor positioning. Then, our main contributions are structured in two parts. Part I (Chapters 3 and 4) introduces our work on time-based indoor positioning for narrow-band signals. Part II (Chapters 5, 6, and 7) introduces our work on indoor positioning and tracking with fine-grained power for WiFi signals.

In Chapter 3, we investigate the methods for synchronization evaluation and high-resolution timestamps. Based on the findings in Chapter 3, we introduce our proposed DTDOA algorithms including LLS and fingerprinting in Chapter 4.

Chapter 5 describes the passive indoor positioning system for WiFi signals using fine-grained power (channel information) and mainly targets in trilateration algorithm for stationary targets. In Chapter 6, we introduce an enhanced particle filter to extend the passive indoor positioning system for WiFi signals to support tracking mobile targets. Chapter 7 describes a second enhanced particle filter to fuse the inertial sensor and fine-grained power information for an active positioning system.

Finally, Chapter 8 concludes the thesis by summarizing the contributions of this work, and discusses interesting and promising future directions of research.

Chapter 2

Background and Related Work

2.1 Introduction

As introduced in Chapter 1, indoor positioning has become an emerging technique to support the increasing demands for location based services, which provide new commercial opportunities based on the locations of users. Intensive research has been conducted in this area in the past few years. In this chapter, we review background and related work for indoor positioning including radio-based and inertial sensor based methods.

In the remainder of this chapter, some knowledge of wireless communications, which is used in our work including GSM, IEEE 802.15.4 and IEEE 802.11, is introduced in Section 2.2. Software defined radio techniques are presented in Section 2.3. In this section, the decoding methods in SDR for IEEE 802.15.4 signals, IEEE 802.11 signals, and GMSK modulated signals that are used in GSM are also reviewed. In Section 2.4, range-free positioning algorithms including fingerprinting and centroid algorithms are introduced. Ranging based on different radio parameters, which is the first step for range-based positioning algorithms, is reviewed in Section 2.5. Trilatration and multilatration algorithms for stationary targets are reviewed in Section 2.6. Bayesian estimation including Kalman filters and particle filters for tracking mobile targets are respectively introduced in Sections 2.7 and 2.8. Section 2.9 presents some related work for indoor tracking based on inertial sensors. In Section 2.10, we review some work to combine the radio-based and inertial sensor based tracking. Finally, Section 2.11 concludes this chapter.

2.2 Wireless Communication Technologies

We first review some knowledge about the wireless communication technologies related to our work, especially for physical layer signal processing.

2.2.1 GSM

Cellular networks provide radio coverage over a wide geographic area by distributing large number of based stations over land areas. The providers of cellular networks provide voice and data services to the end users. At the end of the past century, the second generation (2G) [7] of cellular network, i.e., GSM (Global System for Mobile Communication), was already widely spread [115]. It is mainly used to provide voice communications for mobile users. With the increasing demands for mobile data communication, the third generation (3G) [10] of cellular network provides the additional transmission of non-voice data such as UMTS (Universal Mobile Telecommunications System). Currently, the fourth generation (4G) [12], e.g., LTE (Long Term Evolution), has become the most emerging technique to support higher data traffic for mobile end users.

Frequency Band

Despite the vast development in 3G and 4G networks for data communication, GSM remains the world's most widespread mobile communication standard, especially for less developed countries. The 3rd Generation Partnership Project (3GPP) [2] specifies the GSM frequency bands of 900MHz and 1800MHz as the operating bands in Europe and most of other parts of the world. Since GSM is used for voice communication, whose data rate is only 270.833 kbit/s, the bandwidth of GSM is very narrow. For example, the E-GSM 900 uplink band ranges from 880MHz to 915 MHz. It has 124 channels and each one is only 200kHz wide.

GMSK and MSK Modulation Schemes

GSM adopts a modulation scheme called Gaussian Minimum Shift Keying (GMSK), which is a frequency shift keying modulation scheme with continuous phase. GMSK is a modified version of MSK by using a Gaussian filter before the modulation stage [96]. Figure 2.1 shows a general block diagram of the MSK and GMSK modulators. Bits $(1, 0)$ are converted to Non-Return to Zero (NRZ) $(-1, 1)$ data waveform. Then NRZ data passes a filter for pulse shaping to limit the bandwidth of the signal. A filter with a half-sine pulse shape (Figure 2.2(b)) is used for MSK and a Gaussian pulse shape (Figure 2.2(a)) for GMSK. Compared to MSK, Gaussian pulse shaping in GMSK smooths the trajectory of the MSK signals and hence stabilizes instantaneous frequency variations over time. Then the filtered data is sent to a FM (Frequency Modulation) modulator. The resulting signal is

$$s(t) = e^{j2\pi h \sum_{i=-\infty}^{\infty} a_i q(t-iT)}. \tag{2.1}$$

In this expression, the amplitude of the signal is normalized to one. a_i is the NRZ data and h is the modulation index, which is equal to 0.5 for MSK and GMSK. $q(t)$ is given by $q(t) = \int_{-\infty}^{t} g(\tau)d\tau$, where $g(t)$ is the pulse shape filter as in Figure 2.2(a) and 2.2(b).

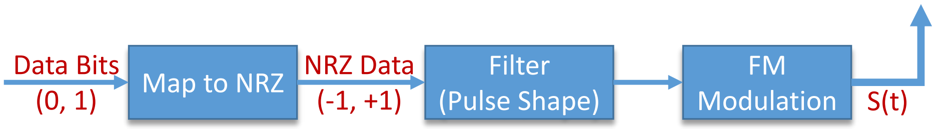
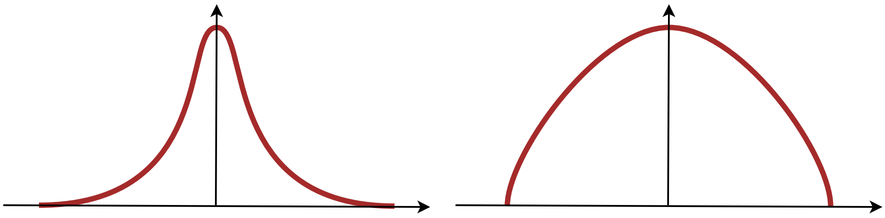


Figure 2.1: MSK Modulation



(a) Gaussian Shape (b) Half-sine Shape

Figure 2.2: Pulse Shapes for GMSK and MSK

At the receiver side, there are different ways to decode a GMSK/MSK signal. A commonly used and efficiently implementable method is N-bits differential detection [128]. Figure 2.3 shows an example of 1-bit differential detection. In 1-bit differential detection, the phase difference between two sequential incoming signals is first calculated as

$$\Delta\varphi(t) = \arg\{r(t)r^*(t-T)\}, \tag{2.2}$$

where $r(t)$ is received signals. This differential phase is sampled at time $t = nT$ and the differential phase at time nT is given by

$$\Delta\varphi(nT) = 2\pi h \int_{t=(n-1)T}^{nT} \sum_{i=-\infty}^{+\infty} a_i g(t-iT) + \Delta\varphi_{\text{noise}}, \tag{2.3}$$

where $\Delta\varphi_{\text{noise}}$ is the additive noise. For MSK, the phase difference only depends on the current bit as

$$\Delta\varphi(nT) = \frac{\pi}{2}a_n + \Delta\varphi_{\text{noise}}. \tag{2.4}$$

For GMSK, the phase difference depends on three consecutive bits as

$$\Delta\varphi(nT) = \frac{\pi}{2}(\alpha_{-1}a_{n-1} + \alpha_0 a_n + \alpha_{+1}a_{n+1}) + \Delta\varphi_{\text{noise}}. \tag{2.5}$$

The three parameters α_{-1}, α_0 and α_{+1} depends on the multiplication of the Gaussian filter's 3dB bandwidth (B) by the bit period of the transmission (T), i.e., BT.

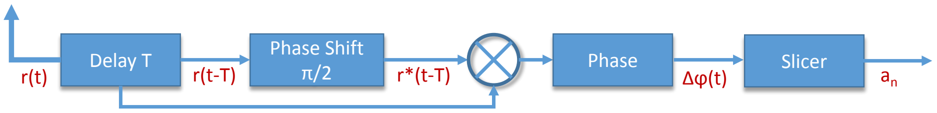


Figure 2.3: MSK Demodulation

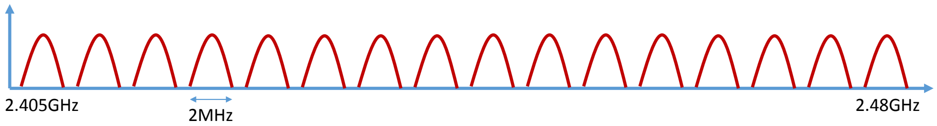


Figure 2.4: IEEE 802.15.4 Operating Band

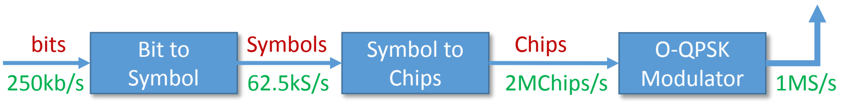


Figure 2.5: IEEE 802.15.4 Modulation and Spreading

For $BT = 0.25$, $\alpha_0 = 0.589$ and $\alpha_{-1} = \alpha_{+1} = 0.205$ [73]. Therefore, the Gaussian filter introduces inter symbol interference. This phase difference is fed into a slicer for decision.

2.2.2 IEEE 802.15.4

The IEEE 802.15.4 standard [11] specifies the physical and MAC layers for low-data rate and low-power applications. It is widely used in home automation [61], wireless sensor network [102] and industry applications [100]. The IEEE 802.15.4 standard is the basis for ZigBee [120] and WirelessHART [34], each of which extends the IEEE 802.15.4 standard by developing the upper layers. IEEE 802.15.4 defines two physical layers including 2.4 GHz and 868/915MHz physical layers. Since 2.4GHz is used worldwide and also adopted in our work, this section only covers the physical layer and some parts of the MAC layer on the 2.4GHz band.

Modulation Schemes and Bandwidth

A total of 16 channels are available in the 2.4GHz band and each channel occupies a bandwidth of 2MHz as shown in Figure 2.4. Data can be transmitted on one of these channels or hopping on different channels [101].

Figure 2.5 shows the block diagram of the modulation and spreading on the 2.4GHz band in a transmitter. Bits from the Physical Protocol Data Unit (PPDU) with a data rate of 250 kbit/s are converted to data symbols of 4 bits. Then, these data symbols get spread according to a set of spreading sequences as in Table 2.1, which generates a stream of chips at 2 MChip/s. This mechanism is called Direct Sequence Spread Spectrum (DSSS) [62]. The output stream is modulated with an Offset-Quadrature Phase Shift Keying (O-QPSK) with half-sine pulse shape [11].

A traditional O-QPSK is a phase modulation scheme with a rectangular pulse shape [103]. If O-QPSK is equipped with a half-sine pulse shape, it is equivalent to MSK [109]. Therefore, we can use the demodulation mechanism for MSK to demodulate the O-QPSK signals, such as one-bit differential detector as in Figure 2.3. The spreading sequences in Table 2.2 is used for MSK decoding (O-QPSK with a half-sine pulse shape) [109].

Table 2.1: Spreading Sequence for O-QPSK in IEEE 802.15.4 [109]

Symbol	Chip Sequence
0	11011001110000110101001000101110
1	11101101100111000011010100100010
2	00101110110110011100001101010010
3	00100010111011011001110000110101
4	01010010001011101101100111000011
5	00110101001000101110110110011100
6	11000011010100100010111011011001
7	10011100001101010010001011101101
8	10001100100101100000011101111011
9	10111000110010010110000001110111
10	01111011100011001001011000000111
11	01110111101110001100100101100000
12	00000111011110111000110010010110
13	01100000011101111011100011001001
14	10010110000001110111101110001100
15	11001001011000000111011110111000

Table 2.2: Spreading Sequence for MSK in IEEE 802.15.4 [109]

Symbol	Chip Sequence
0	x1100000011101111010111001101100
1	x1001110000001110111101011100110
2	x1101100111000000111011110101110
3	x1100110110011100000011101111010
4	x0101110011011001110000001110111
5	x1111010111001101100111000000111
6	x1110111101011100110110011100000
7	x0000111011110101110011011001110
8	x0011111100010000101000110010011
9	x0110001111110001000010100011001
10	x0010011000111111000100001010001
11	x0011001001100011111100010000101
12	x1010001100100110001111110001000
13	x0000101000110010011000111111000
14	x0001000010100011001001100011111
15	x1111000100001010001100100110001

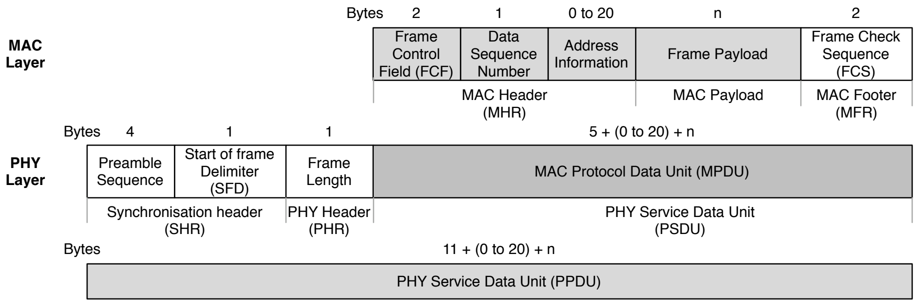


Figure 2.6: IEEE 802.15.4 Frame Format

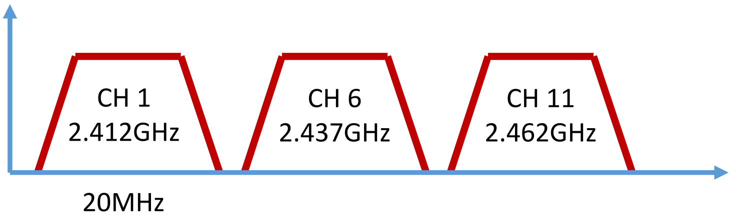


Figure 2.7: IEEE 802.11n Channels

Frame Structure

As shown in Figure 2.6, a PPDU frame includes the preamble sequence, the Start of Frame Delimiter (SFD), the frame length and the MAC Protocol Data Unit (MPDU). The preamble sequence consists of four bytes 0x00, which is used to synchronize to the incoming signal. SFD is defined as one bye of 0xA7, which is used to indicate the end of the preamble. The frame length indicates the length of MPDU.

At MAC layer, a MPDU consists of the Frame Control Field (FCF), the data sequence number, the address information, the frame payload, and the Frame Check Sequence (FCS). FCF indicates the type of received packet and the address information. FCS is the CRC-CCITT 16-bit checksum of the MPDU. It uses the polygon $x^{16} + x^{12} + x^5 + 1$, which is equivalent to the hex number 0x1021. Please find more details about the individual parts in the IEEE 802.15.4 standard [11].

2.2.3 IEEE 802.11

The IEEE 802.11 standard [13] specifies the physical and MAC layers for Wireless Local Area Network (WLAN). IEEE 802.11b [8] is an early version of IEEE 802.11, which was released in 1999. IEEE 802.11b adopts DSSS at 2.4GHz and has a maximum data rate of 11 Mbit/s. IEEE 802.11a and IEEE 802.11g are two versions with OFDM, which can achieve a maximum data rate of 54Mbps. IEEE 802.11a [9] operates on 5GHz and IEEE 802.11g [11] on 2.4GHz. Each channel occupies 20MHz bandwidth as shown in Figure 2.7. IEEE 802.11n was released in 2009, which operates on 2.4GHz and 5GHz. IEEE 802.11n includes many en-

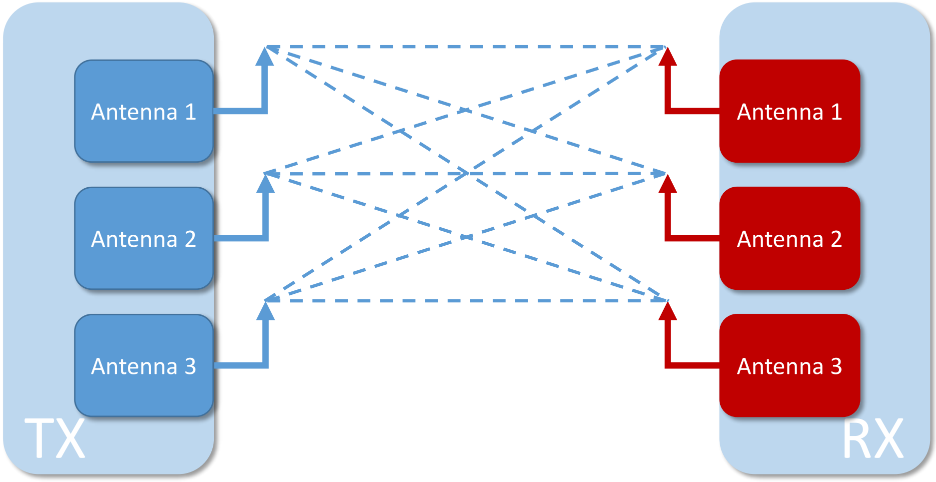


Figure 2.8: 3×3 MIMO

hancements that improve WLAN range, reliability, and throughput. At the physical (PHY) layer, advanced signal processing and modulation techniques have been added to exploit multiple antennas and wider channels (20MHz and 40MHz). At the MAC layer, protocol extensions make more efficient use of available bandwidth. These High Throughput (HT) enhancements can boost data rates up to 600 Mbps. Since we work on IEEE 802.11n, we review some relevant knowledge in IEEE 802.11n in the following subsections.

Physical Layer in IEEE 802.11n

At physical layer, there are two main features in IEEE 802.11n: OFDM and MIMO. OFDM is already used in IEEE 802.11a/g. MIMO was first introduced to IEEE 802.11 standards in the version of IEEE 802.11n.

MIMO is an important physical layer technique in the IEEE 802.11n standard, which uses multiple antennas at transmitter and receiver to increase data throughput and to improve the robustness of the transmission [39]. MIMO transmits different signals over the individual antennas. Because different antennas face different propagation channels, MIMO introduces space diversity to the system. The signals transmitted over different antennas can adopt different coding, delay, and phase control during processing. By adopting these advanced techniques in MIMO, the data rate is increased without introducing more bandwidth. For example, Figure 2.8 shows a 3×3 MIMO system. Each spatial stream in IEEE 802.11n provides a maximum rate of 150 Mbps at 40MHz bandwidth. 3×3 MIMO provides 3 data streams and thus the maximum rate can achieve 450 Mbps [39].

OFDM [39] is adopted at the physical layer in the IEEE 802.11n standard to achieve high data rates. In an OFDM system, the available frequency spectrum is divided into several subcarriers, which are orthogonal to each other as shown in Figure 2.9. In contrast to signal carrier systems such as DSSS, OFDM intro-

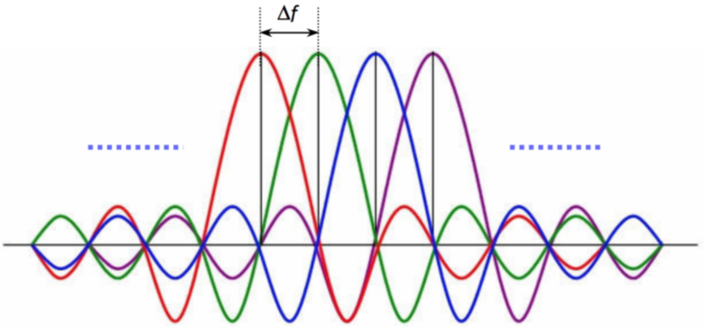


Figure 2.9: OFDM Subcarriers

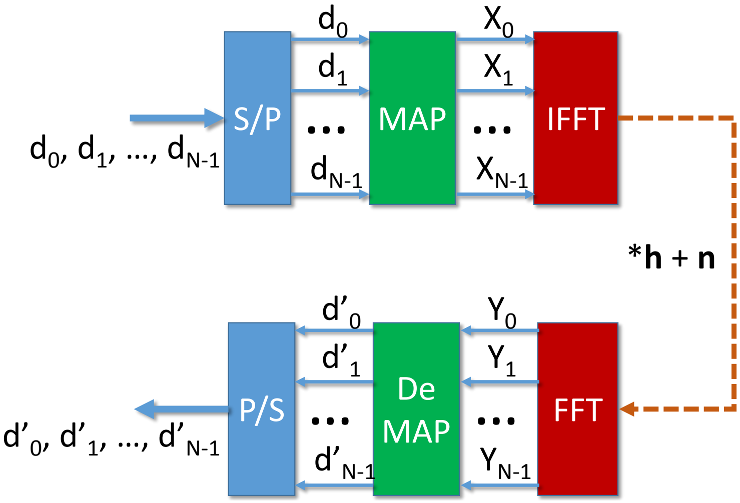


Figure 2.10: OFDM Structure

duces frequency diversity to the communication system because data can be simultaneously sent over multiple frequencies. OFDM has been widely used in some modern wireless communication systems, such as LTE cellular network, IEEE 802.11a/g/n/ac network and Digital Video Broadcasting (DVB) [42] systems.

As shown in Figure 2.10, on the OFDM transmitter side, the data $[d_0, d_1, \cdots, d_{N-1}]$ is first converted from a serial form to a parallel form by S/P (Serial to Parallel conversion) and modulated (MAP) onto subcarriers in the frequency domain. The data is further converted from the frequency domain to the time domain via an IFFT. Upon receiving the signals, a FFT procedure converts the data back to the frequency domain. The data is demodulated (DeMAP) on each subcarrier and converted to serial by P/S (Parallel to Serial conversion).

Let $\mathbf{X} = [X_k]^T$ and $\mathbf{Y} = [Y_k]^T$ $(k = 0, ..., N-1)$ respectively denote the input data of the IFFT block at the transmitter and the output data of FFT block at the receiver in the frequency domain. Let $\mathbf{h} = [h_n]^T$ and $\mathbf{n} = [n_n]^T$ $(n = 0, ..., N-1)$ denote the sampled channel impulse response and Gaussian noise in the time domain. Define the input matrix $\bar{\mathbf{X}} = \text{diag}(\mathbf{X})$ and the received signal in

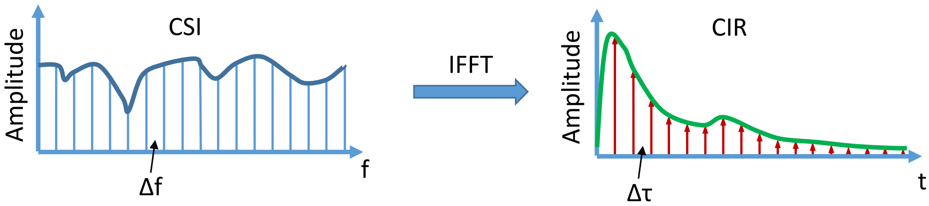


Figure 2.11: CSI and CIR

the frequency domain is derived as

$$\mathbf{Y} = \bar{\mathbf{X}} \cdot \mathrm{FFT}(\mathbf{h}) + \mathrm{FFT}(\mathbf{n}) = \bar{\mathbf{X}}\mathbf{H} + \mathbf{N}, \tag{2.6}$$

where $\mathbf{H} = \mathrm{FFT}(\mathbf{h})$ is the channel state information and $\mathbf{N} = \mathrm{FFT}(\mathbf{n})$ is the noise in the frequency domain. The equation demonstrates that an OFDM system is equivalent to a transmission of data over a set of parallel channels. Channel estimation is important in an OFDM system to achieve high data rates and reliable communication. Channel information is classified as CSI in frequency domain and CIR in time domain. In frequency domain, CSI provides channel information with amplitude and phase over multiple subchannels. As shown in Figure 2.11, the constructive and destructive phases of signals from multiple propagation paths cause frequency-selective fading, i.e., different attenuations in different frequency subcarriers. $\mathbf{H}$ in Equation (2.6) is the CSI information in frequency domain.

In time domain, CIR characterizes amplitude and phase information over multiple propagation paths. CIR is modelled as a temporal linear filter as

$$h(\tau) = \sum_{n=1}^{N} a_n e^{-j\theta_n} \delta(\tau - \tau_n), \tag{2.7}$$

where a_n, θ_n and τ_n are the amplitude, phase and time delay of the nth path. N is the total number of paths and $\delta(\tau)$ is the Dirac delta function. $\mathbf{h}$ in Equation (2.6) is the CIR information in time domain. CIR is converted to CSI by FFT and CSI to CIR by IFFT.

Two basic channel estimation mechanisms are commonly used, i.e., block-type pilot and comb-type pilot channel estimation [39]. Pilot means training symbols with known values at the receiver side. As shown in Figure 2.12, block-type pilot is performed by inserting pilots into all subcarriers of OFDM symbols within a specific period. The comb-type pilot channel estimation inserts pilot tones into certain subcarriers of each OFDM symbol, and interpolates to estimate CSI in other subcarriers. The Least Square (LS) and Minimum Mean Square Error (MMSE) techniques [39] are two widely used for channel estimation with pilot symbols.

- The LS channel estimation method estimates the channel by minimizing the following cost function as

$$\hat{\mathbf{H}}_{\mathbf{LS}} = \underset{\hat{\mathbf{H}}_{LS}}{\mathrm{argmin}} \, ||\mathbf{Y} - \bar{\mathbf{X}}\hat{\mathbf{H}}_{\mathbf{LS}}||^2, \tag{2.8}$$

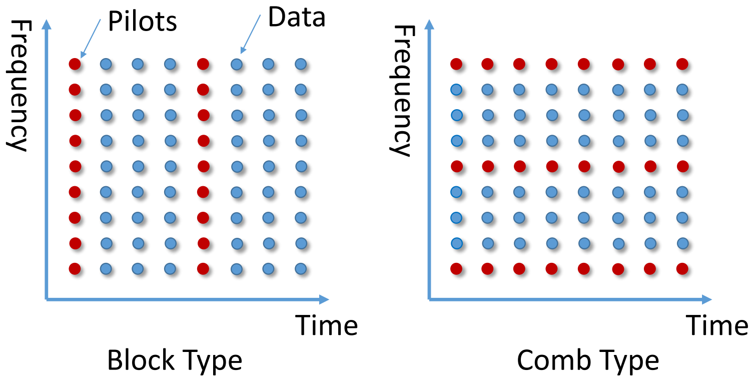


Figure 2.12: Channel Estimation Mechanisms

where $\bar{\mathbf{X}}$ is the transmitted pilot symbols in diagonal matrix and $\mathbf{Y}$ is the received pilot symbols in vector. The solution for this problem is got by setting the derivative of the cost function with respect to $\hat{\mathbf{H}}$ to zero. Finally, the LS solution is got as

$$\hat{\mathbf{H}}_{\mathbf{LS}} = \bar{\mathbf{X}}^{-1}\mathbf{Y}. \tag{2.9}$$

The advantage of this LS solution is its simplicity to implement.

- MMSE further improves the LS solution by introducing a weight matrix $\mathbf{W}$. The MMSE estimation finds the solution as

$$\hat{\mathbf{H}}_{\mathbf{MMSE}} = \underset{\hat{\mathbf{H}}_{\mathbf{MMSE}}}{\operatorname{argmin}} ||\mathbf{H} - \hat{\mathbf{H}}_{\mathbf{MMSE}}||^2, \tag{2.10}$$

where $\hat{\mathbf{H}}_{\mathbf{MMSE}} = \mathbf{W}\hat{\mathbf{H}}_{\mathbf{LS}}$. The MMSE solution (please find more details in [39]) is as

$$\hat{\mathbf{H}}_{\mathbf{MMSE}} = \mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathbf{LS}}}\mathbf{R}^{-1}_{\hat{\mathbf{H}}_{\mathbf{LS}}\hat{\mathbf{H}}_{\mathbf{LS}}}\hat{\mathbf{H}}_{\mathbf{LS}}, \tag{2.11}$$

where $\mathbf{R}_{\mathbf{H}\hat{\mathbf{H}}_{\mathbf{LS}}}$ is the cross-correlation matrix between the true channel vector and LS estimated channel vector. $\mathbf{R}_{\hat{\mathbf{H}}_{\mathbf{LS}}\hat{\mathbf{H}}_{\mathbf{LS}}}$ is the auto-correlation matrix of the LS estimated channel vector. MMSE provides better performance than LS but it requires some prior channel knowledge, such as the correlation matrix of the channel and Signal to Noise Ratio (SNR), which may not be available in practice. Additionally, the implementation complexity is also higher than LS.

Frames in IEEE 802.11n

The IEEE 802.11n standard supports three physical layer models: legacy mode, mixed mode and green field mode [13]. Legacy mode only supports 20MHz bandwidth and supports legacy IEEE 802.11a/g systems. High Throughput (HT) mode is supported in both green field and mixed modes. Green field can only be used

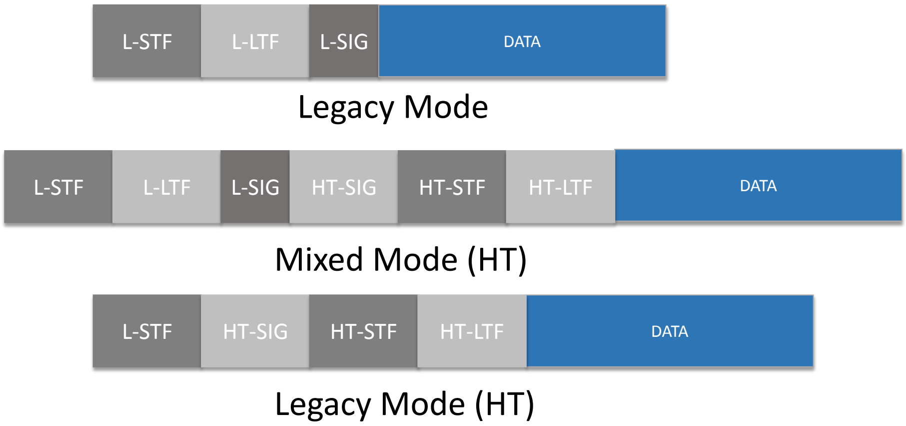


Figure 2.13: IEEE 802.11n Frames

where no legacy system exists. These different modes have different frame format as shown in Figure 2.13. In our work we mainly use the legacy frames, which are reviewed in the remainder of this subsection.

As shown in Figure 2.13, the preamble in the legacy packets consists of L-STF (Legacy Short Training Field) and L-LTF (Legacy Long Training Field). L-STF consists of ten copies of short preambles (also called Short Training Sequences (STS)) and L-LTF consists 2.5 copies long preambles (also called Long Training Sequences (LTS)). L-STF is mainly used for timing acquisition and coarse frequency acquisition. L-LTF is used for channel estimation and fine frequency acquisition. A long preamble consists of 53 subcarriers, which are modulated by the sequence L as

$$L_{-26,26} =$$
$$\{1, 1, -1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 1, 1, -1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 0, 1,$$
$$-1, -1, 1, 1, -1, 1, -1, 1, -1, -1, -1, -1, -1, 1, 1, -1, -1, 1, -1, 1, -1, 1, 1, 1, 1\}$$

These two long preambles in L-LTF are used for block-type pilot channel estimation. Additionally, in IEEE 802.11n, four groups of comb-type pilots are inserted in the subcarriers with the number of -21, -7, 7 and 21.

The preambles are followed by the signal field. This field gives information about the type of modulation and coding rates used in the data field. The encoding of the signal field is based on BPSK modulation and uses convolutional coding at $R = 1/2$. For the data field, please refer to [13] for more details about the modulation and channel coding schemes.

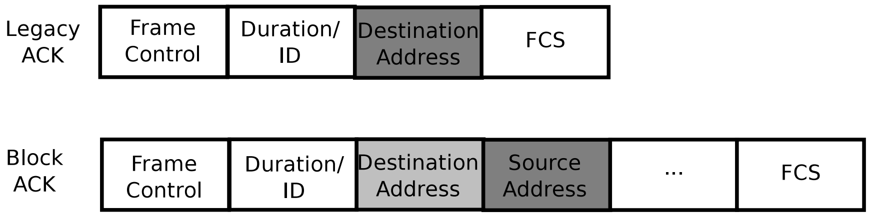


Figure 2.14: ACK Packets in IEEE 802.11n

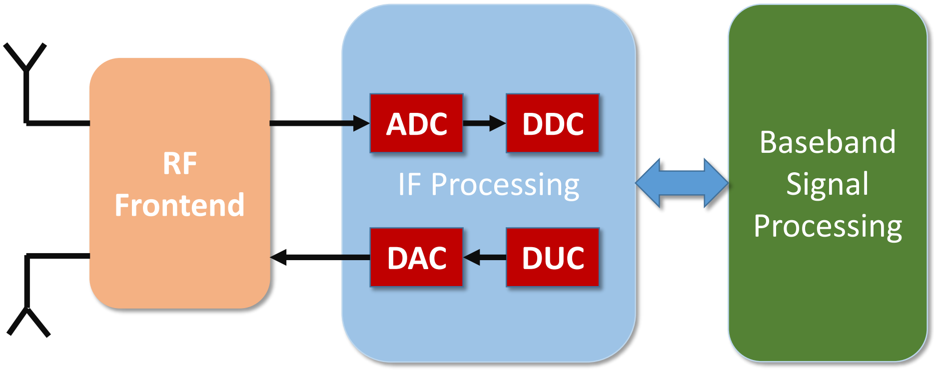


Figure 2.15: A General Structure of Software Defined Radio

ACK Packets in IEEE 802.11n

Acknowledgement (ACK) packets from the mobile users are relevant for the passive positioning system in our work. Therefore, we review ACK packets in IEEE 802.11n. ACK packets are adopted in the IEEE 802.11a/g/n standards to inform a transmitter of the correct reception by a receiver. In IEEE 802.11n, ACK packets that are used in IEEE 802.11a/g are referred to as legacy ACKs. Along with legacy ACKs, IEEE 802.11n introduces block ACKs to acknowledge multiple MAC Protocol Data Units (MPDU) accumulatively by using a single block ACK frame, i.e., Aggregate-MPDU frame aggregation mechanism [13]. Figure 2.14 shows the frame formats of the legacy and block ACKs. In legacy ACKs, the MAC header only includes the destination (receiver) address information, which is the MAC address of the access point if the ACK packets are transmitted from user equipments to the access point, i.e., uplink. Besides destination address, the MAC header in block ACKs includes source (transmitter) address, which is the MAC address of the user equipment in uplink messages.

2.3 Software Defined Radio Systems

Traditional radio devices rely on hardware, which limits cross-functionality and results in inflexibility in supporting multiple waveform standards. For example, an IEEE 802.15.4 receiver like the CC2420 chip [1] can not receive IEEE 802.11 signals even if both signals operate on the same frequency (2.4GHz). Software

Defined Radio (SDR) has been proposed to provide an efficient and inexpensive solution by defining some or all of the physical layer functions as software [111]. Because of the configurable physical layer implementation, SDR techniques allow wireless devices to support multi-mode and multi-band radio signals.

Figure 2.15 shows a general structure of software defined radio system, which normally consists of three components: RF (Radio Frequency) frontend, IF (Intermediate Frequency) processing and baseband signal processing. Take the receiver as an instance. RF-frontend captures analog signals at certain radio frequency (for example, IEEE 802.11a at 5GHz and IEEE 802.11g at 2.4GHz) and convert these radio signals to a lower intermediate frequency. Since this part needs to process the analog signals, it is implemented in hardware circuit but the central radio frequency can be configured by tuning the oscillator. Then the IF analog signals need to be sampled by an ADC (Analog-to-Digital Converter) to digital signals centered at the intermediate frequency. These IF digital signals are converted to baseband complex signals centered at zero frequency by a Digital Down-Converter (DDC). Additionally, DDC is also used to decimate the high sampling rate signal from ADC to a lower sampling rate, which allows the baseband signals to be processed by lower speed processors. Note that DDC or DUC (Digital Up-Converter, which is used in the transmitter to interpolate the low sampling rate baseband signals to the high sampling rate IF signals) [47] are normally implemented in software. Finally, these baseband signals are processed in software.

2.3.1 Hardware Platforms

Nowadays, numerous platforms have been developed for software defined radio. Based on the processors for baseband signals, these platforms are classified as General Purpose Processors (GPP) [5], Digital Signal Processors (DSP) [83], Graphics Processing Units (GPU) [67] and Field-Programmable Gate Array (FPGA) [6].

- **GPPs** have high flexibility because they are not specialized for particular applications. However, GPPs for SDR have low capability for mathematical operations and low energy efficiency.
- **DSPs**, such as Texas Instruments DSP, are specialized for mathematical operations but they are slow for other applications.
- **GPUs** are optimized for vector manipulations and have high capability for signal processing. However, they are difficult to program and have extremely low energy efficiency.
- **FPGAs** are reconfigurable logic devices that enable highly parallel implementations of digital signal processing algorithms. Similar as DSPs, they are slow for the other applications.

WARP [6] from Rice University and USRP [17, 16] from Ettus Research or National Instruments are commonly used SDR platforms. WARP is normally used for research and prototype of wireless networks. The hardware of WARP uses FPGAs for signal processing. In contrast to WARP, USRP relies on GPP for signal processing. Since USRP is used in our research, we introduce more details about these platforms in the reminder of this section.

An USRP device comprises a motherboard and RF daughterboard. The RF daughterboard is a RF frontend in Figure 2.15. Ettus research provides different types of daughterboards for different applications and different frequency bands. For example, the WBX daughterboard [4] supports frequency ranging from 50MHz to 2.2GHz and SBX [3] from 400MHz to 4GHz. Both WBX and SBX support up to 40MHz bandwidth. The motherboard in USRP is used for IF processing and hence it comprises ADC/DAC and DDC/DUC in Figure 2.15. The major computing component on the motherboard is FPGA, which implements DDC and DUC. USRP hardware is available in different categories: such as Gigabit Ethernet networked (USRP N210/200), USB bus (USRP B210/200) and Embedded (USRP E110/100) series. USRP devices are controlled with an open source driver called UHD (USRP Hardware Driver). It supports Linux, MacOS, and Windows platforms. In our work, we use USRP N210 [17] and E110 [16].

USRP N210 [17] is a network based device, in which the baseband samples from the motherboard are sent to a remote PC for signal processing by a Gigabit Ethernet interface with up to 25MS/s (Mega Samples per Second) of 16-bit samples and 50MS/s of 8-bit samples. In the motherboard, it contains two 14-bit ADCs with the sampling rates of 100MS/s and two 16-bit DACs with sampling rates of 400MS/s. FPGA is the key element in the motherboard of USRP N210, which is Xilinx Spartan XC3SD3400A FPGA [14]. FPGA in USRP N210 is controlled by a Reference and System Clock Generation modules (RSCGs), which provides a fixed clock with 100MHz (master clock rate) [17]. RSCG can use its internal clock (oscillator) or lock to an external clock. For example, Ettus research provides a GPSDO (GPS-disciplined oscillator) [15] as an external GPS locked reference oscillator. At the receiver side, DDC in FPGA can only decimate the income signals to a low sampling rate with an integer fraction of the master clock (100MHz).

In contrast to USRP N210, USRP E110 [16] integrates a Texas Instrument embedded processor running a Linux operating system, which does not need to connect to another host machine, i.e., a PC, for signal processing. Therefore, it is ideal for standalone operation as a smart, distributed RF sensor. The drawback of USRP E110 is the limited computation capability because of the low speed embedded processor.

2.3.2 GNU Radio for Different Signals

GNU Radio [5] is a well known SDR software system based on GPP. GNU Radio uses a combination of C++ and Python. The processing blocks requiring high computation are implemented in C++. Python is used to control and coordinate these blocks. GNU Radio combined with USRP provides a runtime environment for signal processing.

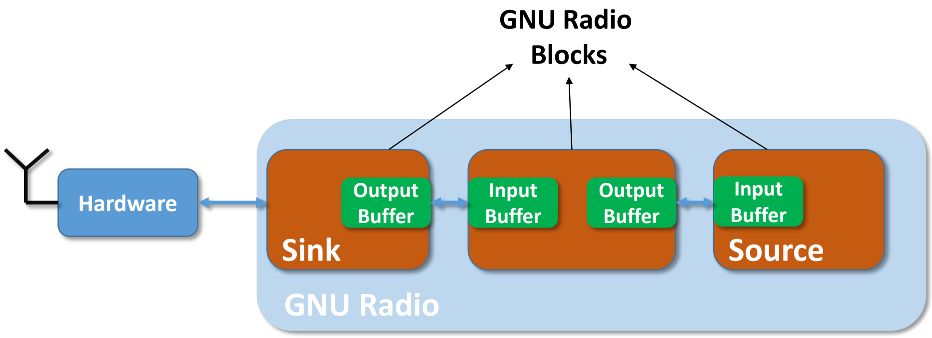


Figure 2.16: GNU Radio Flow Graph

A complete system with GNU Radio is a flow graph [5], which is a combination of connected GNU Radio blocks as shown in Figure 2.16. A GNU Radio block implements various data/signal processing functions, such as a filter or a modulator. Briefly, each GNU Radio block receives one or multiple input sample streams, processes them, and generates one or multiple output sample streams. Blocks in a flow graph are connected by buffers [5]. As shown in Figure 2.16, each GNU Radio block has two buffers: input and output buffers. Output buffer is allocated to hold samples generated by this block. Input buffer is used to read and store the input sample stream. The structure of stream-based signal processing in GNU Radio works well for samples but has drawbacks to pass the control and meta data from one block to another. In GNU Radio, stream tags [5] have been introduced to attached the sample stream with meta/control data, like sampling frequency or timestamps, as an isosynchronous data stream parallel to the main data stream.

In the remainder of this section, we introduce wireless communication encoding or decoding systems based on GNU Radio, which are related to our work, including GMSK transceiver, IEEE 802.15.4 decoding and IEEE 802.11 decoding.

GMSK Transceiver

As introduced in Section 2.2.1, GMSK modulation and demodulation schemes are used in GSM systems. In GNU Radio, a GMSK digital communication system has been provided based on existing GNU Radio blocks [54]. It can transmit and receive GMSK-modulated packets at different bit rates and center frequency. We modify this GMSK digital communication system to design our GSM-like testbed for evaluating positioning algorithms in Chapters 3 and 4.

Figure 2.17 shows the flow graph of the transmitter and receiver. For the transmitter, it uses the same structure as introduced in Section 2.2.1, which consists of NRZ, Gaussian Filter for pulse shaping and frequency modulation.

On the receiver side, it adopts the one-bit differential detection method as introduced in Section 2.2.1 to decode the GMSK signals. FM demodulation is used to detect the phase difference between two sequential incoming signals as in Equation (2.2). As introduced in Section 2.2.1 the GMSK signals are pulse shaped by

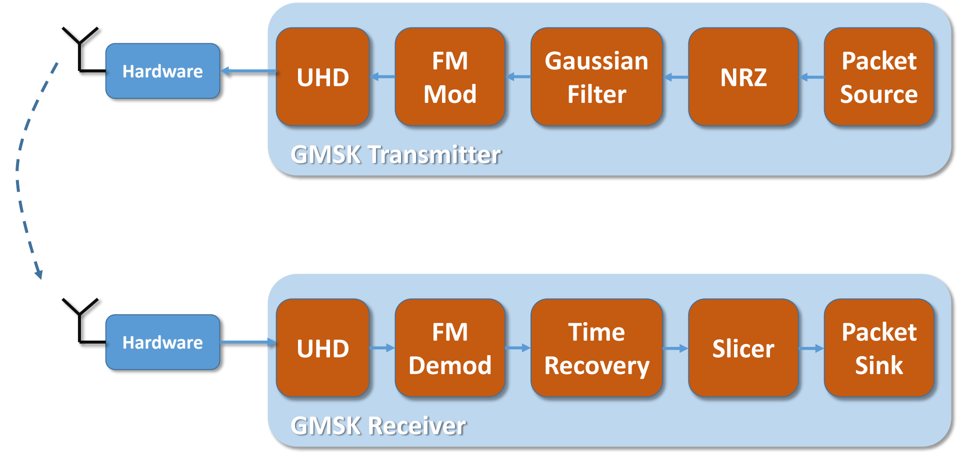


Figure 2.17: GMSK Transceiver

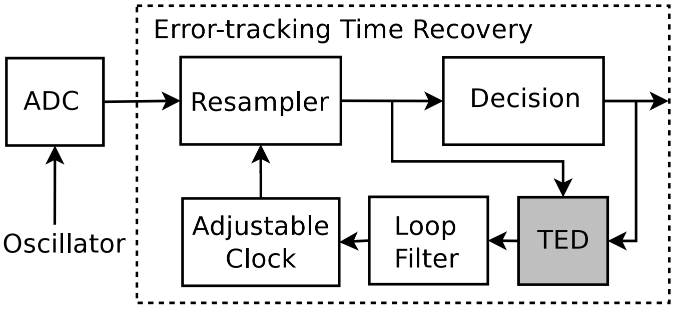


Figure 2.18: Digital Architecture of Time Recovery Loop

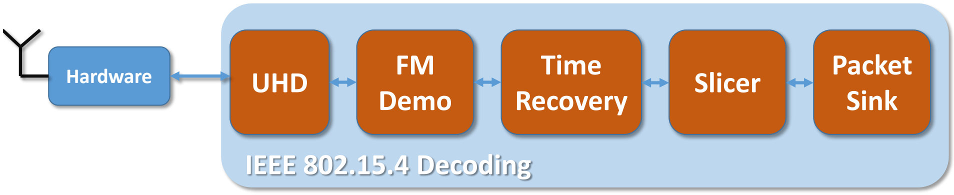


Figure 2.19: IEEE 802.15.4 Decoding

a Gaussian filter. Therefore, to achieve the maximum SNR, the sampling position needs to be taken at the peak of pulse shape. However, in a digital communication system, sampling clock is not synchronized to the incoming signals. Therefore, the sampling position is normally shifted from the peak of the pulse shape. Time recovery [89] is normally used to resample the incoming digital samples to the peak of the pulse shape to achieve maximum SNR and low bit error rate. In the time recovery method whose structure is shown in Figure 2.18, a Timing Error Detection (TED) module is used to extract a timing error term e_k (k is the sample number) between the actual and optimal sample positions based on the symmetrical property of the pulse. Different methods have been proposed to calculate this timing error term, such as early-late gate algorithm, Mueller and Muller algorithm, and Gardner algorithm [89]. In this GMSK receiver, a Mueller and Muller algorithm [95] is used to calculate the timing error term as $e_k = (y_k * \hat{y}_{k-1}) - (\hat{y}_k * y_{k-1})$, where y_k is the value before decision and $\hat{y}_k$ is the value after decision. The timing error term e_k is passed to a loop filter, which outputs the normalized timing error $\mu(k)$ to decide on the correction of the sampling time in the resampler. Subsequently, the sampling position is adjusted to be closer to the optimal one. The outputs of time recovery block are fed into a slicer for symbol decision. After the signals are demodulated to bits, a correlator is used to find the beginning of the packet and finally the packet is reconstructed in the frame sink block.

IEEE 802.15.4 Decoding

In 2006, the author of [109] provided a GNU Radio based IEEE 802.15.4 decoding system for O-QPSK physical layer (2.4GHz). It adopts the demodulation method introduced in Section 2.2.2, which demodulates the O-QPSK signals by a MSK one-bit differential detector. Figure 2.19 shows the flow graph, which has similar structure as GMSK including FM demodulator, time recovery and slicer. In contrast to GMSK with a Gaussian pulse shape, the pulse shape of O-QPSK is half-sine, which is used in time recovery to adapt the sampling position. The main difference between this IEEE 802.15.4 decoding system and GMSK receiver is in the packet sink block. Based on the frame structure of IEEE 802.15.4 packets as in Figure 2.6, the packet sink block detects physical layer frame by searching the synchronization header and then decodes the whole MPDU. Additionally, in this block, Table 2.2 is used to convert the chips to bits. Once a complete MPDU is found, it is added to a message queue. In this system, all components in Figure 2.19 consist of physical layer decoding (FM demo, time recovery and slicer) and MAC layer decoding (packet sink) are implemented in separated GNU Radio blocks. In Chapter 4, based on the same physical layer decoding method, we build up an IEEE 802.15.4 decoding system, which more efficiently passes control data from the physical layer to the MAC layer with a cross-layer structure.

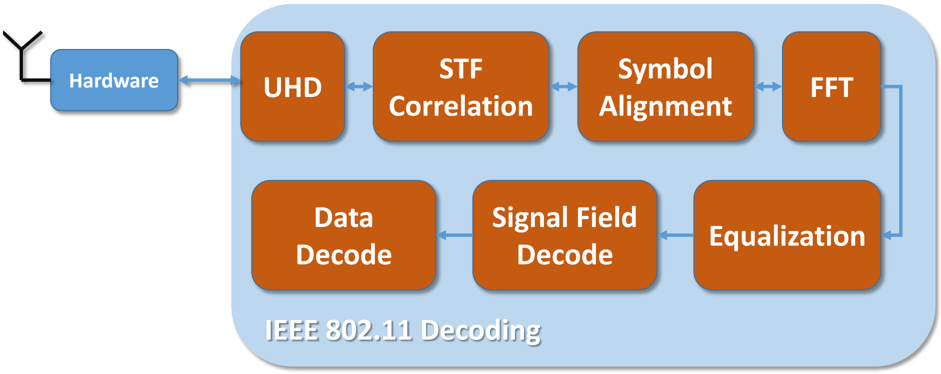
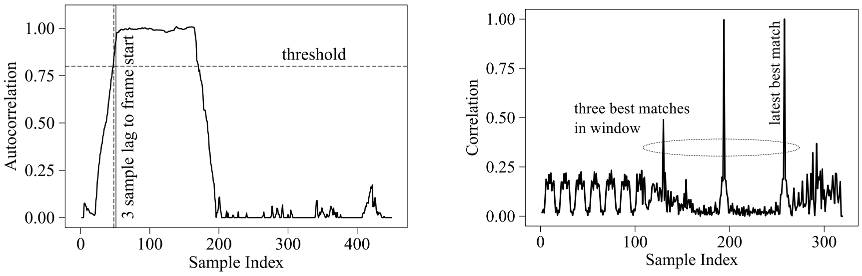


Figure 2.20: IEEE 802.11 Decoding

IEEE 802.11 Decoding

The authors of [28] developed a system to decode IEEE 802.11a/g/p signals based on GNU Radio and USRP N210. This system can be configured to support 10MHz (IEEE 802.11p) and 20MHz (IEEE 802.11a/g and IEEE 802.11n packets with legacy model). The decoding procedure follows the structure in Figure 2.20. Please refer to [28] for more details about GNU Radio blocks in this system. This system is modified to design the WiFi signals sniffers in our passive positioning/tracking system, which is introduced in Chapter 5 and 6.



(a) STF Correlation

(b) LTF Correlation

Figure 2.21: STF and LTF Correlation [28]

In this system, the incoming samples with 20MHz sampling rate first correlate with the pre-defined short preamble (STF). If there is a frame receiving, a plateau will be detected as in Figure 2.21(a). Then the samples in the detected frame are correlated with the long preamble (LTF) to fine align the OFDM symbol for FFT operation. Additionally, the frequency offset is also corrected by long preamble. After aligning OFDM symbols, a FFT is applied to convert the signals from time domain to frequency domain. Then equalization block is adopted to correct phase offset based on the signals from the 4 pilot subcarriers. The signal field, which carries information about the modulation and encoding schemes of the following symbols, is decoded. Based on the modulation schemes and coding rate information obtained from the signal field, the remaining parts of this frame are decoded.

2.4 Range-free Positioning Methods

After briefly introducing some background knowledge about wireless communication, we review some relative work and background about positioning techniques based on radio signals. As mentioned in Section 2.1, radio-based indoor positioning can be classified as range-based and range-free methods. Because of the severe multipath propagation indoor, it is nontrivial to achieve accurately ranging. Hence, range-free indoor positioning has been investigated for decades. In this section, we review two well-known range-free methods including fingerprinting and centroid methods.

2.4.1 Fingerprinting Positioning Methods

Fingerprinting positioning methods have been intensively investigated in the past years and are currently the dominating indoor positioning algorithm because of its high positioning accuracy. Fingerprinting algorithms can be used in both passive positioning and active positioning system. A fingerprinting method comprises two phases including offline training and online positioning. The aim of the offline training phase is to build a fingerprint database, i.e., radio map, which contains different features of radio signals at a set of training locations. In the previous research, the features used for constructing the radio map focus on RSS and channel information. To generate this radio map, a mobile device needs to move through the area of interest at the training locations and record the radio features in a database. In the online positioning phase, the mobile target measures the radio features. Then, pattern matching techniques are applied to map the measured radio features to the training data in the radio map and finally locate the target.

To achieve higher positioning accuracy, the training positions are required to be very dense and cover the whole area of interest. Therefore, it is very labour intensive to build a radio map, which covers a large area of interest. Additionally, fingerprinting methods are typically error prone to layout changes, which make the radio map out of date. Therefore, the radio map needs to be updated frequently as soon as the surrounding environments change. Otherwise, the positioning accuracy will dramatically decrease.

RADAR

A Microsoft research group presented the first fingerprinting system called RADAR [24], which uses RSSI from the received WiFi signals. In the offline phase, RSSIs at multiple ANs are measured to build the RSSI radio map. Assume that at the ith training location, the measured RSSI vector is $\mathbf{RSSI}_i$. In the online phase, after measuring the RSSI vector $\mathbf{s}$ at the current location, the K-Nearest Neighbours

(KNN) algorithm is adopted to match the position of a target device to the RSSI map. They select the K nearest neighbours which have smallest Euclidean distance ($D = ||\mathbf{s} - \mathbf{RSSI}_i||$) from the measured RSSI vector in the radio map. Then they calculate the average of these neighbours' locations as the estimated location of the target as

$$(x', y') = \frac{1}{K}\sum_{i=1}^{K}(x_i, y_i), \tag{2.12}$$

where (x_i, y_i) are the coordinates of the training locations.

To further improve the positioning accuracy, some work [41] introduces weighting techniques to KNN, namely Weighted-KNN (WKNN) , as

$$(x', y') = \sum_{i=1}^{K}[w_i \cdot (x_i, y_i)]. \tag{2.13}$$

A most often used way to set the weights w_i in Equation (2.13) is as

$$w_i = \frac{1/D}{\sum_j^K 1/D}, \tag{2.14}$$

which is inversely proportional to the Euclidean distance ($D = ||\mathbf{s} - \mathbf{RSSI}_i||$).

HORUS

HORUS [129] is another well known RSSI-based fingerprinting system for WiFi systems. It also comprises two phase: offline training and online positioning. In contrast to RADAR with a KNN method in online positioning phase, HORUS uses a joint clustering method for positioning, which uses a probabilistic method. This probabilistic method considers the pattern recognition as a classification problem. Assuming that there are n training positions $L_1, L_2, \cdots, L_n$ and $\mathbf{s}$ is the observed RSSI vector during the online stage, HORUS calculates the posteriori probability of $P(L_i|\mathbf{s})$. Using Bayes' formula and assuming that $P(L_i) = P(L_j)$ for $i, j = 1, 2, \cdots, n$, the likelihood of $P(\mathbf{s}|L_i)$ is equal to $P(L_i|\mathbf{s})$. After obtaining the likelihood $P(\mathbf{s}|L_i)$, the location is estimated by a weighted average of the coordinates of all training positions as

$$(x', y') = \sum_{i=1}^{n} P(L_i|\mathbf{s})(x_i, y_i). \tag{2.15}$$

FIFS

As introduced in Section 2.2.3, CSI in an OFDM system provides channel information with amplitude and phase over multiple subchannels, which is a fine-grained information compared to RSSI. Compared to RSSI, CSI provides more information because of frequency diversity. FIFS [124] is a fingerprinting system for WiFi signals, which adopts CSI instead of RSSI to construct the radio map. The system leverages a commercial WiFi card (IWL5300), which is able to extract CSI from the received packets. To build the radio map, a laptop equipped with the IWL5300 WiFi card needs to scan the CSI information in different training positions. Then, they calculate an effective CSI by summing power of CSI over all the independent subchannels and store this effective CSI information in the database. For online positioning, they adopt the probabilistic method (Equation (2.15)) to locate the target. As reported in [124], the positioning accuracy of FIFS by using CSI is impressively higher than HORUS with RSSI.

CSI-MIMO

CSI-MIMO [33] further considers the frequency diversity and spatial diversity in the measured CSI information by combining multiple sub-carriers and multiple transmit and receiver antennas to generate the location fingerprints. To build up the radio map, first, they average the CSI value of each subchannel over MIMO and obtain an aggregated CSI vector of dimension 1×30. Each component in this vector is an aggregated CSI on one individual subchannel. Second, they extract the amplitude and phase information in CSI and subtract these information between sequential subchannels. The resulting amplitude and phase difference values construct the fingerprints in the radio map. For the online positioning phase, they use both KNN as in RADAR and the probabilistic method as in HORUS and FIFS. Experimental results show that by considering the amplitude and phase information, CSI-MIMO outperforms FIFS by 57%.

2.4.2 Centroid Positioning Methods

Centroid positioning algorithms are another category of range-free positioning and they belong to proximity based positioning methods. They are widely used in wireless sensor networks to roughly estimate a sensor node's location because of its simplicity. The location is estimated by calculating an average value of the coordinates of K nearest neighbour nodes as

$$(x', y') = \frac{1}{K}\sum_{i=1}^{K}(x_i, y_i), \tag{2.16}$$

where (x_i, y_i) are the coordinates of the nearest neighbour nodes. In contrast to fingerprinting, in which the neighbours are the training positions, the neighbour nodes in the centroid algorithm are the real deployed nodes, e.g., sensor nodes. Therefore, to achieve high positioning accuracy, the density of deployed nodes requires to be high.

Weighted Centroid (WC) methods improve the centroid methods by introducing different weights for different neighbour nodes as

$$(x', y') = \sum_{i=1}^{K} [w_i \cdot (x_i, y_i)], \tag{2.17}$$

where w_i is the weight for the ith neighbour node. The weight can be set based on RSSI or the estimated distance. For example, the authors of [29] propose a weighted centroid algorithm for IEEE 802.15.4 signals based on the estimated propagation distance from RSSI. Each weight is inversely proportional to the estimated propagation distance d as

$$w_i = \frac{1/d_i^g}{\sum_{j=1}^{K} 1/d_j^g}, \tag{2.18}$$

where g is a parameter influenced by the surrounding environments and the weights are normalized. The value of g can be manually determined according to [118]. For a propagation range of $10m$, $g = 1$ and for $20m$, $g = 2$. Although ranges, i.e., the propagation distance d in Equation (2.18), are calculated in this weighted centroid algorithm, the range information is only used as weights instead of directly using for positioning like trilateration algorithms, which are introduced in Section 2.6. Therefore, it is less influenced by the ranging accuracy and more robust to ranging errors compared to some trilateration algorithms like LLS [29].

These centroid algorithms are easy to implement but can only provide low positioning accuracy. Therefore, it is suitable to use these algorithms in some applications with low computation capability and low requirements for positioning accuracy such as in wireless sensor networks.

2.5 Ranging Methods

In contrast to range-free positioning methods, radio parameters in range-based positioning methods are used to estimate the propagation distances from the target to different ANs. This procedure is called ranging, which is the first step for range-based indoor positioning. To achieve ranging, we can use different radio parameters such as time, RSSI or channel information.

2.5.1 Time-based Ranging

Arrival Time of Radio Signals

Accurate measurement of the arrival time of radio signals is a common challenge in all the time-based ranging methods. The accuracy of the timestamps depends on several aspects, such as uncertain measurement delays in different receivers and properties of radio signals including bandwidth and modulation/demodulation schemes.

The timestamps for the received signals from a packet source reflect the signal propagation time as well as the processing time in ANs (receivers). For most of the commercial devices like WiFi cards and sensor nodes, a timestamp is normally given at the MAC layer and the accuracy is only on the microseconds level because of processing delay at physical and MAC layers [52]. Consequently, a timestamp is best given close to the physical layer to avoid influences from processing time at the MAC or higher layers at the receiver. To obtain the physical layer timestamps, special devices are required, which can process the physical layer information to achieve high accuracy of timestamps.

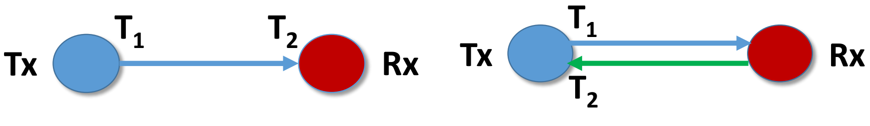
The accuracy of physical layer timestamps is highly dependent on the properties of radio signals including bandwidth and demodulation schemes. Many studies on time-based positioning with radio signals focus on Ultra Wide Band (UWB) signals [56, 85]. Due to its wide bandwidth (larger than 500MHz), UWB can achieve nanosecond accuracy for timestamps and resolve the multipath components. In LOS conditions, positioning systems based on UWB signals can achieve an accuracy of ranging on the level of millimeters. NLOS propagation is the main error source for positioning error by using UWB because the LOS component may not exist [44, 86]. Additionally, UWB signals require a special radio transceiver and hence it is normally used for some specific applications, such as positioning in sensor network [56] and navigation for mobile robot [70].

In our daily life, signals with narrower bandwidth are more often used than UWB signals, such as GSM with 200KHz, IEEE 802.15.4 with 2MHz, WiFi with 20MHz - 40MHz and LTE with up to 20MHz. It becomes extremely challenging to accurately measure the arrival time of these radio signals with narrow bandwidth. Additionally, complex indoor propagation is another aspect limiting the accuracy of time measurement for signals with narrow bandwidth. The sources of errors include multipath propagation and shadowing. Benefiting from its wide bandwidth and correspondingly extremely short pulse interval, UWB can resolve and separate the multipath components. However, for a signal with narrower bandwidth as shown in Figure 1.3, the signals from different paths including the direct path and multipath from reflection overlap in one symbol duration. It is extremely challenging to detect the direct path. Furthermore, shadowing attenuates or completely blocks the direct path, adding another error to the range estimate.

TOA, TDOA and DTDOA

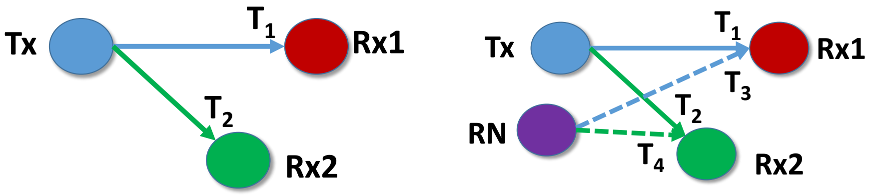
There are different ways to achieve time-based ranging such as TOA (Time of Arrival) and TDOA (Time Difference of Arrival). TOA is used to calculate the propagation distances between a target and different ANs by measuring the propagation delay of the radio signals. In contrast to TOA, TDOA-based ranging is used to calculate the difference of distances between the target and different ANs.

- **TOA** ranging can be achieved by two methods including one-way ranging and two-way ranging. For one-way ranging, as shown in Figure 2.22(a),



(a) One-Way Ranging (TOA) (b) Two-Way Ranging (RTT)

Figure 2.22: Time Of Arrival (TOA)



(a) TDOA (b) DTDOA

Figure 2.23: TDOA and DTDOA

the transmitter needs to measure the transmission time T_1 and receiver measures the time of arrival T_2. Then, the Time of Flight (TOF) is calculated as TOF $= T_2 - T_1$. The propagation distance based on the transmitter and receiver is calculated as $d =$ TOF$\cdot c$, where c is the speed of radio signal (light). This one-way ranging requires synchronization between transmitter and receivers. This one-way ranging method is used in GPS [48]. Round Trip Time (RTT) method measurements can also be used for TOA estimation. As shown in Figure 2.22(b), with RTT, the transmission time (T_1) of a radio signal (request signal) is measured at the transmitter side. After receiving the signal, the receiver needs to send a response signal back to the transmitter, which measures time arrival time (T_2) of this response signal. Then, RTT is calculated as RTT $= T_2 - T_1$. Theoretically, the propagation distance between the transmitter and receiver can be calculated as $d =$ RTT $\cdot\, c/2$. One of the advantages for RTT is that it does not need to synchronize the devices including transmitter and receivers. In practice, one of the critical influence on the ranging accuracy based on RTT is the uncertain processing delay at the receiver side. The authors of [87] modified a firmware for a commercial WiFi card to extract timestamps from the received WiFi packets. They used RTT measurements to calculate the ranges between the target and different ANs. To compensate the uncertain processing delay at the receiver side in the RTT measurements, they proposed to offline calibrate the processing delay at the receiver side and then compensate the delay in RTT. Limited by the bandwidth of WiFi (20MHz), the 80% accuracy of positioning can only reach $3.7 - 5.8m$ as reported in [87]. Because both one-way and two-way ranging TOA methods require the participation of the positioning target for synchronization or packet exchanges, TOA methods are used in active positioning systems, e.g., GPS, but not in passive positioning.

- **TDOA** does not require two-way packet exchange (request and response) between the transmitter and receivers as RTT. As shown in Figure 2.23(a), two receivers measure the arrival time of the same signal from one target (T_1 and T_2) and calculate the difference (TDOA) between the measured time in these two receivers as TDOA $= T_1 - T_2$. The procedure of converting the measured TDOA values to the difference between the propagation distances from the target to the different ANs (receivers) is called TDOA-based ranging, i.e., $\Delta d =$ TDOA $\cdot\, c$. One of the challenges for TDOA measurement is the imperfect synchronization between ANs. Because of the large propagation speed of radio signals ($3 \cdot 10^8 m/s$), the synchronization between ANs is required to be on the level of nanoseconds to achieve high ranging accuracy. Therefore, a large amount of research has been conducted on how to improve synchronization accuracy and how to compensate the remaining synchronization offset. The authors of [85] solved the synchronization problem by wiring all the receivers together to share a single crystal clock. However, this mechanism can only be used for a small scale deployment and is infeasible for a large scale deployment. The authors of [133] analyzed TDOA performance with GSM signals in simulation and provided a ranging accuracy of 11.4m in LOS situations and 23.3m in NLOS situations with 200KHz bandwidth. They did not consider, however, practical factors such as synchronization and sampling rate. TruePosition [116] is a leader in the deployment of location technologies in support of the E911 mandate. They provide Uplink-TDOA (U-TDOA) solution for GSM positioning. The accuracy of positioning is influenced by the mentioned factors, i.e., AN synchronization and multipath propagation, and they declared the positioning accuracy from $50m$ to $150m$ in urban and sub-urban environments in their report.

- **DTDOA** is a modified version of TDOA to compensate the synchronization offset between receivers using a Reference Node (RN) as shown in Figure 2.23(b). The main idea is that two receivers need to calculate TDOA values for the packets from the target (TDOA$_t = T_1 - T_2$) and RN (TDOA$_r = T_3 - T_4$) respectively. By calculating the difference between these two values as DTDOA $=$ TDOA$_t -$ TDOA$_r$, the synchronization offset between these two receivers can be compensated. The authors of [122] have discussed DTDOA in WLAN and evaluated their approach in a MATLAB simulation but no evaluation in real world is done. In practice, the authors of [97] demonstrated that some limitations are posed on the interval between the packet transmissions of the RN and the target. Since RN and the target can not transmit their packets exactly at the same time, which will cause collisions at the receiver side, synchronization offset between the non-perfectly synchronized receivers accumulates during the transmission interval between the RN and target. According to [97] the accumulated synchronization offset can introduce ranging errors as large as $250m$ in a practical system. The authors proposed to estimate the mean clock offset between two unsynchronized local clocks in an off-line phase and to compensate the accumulated synchronization offset during the on-line phase. However, they also showed that the momentary clock offset varies randomly from the mean value, demanding very short interval between the packet transmissions of the RN and the target, e.g., hundreds of microseconds, to meet the strict requirements of TDOA localization. Therefore, in our work (Chapter 4), we propose to combine DTDOA with GPS synchronization, which is able to compensate the momentary clock offset but do not need the offline calibration and do not require very strict limitations on the interval between the packet transmissions of the RN and the target, compared to [97]. Both TDOA and DTDOA can be implemented at the network side to passive locate the target. Therefore, TDOA and DTDOA can be used in both passive and active positioning systems.

Time Synchronization

As mentioned before, one-way TOA ranging and TDOA methods require high synchronization accuracy between devices. We introduce the clock terminology used in this thesis.

Every AN in a time-based positioning system has its own clock. Assuming that the reference time of the whole system is $C(t) = t$, ideally, the local time of the ith AN (AN_i) should be $C_i(t) = t$. However, even if started at exactly the same time, local clocks will drift away from the reference time because of frequency deviations of the oscillator.

Generally, the clock function of AN_i is modeled as

$$C_i(t) = \int_0^t \Delta f_i(t)\,\mathrm{d}t + \Delta\theta_i, \tag{2.19}$$

where the parameters $\Delta f_i(t)$ and $\Delta\theta_i$ are the *clock skew* and *initial clock offset* at AN_i. In this thesis, we use the nomenclature from [93] to define that $\Delta C_{ij}(t) = C_i(t) - C_j(t)$ is the *relative clock offset*, and $\Delta f_{ij}(t) = \Delta f_i(t) - \Delta f_j(t)$ is the *relative clock skew* between AN_i and AN_j.

There are different mechanisms to synchronize the devices, which intend to minimize the relative clock offset $\Delta C_{ij}(t)$. There are numerous ways for device synchronization such as Network Time Protocol (NTP), Precision Time Protocol (PTP) (IEEE 1588) and GPS. NTP [90] and PTP [74] are commonly used to synchronize clocks in the Internet computing infrastructure. NTP is often used to synchronize system clocks in general-purpose workstations and servers. PTP is used most often to synchronize device clocks in special-purpose industrial automation and measurement networks. These devices are connected to dedicated, high-speed Ethernet LAN segments interconnected by switches. NTP can only provide an accuracy on the level of milliseconds and PTP improves the accuracy to microseconds, which is still not accurate enough for positioning.

GPS synchronization offers an accuracy of sub-microseconds or nanoseconds by regularly calibrating the local times of GPS receivers to the signals from satellites. There is a Phase Lock Loop (PLL) with an inaccurate local clock inside each GPS receiver. PLL requires some time to lock to the GPS signals, denoted as an initialization phase. After PLL has locked, PLL steps into a stable phase. In the stable phase, PLL needs to be periodically adjusted based on the received GPS signals. If the GPS receivers can not receive the signals from the satellites, the local clocks will continuously drift away from each other. Hence, indoor devices should be synchronized by using GPS receivers with outdoor antennas. Although GPS synchronization has been reported in some data sheet [15] that it is able to achieve sub-microseconds synchronization accuracy, it is still not clear that how accurate the clock offset and skew of the real deployed GPS-synchronized devices can reach in nanoseconds. This nanosecond accuracy of both clock offset and skew has significant impact on time-based positioning and therefore, in our work (Chapter 3), we propose our methods to investigate the properties of GPS-synchronized clocks in depth.

2.5.2 RSSI-based Ranging

RSSI is another commonly used radio parameter to calculate the propagation distance [131, 57]. In contrast to time information, which requires special devices, RSSI is available for most of the commercial devices for Bluetooth, WiFi and Zigbee. However, RSSI is a coarse MAC layer power indicator, which is prone to multipath and NLOS propagation. In indoor environments, a radio signal propagates to a receiver through multiple propagation paths, in which the signals have different delay, attenuation and phase shift. Then, RSSI is simply calculated as the power of the received signal, which combines multiple versions of the original signal. Therefore, a small change in the multipath propagation may lead to significant changes of the relative phases in constructive or destructive signals. This will cause significant fluctuations in RSSI as shown in Figure 1.2 [126].

To characterizes the variation of received signal strength over distance, a Log Distance Path Loss (LDPL) is commonly used as [20, 21, 49]

$$\mathrm{RSS} = P_t - (\mathrm{PL}(d_0) + 10 \cdot \xi \cdot log_{10}(\frac{d}{d_0}) + X_\theta), \tag{2.20}$$

where RSS is the received signal strength, P_t is the transmission power in dBm, $\mathrm{PL}(d_0)$ is the path loss at reference point d_0 and ξ is the path loss exponent. X_θ is a zero-mean normal random variable reflecting shadowing attenuation in dB. LDPL assumes a linear relation between the path loss and logarithmic distance. Besides the multipath effects, it is challenging to accurately model the power attenuation because of obstacles (shadowing) in a complex indoor environment. Therefore, RSSI-based ranging normally suffers from large ranging errors. To compensate the attenuation of walls, a path loss model with a Wall Attenuation Factor (WAF)

[51] has been proposed as

$$\text{RSS} = P_t - (\text{PL}(d_0) + 10 \cdot \xi \cdot log_{10}(\frac{d}{d_0}) + \text{WAF} + X_\theta), \tag{2.21}$$

where WAF is the factor indicating the power attenuation because of the walls. However, with this WAF model, a correct count of the number of walls between the target and receivers, which is difficult to obtain in practice [135].

2.5.3 Ranging based on Channel Information

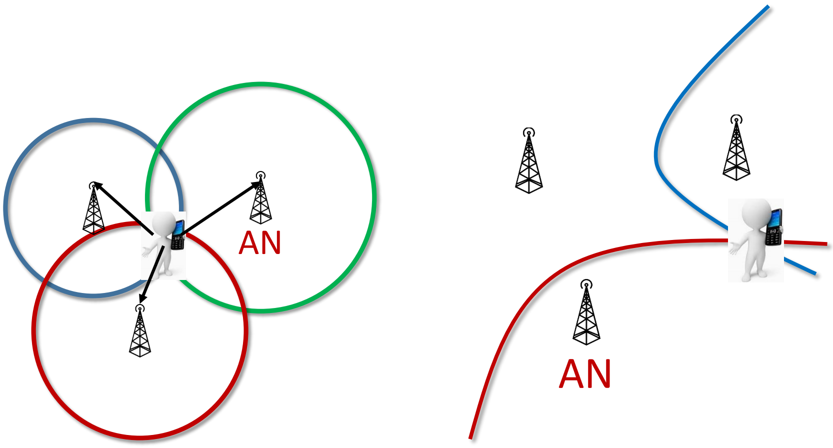
In contrast to RSSI, CSI is a fine-grained physical layer information, which can be used to resolve the signals from multiple paths. As mentioned in Section 2.2.3, CSI characterizes amplitude and phase information over multiple subcarriers in frequency domain. Note that CSI information is only available in OFDM systems with multiple carriers but not in single band systems like DSSS. By applying an IFFT, CSI is converted to CIR in time domain, which characterizes amplitude and phase information over multiple propagation paths in time domain.

As shown in Figure 2.11, CIR is a digitalized channel information in time domain, with a resolution of $\Delta\tau = \tau_n - \tau_{n-1}$. The resolution $\Delta\tau$ depends on the bandwidth of the used signal, i.e., $\Delta\tau = 1/B$, where B is the bandwidth of the signal. If the bandwidth B is infinite, CIR will be the same as the analog channel and it can distinguish all the propagation paths. However, in practice, the bandwidth B is limited. Therefore, the measured CIR with a resolution of $\Delta\tau = 1/B$ can only distinguish several clusters of propagation paths rather than every individual multipath component. The bandwidth of IEEE 802.11n is 20MHz and hence the time resolution of an estimated CIR is $1/20\text{MHz} = 50ns$, i.e., $\Delta\tau = 50ns$ [123, 126, 132].

FILA is a well-known indoor positioning system for WiFi signals by using CSI-based ranging. In FILA, CSI information is extracted from an IWL5300 WiFi card. Then, CSI is converted to CIR by applying an IFFT. FILA takes a threshold-based method to separate the signal power corresponding to the LOS path. More specifically, given a measured CIR $h(\tau)$, the paths with amplitudes smaller than 50% of the first peak value in $h(\tau)$ are filtered out, thus retaining the LOS or the shortest NLOS paths. The filtered CIR samples are once again converted into the frequency domain. FILA adopts a weighted summation on the filtered CSIs to normalize the power to the central frequency within the band as

$$\text{CSI}_{\text{eff}} = \frac{1}{N} \sum_n \frac{f_n}{f_c} \cdot a_n, \tag{2.22}$$

where CSI_{eff} is the final input for distance estimation. N is the total number of subcarriers.f_c is the central frequency, and a_n is the amplitude of the filtered CSI on the nth subcarrier. Experimental results in [123] demonstrate that CSI can mitigate multipath propagation and significantly improve the positioning accuracy



(a) Trilateration (b) Multilateration

Figure 2.24: A Geometric Explanation of Trilateration and Multilateration

compared to RSSI. The authors of [126] proposed a positioning system with nomadic access points, named NomLoc, which proposed to estimate the power of direct path with the maximum power in the measured CIR samples and consider the other CIR samples as multipath components from reflection and diffraction. With this method, it is able to mitigate the influence of multipath propagation. This work also relies on the IWL5300 WiFi card for CSI extraction. This multipath mitigation method [126] is also used in our work as a part of our ranging methods based on channel information (in Chapters 5, 6, and 7).

Both [126] and [123] rely on the specific card IWL5300 WiFi card for CSI extraction. However, there are some limitations on this card for CSI extraction. First, the firmware [60] only supports CSI extraction from the connections using an IEEE 802.11n HT (High-Throughput) rate but not legacy mode. Second, it does not support extracting CSI from overheard packets. Hence, the network cards with the firmware [60] can not be used for a passive positioning system, in which the ANs need to obtain CSI from the overheard packets.

2.6 Trilateration and Multilateration Algorithms

After obtaining ranging information based on radio parameters, range-based positioning algorithms such as trilateration and multilateration are used to estimate the location of the target. Figures 2.24(a) [36] and 2.24(b) [127] show a geometric explanation of trilateration and multilateration. For trilateration, the target is located at the intersecting area of circles, whose centers are the locations of ANs and radii are the estimated ranges. Trilateration is suitable for TOA, RSSI and CSI based ranging, which can directly obtain the propagation distances from the target to different ANs. In contrast to trilateration, multilateration is based on the measurement of the differences in distances between different ANs, which is suitable for TDOA-based ranging. As shown in Figure 2.24(b), the target is located in the intersecting area of multiple hyperbolic lines. Geometric explanations as in Figure 2.24(a) and Figure 2.24(b) provide us a clear view of these two methods but geometric solutions are difficult to be implemented in practice. In practice, the location of the target is numerically calculated.

The range-based positioning problem can be formulated as a convex optimization problem to find the optimal solution (location of the target), which meets the range measurements. The commonly used methods like Linear Least Square (LLS) [114], Weighted Least Square (WLS) [35], Constrained Weighted Least Square (CWLS) [37] and Nonlinear Least Square (NLS) [125] formulate the positioning problem as the least square problem.

- **Trilateration:** Considering a set of ranges r_k estimated by RTT, RSSI or CSI and coordinates of ANs (x_k, y_k) where k indicates the kth AN, we obtain

  $$r_k = \sqrt{(x - x_k)^2 + (y - y_k)^2} + n_{\text{noise}}. \tag{2.23}$$

  Assuming N ANs, to get the solution (x', y') for this overdetermined nonlinear system of equations, we have to solve the following least square problem,

  $$(x', y') = \underset{x,y}{\operatorname{argmin}} \sum_{k=1}^{N} [\sqrt{(x - x_k)^2 + (y - y_k)^2} - r_k]^2. \tag{2.24}$$

- **Multilateration:** Considering a set of range differences Δr_k, which indicates the range difference between the kth and the first ANs, we obtain

  $$\Delta r_k = \sqrt{(x - x_k)^2 + (y - y_k)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2} + n_{\text{noise}}. \tag{2.25}$$

  To get the solution (x', y') for these overdetermined equations, we have to solve the following least square problem,

  $$(x', y') = \underset{x,y}{\operatorname{argmin}} \sum_{k=1}^{N} [\sqrt{(x - x_k)^2 + (y - y_k)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2} - \Delta r_k]^2. \tag{2.26}$$

There are different ways to solve the aforementioned least square problems for trilateration and multilateration, such as NLS, LLS, WLS and CWLS.

2.6.1 Non-linear Least Square

Consider the unconstrained minimization problem

$$\text{minimize} f(\mathbf{x}), \tag{2.27}$$

where $\mathbf{x} \in \mathbf{R}^n$ and $f : \mathbf{R}^n \longrightarrow \mathbf{R}$ is continuously differentiable. For two dimensional indoor positioning, n is equal to 2. Numerical methods based on NLS for nonlinear optimization problems are iterative. Trust region and line search are two commonly used methods to find the solution for the problem in Equation (2.27).

The line search approach first finds a descent direction along which the objective function f will be reduced and then computes a step size that determines how far $\mathbf{x}$ should move along that direction. The descent direction can be computed by various methods, such as gradient descent, Newton's method and Quasi-Newton method [98]. The step size can be determined either exactly or inexactly. Take a quasi-Newton method as an instance. A Quasi-Newton method generates a sequence of iterates $\{\mathbf{x}_k\}$ and steps $\{\mathbf{s}_k\}$ with $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$. At $\mathbf{x}_k$, a quadratic model of $f(\mathbf{x}_k + \mathbf{s}_k)$ is formed as

$$\Phi_k(\mathbf{s}_k) = f(\mathbf{x}_k) + \mathbf{g}_k^T\mathbf{s}_k + \frac{1}{2}\mathbf{s}_k^T\mathbf{H}_k\mathbf{s}_k. \tag{2.28}$$

$\mathbf{g}_k = \nabla f(\mathbf{x}_k)$ and $\mathbf{H}_k$ introduce curvature into the model. To minimize the $\Phi_k(\mathbf{s}_k)$, $\mathbf{s}_k = -\mathbf{H}_k^{-1}\mathbf{g}_k$. A line search strategy considers $-\mathbf{H}_k^{-1}\mathbf{g}_k$ to be a search direction. The step $\mathbf{s}_k$ needs to follow $-\lambda_k\mathbf{H}_k^{-1}\mathbf{g}_k$, where λ_k is for step length and it needs to be chosen in an appropriate way.

In contrast to line search strategies, trust region techniques do not necessarily choose the searching direction. In trust region, the step is an approximate solution of the following trust region subproblem

$$\text{minimize } \Phi_k(\mathbf{s}_k), \quad \text{subject to } ||\mathbf{s}_k|| \leq \theta_k, \tag{2.29}$$

where θ_k is the trust radius and $||\cdot||$ is the norm of a vector. A merit function is normally used to test whether the trial step is accepted or the trust region radius needs to be adjusted. If a trial step is accepted at some iteration then we call the related iteration an effective iteration, otherwise we call the iteration an ineffective iteration. Please refer to [94] for more details about the trust region solution for the unconstrained minimization problem (Equation (2.27)).

Such NLS techniques face high computational complexity and require good initialization in order to avoid converging to the local minima of the cost function [55].

2.6.2 Linear Least Square

LLS is an alternative of NLS to solve the problems of Equation (2.24) and (2.26), which linearizes the nonlinear equations by introducing a constraint. Compared to NLS, LLS has less computational complexity but is prone to the influence of noise.

Trilateration: Without measurement errors, Equation (2.23) is written as

$$\begin{aligned} r_k^2 &= (x - x_k)^2 + (y - y_k)^2 \\ &= R_1^2 - 2xx_k - 2yy_k + (x_k^2 + y_k^2) \\ \Longrightarrow & \\ & x_kx + y_ky - 0.5R_1^2 = 0.5(x_k^2 + y_k^2 - r_k^2) \end{aligned} \tag{2.30}$$

where $R_1 = \sqrt{x^2 + y^2}$ is the constraint introduced for linearization. Then, Equation (2.30) expressed in matrix-vector form is

$$\mathbf{G}\theta = \mathbf{h}, \tag{2.31}$$

where $\theta = [x, y, R_1^2]^T$,

$$\mathbf{G} = \begin{pmatrix} x_1 & y_1 & -0.5 \\ \vdots & \vdots & \vdots \\ x_N & y_N & -0.5 \end{pmatrix},$$

$$\mathbf{h} = \frac{1}{2} \begin{pmatrix} x_1^2 + y_1^2 - r_1^2 \\ \vdots \\ x_N^2 + y_N^2 - r_N^2 \end{pmatrix}.$$

To find the solution of (x, y), we minimize the sum of squares of the residuals as

$$\begin{aligned} \hat{\theta} &= \underset{\theta}{\operatorname{argmin}}[(\mathbf{G}\theta - \mathbf{h})^T(\mathbf{G}\theta - \mathbf{h})] \\ &= (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{h}, \end{aligned} \tag{2.32}$$

and $\hat{\theta} = [x', y', R_1^2]^T$. After obtaining $\hat{\theta}$, we get the LLS solution (x', y') for trilateration.

Multilateration: With similar procedures, we derive the LLS solution for multilateration. Without measurement errors, Equation (2.25) is rewritten as

$$\begin{aligned} &\Delta r_k = \sqrt{(x - x_k)^2 + (y - y_k)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2} \\ &\Longrightarrow \\ &\Delta r_k + \sqrt{(x - x_1)^2 + (y - y_1)^2} = \sqrt{(x - x_k)^2 + (y - y_k)^2} \\ &\Longrightarrow \\ &(x - x_1)(x_k - x_1) + (y - y_1)(y_k - y_1) + \Delta r_k R_2 = \frac{1}{2}[(x_k - x_1)^2 + (y_k - y_1)^2 - \Delta r_k^2] \end{aligned} \tag{2.33}$$

where $R_2 = \sqrt{(x - x_1)^2 + (y - y_1)^2}$ is the constraint introduced for linearization. Then Equations (2.33) expressed in matrix-vector form is

$$\mathbf{G}\theta = \mathbf{h}, \tag{2.34}$$

where $\theta = [x - x_1, y - y_1, R_2]^T$,

$$\mathbf{G} = \begin{pmatrix} x_2 - x_1 & y_2 - y_1 & \Delta r_2 \\ \vdots & \vdots & \vdots \\ x_N - x_1 & y_N - y_1 & \Delta r_N \end{pmatrix},$$

$$\mathbf{h} = \frac{1}{2}\begin{pmatrix} (x_2 - x_1)^2 + (y_2 - y_1)^2 - \Delta r_2^2 \\ \vdots \\ (x_N - x_1)^2 + (y_N - y_1)^2 - \Delta r_N^2 \end{pmatrix}.$$

To find the solution of (x, y), we minimize the sum of squares of the residuals as

$$\begin{aligned}\hat{\theta} &= \underset{\theta}{\operatorname{argmin}}[(\mathbf{G}\theta - \mathbf{h})^T(\mathbf{G}\theta - \mathbf{h})] \\ &= (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{h},\end{aligned} \tag{2.35}$$

and $\hat{\theta} = [x' - x_1, y' - y_1, R_2]^T$. After obtaining $\hat{\theta}$, we get the LLS solution (x', y') for multilateration.

2.6.3 Weighted Least Square

In LLS, all the measured ranges are treated equally. However, ranges from different ANs normally face different ranging errors. Therefore, this oversimplicity introduces large positioning errors. Weighted least square improves LLS by introducing a weighting technique, which sets different weights to the measured ranges from different ANs. WLS is an efficient way to mitigate the ranging errors. By introducing weights in LLS, Equations (2.32) and (2.35) are rewritten as

$$\begin{aligned}\hat{\theta} &= \underset{\theta}{\operatorname{argmin}}[(\mathbf{G}\theta - \mathbf{h})^T\mathbf{W}(\mathbf{G}\theta - \mathbf{h})] \\ &= (\mathbf{G}^T\mathbf{W}\mathbf{G})^{-1}\mathbf{G}^T\mathbf{W}\mathbf{h},\end{aligned} \tag{2.36}$$

where $\mathbf{W}$ is the weighting matrix. Theoretically, $\mathbf{W}$ for trilateration is set as

$$\mathbf{W} = \text{diag}[\frac{1}{\sigma(r_1)}, \cdots, \frac{1}{\sigma(r_N)}] \tag{2.37}$$

and for multilateration as

$$\mathbf{W} = \text{diag}[\frac{1}{\sigma(\Delta r_2)}, \cdots, \frac{1}{\sigma(\Delta r_N)}], \tag{2.38}$$

where $\sigma(\cdot)$ indicates the variance.

2.6.4 Constrained Weighted Least Square

In contrast to WLS, CWLS further considers the constraint of $R_1 = \sqrt{x^2 + y^2}$ for trilateration and $R_2 = \sqrt{(x - x_1)^2 + (y - y_1)^2}$ for multilateration. We introduce the trilateration in this section because it is used in our work (Chapter 5). Please find a similar procedure for multilateration in [37].

After considering the constraint, the unconstrained problem of Equation (2.36) becomes a constrained problem as

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}(\mathbf{G}\boldsymbol{\theta} - \mathbf{h})^T\mathbf{W}(\mathbf{G}\boldsymbol{\theta} - \mathbf{h}), \tag{2.39}$$

subject to

$$\mathbf{q}^T\boldsymbol{\theta} + \boldsymbol{\theta}^T\mathbf{P}\boldsymbol{\theta} = 0,$$

where $\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$, and $\mathbf{q} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}$.

The constrained optimal problem is equivalent to minimize the Lagrangian equation,

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, L(\boldsymbol{\theta}, \lambda) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmin}}((\mathbf{G}\boldsymbol{\theta} - \mathbf{h})^T\mathbf{W}(\mathbf{G}\boldsymbol{\theta} - \mathbf{h}) + \lambda(\mathbf{q}^T\boldsymbol{\theta} + \boldsymbol{\theta}^T\mathbf{p}\boldsymbol{\theta})), \end{aligned} \tag{2.40}$$

where λ is the Lagrange multiplier. To obtain the final estimation of the target location, please find more details in [37] to solve the Equation (2.40) and to obtain the final target location (x', y') in $\hat{\boldsymbol{\theta}}$.

2.6.5 Trilateration and Multilateration in Range-based Positioning

Since LLS and NLS do not require weights on each ranging information as in WLS and CWLS, whose optimal solution requires ground truth information, they are widely used in practice. The authors of [125] investigated both NLS and LLS based on RSSI-based ranging using an IEEE 802.11 network and IEEE 802.15.4 network in a real office building environment. The experimental results show that NLS significantly outperforms LLS. LLS is prone to ranging errors. In FILA [123], because the CSI information mitigates multipath effects and achieves high ranging accuracy, the authors only adopt LLS to locate the target and achieve a median error of $1.2m$ in a corridor environment. Although a very high accuracy has been reported [123], they only evaluated their system in a rather simple scenario (only in a corridor for multi-room environments) with dense deployment of ANs.

To achieve the optimal solution, WLS and CWLS algorithms require ground truth information or variance of estimated ranges, which are difficult to obtain in practice. Therefore, large amount of research is conducted to investigate the performance of WLS and CWLS in simulation with some unpractical assumptions, such as knowing ground truth positions [35, 37, 59].

2.7 Kalman Filters for Tracking

The aforementioned trilateration or multilateration algorithms consider the range-based positioning as a convex optimization problem but do not consider the additional information about the relation of movement states between sequential moments for mobile targets. Therefore, it is more suitable for positioning stationary targets. To track a mobile target, Bayesian filters, which consider the tracking problem as a Hidden Markov Model (HMM) as in Figure 2.25, can provide higher accuracy.

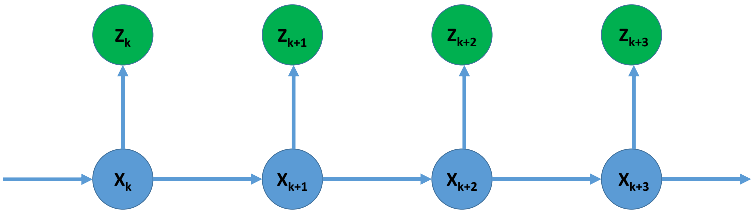


Figure 2.25: Hidden Markov Model

We consider the problem of tracking the locations of a mobile wireless device over time and in two-dimensional space, given a stream of noisy range measurements from at least three ANs. Thus, at time k, we have:

- an unknown system state vector $\mathbf{x}_k$ including the target's location (or velocity and accelerated velocity in addition),
- a discrete sequence of noisy measurement vectors $\mathbf{z}_{1:k}$, taken at times $1, \dots, k$ including the distances to the different ANs, obtained from the RSS or time information.

The target moves according to a function:

$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{v}_k,$ (**dynamic equation** or **system equation**)

and the measurement system observes the target according to another function:

$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{u}_k,$ (**observation equation** or **measurement equation**)

where $\mathbf{v}_k$ and $\mathbf{u}_k$ are the system and measurement noise, respectively. Therefore, in HMM, the hidden variable $\mathbf{x}_k$ depends only on the value of the hidden variable $\mathbf{x}_{k-1}$. The values at time $k-2$ and before have no influence, which is called Markov property. The value of the observed variable $\mathbf{z}_k$ only depends on the value of the hidden variable $\mathbf{x}_k$.

From a Bayesian perspective, the goal is to calculate the "degree of belief" $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ in the current state of the system $\mathbf{x}_k$, based on the available measurements $\mathbf{z}_{1:k}$ and an initial Probability Density Function (PDF) $p(\mathbf{x}_0)$ [23]. This degree of belief is the posterior PDF over the state space of our system. The posterior PDF is calculated via Bayes's rule when the measurement z_k is available:

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})}{p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}. \tag{2.41}$$

$p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is the transition probability of a Markov process based on the dynamic equation. The Markov model (Equation (2.41)) indicates that the current state only depends on the previous state. Based on this concept, prediction is based on the dynamic equation. $p(\mathbf{z}_k|\mathbf{x}_k)$ is the likelihood based on the observation function. $p(\mathbf{z}_k|\mathbf{z}_{1:k-1})$ is the normalizing constant, which follows:

$$p(\mathbf{z}_k|\mathbf{z}_{1:k-1}) = \int p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})d\mathbf{x}_k. \tag{2.42}$$

The recursive PDF estimation based on Equation (2.41) is a general conceptual solution, which can not be derived analytically. Based on certain assumptions, some algorithms still can achieve possible optimal and sub-optimal solutions such as Linear Kalman Filter (LKF), Extended Kalman Filter (EKF) and Particle Filter (PF).

2.7.1 Linear Kalman Filter

Linear Kalman Filter was first introduced by Kalman in 1960. It assumes that both dynamic and observation equations are linear and all the distributions including the posterior PDF, noise in dynamic and observation equations are Gaussian. Linear Kalman Filter is the optimal solution to estimate the state under the aforementioned assumption.

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{v}_{k-1}, \tag{2.43}$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{u}_k. \tag{2.44}$$

The system ($\mathbf{v}_{k-1}$) and measurement noise ($\mathbf{u}_k$) are assumed to be independent and Gaussian with zero mean as

$$p(\mathbf{v}) \sim N(0, \mathbf{Q}), \tag{2.45}$$

$$p(\mathbf{u}) \sim N(0, \mathbf{R}). \tag{2.46}$$

$\mathbf{Q}$ and $\mathbf{R}$ are the noise covariance matrix.

To get the solution for $\mathbf{x}_k$, LKF is based on a two-step procedure.

- The first step is called state prediction and consists of the following two equations:

  $$\hat{\mathbf{x}}_k = \mathbf{F}\hat{\mathbf{x}}_{k-1} + \mathbf{v}_{k-1}, \tag{2.47}$$

  $$\mathbf{P}_k = \mathbf{F}\mathbf{P}_{k-1}\mathbf{F}^T + \mathbf{Q}. \tag{2.48}$$

  Equation (2.47) predicts the current state based on the previous moment and the dynamic equation. Equation (2.48) projects the covariance estimate forward.

- The second step consists of the following three equations:

  $$\mathbf{K}_k = \mathbf{P}_k\mathbf{H}^T(\mathbf{H}\mathbf{P}_k\mathbf{H}^T + \mathbf{R})^{-1} \tag{2.49}$$

  $$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\hat{\mathbf{x}}_k) \tag{2.50}$$

  $$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_k \tag{2.51}$$

  Equation (2.49) calculates the Kalman Gain, which controls the discrepancy between the actual measurement $\mathbf{z}_k$ and the predicted state based on the system model $\mathbf{H}\hat{\mathbf{x}}_k$. Equation (2.50) actually takes the current measurement $\mathbf{z}_k$, Kalman Gain and the predicted state based on the dynamic equation to further adjust the estimation of the current state. Equation (2.51) updates the state error covariance.

2.7.2 Extended Kalman Filter

Although Linear Kalman Filter is very powerful for the linear systems, its performance gets dramatically deteriorated with nonlinear dynamic and observation equations, which are very common in real-world tracking. EKF extends LKF to support nonlinear dynamic and observation equations as

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{v}_k; \tag{2.52}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{u}_k, \tag{2.53}$$

where $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$ are non-linear functions. In EKF, all the noise and posterior PDF are still assumed to be Gaussian distributed. For mildly nonlinear and smooth (differentiable) functions, analytical methods are used to linearize the nonlinear functions. This linearization makes the LKF structure available for use with nonlinear dynamic and/or observation equations.

The nonlinear dynamic and observation equations are linearized as follows

$$\mathbf{f}(\mathbf{x}_{k-1}) \approx \mathbf{f}(\mathbf{x'}_{k-1}) + \mathbf{J_f}(\mathbf{x'}_{k-1})(\mathbf{x}_{k-1} - \mathbf{x'}_{k-1}), \tag{2.54}$$

$$\mathbf{h}(\mathbf{x}_k) \approx \mathbf{h}(\mathbf{x'}_k) + \mathbf{J_h}(\mathbf{x'}_k)(\mathbf{x}_k - \mathbf{x'}_k), \tag{2.55}$$

where $\mathbf{J_f}(\cdot)$ and $\mathbf{h_f}(\cdot)$ are Jacobian of $\mathbf{f}(\cdot)$ and $\mathbf{f}(\cdot)$ as

$$\mathbf{J_f} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \vdots & \vdots & \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix},$$

$$\mathbf{J_h} = \begin{pmatrix} \frac{\partial h_1}{\partial x_1} & \frac{\partial h_1}{\partial x_2} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \vdots & \vdots & \\ \frac{\partial h_m}{\partial x_1} & \frac{\partial h_m}{\partial x_2} & \cdots & \frac{\partial h_m}{\partial x_n} \end{pmatrix}.$$

Equations (2.54) and (2.55) neglect the cross-terms with higher order than one.

After calculating the Jacobian matrix of $\mathbf{J_f}$ and $\mathbf{J_h}$, EKF adopts a similar two-step procedure as in LKF to estimate the state of the system.

- The first step for state prediction follows the following two equations:

  $$\hat{\mathbf{x}}_k = \mathbf{f}(\hat{\mathbf{x}}_{k-1}), \tag{2.56}$$

  $$\mathbf{P}_k = \mathbf{J_f}(\mathbf{x}_{k-1})\mathbf{P}_{k-1}\mathbf{J_f}^T(\mathbf{x}_{k-1}) + \mathbf{Q}. \tag{2.57}$$

  Equation (2.56) predicts the current state from the previous moment based on the dynamic equation and Equation (2.57) projects the covariance estimate forward.

- The second step for correction based on the observations consists of the following three equations:

$$\mathbf{K}_k = \mathbf{P}_k \mathbf{J_h}^T(\hat{\mathbf{x}}_\mathbf{k})(\mathbf{J_h}(\hat{\mathbf{x}}_\mathbf{k})\mathbf{P}_k \mathbf{J_h}^T(\hat{\mathbf{x}}_\mathbf{k}) + \mathbf{R})^{-1}, \tag{2.58}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{z}_k - \mathbf{h}(\hat{\mathbf{x}}_k)), \tag{2.59}$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{J_h}(\hat{\mathbf{x}}_\mathbf{k})\mathbf{P}_k. \tag{2.60}$$

  Similar as in LKF, Equation (2.58) calculates the Kalman Gain. Equation (2.59) further adjusts the estimation of the current state. Equation (2.60) updates the state error covariance.

2.8 Particle Filter

The aforementioned LKF and EKF analytically find the solution based on the assumption that the posterior PDF is a Gaussian distribution because the first two moments. i.e., mean and variance, fully represent a Gaussian distributed state. To deal with a non-Gaussian posterior PDF, particle filters are adopted by using Monte Carlo simulation to represent the required posterior PDF by a set of random samples with *associated weights*. Additionally, the dynamic and observation equations in a particle filter can be nonlinear and noise in both equations is not necessary to be Gaussian. The main components of a particle filter consist of sequential important sampling and resampling.

2.8.1 Monte Carlo Methods

Before introducing sequential important sampling and resampling, we briefly introduce some basic knowledge about Monte Carlo Methods including density approximation from Monte Carlo samples and importance sampling.

Density Approximation from Monte Carlo Samples

Monte Carlo simulation adopts a set of discrete samples to approximate a probability distribution. With increasing the number of samples, the distribution/density estimated by Monte Carlo simulation approaches the original/target distribution.

As described in [63], given a set of samples, a multidimensional histogram density estimate is defined by

$$\hat{p}(\mathbf{x}) = \frac{v_i}{Nh^{n_x}}, \tag{2.61}$$

where v_i is the number of samples falling into the ith hypervolume bin, N is the total number of samples, h is the bin width and n_x means n_x dimensional. With

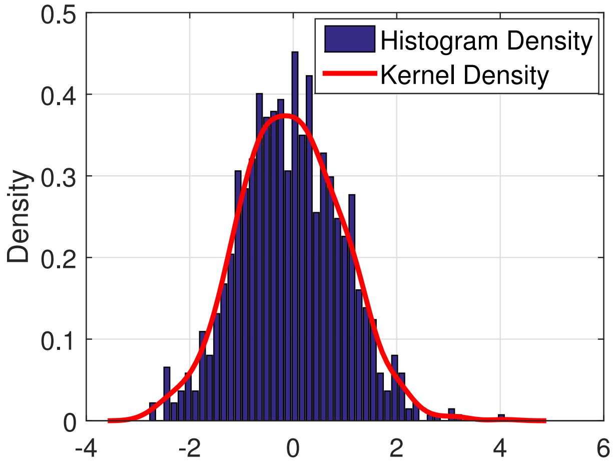


Figure 2.26: Histogram and Kernel Densities

Table 2.3: Common Univariate Kernel Functions [63]

Kernel	$K(u)$
Uniform	$\frac{1}{2}I(\|u\| \leq 1)$
Triangle	$(1-\|u\|)I(\|u\| \leq 1)$
Cosine	$-\frac{\pi}{4}\cos(\frac{\pi}{2}u)I(\|u\| \leq 1)$
Gaussian	$\frac{1}{\sqrt{2\pi}}\exp(-\frac{1}{2}u^2)$

this histogram density estimate, when the size of bin goes to zero, we rewrite the sample density function as

$$\hat{p}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{x}-\mathbf{x}^i), \tag{2.62}$$

where $\delta(\cdot)$ is the Dirac delta function and $\mathbf{x}^i$ is the ith sample. Figure 2.26 shows an one-dimensional histogram for Gaussian distribution. The histogram shown in Figure 2.26 has discontinuities at each bin boundary [63].

To smooth the histogram density, we convolves $\hat{p}(\mathbf{x})$ with a smoothing scaled kernel density function, $K_h(x)$ as [63]

$$\begin{aligned} p_K(\mathbf{x}) &= (\hat{p} * K_h)(\mathbf{x}) \\ &= \int_{\mathbf{R}^{n_x}} \hat{p}(\mathbf{u})K_{\mathbf{H}}(\mathbf{x}-\mathbf{u})d\mathbf{u} \\ &= \int_{\mathbf{R}^{n_x}} \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{u}-\mathbf{x}^i)K_{\mathbf{H}}(\mathbf{x}-\mathbf{u})d\mathbf{u} \\ &= \frac{1}{N}\sum_{i=1}^{N}K_{\mathbf{H}}(\mathbf{x}-\mathbf{x}^i), \end{aligned} \tag{2.63}$$

where the components of $\mathbf{H}$ are the bandwidth parameters of the kernel. This density approximation method is called Kernel Destiny Estimation (KDE). In KDE, there are multiple kernel functions and Table 2.3 lists some common univariate kernel functions. Among these kernel functions, Gaussian kernel is most commonly used, and the kernel density (one-dimensional) with Gaussian Kernel is estimated as

$$\begin{aligned} p_K(x) &= \frac{1}{N}\sum_{i=1}^{N} K_h(x - x^i) \\ &= \sum_{i=1}^{N} \frac{1}{N}\frac{1}{\sqrt{2\pi}h}\text{exp}[-\frac{1}{2}(\frac{x - x^i}{h})^2], \end{aligned} \tag{2.64}$$

where the bandwidth parameter h is the standard deviation of the Gaussian density. Figure 2.26 indicates the kernel density of the histogram density. We can find that the kernel density is more smooth than histogram density.

Importance Sampling

Monte Carlo methods provide a solution to estimate a density or distribution by a set of samples. One of the important usage for Monte Carlo simulation is density weighted integration [63]. Take the following integral as an example:

$$\mathbf{I}(\mathbf{x}) = \int \mathbf{f}(\mathbf{x})p(\mathbf{x})d\mathbf{x}. \tag{2.65}$$

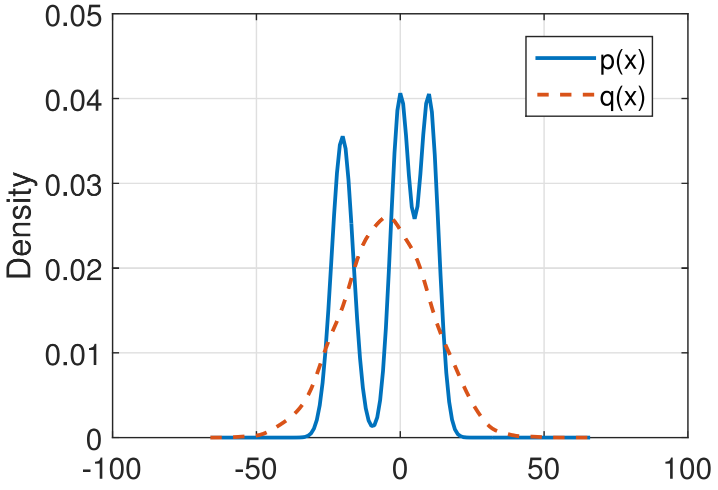
Consider the density $p(\mathbf{x})$ is Gaussian. We can replace $p(\mathbf{x})$ with a set of discrete samples (the same as Equation (2.62)) as

$$\begin{aligned} p(\mathbf{x}) &\simeq \frac{1}{N}\sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}^i) \\ &= \sum_{i=1}^{N} w_i \delta(\mathbf{x} - \mathbf{x}^i), \end{aligned} \tag{2.66}$$

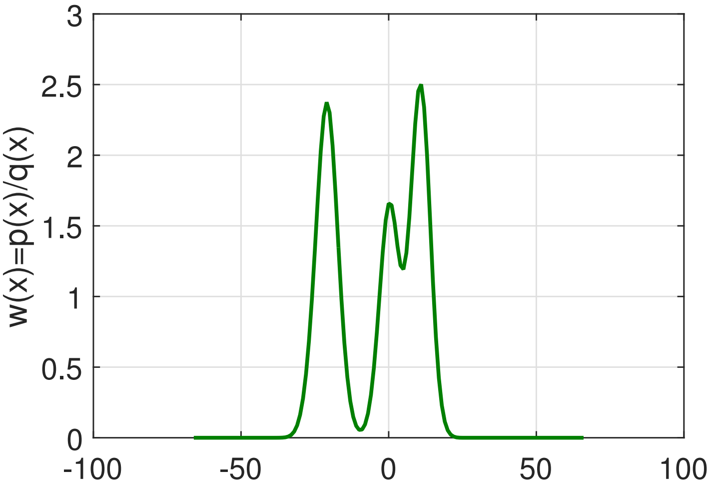
where $w_i = 1/N$. The integral in Equation (2.65) becomes

$$\begin{aligned} \mathbf{I}(\mathbf{x}) &\simeq \int \mathbf{f}(\mathbf{x}) \sum_{i=1}^{N} w_i \delta(\mathbf{x} - \mathbf{x}^i) d\mathbf{x} \\ &= \sum_{i=1}^{N} w_i \mathbf{f}(\mathbf{x}^i). \end{aligned} \tag{2.67}$$

$\mathbf{f}(\mathbf{x})$ can represent any moment generating function [63]. For example, the mean ($\bar{\mathbf{g}}(\mathbf{x})$) of a function $\mathbf{g}(\mathbf{x})$ is calculated if $\mathbf{f}(\mathbf{x}) = \mathbf{g}(\mathbf{x})$ leading to $\mathbf{I}(\mathbf{x}) = \bar{\mathbf{g}}(\mathbf{x})$.



(a) Densities



(b) Weights

Figure 2.27: Example of the Creation of $w(x)$

If we wish to compute the covariance of $\mathbf{g}(\mathbf{x})$, then we let $\mathbf{f}(\mathbf{x}) = [\mathbf{g}(\mathbf{x}) - \bar{\mathbf{g}}(\mathbf{x})][\mathbf{g}(\mathbf{x}) - \bar{\mathbf{g}}(\mathbf{x})]^T$ [63].

w_i does not represent the probability of $\mathbf{x}^i$ but represents its importance to the weighted sum approximation to the moment integral in Equation (2.67) [63]. Since in Equations (2.66) and (2.67), samples are directly from a known Gaussian density, all the samples are equally important to the sample mean and therefore they are with equal values $\frac{1}{N}$.

We can easily get samples from a known density as in Equation (2.66). However, if $p(\mathbf{x})$ is an unknown density or one that is difficult to sample, it becomes difficult to draw samples directly from $p(\mathbf{x})$. *Importance Sampling* is a key method in Monte Carlo methods to draw samples from an unknown density. Assume that $p(\mathbf{x})$ is an unknown density and $q(\mathbf{x})$ is a known density, e.g., a Gaussian density. Now we define a weighting factor [63] as

$$w(\mathbf{x}) = \frac{p(\mathbf{x})}{q(\mathbf{x})}. \tag{2.68}$$

There $p(\mathbf{x})$ is represented as a scaled version of $q(\mathbf{x})$ with different scaling factor at each $\mathbf{x}$. Now, Assuming that $q(\mathbf{x})$ is a Gaussian density, which is represented as Equation (2.66), $p(\mathbf{x})$ is written as

$$p(\mathbf{x}) \simeq \sum_{i=1}^{N} w_i \delta(\mathbf{x} - \mathbf{x}^i). \tag{2.69}$$

However, w_i in Equation (2.69) are no longer uniformly equal to $\frac{1}{N}$. Figure 2.27 indicates the density and weights, which clearly indicates that weights $w(x)$ are different from the density $p(x)$.

2.8.2 Sequential Important Sampling

Now, we consider the goal for HMM, which is mentioned at the beginning of Section 2.7, to calculate the "degree of belief" $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ (posterior PDF). Sequential importance sampling (SIS) is the most basic Monte Carlo method used for dynamic systems (HMM) [63]. The idea of the SIS algorithm is to approximate the posterior PDF $p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$ at time k-1 with a weighted set of samples $\{\mathbf{x}^i_{k-1}, w^i_{k-1}\}^{N_s}_{i=1}$, where $\mathbf{x}^i_{k-1}$ is the ith particle (sample), w^i_{k-1} is the *associated weight* and N_s is the total number of samples. These particles together with the associated weights are recursively updated to obtain an approximation of the posterior distribution at the next time, i.e., $\{\mathbf{x}^i_k, w^i_k\}^{N_s}_{i=1}$.

Since $p(\mathbf{x}_k|\mathbf{z}_{1:k})$ is an unknown PDF, we adopt the important sampling as mentioned in Section 2.8.1 to represent this unknown PDF as

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = w(\mathbf{x}_k) q(\mathbf{x}_k|\mathbf{z}_{1:k}), \tag{2.70}$$

where $q(\mathbf{x}_k|\mathbf{z}_{1:k})$ is the easily sampled importance density and the weighted function $w(\mathbf{x}_k)$ is defined by

$$w(\mathbf{x}_k) = \frac{p(\mathbf{x}_k|\mathbf{z}_{1:k})}{q(\mathbf{x}_k|\mathbf{z}_{1:k})}. \tag{2.71}$$

Using Baye's rule, we obtain that

$$p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{p(\mathbf{z}_k|p(\mathbf{z}_{1:k-1}))}, \tag{2.72}$$

where $p(\mathbf{z}_k|p(\mathbf{z}_{1:k-1}))$ is just a normalization term as mentioned in Section 2.7. The weights are written as

$$w(\mathbf{x}_k) \propto \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{z}_{1:k-1})}{q(\mathbf{x}_k|\mathbf{z}_{1:k})}. \tag{2.73}$$

Using the Chapman-Kolmogorov theorem [63] and considering $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_{k-1}) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_{1:k}) = q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$ in a first order HMM, we further rewrite the weights as follows

$$w(\mathbf{x}_k) \propto \frac{p(\mathbf{z}_k|\mathbf{x}_k) \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})d\mathbf{x}_{k-1}}{\int q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)q(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})d\mathbf{x}_{k-1}}. \tag{2.74}$$

Assume that we have a set of samples $\mathbf{x}_{k-1}^i$ from $q(\mathbf{x}_{k-1}|\mathbf{z}_{1:k-1})$. Equation (2.74) is written as

$$\begin{aligned} w(\mathbf{x}_k) &= \sum_{i=1}^{N_s} \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)p(\mathbf{x}_{k-1}^i|\mathbf{z}_{1:k-1})}{q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)q(\mathbf{x}_{k-1}^i|\mathbf{z}_{1:k-1})} \\ &= \sum_{i=1}^{N_s} w_{n-1}^i \frac{p(\mathbf{z}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_{k-1}^i|\mathbf{z}_{1:k-1})}, \end{aligned} \tag{2.75}$$

where

$$w_{n-1}^i = \frac{p(\mathbf{x}_{k-1}^i|\mathbf{z}_{1:k-1})}{q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)}. \tag{2.76}$$

Considering Equation (2.70) and generating samples $\mathbf{x}_n^i$ from $q(\mathbf{x}_k|\mathbf{x}_{k-1}^i, \mathbf{z}_k)$, we finally obtain that (please refer to [63] for more details)

$$\boxed{p(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{i=1}^{N} w_k^i \delta(\mathbf{x}_k - \mathbf{x}_k^i),} \tag{2.77}$$

where the weights are recursively updated as

$$\boxed{w_k^i \propto w_{k-1}^i \cdot \frac{p(\mathbf{z}_k|\mathbf{x}_k^i)p(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i|\mathbf{x}_{k-1}^i, \mathbf{z}_k)}.} \tag{2.78}$$

To get the mean of $\mathbf{x}_k$ with respect to $p(\mathbf{x}_k|\mathbf{z}_{1:k})$, we use Equation (2.67) as

$$\bar{\mathbf{x}}_k = \sum_{i=1}^{N} w_k^i \mathbf{x}_k^i. \tag{2.79}$$

One of the problems in the SIS particle filter is how to analytically define the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k)$, the transition density $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, and the importance density $q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k)$. There are several versions of particle filters proposed based on different kinds of importance density. One of the most widely used and efficiently implementable particle filter is *Bootstrap Particle Filter* (BPF) [63], in which the importance density is chosen to be equal to the transition density as

$$q(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{z}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1}). \tag{2.80}$$

Hence, the associated weights are calculated as

$$\boxed{w_k^i \propto w_{k-1}^i \cdot p(\mathbf{z}_k|\mathbf{x}_k^i)}, \tag{2.81}$$

in which the associated weights are only determined by the likelihood function of $p(\mathbf{z}_k|\mathbf{x}_k^i)$. An efficient and accurate derivation of the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k^i)$ is important for accurate tracking by BPF. Our tracking algorithms proposed in Chapters 6 and 7 are based on BPF, in which we work on improving the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k^i)$ to more accurately update the associated weights in Equation (2.81).

2.8.3 Resampling

In particle filters, the particles are propagated based on Equation (2.77) and (2.78) over and over again, with noise added at each iteration. This causes the estimated discrete posterior density based on the spread of the particles diverging from the true posterior density. The weights (importance) are reduced to zero for of all but a few central particles, which is called degeneracy problem [63]. Therefore, fewer and fewer particles have large enough weights (importance) to contribute the estimation of moments, e.g., mean, and correspondingly the variances of the moment estimates increase. A suitable measure of degeneracy is the effective sample size $N_{\text{eff}} = 1/\sum_{i=1}^{N_s}(w_k^i)^2$, where a smaller N_{eff} means a larger variance for the weights and correspondingly more degeneracy.

To deal with the sample degeneracy problem, resampling is used [63]. With resampling, the particles with large weights are replicated and the ones with negligible weights are removed. Then, equal weights are reassigned to the new particle set, which is $w_k^i = \frac{1}{N_s}$. More formally, before resampling, the posterior PDF and associated weights are as in Equation (2.77) and (2.78). After resampling, the posterior PDF becomes

$$\hat{p}(\mathbf{x}_k|\mathbf{z}_{1:k}) = \sum_{j=1}^{N_s} \frac{1}{N_s}\delta(\mathbf{x}_k - \mathbf{x}_k^{j*}) = \sum_{i=1}^{N_s} \frac{n_i}{N_s}\delta(\mathbf{x}_k - \mathbf{x}_k^i), \tag{2.82}$$

where n_i is the number of copies of particle $\mathbf{x}_k^i$ in the new set of particles $\mathbf{x}_k^{j*}$. Weights are uniformly equal to $\frac{1}{N_s}$.

There are different ways for resampling, such as *systematic resampling*, *residual resampling*, and *multinomial resampling*, in which the systematic resampling algorithm introduced in [63] is the most widely-used algorithm in the literature as it is easy to implement and outperforms other resampling schemes in most scenarios. Please refer to [63] for more details about resampling.

2.8.4 Dynamic Equations for Indoor Tracking

The dynamic equation, $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{v}_k$, in Bayesian estimation is used to update the first two moments of a Gaussian distribution in Kalman Filters or the particles in particle filter.

The Constant Velocity Model

A Constant Velocity (CV) model is commonly used as the dynamic equation for target tracking. In this model, the state vector is defined as

$$\mathbf{x} = [x, y, \hat{x}, \hat{y}]^T, \tag{2.83}$$

where (x, y) are the Cartesian coordinates of the target and $(\hat{x}, \hat{y})$ is a two-dimensional moving speed vector. Under the CV model, the prediction function is written as

$$\mathbf{x}_k = \mathbf{F} \cdot \mathbf{x}_{k-1} + \boldsymbol{\eta}\mathbf{w}, \tag{2.84}$$

where

$$\boldsymbol{\eta}_{\mathrm{CV}} = \begin{pmatrix} \Delta T^2/2 & 0 \\ 0 & \Delta T^2/2 \\ \Delta T & 0 \\ 0 & \Delta T \end{pmatrix}, \mathbf{F}_{\mathrm{CV}} = \begin{pmatrix} 1 & 0 & \Delta T & 0 \\ 0 & 1 & 0 & \Delta T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

ΔT is the time interval between two subsequent estimations of the target location and $\mathbf{w}$ is a 2×1 independent and identically distributed (i.i.d.) process noise vector.

The Coordinated Turn Model

The aforementioned CV model does not consider the relation between the two components $(\hat{x}, \hat{y})$ and has low tracking accuracy especially when there is a sudden change in the moving direction of the target. The authors of [88] first proposed to adopt a multi-model including a CV model and two coordinated turn models in range-based bootstrap particle filter for positioning, which is referred to as MM-BPF (Multi-Model BPF). Multi-model has been widely investigated in aircraft tracking [22]. In MM-BPF, an integer parameter $r(t)$ (also called regime) is used to switch between different models. Since there are three models in MM-BPF, $r(t)$ is chosen from $[1, 2, 3]$ and modelled as a Markov chain with transitional probability matrix with $\pi_{mn} = \text{Prob}\{r(t) = m|r(t-1) = n\}$. With different $r(t)$, $\mathbf{F}$ is no longer a determined matrix but varies with time. If $r(t) = 1$, the CV model is used ($\mathbf{F}_1 = \mathbf{F}_{\text{CV}}$). If $r(t)$ = 2, the coordinated turn model is used to model turning in the anticlockwise direction as

$$\mathbf{F}_2 = \begin{pmatrix} 1 & 0 & sin(\Delta T\omega)/\omega & (cos(\Delta T\omega)-1)/\omega \\ 0 & 1 & (1-cos(\Delta T\omega))/\omega & sin(\Delta T\omega)/\omega \\ 0 & 0 & cos(\Delta T\omega) & -sin(\Delta T\omega) \\ 0 & 0 & sin(\Delta T\omega) & cos(\Delta T\omega) \end{pmatrix}. \tag{2.85}$$

For $r(t)$ = 3, the coordinated turn model is used to model turning in the clockwise direction as

$$\mathbf{F}_3 = \begin{pmatrix} 1 & 0 & sin(\Delta T\omega)/\omega & (1-cos(\Delta T\omega))/\omega \\ 0 & 1 & (cos(\Delta T\omega)-1)/\omega & sin(\Delta T\omega)/\omega \\ 0 & 0 & cos(\Delta T\omega) & sin(\Delta T\omega) \\ 0 & 0 & -sin(\Delta T\omega) & cos(\Delta T\omega) \end{pmatrix}. \tag{2.86}$$

ω is the turning rate in rad/s, which is considered to be constant and needed to be predefined. In MM-BPF, before updating the particles for **x**, the regime $r(t)$ needs to be estimated to determine the model **F**. Therefore, another set of particles are sampled according to the transitional probability matrix π_{mn} and used to estimate the regime $r(t)$.

2.8.5 Application of Bayesian Filters in Radio-based Tracking

Bayesian filters have been investigated in both fingerprinting algorithms [117, 69] and range-based positioning algorithms to track users [108, 110].

- The fingerprinting methods introduced in Section 2.4, including KNN in RADAR and the probabilistic method in HORUS, FIFS and CSI-MIMO, are designed for stationary target and do not consider the movement of target. The authors of [117] adopt particle filters to smooth the output of KNN for outdoor tracking in cellular networks. They compared their proposed KNN/PF method (combining KNN and PF) to KNN and KNN/LKF (combining KNN and LKF). The experimental results show that after smoothing the tracking path, KNN/PF outperforms KNN/LKF and KNN.
- The authors of [108] experimentally evaluated multiple versions of Kalman filters using RSSI-based ranging in a wireless sensor network. Their experiment results showed that Kalman filters allow for sufficiently accurate localization results and are robust against inaccurate range measurements. We found few works to investigate particle filters exclusively using power-based ranging for indoor positioning. The authors of [110] investigated particle filters using RSSI-based ranging in an outdoor environment but their results showed an accuracy of $4m$ to $6m$, which is not accurate enough for indoor tracking.

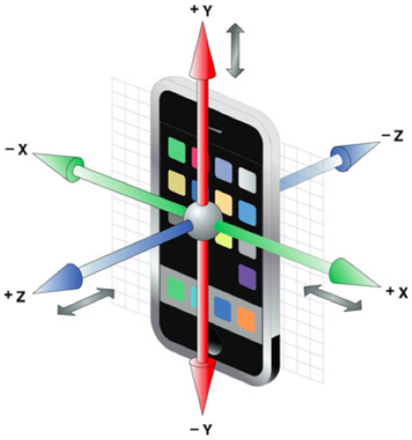


Figure 2.28: Accelerometer in 3 Dimensions

2.9 Inertial Sensor based Tracking

Besides radio-based indoor positioning methods, inertial sensors have attracted much research [66, 104, 64] in indoor tracking due to the fast development of smart phones. In this section, we review some indoor tracking work based on inertial sensors.

2.9.1 Inertial Sensors

Nowadays, various inertial sensors consisting of accelerometers, magnetometers, and gyroscopes have been integrated into modern smart phones.

Accelerometer

Accelerometers in a smart phone are used to measure the acceleration applied to the device. The acceleration $(\mathbf{A'}_d)$ are measured by the forces on the sensor $(\mathbf{F})$ as

$$\mathbf{A'}_d = -\sum \frac{\mathbf{F}}{m}, \tag{2.87}$$

where m is the mass of the sensor. Note that the force of gravity is always applied on the device and therefore the measured acceleration needs to be corrected as

$$\mathbf{A}_d = -\mathbf{g} - \sum \frac{\mathbf{F}}{m}, \tag{2.88}$$

where $\mathbf{g}$ is the gravity acceleration. The acceleration is a vector value in 3 dimensions as shown in Figure 2.28. When the device is lying on a table, the accelerometer reports $|\mathbf{A}_d| = |g| = 9.81m/s^2$. Similarly, when the device is in free fall, the accelerometer reports $|\mathbf{A}_d| = 0m/s^2$. Therefore, to measure the real acceleration of the device, the contribution of the force of gravity must be removed from the accelerometer data.

Magnetometer

A magnetometer is used to measure the strength of the magnetic field with a unit of Tesla in the surrounding area of the device. For example, the Earth's magnetic field on the equator is $31\mu T$ and a typical fridge magnet is $5mT$. When the device is lying on a table, its pointing head direction can be estimated by using magnetometer. The computed direction is indicated in degrees like 0 North, 90 East, 180 South and 270 West. The problem by magnetometer measuring is mainly the errors caused by iron materials in the environment [53].

Gyroscope

Generally, a gyroscope is a sensor for measuring the rotation rate of the device's orientation. This rate of rotation can be measured along any of the three axes expressed in $\mathrm{rad/s}$. The rotation around the x, y and z-axis is also referred to as roll, pitch and yaw. For example, when the device is lying on a table, its axes indicate an angular velocity of zero.

2.9.2 Pedestrian Dead Reckoning

Pedestrian dead reckoning is a radio-free positioning method, which determines the positions of people based on inertial sensors such as accelerometers, gyroscopes and magnetometers. The principle of PDR is to update the positions of the subject after a step has been taken. The update is performed recursively using the last known position. A PDR algorithm normally consists of two components: moved distance estimation and heading orientation estimation.

Moved Distance Estimation

Moved distance is the displacement of a person between sequential sampling time intervals. It can be estimated based on two procedures: step detection and step length estimation.

- **Step detection** using inertial sensors has been investigated for years. Steps are often detected by analyzing the outputs of accelerometers [31, 65, 84]. Considering a walking person who flat holds a smart phone, every step generates one peak and dip in the measured vertical acceleration from the accelerometer, which form the basis of the peak detection method of step counting. The peak detection method analyzes a window (or buffer) of sensor values and determines a step based on the detection of a peak. This determination is usually based on threshold values. Peak detection [68, 105] and zero-crossing counting [26, 58] are two commonly used methods to find specific steps based on low-pass filtered accelerometer signals. Additionally, the cycle nature of walking leads to periodicity in the sensor data. Therefore, autocorrelation [25] and spectral analysis [104] methods are used to detect this periodicity respectively in time domain and frequency domain for step detection.

- **Step length** are estimated to determine the distance travelled in a detected step. One coarse way for step length is to map the averaged step length to height and gender of a person. However, step lengths normally vary over people and an average step length will inevitably introduce errors accumulating over a number of estimates. Another approach to estimate step length is based on accelerometers. Different step lengths generate different variance between the peak and dip values in the measured acceleration values. The authors of [121] proposed the following equation to estimate the step length

  $$l = K(\hat{a}_{v,max} - \hat{a}_{v,min})^{1/4}, \tag{2.89}$$

  where l is stride length, $\hat{a}_{v,max}$ and $\hat{a}_{v,min}$ are the peak and dip values of $\hat{a}_v$ on each step respectively, and K is a coefficient calibrated for individuals. This method has been widely used in PDR systems and also used in our work (Chapter 7) to estimate the step length.

Heading Orientation Estimation

Most of the approaches to estimate heading orientation is based on magnetometers. If the smart phone remains flat, heading orientation can be computed from the arctangent of the ratio of the two horizontal magnetic field components. Since, in general, the smart phone has an arbitrary orientation, accelerometer values are normally investigated to compensate this tilt. In modern smart phones, electronic compasses are normally integrated, which adopts 3-axis accelerometer and magnetometer sensors. The heading orientation is a function of all three accelerometer readings and all three magnetometer readings. Please find more details about the electronic compass in [99]. Heading orientation estimation based on magnetometers is normally influenced by structural elements of the building and electronic devices. Gyroscope is another sensor, which can estimate the heading orientation but it is prone to accumulative errors. Some approaches have been proposed to fuse both gyroscope and compass values, which can compensate the noise from the magnetometers and accumulative errors from gyroscopes [68].

2.10 Hybrid Approaches of Inertial Sensors and Radio-based Indoor Positioning

Instead of using standalone radio signals or inertial sensors for indoor positioning, numerous hybrid methods have been proposed to take advantage of both radio signals and inertial sensors for high positioning accuracy. The main advantages for these hybrid approaches are 1) mitigating accumulative errors in PDR, 2) facilitating training in radio based positioning, and 3) improving tracking accuracy.

Wap is a system [64], which uses a particle filter to fuse inertial sensor information and RSSI of WiFi signals for tracking. Wap relies on PDR algorithms to update the particles in the particle filter. As mentioned before, PDR is prone to accumulated errors. Therefore, the relative changes of RSSI is used to discover the direction changes and improve room distinguishing algorithms, which are used to resample the particles and mitigate the accumulative errors.

Zee is a SLAM (Simultaneous Localization And Mapping) system [104] with WiFi signals. The system adopts inertial sensors with the constraints imposed by the map to filter out infeasible locations over time and converging on the true location. Concurrently with estimating location with PDR, Zee also scans and records RSSI information of WiFi signals. As soon as the location information is available, the system builds up the radio map with the recorded WiFi RSSI training set. Therefore, Zee achieves a calibration free WiFi-based positioning system.

Inertial sensors and RSSI fingerprinting are fused by a particle filter in [50]. In this work, the authors adopted PDR to estimate the relative locations between sequential time intervals and used these information to update the particles in the particle filter. Furthermore, the RSSI measurements are further used to construct the observation vector to correct the particles based on a pre-calibrated radio map.

2.11 Conclusions

In this chapter, we review the background and relative work of indoor positioning. In Section 2.2, we introduce three wireless communication systems including GSM, IEEE 802.15.4 and IEEE 802.11, especially targeted at the physical layer signal processing. In Section 2.3, software defined radio techniques and the decoding methods in SDR for the aforementioned three wireless communication systems are introduced. In Section 2.4, we introduce the range-free algorithms including fingerprinting and centroid algorithms. In Section 2.5, we review ranging methods based on three radio parameters including RSSI, time information and channel information. In Section 2.6, the range-based positioning methods for stationary targets including trilateration and multilateration are reviewed including NLS, LLS, WLS and CWLS. In Section 2.7, we review Kalman filters for tracking mobile targets including LKF and EKF. In Section 2.8, we introduce some key techniques in particle filters including Monte Carlos methods, sequential importance sampling and resampling. In Section 2.9, we introduce the methods and related work for indoor positioning based inertial sensors. In Section 2.10, we introduce some hybrid approaches to combine the radio-based positioning and PDR.

Part I

Time-based Indoor Positioning

Nowadays, radio-based indoor positioning has become an emerging research area because of the increasing commercial demands for location-based services. Although LTE and WiFi techniques with wide bandwidth have become the most emerging techniques for data communication, narrow-band signals are still widely used in many applications. For example, GSM signals with a bandwidth of 200KHz is still widely adopted by end users in cellular networks, especially in less developed countries. IEEE 802.15.4, e.g., ZigBee, is a wireless standard with narrow bandwidth (2MHz) for short-range communication. It is widely used in home automation, wireless sensor network and industry applications. It has attracted interests of researchers as an alternative to WiFi positioning. Therefore, in Part I (Chapter 3 and 4), we investigate how to locate a narrow-band signal emitter by time information. In Chapter 3, we investigate two key problems in time-based indoor positioning: designing timestamps and evaluating synchronization between ANs. We provide our enhanced mechanisms to improve the resolution of timestamps by extracting physical layer information, which is so called timing error in time recovery. We further investigate and evaluate the performance of GPS synchronization in depth by using the timing error information in time recovery. The timestamps and findings for GPS synchronization in Chapter 3 form the basis of the time-based positioning algorithms designed in Chapter 4. In Chapter 4, we propose to combine DTDOA together with GPS synchronization to completely eliminate the momentary synchronization offset. Then, we propose both multilateration and fingerprinting algorithms based on DTDOA. In this part, our algorithms are evaluated in both indoor and outdoor environments on two positioning testbeds for GSM and IEEE 802.15.4 signals.

Chapter 3

Timestamps and Synchronization for Narrow-band Signals

3.1 Introduction

To achieve time-based positioning for narrow-band signals, the first challenge is the accuracy of timestamps, which is influenced by two factors namely bandwidth of signals and multipath propagation. At physical layer, narrower bandwidth leads to longer duration of a signal's symbol. As mentioned in Section 2.5.1, a timestamp is given on the time when a reference symbol in a packet is received. Due to the long symbol duration for narrow-band signals, an accurate timestamp needs to correctly estimate the arrival time of a signal within one symbol duration. For example, the bandwidth of a GSM signal is 200KHz and the symbol duration is $5\mu s$. Therefore, an accurate timestamp is required to estimate the arrival time of a GSM signal within $5\mu s$. Additionally, replicas of the original signal from different propagation paths arrive at the receiver with different delays. Due to the long symbol duration, the waveforms of original signal from the direct path and replicas from reflection paths overlap with each other within one symbol duration. Therefore, it is very challenging to generate the timestamp on the signal from the direct path because of the multipath effects (as introduced in Section 2.5.1).

Furthermore, we are interested in passive positioning systems, which can produce user location information for third-party providers of positioning services and monitor multiple users' locations simultaneously. To achieve these passive positioning systems, synchronization between anchor nodes is a critical challenge for time-based positioning. Because the target does not participate in the positioning process in a passive positioning system, RTT can not be utilized, which requires two-way packet exchanges between the target and ANs. TDOA is one of the promising time-based positioning algorithms for passive positioning systems. In reality, the process is challenged by imperfect time synchronization among ANs due to the high propagation speed of radio signals. For passive indoor positioning systems, ANs need to be distributed in the area of interest. In a large testing area where ANs can not be synchronized by a shared cable, GPS offers the most promising synchronization solution as introduced in Section 2.5.1. Indoor deployed ANs can be synchronized with outdoor antennas. Before utilizing the GPS synchronization for TDOA, we quantify the achieved degree of synchronization. To quantify the synchronization accuracy between two ANs, the easiest way is to compare their timestamps for the same receiving message emitted by an equidistant reference node. However, because of the limited accuracy of timestamps, it is still challenging to accurately evaluate the degree of synchronization between GPS synchronized ANs.

In this chapter, we address two challenges in a TDOA positioning system for narrow-band signals.

- First, we investigate how to design a high-resolution timestamp to accurately estimate the arrival time of a signal within one symbol duration. The method is based on a normalized timing error information in the signal processing procedure (time recovery) for narrow-band signals, e.g., GSM and IEEE 802.15.4. With this method, we are able to achieve timestamps with nanosecond resolution.

- Second, we further exploit this normalized timing error information to quantify the clock offsets and skews among GPS synchronized receivers. Additionally, the measurements of clock offsets are typically noisy in practice. It is challenging to calculate its derivation, i.e., clock skew. We propose a mechanism with two steps to accurately calculate the clock skews from noisy measurements of clock offsets. First, we adopt Savitzky-Golay (S-G) filter [107] to smooth the noisy measured clock offsets. Second, we calculate the derivation of the filtered clock offsets as the clock skews.

We have conducted a set of comprehensive experiments to evaluate our designed timestamps and GPS synchronization accuracy. Through our experiments, we find that our proposed timestamps achieve nanosecond resolution but it is still prone to multipath propagation because of the narrow-band property of GSM signals. For synchronization, we find that two GPS synchronized devices have up to $171ns$ synchronization offsets, which set a limit on positioning accuracy to tens of meters for TDOA. However, we find that clock skews between the two GPS synchronized devices are negligible. All the achievements and findings in this chapter form the basis to achieve time-based positioning for narrow-band signals, which is introduced in Chapter 4.

In the remainder of this chapter, Section 3.2 introduces the method for timestamp design by using a time recovery method. The method to exploit the time recovery method for synchronization evaluation is described in Section 3.3. A SDR testbed to mimic the physical layer of GSM signals is introduced in Section 3.4, in which the timing error information is extracted for timestamps and synchronization evaluation. The evaluation results for synchronization are presented in Section 3.5. Section 3.6 concludes this chapter.

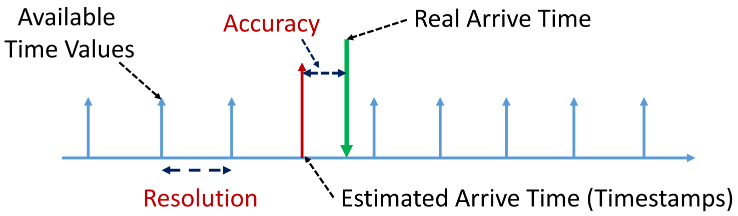


Figure 3.1: Resolution vs Accuracy

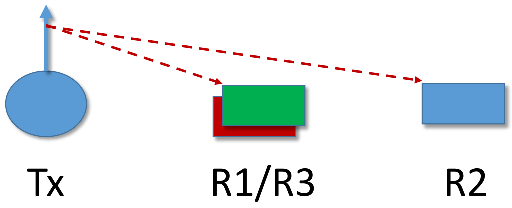
3.2 Nanosecond Timestamps via Time Recovery

Accurate timestamps are the prerequisite to achieve high positioning accuracy based on time information. Before introducing timestamps, we clarify two definitions, *accuracy* and *resolution* of timestamps. As shown in Figure 3.1, accuracy of a timestamp is defined as how long the estimated arrival time shifts from the real arrival time of a signal. In practice, an arrival signal is timestamped based on discrete clock samples. The resolution of a timestamp is defined as the time interval between successive time values (blue samples in Figure 3.1), which the clock can give. For example, for a clock with frequency of 2MHz, the time interval between two successive clock samples (time values) is $0.5\mu s$ and hence the resolution of the timestamp relying on this clock is $0.5\mu s$. Therefore, a high resolution is a prerequisite for high accuracy of a timestamp.

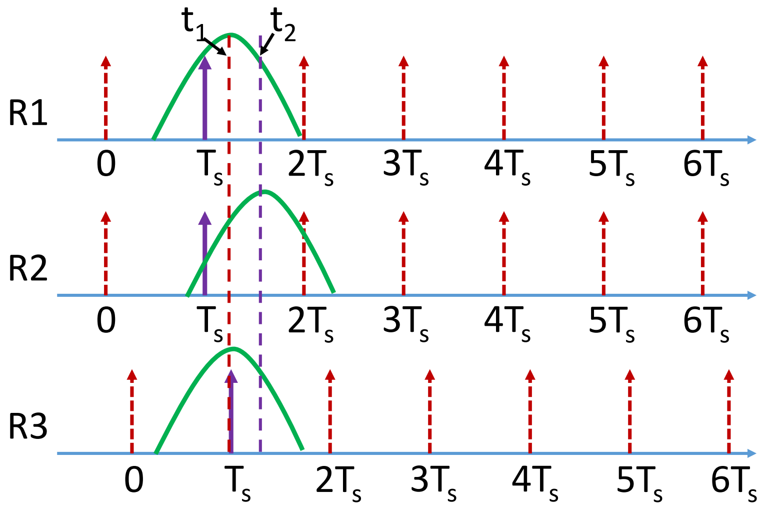
3.2.1 Sample-based Timestamps

The timestamp of a received packet is defined as the time of receiving one reference symbol in the packet. In a fully digital communication system, an Analog-to-Digital Converter (ADC) is used to convert the analog signal to digital samples at the receiver side. Considering the case of one sample per symbol for digitalization at the receiver side, the sample-based timestamp for the received packet is defined as the time for the sample, which represents the reference symbol. In a digital receiver, the output samples from ADC are constantly aligned with a sample interval of T_s. Therefore, the resolution of this sample-based timestamp is T_s.

To better introduce the influence of sampling frequency on the time estimation and synchronization evaluation, we take the scenarios in Figure 3.2 as an example. The transmitter in Figure 3.2(a) sends a packet to receivers 1, 2, and 3. Receivers 1 and 3 are co-located with each other, which are closer to the transmitter than receiver 2. Figure 3.2(b) shows the digital samples inside each receiver. In Figure 3.2(b), we find that receivers 1 and 2 are perfectly synchronized with each other but receivers 1 and 3 have a synchronization offset. Considering receivers 1 and 2, because the signal propagation distances from the transmitter to receivers 1 and 2 are different, the reference symbol arrives at receiver 1 at t_1 and at receiver 2 at t_2. However, because $t_2 - t_1 < T_s$ meaning that the difference between the time delays from the target to the two receivers is smaller than the resolution, the



(a) Transmitter and Receivers Setup



(b) Digital Samples

Figure 3.2: Sample-based Timestamps and Sub-sample Timestamps

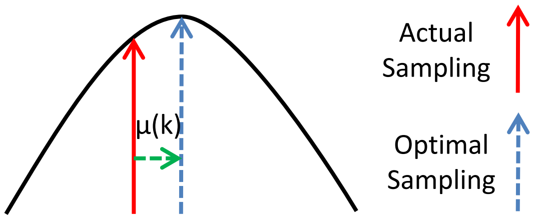


Figure 3.3: Sub-sample Timestamps via Time Recovery

sample-based timestamps generated at these two receivers can not distinguish such small time difference and issue T_s as their sample-based timestamps for this packet. Furthermore, considering receivers 1 and 3, which are co-located with each other, the reference symbol arrives at these two receivers at the same time. Because there is a synchronization offset between receivers 1 and 3, these two receivers should issue two different timestamps for this reference symbol and the synchronization offset can be calculated by subtracting these two timestamps. However, because the synchronization offset is smaller than T_s, the low resolution of sample-based timestamps can not distinguish such small time difference and both two receivers give the same timestamp T_s to this packet. Therefore, it is not able to calculate the synchronization offset based on this sample-based timestamp.

3.2.2 Sub-sample Timestamps

If a timestamp is given on the peak of the reference symbol's pulse shape (t_1 and t_2 in Figure 3.2(b)) instead of each digital sample (red arrows in Figure 3.2(b)), the different time delays from the transmitter to receiver 1 and 2 can be distinguished. Because this timestamp achieves sub-sample resolution, we refer to this timestamp as sub-sample timestamp. To achieve sub-sample timestamps, we calculate the fractional time delay from the digital sample to the peak of the pulse shape (optimal sampling position).

As shown in Figure 3.3, the digital samples are often before or after the peak, displaced at $\mu(k)$. $\mu(k)$ is the normalized timing error given by

$$\mu(k) = \frac{\Delta T(k)}{T_s}, \tag{3.1}$$

where k is the sample number, $\Delta T(k)$ is the offset between actual and optimal sampling position, i.e., peak of the pulse shape. Estimation of $\mu(k)$ is required to calculate the sub-sample timestamps.

As mentioned in Section 2.3.2, a method called time recovery in digital communication systems can be adopted to estimate the normalized timing error $\mu(k)$. Time recovery [89] as shown in Figure 2.18 is often used in receivers to correct the shift in the sampling position during signal recovery. The method synchronizes the sampler with the pulses of the received analog waveform. The shapes of the pulses depend on the signals and the corresponding demodulation methods. For example, for GSM signals with a GMSK modulation scheme, the pulse has a Gaussian shape. For IEEE 802.15.4 signals, a MSK receiver is used to decode the signals and the pulse is with a sine shape. Both shapes are symmetric. In the time recovery method, a Timing Error Detection (TED) module is used to extract a timing error term e_k (k is the sample number) between the actual and optimal sample positions based on the symmetric property of the pulse. Different methods have been proposed to calculate this timing error term, such as the early-late gate algorithm, Mueller and Muller algorithm, and Gardner algorithm [89]. We work on the Mueller and Muller algorithm to calculate the timing error term as

$e_k = (y_k * \hat{y}_{k-1}) - (\hat{y}_k * y_{k-1})$, where y_k is the value before the decision and $\hat{y}_k$ is the value after the decision in Figure 2.18. The timing error term e_k is passed to a loop filter, which outputs the normalized timing error $\mu(k)$ to decide on the correction of the sampling time in the re-sampler. Subsequently, the sampling position is adjusted to be closer to the optimal one.

The normalized timing error $\mu(k)$ of sample k is adopted to design a sub-sample timestamp. Once the receiver starts to receive packets and generates samples, we count the generated samples and obtain the sample-based timestamp $T'(k)$ for the kth sample as follows:

$$T'(k) = T'(1) + T_s * (k-1). \qquad (3.2)$$

$T'(1)$ is the sample-based timestamp for the first sample. In Equation (3.2) the resolution of the sample-based timestamp is limited by T_s. With $\mu(k)$ obtained by the time recovery, we improve the timestamps as

$$T(k) = T'(k) + \mu(k) \cdot T_s, \qquad (3.3)$$

where $T(k)$ is the sub-sample timestamp. Now, assuming that the kth sample in the sample stream is the sample representing the reference symbol of the received packet, $T(k)$ is the sub-sample timestamp for this packet.

3.2.3 Influence of Multipath Propagation on Sub-sample Timestamps

To calculate sub-sample timestamps, one important assumption is that the pulse shape of the received symbol is symmetric. As mentioned in Section 2.2.1, both the Gaussian shape in GMSK and the half-sine shape in MSK are symmetric. In an indoor environment, multiple replicas of the original signal arrive at the receiver with very short delays after the signal from the direct path as shown in Figure 1.3. All these signals from refection paths, whose delays from the direct path are shorter than the symbol duration, overlap with the signal from the direct path and this results in the distortion of the received pulse shape. The received pulse shape is not symmetric (Figure 1.3) and this introduces an error to the estimated timestamps.

3.3 Quantify GPS Synchronization via Time recovery

In Section 3.2, we have introduced how to use the timing error information from time recovery to improve timestamps. In this section, we exploit time recovery to achieve highly accurate calculation of the synchronization misalignment among GPS synchronized nodes. We derive the relation between the normalized timing error $\mu(k)$, the clock offset and skew.

3.3.1 Relative Clock Offset without Long Term Clock Drift

It is commonly known that GPS-synchronized clocks can prevent long-term clock drift by periodically calibrating to the signals from satellites. Considering that GPS synchronization offsets are typically on the level of sub-microseconds and the sampling interval T_s is in the range of microseconds, the clock offsets between two GPS-synchronized receivers are always smaller than the sampling interval T_s at any moment. In such case, we derive the relation between the relative clock offset and the relative normalized timing error. Based on Equation (3.3) and the definition of relative clock offset (introduced in Section 2.5.1), we obtain the relative clock offset between the ith and jth receivers as

$$\begin{aligned}\Delta C_{ij}(k) &= T_i(k) - T_j(k)\\ &= (T_i'(k) + \mu_i(k) \cdot T_s) - (T_j'(k) + \mu_j(k) \cdot T_s).\end{aligned} \tag{3.4}$$

Consider the aforementioned case that the relative clock offsets of the GPS synchronized devices are always smaller than the sampling interval T_s. As the example introduced in Section 3.2.1, when the clock offset is smaller than the resolution of the sample-based timestamps, two receivers give the same sample-based timestamps to the received packet and hence we get that $T_i'(k) = T_j'(k)$ for the same sample number k. Therefore, we further simplify Equation (3.4) as

$$\begin{aligned}\Delta C_{ij}(k) &= \mu_i(k) \cdot T_s - \mu_j(k) \cdot T_s\\ &= \Delta\mu_{ij}(k) \cdot T_s,\end{aligned} \tag{3.5}$$

where $\Delta\mu_{ij}(k) = \mu_i(k) - \mu_j(k)$. In the above equation, T_s is constant and, thus, the relative clock offset is determined by $\Delta\mu_{ij}(k)$.

Furthermore, considering Equation (2.19) and $t = k \cdot T_s$, we calculate

$$\begin{aligned}\Delta C_{ij}(t) &= T_s \cdot \Delta\mu_{ij}(t)\\ &= \int_0^t \Delta f_{ij}(t)\mathrm{d}t + \Delta\theta_{ij},\end{aligned} \tag{3.6}$$

where $\Delta\theta_{ij}$, the relative initial clock offset between AN_i and AN_j, is constant. Therefore, we can analyze the short-term (momentary) clock offsets through the relative normalized timing error $\Delta\mu_{ij}(t)$. If there is no short-term clock offset, $\Delta\mu_{ij}(t)$ is constant. Otherwise, $\Delta\mu_{ij}(t)$ fluctuates. We use this to identify the presence of short-term clock offsets by monitoring the behaviour of $\Delta\mu_{ij}(t)$.

3.3.2 Relative Clock Skew via Savitzky-Golay Filter

Here we derive how to obtain the relative clock skew between two ANs. Based on Equation (3.6), we obtain the relative clock skew in an ideal case as

$$\Delta f_{ij}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\big(\Delta C_{ij}(t)\big) = T_s \cdot \frac{\mathrm{d}}{\mathrm{d}t}\big(\Delta\mu_{ij}(t)\big). \tag{3.7}$$

The relative clock skew is calculated by the derivative of $\Delta C_{ij}(t)$ or $\Delta\mu_{ij}(t)$.

However, in practice, the measurement of $\Delta\mu_{ij}(t)$ is noisy and it is hard to accurately measure its derivative. Therefore, in our work, we propose to estimate the relative clock skew in two steps.

At the first step, a S-G filter [71] is applied to smooth the measured $\Delta\mu_{ij}(t)$ and suppress the influence of noise. The S-G filter applies a moving window smoothing technique based on least squares polynomial fitting. We take the group of $2M+1$ samples of $\Delta\mu_{ij}(t)$ centered at n, which is moving from 0 to the end of the samples. We obtain the coefficients of a polynomial,

$$\Delta\mu'_{ij}(n) = \sum_{j=0}^{N} a_j n^j, \tag{3.8}$$

which minimizes the mean-squared approximation error,

$$\begin{aligned}\varepsilon_N &= \sum_{n=-M}^{M} (\Delta\mu'_{ij}(n) - \Delta\mu_{ij}(n))^2 \\ &= \sum_{n=-M}^{M} (\sum_{j=0}^{N} a_j n^j - \Delta\mu_{ij}(n))^2,\end{aligned} \tag{3.9}$$

where N is the order of polynomial and $2M+1$ is the size of the moving window. The coefficients $[a_0, a_1, ..., a_N]$ are adaptive in each window to find the optimal polynomial to fit the samples in the window. $\Delta\mu'_{ij}(n)$ is the output of the filter and sampled $\Delta\mu'_{ij}(t)$ with sampling space of T_s.

At the second step, the derivative of the filtered value $\Delta\mu'_{ij}(t)$ or $\Delta C'_{ij}(t)$ is calculated to obtain the relative clock skew as follows,

$$\Delta f_{ij}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\big(\Delta C'_{ij}(t)\big) = T_s \cdot \frac{\mathrm{d}}{\mathrm{d}t}\big(\Delta\mu'_{ij}(t)\big). \tag{3.10}$$

3.4 A SDR-based Testbed for GSM-like Signals

To quantify the clock offset and skew between GPS synchronized receivers based on the timing error information and investigate the time-based positioning algorithms for GSM signals, we have set up a testbed based on Software Defined Radio (SDR) devices, in which we construct our own transmitter to continuously generate GMSK-modulated signals with 250KHz symbol rate. At the receiver side, the sampling rate is 500KHz (two samples per symbol) resulting in a sample interval of $2\mu s$. The signal is with the same modulation scheme as GSM but operated in the ISM band (433MHz). In the remainder of this thesis, we refer to this testbed as the GSM-like testbed.

3.4.1 Hardware

We use the embedded USRP E110 model [16] as the signal transmitter. USRP E110 integrates an ARM (Advanced RISC Machine) processor running a Linux operating system, which does not need to connect to another host machine, i.e., a PC, for signal processing. We choose USRP E110 as the transmitter because of its mobile convenience. The drawback of USRP E110 is limited computation capability because of the ARM processor. However, the signal processing effort in the transmitter is not high and USRP E110 is powerful enough. The receivers require much higher computation capability than the transmitter. Therefore, we use the networked USRP N210 models [17] for receivers, which are connected to a PC by Gigabit Ethernet. The signal processing is implemented in a desktop or laptop PC.

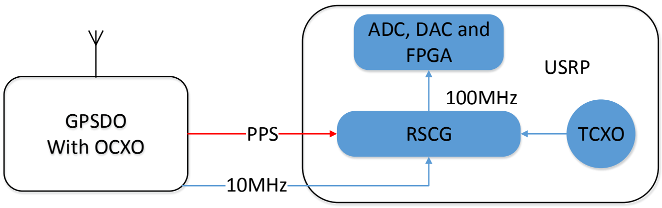
The operations of USRP N210 as ANs are controlled by Reference and System Clock Generation modules (RSCGs) as shown in Figure 3.4(a). Ettus Research provides a GPS Disciplined Oscillator (GPSDO) kit [15] with an Oven-Controlled Crystal Oscillator (OCXO), allowing the device to synchronize to GPS signals.

3.4.2 Software Realization

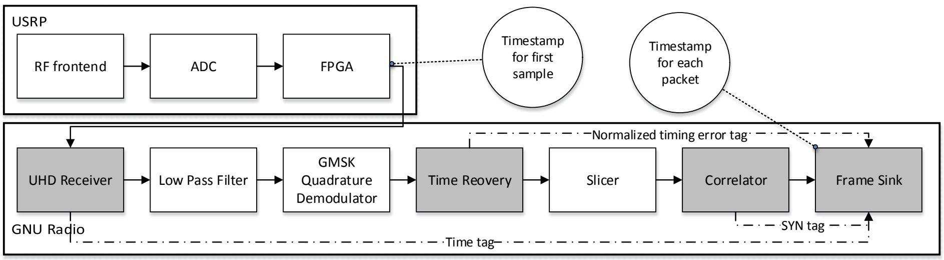
We use the components in GNU radio software to construct the transmitter, which continuously generates GMSK-modulated signals as introduced in Section 2.3.2. The structure is shown as in Figure 2.17. For the receiver, we also mainly rely on the components in GNU radio software to construct the GMSK receiver. Figure 3.4(b) illustrates the components of the SDR system in detail and please find the functionality of each component in Section 2.3.2.

The main task in this testbed is to extract the normalized timing error information from the time recovery components to design the sub-sample timestamps and evaluate the degree of synchronization. As mentioned in Section 3.3.1, to design the sub-sample timestamps, we require three values: 1) the timestamp for the first sample in the sample stream, 2) sample number of the reference symbol and 3) the normalized timing error $\mu(t)$ for each sample. The sample-based timestamps are calculated based on the first two values. Then, the sub-sample timestamps are further calculated based on the normalized timing error $\mu(t)$ for each sample. For this reason, we need to ensure that the normalized timing error $\mu(t)$ is associated to each sample to calculate the sub-sample timestamps. Therefore, some modifications of the GMSK receiver in GNU Radio are needed and the modified blocks are indicated in gray in Figure 3.4(b). We make use of stream tags in GNU Radio to attach tags with control information to samples of the data stream.

- First, a time tag, which indicates the starting time of the stream, is attached to the first sample of the stream in the USRP Hardware Driver (UHD) receiver. All following samples are numbered relative to the first sample.



(a) GPS Synchronization Structure



(b) GMSK Receiver with Enhanced Timestamp

Figure 3.4: Hardware and Software Realization

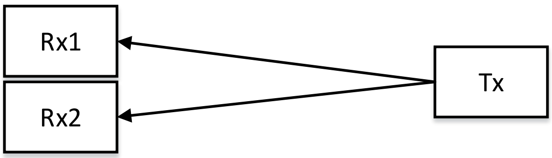
- Second, we modify the time recovery block to allow forwarding the normalized timing error $\mu(k)$ for each output sample.
- Third, a SYN tag indicating the first sample (as the reference symbol) in the packet is passed to the frame sink.
- Finally, in the frame sink component, we first store the timestamp for the first sample in the sample stream. Second, based on the SYN tag, we get the sample number for this reference symbol in the received packet. Then we calculate the sample-based timestamp for this received packet. Finally, the normalized timing error $\mu(k)$ are used to calculate the sub-sample timestamps according to Equation (3.3).

3.5 Time Synchronization Evaluation

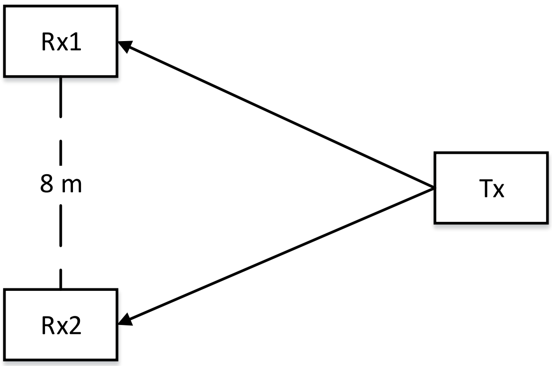
Based on the GSM-like testbed, we evaluate the synchronization accuracy of GPS synchronized receivers by using the normalized timing error information in different scenarios. Additionally, we investigate the influence of synchronization offset on the accuracy of TDOA algorithm in an outdoor environment.

3.5.1 Indoor Measurements

Several experiments in indoor environments are conducted to evaluate the synchronization accuracy and the influence of multipath propagation on the estimated timestamps.



(a) Measurement Setup 1



(b) Measurement Setup 2

Figure 3.5: Measurement Setup

Measurement Setup

To investigate the performance of GPS synchronization, the setup in Figure 3.5(a) is adopted in indoor environments. Assuming the same delays in hardware two factors can cause time offset between two receivers, namely, multipath propagation and synchronization offset. In order to isolate only the effect of the synchronization component on our measurement, we co-locate the two receivers. This setup ensures as much as possible the same propagation path of the signals. The setup of Figure 3.5(b) is used to investigate the effect of multipath propagation.

GPS Synchronization Evaluation

Taking the setup of Figure 3.5(a), we first try to calculate the GPS synchronization offset between the two receivers based on the sample-based timestamps (Equation (3.2)). The measurements based on these sample-based timestamps do not show a long-term clock offset, at least not larger than the sampling interval ($2\mu s$). This observation demonstrates the assumption in Section 3.3.1 that clock offsets between two GPS synchronized ANs are not larger than the sampling interval T_s at any moment. However, due to the limited resolution, we are not able to determine whether there is a short-term (momentary) clock offset between the receivers based on the sample-based timestamps.

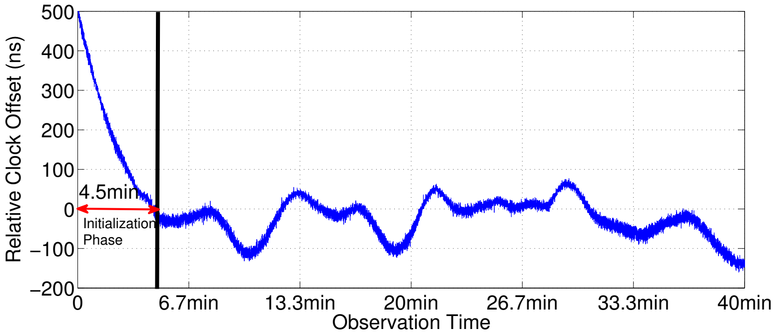
To analyze the performance of GPS synchronization in depth, we adopt the proposed method (in Section 3.3) that uses the normalized timing error $\overline{\mu}$, which is calculated by averaging the normalized timing error $\mu(k)$ over all the samples in each received packet. This averaged value is able to mitigate the influence of random noise in the measurement. In particular, we investigate the duration of initialization phase, clock offset, and clock skew.

Initialization Phase. It is important to measure the length of the initialization phase, during which applications with high requirements of accurate synchronization, e.g., localization, should be avoided. The first measurement denoted as M1 is conducted to measure the required length of the initialization phase. With our proposed method, we start to record the clock offset as soon as the two GPS receivers are powered on. Figure 3.6(a) indicates the changes of the relative clock offset between two GPS receivers over 40 minutes. We find that the GPS receivers need around 4.5 minutes to synchronize with each other. There are two reasons for this initialization duration. First, the stability of the oscillator inside each GPS is affected by the temperature. After USRPs powering up, the temperature inside increases and gets stable after a while. After the temperature getting stable, the oscillator gets stable. Second, as shown in Figure 3.4(a), it requires two steps to synchronize an USRP's clock to the satellite including GPS receiver synchronized to the signals from satellite and USRP synchronized to the signals from the GPS receiver. The phase lock loops inside both the GPS receiver and USRP need some time to lock to the incoming signals. After the initialization phase, the clock offset between two GPS receivers becomes stable.

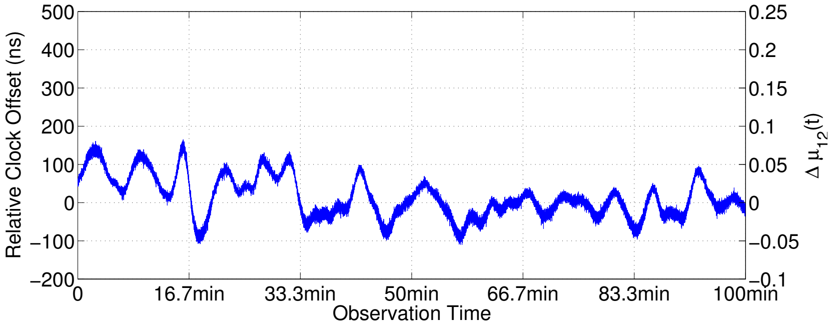
Clock Offset. We take a second measurement with duration of 18 hours to analyze the GPS clock offset in the stable phase. The measurement is denoted as M2. In the measurement, the $\Delta\mu_{12}(t)$ values are collected 5 minutes after the devices have been powered on to avoid the initialization phase in the GPS receivers.

Figure 3.6(b) indicates the measurement results of the first 100 minutes, where the left Y-axis indicates the relative clock offset and the right Y-axis is the value of $\Delta\mu_{12}(t)$. The relative clock offset between two GPS receivers is calculated based on Equation (3.5) where the sampling interval T_s is $2\mu s$. From this figure, first, we observe that the initial relative clock offset (at the time of 0s) is smaller than 100ns and there is no initialization phase as in M1 (Figure 3.6(a)). Second, fluctuations in the $\Delta\mu_{12}(t)$ value corresponding to the variation in the relative clock offset is well visible.

Figure 3.7 summarizes the Cumulative Distribution Function (CDF) of the absolute value of the relative clock offset ($|\Delta C_{i,j}|$). The first row in Table 3.1 summarizes the relative clock offset between two GPS receivers in the measurement. Mean and Standard Deviation (SD) are calculated for the value of $\Delta C_{i,j}$, while the maximum and 90% accuracy are for the absolute value ($|\Delta C_{i,j}|$). Our findings show that the maximum measured clock offset between the two receivers is 171ns, resulting in more than 51m time-based ranging error. The 90% synchronization accuracy is 66ns, corresponding to 19.8m time-based ranging error. Furthermore, Figure 3.8 indicates the histogram (density) of the clock offsets in the entire duration of the experiments M2 and M3. Figure 3.9 indicates the histogram of the clock offsets in experiment M2 for 2 minutes. In Figure 3.8, the distribution of



(a) Initialization Phase of GPS Synchronization (M1)



(b) Clock Offset in the Stable Phase (M2)

Figure 3.6: Clock Offset of GPS Synchronization with Co-Located Receivers

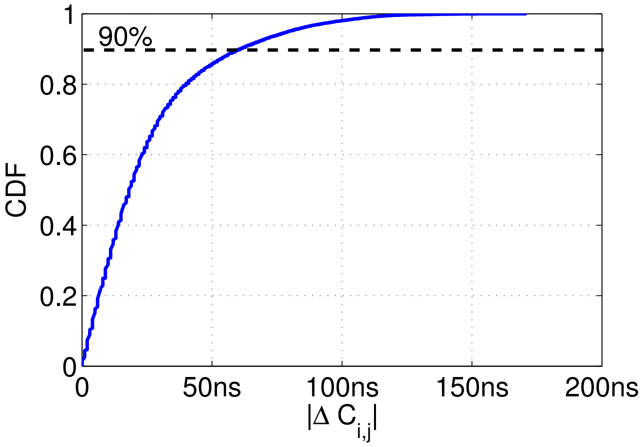


Figure 3.7: CDF of the Relative Clock Offset (M2)

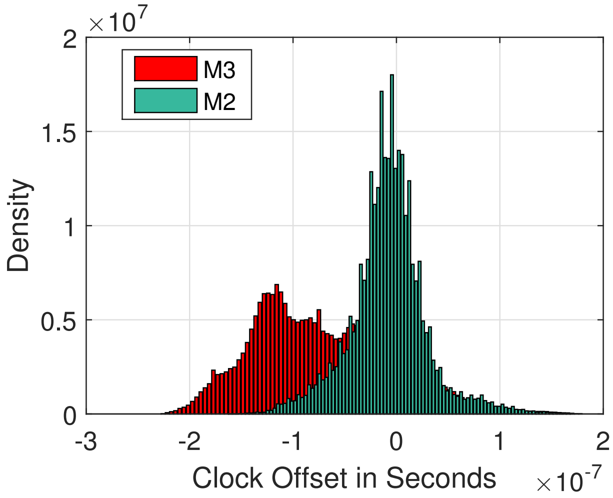


Figure 3.8: Histogram of Clock Offsets in M2 and M3

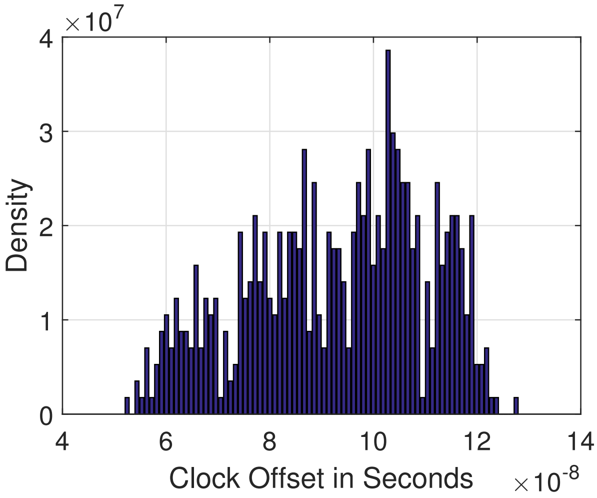
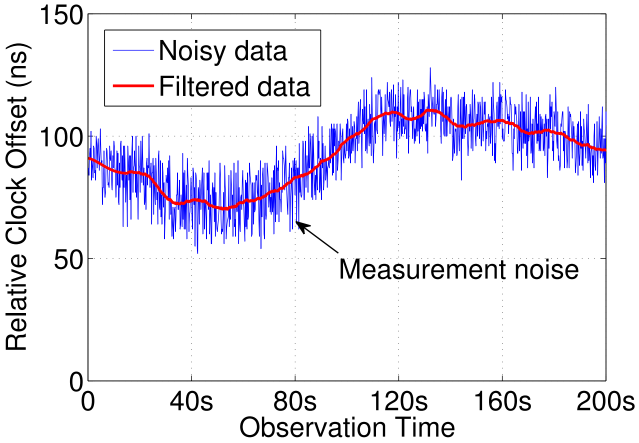
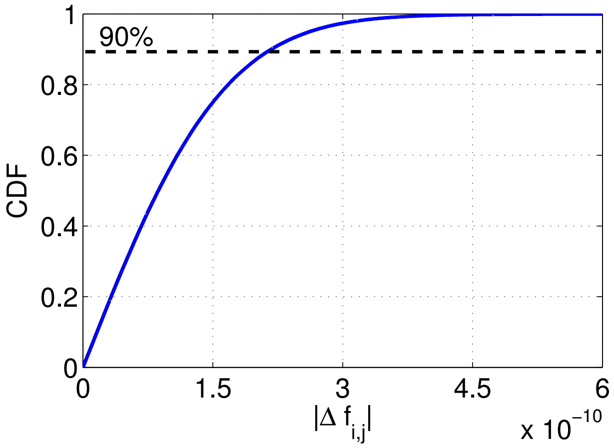


Figure 3.9: Histogram of Clock Offsets in M2 for 2 Minutes



(a) Data Filtering with S-G Filter (M2)



(b) CDF of the Relative Clock Skew (M2)

Figure 3.10: Measurement Results for Clock Skew

clock offsets in a long measurement period (18 hours) fits better to a Gaussian distribution with a symmetric shape. The median/mean value is around 0 (7ns in Table 3.1). Therefore, for TDOA ranging, we can take a long measurement (over several hours) and average the data to mitigate the influence of synchronization. However, it is unrealistic to take a measurement for hours in practice. For a short measurement period (2 minutes) as in Figure 3.9, the distribution of clock offsets is in an uncertain shape and difficult to find a model to fit. The mean/median error is not around 0. Therefore, the large clock offset between two GPS synchronized receivers introduces large error to time-based positioning, i.e., TDOA, and for a reasonable measurement duration this error is challenging to be eliminated by a predefined model. Hence GPS synchronization is not good enough for TDOA to achieve accurate positioning.

Table 3.1: Relative Clock Offset ($\Delta C_{i,j}$) between Two GPS Receivers

Measurement	Mean $\Delta C_{i,j}$	SD $\Delta C_{i,j}$	Maximum $\lvert\Delta C_{i,j}\rvert$	90% Accuracy $\lvert\Delta C_{i,j}\rvert$
M2	7ns	37.1ns	171ns	66ns
M3	-69.4ns	65.4ns	228ns	150ns

Clock Skew. Figure 3.10(a) shows part of the results from measurement M2. As it can be seen in the figure, measurements introduce noise to the data. As introduced in Section 3.3.2, noisy data prevents us from accurately calculating the relative clock skew. Therefore, we first apply the S-G filter to smooth the data and get the filtered data as the red line in Figure 3.10(a). The filtered data follows the trend of clock offset between two GPS synchronized ANs. We calculate the relative clock skew from the differentiation of the filtered clock offset. Figure 3.10(b) shows CDF of the clock skews. 90% of the measured relative clock skews are smaller than $2.18 \cdot 10^{-10}$ and the maximum is $1.37 \cdot 10^{-9}$. It means that during short intervals the change of the relative clock offset can be ignored. For example, during 10 seconds, 90% of the relative clock offsets vary within 2.18ns.

Influence of Multipath Propagation

To demonstrate that multipath propagation may affect the observed synchronization accuracy, we have taken a third measurement (M3) for 18 hours where the two receivers with line of sight connection to the transmitter were separated by eight meters in an indoor environment as shown in Figure 3.5(b). The measurement results are summarized in the second row of Table 3.1. We observe that there is a short-term clock offset between the receivers with deviation of 65.4ns. More importantly, the mean value of the synchronization offset is -69.4ns, which is shifted at a 76.4ns offset compared to the mean value of 7ns in the measurement M2. Figure 3.8 clearly shows that there is a shift between the maximum density in the distributions of clock offsets between M2 and M3. This shift is caused by multipath propagation. Signals from different propagation paths overlap with each other, which introduces a distortion in the shape of the received signal. This distortion influences the time recovery method on the accuracy of searching the pulse shape peak and calculating the normalized timing error information. Because the two receivers are located in different positions in an indoor environment, these two receivers face different effects of multipath propagation, which introduce different distortions on the shapes of the received pulses. These different distortions lead to different errors to estimate the normalized timing error information $\bar{\mu}$ and therefore the mean values of clock offsets are shifted. In contrast to M3, the two receivers are co-located in M2. Although the multipath propagation introduces distortions on the pulse, the two co-located receivers face the same multipath propagation effect and hence the error to estimate the normalized timing error information $\bar{\mu}$ is same. Therefore, the difference, $\Delta\mu_{12}(t)$, can eliminate this error and is not influenced by multipath propagation. Correspondingly, the measured relative clock offset based on $\Delta\mu_{12}(t)$ should have a mean value of 0ns and the real measured mean value is 7ns.

3.5.2 Outdoor Measurements

We further conducted experiments to test TDOA accuracy with GPS synchronization. The measurements were conducted in an open space football field to minimize the influence of multipath propagation as in Figure 3.11.

Measurement Setup

The layout of the measurement is given in Figure 3.12. The signal emitter (target) was an USRP E310, which transmits GMSK signals. Three ANs were deployed as a triangle with equal distances (30m). Each USRP N210 (as an AN) was connected to a laptop for signal processing and powered by a battery. Sub-sample timestamps based on Equation (3.3) are used to obtain high resolution timestamps of nanoseconds. A Reference Node (RN) was set in the center of the triangle and periodically transmitted packets to allow the receivers to check their synchronization offset. After collecting the timestamps in each AN, we offline calculate the TDOA values at the two different positions of the object.

TDOA Measurements with GPS Synchronization

Table 3.2 summarizes the TDOA ranging errors, i.e., the absolute values of the offset between the estimated distance and the ground truth distance. The rows of SYN offset indicate the average synchronization offset between two receivers. For example, for experiment 1, the synchronization offset of -38ns between receiver 1 and 2 means that the clock of receiver 1 is 38ns later than receiver 2. The maximum TDOA ranging error with GPS synchronization reaches 39m. Such large ranging errors are caused by imperfect synchronization between ANs and the remaining multipath propagation (even outdoor with open space, reflections from the ground

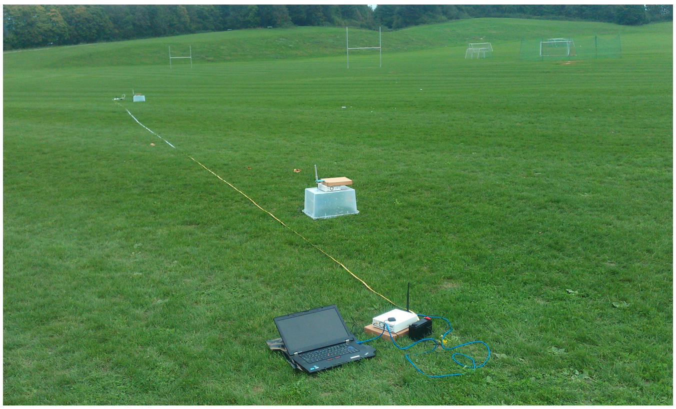


Figure 3.11: Experiment Environment

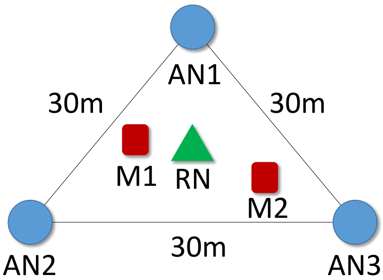


Figure 3.12: Experiment Layout

are still remaining). Furthermore, based on the packets from RN, we find that the maximum synchronization offset between the receivers is 161ns, which is the main reason for TDOA ranging errors in this open space environment. Therefore, calculating and compensating for the remaining GPS synchronization offset is very important for accurate positioning.

Table 3.2: Ranging Error in Open Space Environment

EXP No.	Algorithms	Rx1 and 2	Rx2 and 3	Rx1 and 3
EXP1	TDOA	5.6m	22.6m	23.1m
	SYN offset	-38ns	-50ns	-77ns
EXP2	TDOA	5.5m	39m	33.5m
	SYN offset	-58ns	161ns	105ns

3.6 Conclusions

In this chapter, we designed high resolution timestamps (sub-sample timestamps) based on the time recovery for GMSK-modulated signals (GSM). Our designed sub-sample timestamps are able to improve the resolution of sample-based timestamps with $2\mu s$ to nanoseconds. This nanosecond resolution is a prerequisite for accurate time-based positioning. According to our indoor evaluation, this sub-sample timestamps are still error prone to multipath propagation due to the distortion of pulse shapes.

The main findings for synchronization are that (1) the maximum clock offset is up to 171ns but (2) the maximum clock skew is $1.37 \cdot 10^{-9}$ for GPS synchronized receivers. Therefore, we can conclude that GPS synchronization introduces a large error for TDOA measurements, which has been verified in our outdoor experiments. However, the measured clock skews among GPS synchronized receivers are very small, which forms the basis for our synchronization offset compensation methods in Chapter 4.

Chapter 4

Time-based Indoor Positioning for Narrow-band Signals

4.1 Introduction

As discussed in Chapter 3, the accuracy of time-based indoor positioning with TDOA for narrow-band signals is limited because of imperfect synchronization and multipath propagation. In this chapter, we propose our solutions to eliminate the synchronization offset and locate the target based on time information for narrow-band signals. We mainly work on IEEE 802.15.4 signals with a bandwidth of 2 MHz, which is one of the most frequently used narrow-band signal in indoor environments. We provide a passive positioning system, which locate the IEEE 802.15.4 signal emitter by adopting the time information from the physical layer. Additionally, based on the GSM-like testbed described in Chapter 3, we also investigate our proposed methods for GSM signals with extremely narrow bandwidth (200 KHz) in an outdoor environment with weak multipath effects. We summarize our contributions in this chapter as follows.

- As introduced in Chapter 3, although the relative clock offsets between two GPS synchronized ANs are too large to achieve accurate TDOA-based ranging, the relative clock skews are very small. Based on this observation, we propose to eliminate the negative influence of imperfect synchronization between ANs by combining DTDOA and GPS synchronization.

- We propose two novel positioning algorithms based on DTDOA for narrow-band signals. Our first contribution adds to range-based algorithms and employs a Linear Least Square (LLS) algorithm to convert the DTDOA-based ranges to the location of the target by introducing a new intermediate parameter. Our second contribution (a DTDOA-based fingerprinting) adds to range-free positioning. To our knowledge, we are the first to propose and demonstrate the feasibility of DTDOA-based fingerprinting in indoor environments.

DTDOA and two positioning algorithms (LLS and fingerprinting based on DTDOA) are supported by theoretical analysis and experiments in a real-world IEEE 802.15.4 testbed. The high reliability of our findings is backed-up by a method to minimize the influence of outliers. The testbed is designed and implemented using software defined radio to overhear IEEE 802.15.4 signals and extract physical layer fine-grained information used for positioning. Through a set of comprehensive measurements, we find that DTDOA-based LLS significantly improves the accuracy compared to TDOA-based LLS. However, the accuracy is still limited in indoor environments because of severe multipath influence on the timestamps for narrow-band signals. TDOA-based fingerprinting achieves much better accuracy than LLS and is suitable to locate the target in indoor environments. Furthermore, we compare the performance of DTDOA-based and RSS-based fingerprinting in different scenarios. The measurement results show that DTDOA-based fingerprinting achieves quite similar performance as power-based fingerprinting. However, in Non-Line Of Sight (NLOS) conditions, DTDOA-based fingerprinting performs better than RSS. Additionally, we also adopt DTDOA methods to improve the ranging accuracy on the GSM-like testbed, which is proposed in Chapter 3. We evaluate the performance of DTDOA-based ranging on the GSM-like testbed in an outdoor environment with open space. The DTDOA-based ranging accuracy is significantly improved compared to TDOA-based ranging.

In the remainder of this chapter, Section 4.2 theoretically analyzes the difference between TDOA and DTDOA and introduces how to eliminate the synchronization offset by adopting DTDOA and GPS synchronization. Based on the proposed DTDOA methods, two positioning algorithms, i.e., LLS and fingerprinting, are introduced in Section 4.3. Section 4.4 introduces our proposed SDR-based passive positioning system for IEEE 802.15.4 signals based on time information. The evaluation results are presented in Section 4.5. Section 4.6 concludes this chapter.

4.2 Differential Time Difference Of Arrival

In this section, first, we theoretically investigate the factors influencing the performance of TDOA. Second, based on the quantification of GPS synchronization in Chapter 3, we propose to combine the DTDOA mechanism with GPS synchronization to efficiently eliminate the influence of synchronization offset.

4.2.1 TDOA Model

TDOA is defined as the difference of arriving times between ANs for the same packet from a target. Assume all ANs are synchronized by GPS signals, which have large synchronization offsets as introduced in Chapter 3. Given the factors influencing the accuracy of TDOA, we propose to decompose TDOA into four components as

$$T_{\text{TDOA}} = T_{\text{d}} + T_{\text{sh}} + T_{\text{sy}} + n, \tag{4.1}$$

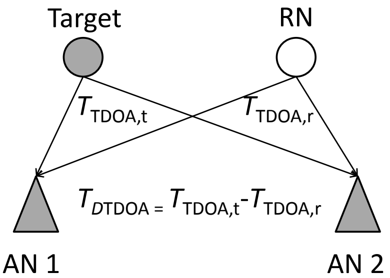


Figure 4.1: DTDOA Operation

where T_{TDOA} is the overall TDOA value, T_{d} is the TDOA component related to the geometric distance, T_{sh} is the shadowing and multipath component, T_{sy} is the synchronization offset component and n is Gaussian noise.

The components, T_{d} and T_{sh}, are determined by the location of the target and remain constant in a static environment. However, with GPS synchronization, the component T_{sy} is unpredictably variable over time as introduced in Chapter 3. Therefore, the overall T_{TDOA} in Equation (4.1) is unpredictably variable as well, even if the target stays at the same location. Hence, T_{TDOA} is not only determined by the location of the target but also influenced by synchronization. Therefore, T_{TDOA} is not a good measure for ranging and fingerprinting. Different from ranging in RTT, which converts the signal propagation time to the propagation distance between the target and an AN, TDOA-based ranging is defined as the multiplication of the measured TDOA value and speed of radio signal c, i.e., $T_{\text{TDOA}} \cdot c$, which is actually the difference between the two propagation distances from the target to two ANs.

4.2.2 DTDOA Model with GPS Synchronization

DTDOA is defined as the difference of TDOA values for the target and a Reference Node (RN) between the same pair of ANs as introduced in Section 2.5.1. Figure 4.1 illustrates the calculation of DTDOA. Compared to a TDOA system, a RN is added to compensate for imperfect synchronization. RN transmits a control packet after overhearing the target's transmission.

Based on the TDOA model in Equation (4.1), the T_{TDOA} values for the target and RN between the same pair of ANs are respectively defined as

$$T_{\text{TDOA,t}} = T_{\text{d,t}} + T_{\text{sh,t}} + T_{\text{sy,t}} + n, \tag{4.2}$$

$$T_{\text{TDOA,r}} = T_{\text{d,r}} + T_{\text{sh,r}} + T_{\text{sy,r}} + n, \tag{4.3}$$

where $T_{\text{d,t}}$, $T_{\text{sh,t}}$ and $T_{\text{sy,t}}$ are the components for the target, and $T_{\text{d,r}}$, $T_{\text{sh,r}}$ and $T_{\text{sy,r}}$ are for RN. DTDOA (T_{DTDOA}) for the target and RN between the same pair of ANs is calculated as

$$\begin{aligned} T_{\text{DTDOA}} &= T_{\text{TDOA,t}} - T_{\text{TDOA,r}} \\ &= (T_{\text{d,t}} - T_{\text{d,r}}) + (T_{\text{sh,t}} - T_{\text{sh,r}}) + (T_{\text{sy,t}} - T_{\text{sy,r}}) + n. \end{aligned} \tag{4.4}$$

To eliminate the influence of synchronization effects on DTDOA, the following condition are required that $T_{\text{sy,t}} - T_{\text{sy,r}} = 0$, i.e., $T_{\text{sy,t}} = T_{\text{sy,r}}$. This means that the synchronization offset between two ANs remains constant in the transmission interval between the two packets from the target and RN respectively. Let T_{TR} denote this duration and Δf denote the relative clock skew between two ANs. We obtain that

$$T_{\text{sy,t}} - T_{\text{sy,r}} = \Delta f \cdot T_{\text{TR}}. \tag{4.5}$$

As measured in Chapter 3, the maximum clock skew between two GPS synchronized receivers is $1.37 \cdot 10^{-9}$, i.e., $\Delta f = 1.37 \cdot 10^{-9}$. Therefore, if we set T_{TR} to few seconds, the accumulated synchronization offset ($T_{\text{sy,t}} - T_{\text{sy,r}} = \Delta f \cdot T_{TR}$) can be ignored. For example, with $T_{TR} = 1s$, the accumulated GPS synchronization offset ($T_{\text{sy,t}} - T_{\text{sy,r}}$) is smaller than $1.37ns$, which allows us to assume $T_{\text{sy,t}} - T_{\text{sy,r}} = 0$.

Therefore, T_{DTDOA} is rewritten as

$$T_{\text{DTDOA}} = (T_{\text{d,t}} - T_{\text{d,r}}) + (T_{\text{sh,t}} - T_{\text{sh,r}}) + n. \tag{4.6}$$

In Equation (4.6), besides the Gaussian noise, there are only components that are determined by the location of the target. Therefore, the distribution of the T_{DTDOA} is expected to be Gaussian with mean value of $(T_{\text{d,t}} - T_{\text{d,r}}) + (T_{\text{sh,t}} - T_{\text{sh,r}})$, which is only determined by the locations of the target and RN. In consequence, T_{DTDOA} is a better candidate for ranging and fingerprinting. Additionally, compared to [97], our proposed DTDOA combining with GPS synchronization is able to compensate the momentary clock offset but do not need the offline calibration and do not require very strict limitations on T_{TR} (described in Section 2.5.1). Similar as TDOA-based ranging, DTDOA-based ranging is defined as $T_{\text{DTDOA}} \cdot c$.

4.3 DTDOA-based Positioning

After measuring DTDOAs between different ANs, we design and apply positioning algorithms to locate the target. In our work, we investigate two kinds of positioning algorithms, i.e., multilateration and fingerprinting, based on the measured DTDOA. Before describing the positioning algorithms, we first present a kernel based data aggregation method to mitigate the influence of outliers. Then, we derive a new LLS algorithm for DTDOA-based multilateration with a RN. Finally, we propose a new fingerprinting method, in which the fingerprints are built based on the DTDOA values in different locations.

4.3.1 KDE-based Data Aggregation

Outliers that are caused by measurement errors can obviously deteriorate the performance of positioning algorithms. We propose to apply Kernel Density Estimation (KDE) (introduced in Section 2.8.1) to aggregate the T_{DTDOA} values from different packets to mitigate the influence of outliers. The benefit of KDE is that it can estimate the density directly from the data without assuming a particular form for the underlying distribution.

Assuming n independent observations of T_{DTDOA}, $\mathbf{X} = [x_1, x_2, \ldots, x_n]$, the kernel density estimator $\widehat{f}_h(x)$ of the density value $f(x)$ at point x is defined as

$$\widehat{f}_h(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x_i - x}{h}\right), \tag{4.7}$$

where $K(u)$ denotes the kernel function, and h denotes the bandwidth. In our work, we adopt a Gaussian kernel, $K(u) = \frac{1}{\sqrt{2\pi}}\exp\bigl(-\frac{1}{2}u^2\bigr)$.

Because of its Gaussian distribution with symmetric shape, DTDOA is estimated by finding the DTDOA value with maximum density in KDE. Therefore, we propose to first adopt the KDE method to calculate $\widehat{f}_h(x)$ as in Equation (4.7). The DTDOA is then estimated as

$$\hat{T}_{\text{DTDOA}} = \underset{T_{\text{DTDOA}}}{\operatorname{argmax}} \widehat{f}_h(T_{\text{DTDOA}}). \tag{4.8}$$

$\hat{T}_{\text{DTDOA}}$ is the aggregated DTDOA value.

4.3.2 DTDOA-based Multilateration

For range-based algorithms, $\hat{T}_{\text{DTDOA}}$ is first converted into distance as $r_{\text{DTDOA,k}} = \hat{T}_{\text{DTDOA,k}} \cdot c$, where $\hat{T}_{\text{DTDOA,k}}$ is $\hat{T}_{\text{DTDOA}}$ between the kth and the first ANs. After ranging, the estimated position (x', y') of the target is found by the LLS algorithm, given the position (x_k, y_k) of the kth AN and position (x_R, y_R) of RN.

Without considering noise, we obtain that

$$\begin{aligned} &r_{\text{DTDOA,k}} = \\ &[\sqrt{(x - x_k)^2 + (y - y_k)^2} - \sqrt{(x - x_1)^2 + (y - y_1)^2}] - \\ &[\sqrt{(x_R - x_k)^2 + (y_R - y_k)^2} - \sqrt{(x_R - x_1)^2 + (y_R - y_1)^2}], \end{aligned} \tag{4.9}$$

where (x, y) is the ground truth location of the target. Two intermediate parameters, R_k and d_1 are introduced. R_k is defined as the difference between the distances from the RN node to the kth AN and the first AN:

$$R_k = \sqrt{(x_R - x_k)^2 + (y_R - y_k)^2} - \sqrt{(x_R - x_1)^2 + (y_R - y_1)^2}.$$

d_1 is defined as the distance from the location of the target to the first AN as $d_1 = \sqrt{(x - x_1)^2 + (y - y_1)^2}$. Equation (4.9) is rewritten as the following linear equation,

$$\begin{aligned} &(x - x_1)(x_k - x_1) + (y - y_1)(y_k - y_1) + (r_{\text{DTDOA,k}} + R_k)d_1 \\ &= \frac{1}{2}[(x_k - x_1)^2 + (y_k - y_1)^2 - (r_{\text{DTDOA,k}} + R_k)^2]. \end{aligned} \tag{4.10}$$

Writing Equation (4.10) in matrix form gives

$$\mathbf{G}\theta = \mathbf{h}, \tag{4.11}$$

$$\mathbf{G} = \begin{pmatrix} x_2 - x_1 & y_2 - y_1 & r_{TDOA,2} + R_2 \\ \vdots & \vdots & \vdots \\ x_M - x_1 & y_M - y_1 & r_{TDOA,M} + R_M \end{pmatrix},$$

$$\mathbf{h} = \frac{1}{2}\begin{pmatrix} (x_2 - x_1)^2 + (y_2 - y_1)^2 - (r_{TDOA,2} + R_2)^2 \\ \vdots \\ (x_M - x_1)^2 + (y_M - y_1)^2 - (r_{TDOA,M} + R_M)^2 \end{pmatrix},$$

where $\theta = [x - x_1, y - y_1, d_1]^T$ and M is the number of ANs.

To find the solution of (x, y), we minimize the sum of squares of the residuals,

$$\begin{aligned} \hat{\theta} &= \underset{\theta}{\operatorname{argmin}}[(\mathbf{G}\theta - \mathbf{h})^T(\mathbf{G}\theta - \mathbf{h})] \\ &= (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{h}, \end{aligned} \tag{4.12}$$

and $\hat{\theta} = [x' - x_1, y' - y_1, R]^T$. After obtaining $\hat{\theta}$, we get the LLS solution (x', y') based on the DTDOA measurements.

4.3.3 DTDOA-based Fingerprinting

The accuracy of multilateration highly depends on the ranging accuracy. However, as our preliminary experiments introduced in Chapter 3, timestamps for narrowband signals are error prone to multipath propagation. Therefore, it is still challenging to obtain high ranging accuracy. Therefore, we propose a time-based fingerprinting method to achieve high accuracy of indoor positioning with time information.

The proposed fingerprinting method relies on two phases including an offline phase and an online phase. In the offline phase, a mobile signal emitter moves through the area of interest in different training positions and ANs record the radio parameters at each training location to form a radio map (database). In our work, DTDOA vectors ($\mathbf{DTDOA}_w$) are stored in the radio map. $\mathbf{DTDOA}_w$ comprises T_{DTDOA} values between different ANs as

$$\mathbf{DTDOA_w} = [T^w_{\text{DTDOA}(1,2)}, T^w_{\text{DTDOA}(1,3)}, \cdots, T^w_{\text{DTDOA}(M-1,M)}] \tag{4.13}$$

where $T^w_{\text{DTDOA}(i,j)}$ with $i < j$ is the T_{DTDOA} value between the ith and jth ANs at the wth training position.

Once the offline training phase is complete, the location of the target is estimated by performing a radio scan and the measured $\mathbf{DTDOA}$ vector is then passed to a fingerprinting algorithm. Weighted KNN (WKNN) (introduced in Section 2.4.1) is a simple fingerprinting algorithm, which selects the K most nearest neighbours based on the Euclidean distance and then returns the weighted average of these K locations as an estimate of the current location of the target. For this WKNN method, first, the Euclidean distances e_w between $\mathbf{DTDOA}$ values at the current location and all the training locations are calculated as

$$e_w = \|\mathbf{DTDOA} - \mathbf{DTDOA}_w\|. \tag{4.14}$$

$\|\cdot\|$ indicates the norm value of a vector. Smaller Euclidean distance e_w means that the measured $\mathbf{DTDOA}$ at the current location better matches to $\mathbf{DTDOA_w}$ at the wth training location. Second, we set $K = 3$, which means that three locations in the radio map with the minimum e_w are selected. Finally, based on these three selected locations, we calculate a weighted average of their coordinates as the estimation of the target's location (x', y') as

$$(x', y') = \sum_{i=1}^{3} \frac{L_i}{\sum_{j=1}^{3} L_j}(x_i, y_i), \tag{4.15}$$

where (x_i, y_i) are the coordinates of the ith training position. Since smaller Euclidean distance e_i means better matching to the ith training position, each weight L_i is inversely proportional to the Euclidean distance e_i, i.e., $L_i = \frac{1}{e_i}$.

4.4 Implementation of Positioning Algorithms in a Positioning System for IEEE 802.15.4 Signals

We have implemented our proposed time-based positioning algorithms for narrowband signals in two systems. The first system is introduced in Chapter 3 (Section 3.4), which uses USRP devices to mimic the GSM signals. In this section, we focus on the implementation of the second system, which is a passive system based on software defined radio techniques to overhear IEEE 802.15.4 signals and accurately timestamp the captured messages.

4.4.1 System Overview

Figure 4.2 shows an overview of the passive positioning system for IEEE 802.15.4 signals. A TelosB node [102] as a signal emitter, which includes a CC2420 radio transceiver [1], is the target for positioning. To implement the DTDOA method, another TelosB node as a RN is deployed at a known location, which overhears the packet from the target and then retransmits the packet. Both TelosB nodes run Contiki system [46] to send packets.

At the receiver side, the system comprises three main components, i.e., USRP for signal capturing, GNU Radio for signal processing, and MATLAB for the positioning algorithms as shown in Figure 4.3. USRP with model N210 is used as signal capturing hardware. All USRPs in our work are synchronized by GPS receivers. All USRPs are connected to a desktop PC with i5 CPU (3.3GHz) as a

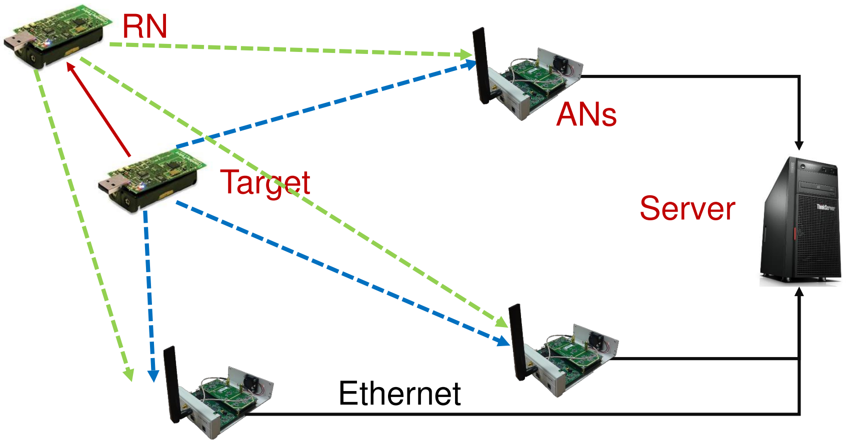


Figure 4.2: An Overview of Passive Positioning System for IEEE 802.15.4 Signals

server, which decodes the packets and runs positioning algorithms. Signal processing is implemented in GNU Radio [5], which is utilized for demodulation and packet reconstruction. The physical layer parameters for positioning, e.g., RSS and timestamp, are extracted in GNU Radio and passed to MATLAB processing. We adopt the advanced method to achieve sub-sample timestamps, which estimate the sub-sample time delay between two output samples from one USRP by using symbol timing recovery from the signal processing chain (as in Section 3.2.2). KDE-based data aggregation, DTDOA-based LLS and fingerprinting as introduced in Section 4.3 are implemented in MATLAB as positioning algorithms. In the system, multiple targets are separated by their node IDs. Additionally, a graphical user interface designed by MATLAB runs in the server to indicate the location of the target on the floor plan in real time.

4.4.2 Packet Decoding and Physical Layer Information Extraction

In this subsection, we introduce the methods in our work for signal processing implemented in GNU Radio. As introduced in Section 2.3.2, the authors of [40] provided an IEEE 802.15.4 decoding system, in which physical and MAC layers (packet sink in Figure 2.19) are separated into different GNU Radio blocks. Since the packet is detected and reconstructed on MAC layer, which is after demodulation, the physical layer information (RSS and sub-sample timestamps) for each packet is only available before or during the demodulation of the digital samples. Therefore, such physical layer information needs to be attached to each sample, passed to the MAC layer, and aggregated when the packet is detected. To pass the physical layer information through different GNU Radio blocks, the stream tags as introduced in the GSM-like testbed (Section 3.4) could be adopted. However, this method significantly increases the processing load and results in overflows in the server (a desktop PC with i5 CPU (3.3GHz)) because of the high sampling rate (4MHz) (in contrast to 500KHz in the GSM-like testbed). Because of overflows,

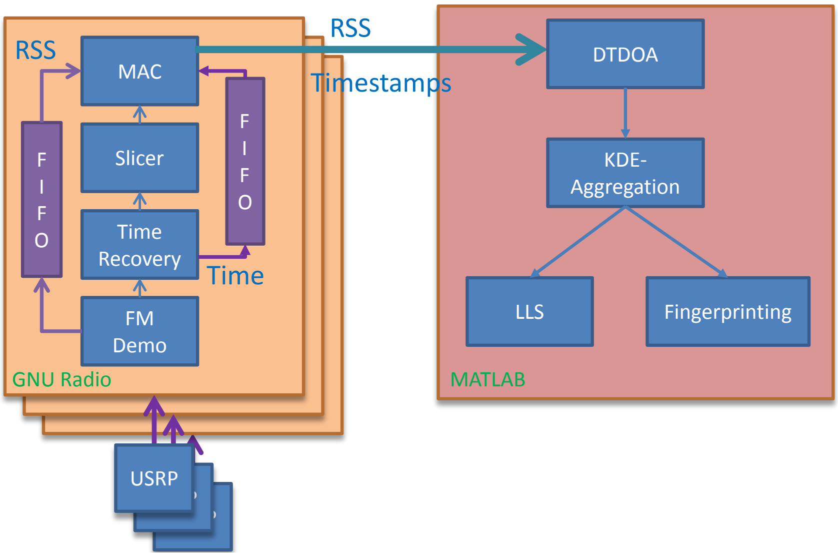


Figure 4.3: The Structure of Passive Positioning System for IEEE 802.15.4 Signals (Receiver Side)

many samples are lost and hence the packet decoding rate is reduced dramatically.

In our work, we implement a cross-layer structure integrating physical and MAC layers into one GNU Radio block, which more efficiently passes the parameters from physical layer to MAC layer and solves the overflow problem. At the physical layer, we implement similar decoding methods for IEEE 802.15.4 signals as in [40]. Figure 4.3 indicates the structure, where a MSK demodulation method is implemented to decode IEEE 802.15.4 signals. The power for each sample is obtained before FM (Frequency Modulation) demodulation and sub-sample timestamps are obtained in the time recovery part after extracting the normalized timing error information. The power and sub-sample timestamp for each sample are stored in two FIFO (First Input First Output) buffers. As soon as the preamble in each IEEE 802.15.4 packet is detected, we search in the two FIFO buffers for the RSS values and sub-sample timestamps corresponding to the first sample in the preamble. In order to mitigate the influence of measurement noise, the power and sub-sample timestamp are obtained by averaging all the samples in a packet.

4.5 Evaluation of Time-based Positioning Algorithms

We have conducted a set of experiments to evaluate the proposed time-based positioning algorithms for narrow-band signals. First, we conducted some preliminary experiments based on the GSM-like testbed in an outdoor environment to investigate the performance of DTDOA with GSM-like signals. Then, we mainly focus on the experiments in indoor environments based on the passive indoor positioning system for IEEE 802.15.4 signals. In these experiments, we first evaluate our proposed IEEE 802.15.4 decoding system for CPU usage and packet decoding rate.

Second, we analyze the distribution and variance of DTDOA values in indoor environments and investigate the Euclidean distance of the DTDOA vector among neighbours. Third, we evaluate the performance of the DTDOA-based LLS. Finally, we evaluate the performance of DTDOA-based fingerprinting and compare it to RSS-based fingerprinting.

4.5.1 Outdoor Experiments for GSM-like Signals

In Section 3.5.2, we have conducted some experiments in an outdoor environment with three ANs, a RN and a target using the GSM-like testbed. The measurement setup is shown in Figure 3.12, where 3 ANs are separated by 30m. As introduced in Section 3.5.2, TDOA faces large ranging errors because of the synchronization offset between GPS-synchronized ANs. With the same measurement setup (Figure 3.12), we adopt the DTDOA algorithm to compensate the synchronization offset and Table 4.1 summarizes the ranging errors between different ANs at the two test locations.

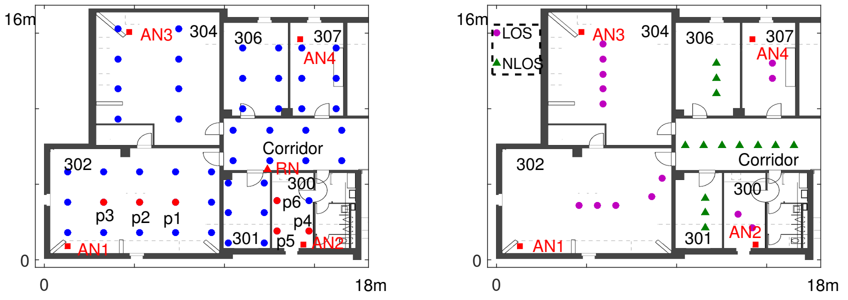
From Table 4.1, we find that the ranging errors based on DTDOA are significantly smaller than TDOA because the synchronization offsets between ANs are eliminated. For example, the maximum ranging error based on TDOA is 39m. In contrast, the maximum ranging error based on DTDOA is reduced to 12.5m and most of the ranging errors are smaller than 10m. The ranging errors are caused by the remaining weak multipath effects, e.g., multipath from the ground. Additionally, noise in the estimation of the normalized timing error from time recovery also introduces errors to the timestamps.

Table 4.1: Ranging Error in Open Space Environment (TDOA and DTDOA)

EXP No.	Algorithms	Rx1 and 2	Rx2 and 3	Rx1 and 3
EXP1	TDOA	5.6m	22.6m	23.1m
	DTDOA	5.2m	6.4m	1.2m
EXP2	TDOA	5.5m	39m	33.5m
	DTDOA	12.5m	12m	2m

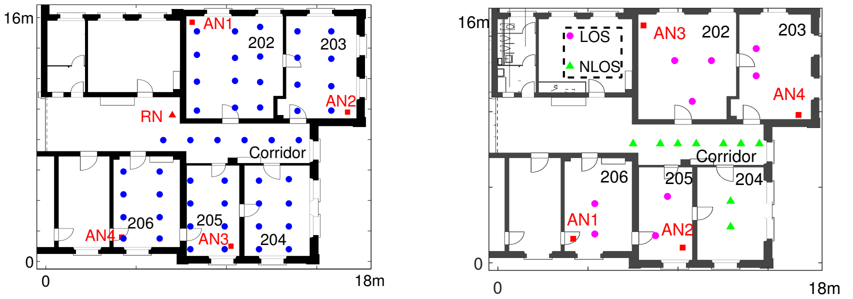
4.5.2 Indoor Experiments for IEEE 802.15.4 Signals

The experiments in Section 4.5.1 have demonstrated that our proposed DTDOA methods efficiently improve ranging accuracy by eliminating the synchronization offset in an open space environment with weak multipath propagation. IEEE 802.15.4 signals at 2.4GHz are designed for short range communication and often used in indoor environments. Therefore, in the remainder of this section, we focus on the evaluation of DTDOA algorithms based on the passive positioning system for IEEE 802.15.4 signals in indoor environments.



(a) Radio Map on the Third Floor and Testing Positions for LLS

(b) Testing Positions on the Third Floor



(c) Radio Map on the Second Floor

(d) Testing Positions on the Second Floor

Figure 4.4: Measurement Setup

Evaluation of Packet Decoding

Before conducting the experiments for indoor positioning, we first evaluate the performance of our implemented IEEE 802.15.4 decoding system with cross-layer design. First, with this cross-layer design, we are able to reduce CPU usage by more than half and avoid overflows. Second, we test our decoding methods with a TelosB node as a signal emitter. We configured the CC2420 radio transceiver to the maximum transmission power level (level 31). To test the performance of the receiver, the distance between the USRP receiver and TelosB emitter with LOS connection was changed from $0.5m$ to $11.5m$ at $1m$ steps. For each distance 600 packets of the target are analyzed and we are able to decode more than 99.4% of the transmitted packets. A small fraction of packets can not be decoded because bit errors happen in the preamble decoding, which causes the receiver can not be synchronized to the incoming packets.

Measurement Setup for Indoor Positioning

Indoor experiments were conducted at two test scenarios, i.e., the second and third floor of the INF building at University of Bern. These two scenarios have different layouts as shown in Figure 4.4(d) and Figure 4.4(b). The second floor is our working area and people are moving during working hours. The third floor is mainly occupied by less frequently used seminar rooms and laboratories. Hence, the environment on the third floor is more stable.

Four ANs are deployed to capture IEEE 802.15.4 signals from TelosB nodes. A RN (TelosB node) is placed in the system and used to calculate T_{DTDOA} values. The target TelosB node periodically broadcasts packets. As soon as RN has received one packet from the target, it retransmits the packet. ANs overhear packets from both the target and RN to calculate the corresponding T_{DTDOA}. The target was configured to transmit five packets per second. The data collection time at each test position was 1 minute.

- First, as a proof of concept, several preliminary measurements were conducted on the third floor of the INF building to analyze DTDOA and TDOA regarding the following three aspects: distribution, variation over space, and Euclidean distance among neighbours.

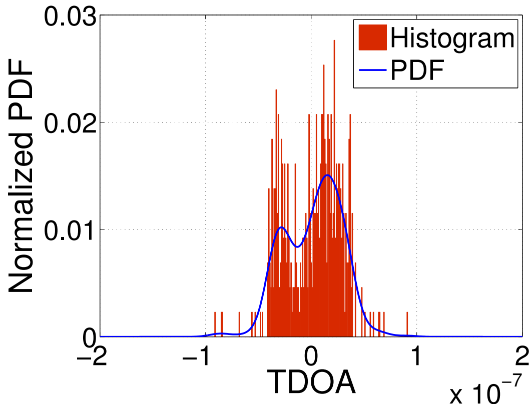
- Second, 53 locations (Figure 4.4(a)) on the third floor were tested to analyze the performance of DTDOA-based ranging and LLS. Moreover, we compare the performance of DTDOA-based LLS to TDOA-based LLS.

- Third, the proposed DTDOA-based fingerprinting was tested on both the second and third floor. Before testing, two radio maps have been created in these two scenarios during weekend when there was no change of the layouts and no people movement. Figure 4.4(a) shows the radio map (including the blue and red points) on the third floor, in which training positions are separated by approximately $2m$ and Figure 4.4(c) shows the radio map on the second floor.

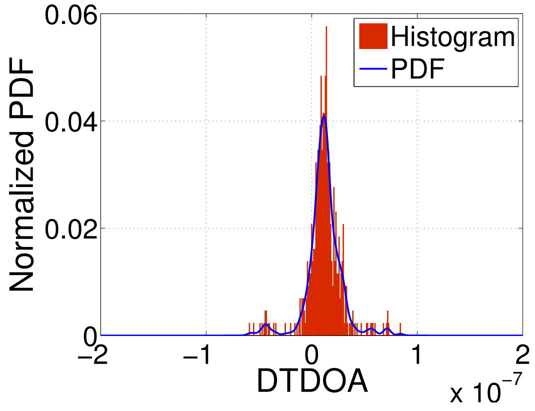
- Fourth, for online tests of the fingerprinting algorithm, the first measurements in both scenarios were taken one day after the creation of radio map at the same weekend. The second measurements were conducted five days later during working hours. 18 test locations were randomly selected on the second floor as shown in Figure 4.4(d). Among them, 9 locations were located in areas, where the target has LOS connection to one of ANs, and are thus referred to **LOS areas**. 9 locations were in areas, where the target has no LOS connection to any AN, and are referred to **NLOS areas**. 27 locations on the third floor were tested as shown in Figure 4.4(b). Among them, 14 locations were in the LOS areas and 13 locations were in the NLOS areas.

Preliminary DTDOA Analysis based on Measurements

- **Distribution:** We analyze the distribution of T_{TDOA} and T_{DTDOA} values for a single location. Figure 4.5(a) and 4.5(b) show the normalized histogram and estimated Probability Density Function (PDF) of T_{TDOA} and T_{DTDOA} values respectively. As measured in Chapter 3, the distribution (PDF) of



(a) Distribution of TDOA Values



(b) Distribution of DTDOA Values

Figure 4.5: Distribution of TDOA and DTDOA Values

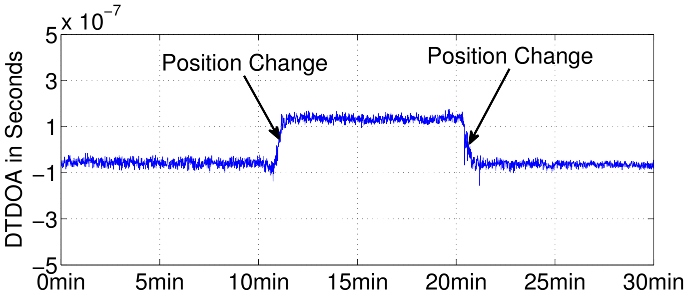


Figure 4.6: DTDOA Change at Different Positions

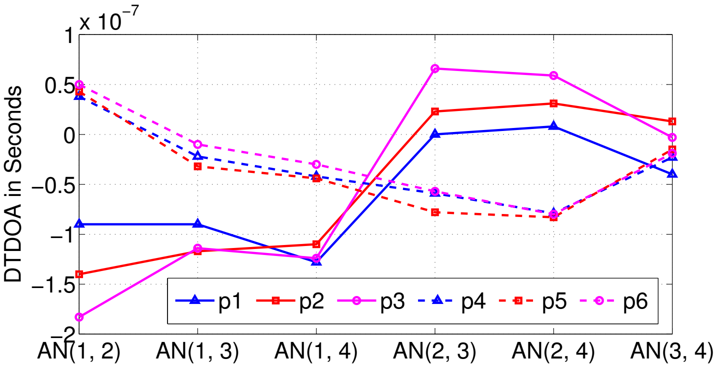


Figure 4.7: DTDOA Vectors at Different Positions

clock offsets between two GPS synchronized ANs during short periods e.g., 1 minute, does not follow a certain model, e.g., Gaussian. Therefore, PDF of T_{TDOA}, which is influenced by imperfect synchronization, has an uncertain shape as shown in Figure 4.5(a). The distribution of T_{DTDOA} fits better to a Gaussian distribution than T_{TDOA} because it eliminates the influence of imperfect synchronization. This finding supports our claim in Section 4.2.2.

- **Variation over space:** Figure 4.6 shows a DTDOA measurement between two ANs for thirty minutes with the target moving from one location to another and then back but remaining stationary for ten minutes at each location. As shown in the figure, the mean DTDOA value does not change significantly for a single location but visibly differs from one location to the next. This finding further supports our analysis from Section 4.2.2 that the DTDOA values are only influenced by the location of the target.

- **Euclidean Distance among Neighbours:** In order to apply fingerprinting (WKNN), the DTDOA vectors ($\mathbf{DTDOA}_w$ in Section 4.3.3) among the neighbours should have lower Euclidean distance than the DTDOA vectors at the locations in far away areas. Therefore, we analyze DTDOA vectors at six different locations (p1 to p6 in Figure 4.4(a)). Figure 4.7 shows the DTDOA vectors at the six locations, where y-axis is the T_{DTDOA} values and x-axis indicates the ANs from which the T_{DTDOA} is obtained. For example, AN(1,2) means that the T_{DTDOA} value is calculated between the first AN and the second AN. Each single vector comprises six T_{DTDOA} values between different ANs. As shown in Figure 4.7, the DTDOA vectors at nearby locations have smaller gaps among each other, which means shorter Euclidean distance.

Based on these preliminary evaluations, T_{DTDOA} is only determined by the location of the target but not influenced by synchronization. Furthermore, DTDOA vectors at nearby locations have shorter Euclidean distance between each other. Both findings support the feasibility of DTDOA fingerprinting.

Table 4.2: Measurement Notation

Notation	Floor No.	Time	Environment
M1	Second	1 day later	Static
M2	Second	5 days later	Dynamic
M3	Third	1 day later	Static
M4	Third	5 days later	Static

Table 4.3: Localization Errors Statistics

	RSS-based WKNN		DTDOA-based WKNN	
Measurements	Mean	SD	Mean	SD
M1	2.13m	1.35m	1.92m	1.15m
M2	2.59m	2m	2.65m	1.47m
M3	2.72m	1.72m	2.89m	1.65m
M4	2.8m	1.6m	3m	1.4m

Measurement Results for DTDOA-based LLS

Figure 4.8 shows a CDF of ranging errors based on the DTDOA and TDOA methods. As discussed in Section 4.3, DTDOA eliminates the influence of synchronization offsets, which can introduce up to 50m ranging error for TDOA. As shown in Figure 4.8(a), DTDOA significantly improves the ranging error compared to TDOA. For example, DTDOA improves the median ranging error by 50% (from $22m$ to $11m$) compared to TDOA-based ranging. This finding is consistent with the results in the outdoor measurement based on the GSM-like testbed.

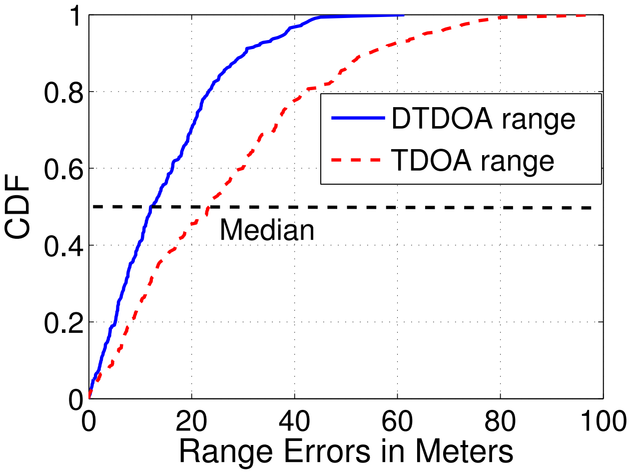
Figure 4.8(b) indicates the CDF of positioning errors for DTDOA and TDOA-based LLS. As shown in Figure 4.8(b), the DTDOA-based LLS significantly outperforms TDOA-based LLS because DTDOA achieves higher ranging accuracy. The improvement of median error by DTDOA-based LLS is around 57% (from $30m$ to $13m$) compared to TDOA-based LLS.

We show that DTDOA achieves higher accuracy than TDOA by eliminating the influence of imperfect synchronization. However, multipath and NLOS propagation still introduce a large error to DTDOA-based ranging for narrow-band signals. DTDOA-based LLS is sensitive to ranging errors and hence it is still challenging to achieve accurate positioning in complex indoor environments.

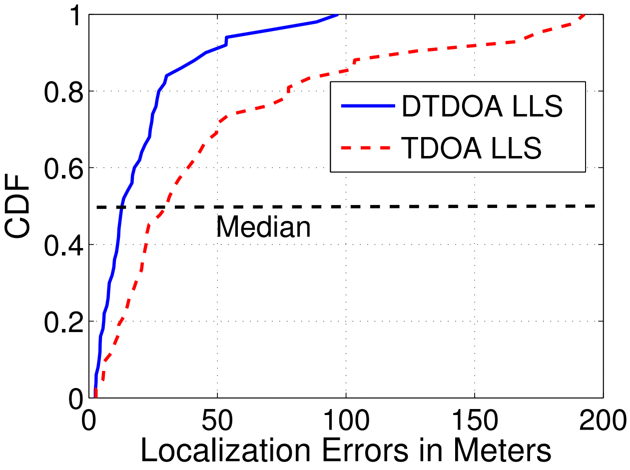
Measurement Results for DTDOA-based Fingerprinting

To achieve accurate indoor positioning with time-based information, i.e., DTDOA, we further evaluate our proposed DTDOA-based fingerprinting algorithm and compare the performance to RSS-based fingerprinting. The measurement notations in Table 4.2 are used in the following. To evaluate the performance of DTDOA-based fingerprinting, we mainly consider two aspects: accuracy and stability.

Accuracy: Table 4.3 summarizes the mean positioning errors and Standard Deviations (SD) of the four measurements. For DTDOA-based fingerprinting, the

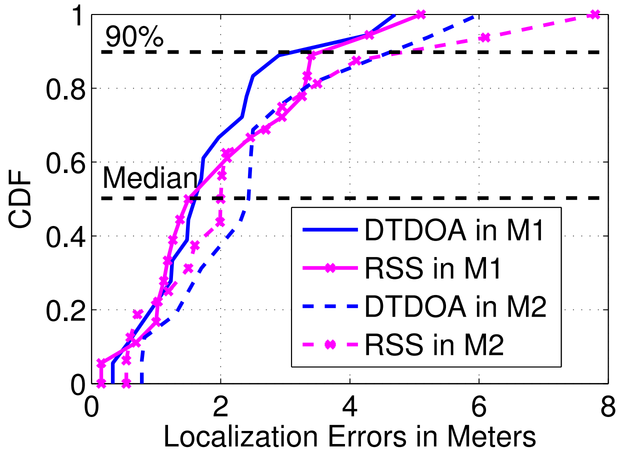


(a) CDF of Ranging Errors

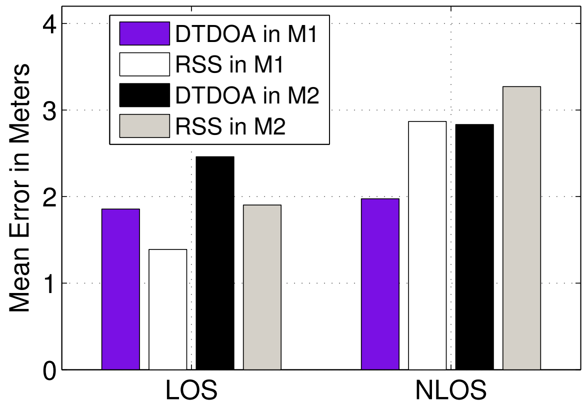


(b) CDF of Positioning Errors

Figure 4.8: Ranging and Positioning Errors

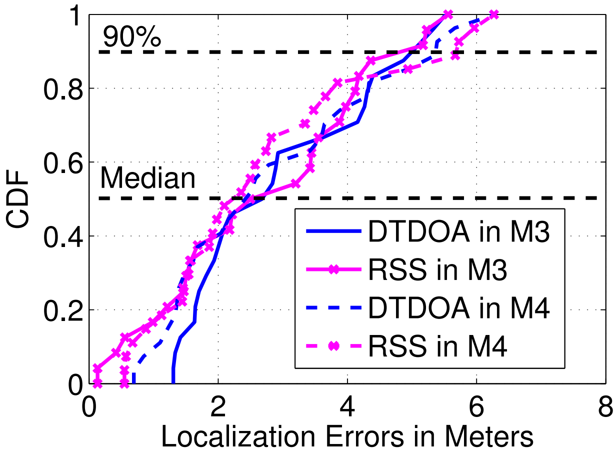


(a) CDF of Positioning Errors

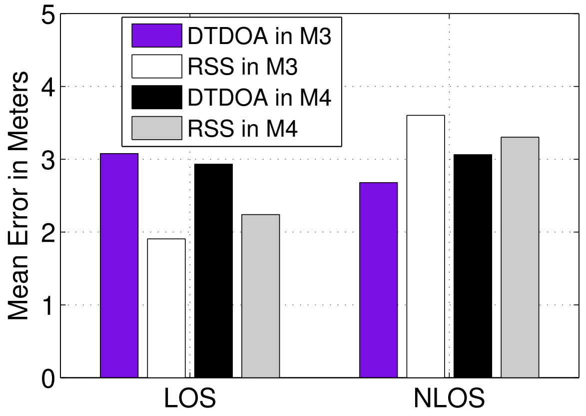


(b) Mean Errors in LOS and NLOS

Figure 4.9: Positioning Errors with WKNN on the Second Floor



(a) CDF of Positioning Errors



(b) Mean Errors in LOS and NLOS

Figure 4.10: Positioning Errors with WKNN on the Third Floor

mean errors in the four measurements are smaller than $3m$. Compared to range-based positioning (DTDOA-based LLS), DTDOA-based fingerprinting achieves much better accuracy and is thus better suited for indoor positioning. DTDOA-based and RSS-based fingerprinting algorithms achieve quite similar performance in terms of mean error in all four measurements. Figures 4.9(a) and 4.10(a) summarize CDFs of positioning errors. Similar as the observations based on mean error, CDFs indicate that DTDOA-based and RSS-based fingerprinting achieve similar performance in all four measurements.

We further investigate the performance of DTDOA-based and RSS-based fingerprinting in LOS and NLOS areas. Figure 4.9(b) shows the mean errors for DTDOA-based and RSS-based fingerprinting in LOS and NLOS areas on the second floor. Figure 4.10(b) is for the third floor. As shown in Figures 4.9(b) and 4.10(b), DTDOA-based and RSS-based fingerprinting achieve different performance in LOS and NLOS areas. In LOS areas, RSS-based fingerprinting achieves higher accuracy than DTDOA-based fingerprinting. It's because in the RSS vector there is a strong RSS component from the AN with LOS connection to the target, and therefore RSS-based fingerprinting can more accurately select the neighbours. However, in NLOS areas, there is no strong RSS component in the RSS vector and DTDOA-based fingerprinting achieves better performance in all four measurements. Take measurement M1 as an example. In LOS areas, RSS-based fingerprinting with mean error of $1.39m$ outperforms DTDOA-based fingerprinting with mean error of $1.86m$. In NLOS areas, DTDOA-based fingerprinting achieves mean error of $1.98m$, which is $0.89m$ better than RSS-based fingerprinting.

Stability: We analyze the stability of fingerprinting algorithms in the same scenario but at different measurement time. For M1 and M2 on the second floor, the environment is dynamic during working hours. Measurement M1 was conducted one day after creating the radio map but M2 was five days later. As shown in Figure 4.9(a), both DTDOA-based and RSS-based fingerprinting in measurement M2 achieve 90% positioning error below $4.6m$, which significantly deteriorates compared to the measurement M1 ($3.2m$ for DTDOA-based fingerprinting and $3.6m$ for RSS-based fingerprinting). The reason is that during working hours people move in offices and the layout of the surrounding environment also changes, e.g., doors become open and closed. These factors influence the accuracy for matching algorithms, i.e., WKNN, to find the correct neighbours based on the original fingerprinting database (radio map). Different from the measurements M1 and M2, the performance of DTDOA-based and RSS-based fingerprinting does not significantly deteriorate in the measurement M4 compared to M3 (Figure 4.10(a)), because the environment on the third floor is quite stable and the floor layout does not change.

4.6 Conclusions

In this chapter, we proposed some novel algorithms for time-based indoor positioning using narrow-band signals. Based on the findings for GPS synchronization in Chapter 3, we proposed to combine DTDOA with GPS synchronization to eliminate the influence of imperfect synchronization. According to a set of evaluations, we find that by combining DTDOA with GPS synchronization we are able to eliminate the momentary clock offsets and do not need offline calibration and strict limitation on the transmission interval between the packets from the target and reference node. With this method, our ranging accuracy gets improved compared to TDOA. This has been verified by the experiments in both outdoor and indoor with GSM and IEEE 802.15.4 signals. Correspondingly, our proposed DTDOA-based LLS significantly improves positioning accuracy, compared to TDOA-based LLS. However, in a complex indoor environment, NLOS propagation still introduces large errors. We further evaluated our proposed DTDOA-based fingerprinting and demonstrated that with narrow-band signals DTDOA-based fingerprinting significantly outperforms LLS and is able to locate the target in an indoor environment with a mean error of 3m. DTDOA-based and RSS-based fingerprinting algorithms achieve different performance in different areas. RSS-based fingerprinting achieves higher accuracy when there is one AN with LOS connection to the target. However, DTDOA-based fingerprinting outperforms RSS-based fingerprinting in the area, where there is no any AN with LOS connection to the target.

Part II

Indoor Positioning and Tracking using Fine-Grained Power

Part I (Chapter 3 and 4) presents our solutions for indoor positioning with narrowband signals, e.g., IEEE 802.15.4 signals, which are commonly used in some specific applications such as logistics and industrial applications. In our daily life, WiFi (IEEE 802.11) is the most commonly used short-range communication technology, which is ubiquitous indoor. Compared to IEEE 802.15.4 signals, IEEE 802.11 signals have wider bandwidth and more advanced physical-layer schemes, i.e., OFDM and MIMO. In the remainder of this thesis, i.e., part II, we introduce our indoor positioning and tracking solutions for WiFi signals. We propose a passive positioning/tracking system based on software defined radio techniques and an active network-based tracking system with commercial WiFi cards. **In this work, we aim to provide solutions, which can achieve high positioning accuracy and meanwhile reduce the calibration efforts (deployment efforts)**. Therefore, we work on range-based solutions instead of fingerprinting algorithms, which are very labour intensive and error prone to the changes of surrounding environments as observed in Chapter 4. In Chapter 5, we introduce our passive positioning system for WiFi signals based on software defined radio techniques, which is mainly designed for locating stationary targets. We introduce our ranging methods based on the channel information and an enhanced trilateration algorithm. In Chapter 6, we further extend the passive positioning system to support tracking mobile targets by designing an enhanced particle filter. In Chapter 7, we introduce an active tracking system for WiFi targets based on commercial WiFi cards. This system is able to extract channel information and inertial sensor information, and fuse them by a particle filter to achieve accurate tracking.

Chapter 5

A Passive WiFi Positioning System using Trilateration based on Channel Information

5.1 Introduction

In this chapter, we introduce a passive positioning system for WiFi targets, which extracts Channel Impulse Response (CIR) from overheard packets to design enhanced methods for ranging and positioning. Special merit of the system is its ability to operate independently of the tracked device and WiFi infrastructure, which is achieved by implementation using software defined radio (SDR) techniques. The SDR-based ANs are able to passively sniff the packets from the WiFi target device and transfer the collected physical layer information of the received packets, i.e., CIR, to a central server, which runs some enhanced positioning algorithms and indicates the location of the target.

In the system, we have proposed some enhanced range-based positioning algorithms as follows.

- First, instead of relying on RSSI, which is the most commonly used radio parameter for WiFi positioning, we work on CIR, which is a fine-grained power information. We use CIR to extract the power from the direct propagation path and mitigate the influence of multipath propagation. The key accuracy enabler in our system is signal recovery at the physical layer, which allows us to obtain fine-grained information such as Channel State Information (CSI) and CIR.

- Second, as mentioned in Section 2.5.2, LDPL (Log-Distance Path Loss model) is not accurate for indoor positioning because of multipath propagation and NLOS effects. To achieve high ranging accuracy, we propose a novel Nonlinear Regression (NLR) method to map the measured power information to propagation distance. With this ranging method, we achieve high positioning accuracy with much less calibration efforts compared to fingerprinting.

- Third, after obtaining range information, we propose an enhanced trilateration algorithm, which is robust to remaining ranging errors. This new trilateration approach combines the Weighted Centroid (WC) and Constrained Weighted Least Square (CWLS) algorithms to mitigate the impact of ranging errors.

We evaluate our system by conducting a comprehensive set of measurements including stationary and mobile targets under complex indoor propagation conditions. Experimental results demonstrate that our proposed NLR model achieves higher ranging accuracy than the LDPL model. Furthermore, compared to the LLS and WC algorithms, the WC-CWLS positioning algorithm achieves higher accuracy and is more robust to ranging errors. By combining the NLR model for ranging and the WC-CWLS algorithm for trilateration, the mean positioning accuracy of the system achieves $2.4m$. We further investigate the distribution of positioning accuracy with different positioning algorithms over the testing area. Note that in this chapter, we mainly focus on trilateration for stationary targets. Although we conduct some preliminary experiments to evaluate the proposed algorithms for a mobile target in this chapter, more comprehensive experiments for tracking a mobile target are given in Chapter 6.

In Section 5.2, the positioning algorithms including multipath mitigation via CIR, the NLR model for ranging and the WC-CWLS trilateration algorithm are introduced. Section 5.3 introduces the implementation of our proposed passive positioning system for WiFi targets. The evaluation results of our system in a complex indoor environment including stationary and mobile target experiments are introduced in Section 5.4. Finally, Section 5.5 concludes this chapter.

5.2 Positioning Algorithms with CIR Information

One of the main contributions in this work is a set of novel positioning algorithms by adopting CIR information, including ranging and trilateration. Figure 5.1 indicates positioning procedures in our work, which comprises three steps: multipath mitigation via CIR, ranging and trilateration.

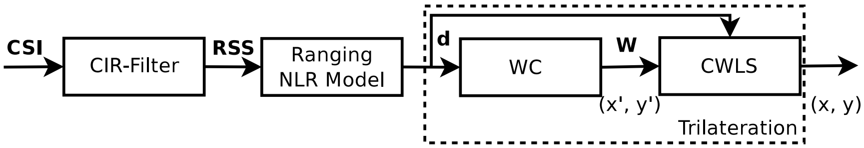


Figure 5.1: Overall Structure of Positioning Algorithms

5.2.1 Multipath Mitigation via CIR

As introduced in Section 2.5.3, CSI reveals a set of channel measurements depicting the amplitudes and phases of every subcarrier in the frequency domain. CIR characterizes the individual paths of the communication channel in the time domain as a set of temporal linear filters. CSI in the frequency domain can be converted into CIR in the time domain via IFFT. In the time domain, CIR can be modeled as a temporal linear filter as

$$h(\tau) = \sum_{n=1}^{N} a_n e^{-j\theta_n} \delta(\tau - \tau_n) \tag{5.1}$$

where a_n, θ_n and τ_n are the amplitude, phase and time delay of the nth path. N is the total number of paths and $\delta(\tau)$ is the Dirac delta function. CIR is a digitalized channel information in time domain, with a resolution of $\Delta\tau = \tau_n - \tau_{n-1}$. The resolution $\Delta\tau$ depends on the bandwidth of the used signal, i.e., $\Delta\tau = 1/B$, where B is the bandwidth of the signal. If the bandwidth B is infinite, CIR would be the same as the analog channel and it can distinguish all the propagation paths. However, in practice, the bandwidth B is limited. Therefore, the measured CIR with a resolution of $\Delta\tau = 1/B$ can only distinguish several clusters of propagation paths rather than every individual multipath component. The bandwidth of IEEE 802.11n is 20MHz and hence the time resolution of an estimated CIR is $1/20\text{MHz} = 50ns$, i.e., $\Delta\tau = 50ns$. [123, 126].

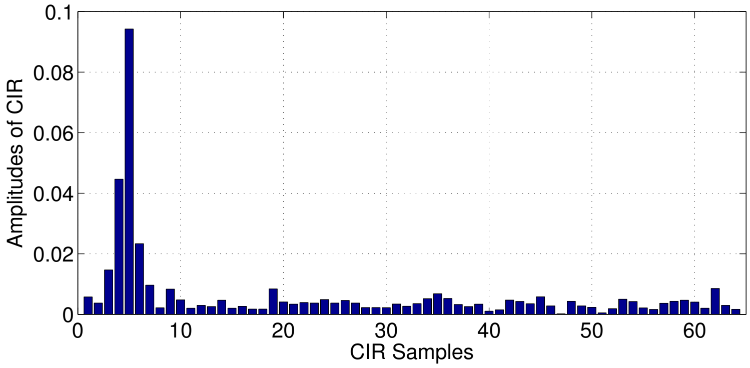


Figure 5.2: Channel Impulse Response

Figure 5.2 indicates the measured CIR with LOS connection between a target and a receiver. In the figure, Y-axis indicates the amplitude of CIR, i.e., a_n in Equation (5.1) and X-axis indicates the sample numbers, i.e., n in Equation (5.1). The time delay between two samples is $\Delta\tau = 50ns$. As shown in Figure 5.2, there is a path with strongest power (amplitude) in the CIR samples and the power in the other channels are much weaker. It is commonly known that the signal from the direct path is stronger than the other signals from multipath propagation. Hence, to mitigate the influence of multipath propagation, the path with the strongest power is selected as the direct path and the power in this path is chosen as the estimated power. The final estimated power is

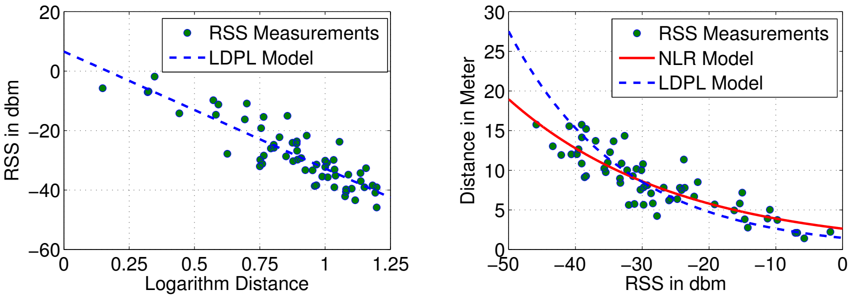
$$\text{RSS} = 10 \cdot \log_{10}[\max(|h(\tau)|)^2]. \tag{5.2}$$

$|h(\tau)|$ indicates the amplitudes of CIR over 64 samples. In case that no direct path exists (i.e., NLOS condition), this method still selects the strongest power in CIR as the estimated RSS, which is normally from the shortest propagation path. Therefore, with Equation (5.2), we are able to extract the shortest propagation path and mitigate the influence of other propagation paths. However, the power from the shortest propagation path is still influenced by the attenuation of obstacles, e.g., NLOS. Therefore, RSS with Equation (5.2) is less reliable and faces larger ranging errors in NLOS conditions than in LOS conditions.

Theoretically, in the measured CIR, the first sample should be the direct path because it has the shortest time delay. However, as shown in Figure 5.2, the sample in CIR with strongest power (direct path) is not consistent with the first sample in CIR. There is an uncertain delay between the direct path sample and the first sample in CIR. The main reason for this uncertain delay is that because of the unstable clock in the receiver, the receiver can not perfectly synchronize with the long preambles in the incoming packets, which generates an uncertain delay in the estimation of CIR based on the long preambles. Therefore, instead of relying on the first sample in CIR as the direct path, we consider the sample with the maximum power in CIR as the direct path.

5.2.2 Ranging

After extracting the fine-grained power, i.e., RSS in Equation (5.2), a model is required to convert the measured RSS into propagation distances (ranging) for range-based positioning. The LDPL model in Equation (2.20) is a generic model to predict the path loss for a wide range of environments. However, the LDPL model has been demonstrated to be inaccurate for indoor environments. A typical method to obtain the LDPL model is based on linear regression [21, 49], which models the relationship between the RSS values and logarithmic propagation distances ($\log_{10}(d)$) as a linear function as shown in Figure 5.3(a).



(a) LDPL in Logarithmic Distance (b) The NLR and LDPL Model

Figure 5.3: The LDPL and NLR Models

In our work, we propose to model the relationship between RSS values and propagation distances as a nonlinear curve fitting problem. Hence, we provide a nonlinear regression (NLR) model as

$$d_i = \alpha_i \cdot e^{\beta_i \cdot \mathrm{RSS}_i}, \tag{5.3}$$

where d_i is the distance between the target and ith AN, RSS_i is the RSS values obtained at the ith AN, α_i and β_j are two unknown parameters in the model that need to be obtained from some initial measurements. Depending on the layout of the test environment and locations of ANs, different ANs face different propagation channels. Therefore, we adapt different (α, β) pairs for different ANs to match different propagation channels.

In order to obtain (α, β) pairs, which can be adapted to fit different testing environments, we need to conduct training tests. Given K training locations in the initial measurements, $(d_{ij}, \mathrm{RSS}_{ij})$ are collected at the jth training location from the ith AN. We apply the nonlinear least square criterion, in which the sum of squared residuals is minimized as

$$\underset{(\alpha_i,\beta_i)}{\operatorname{argmin}} \sum_{j=1}^{K} (\alpha_i \cdot e^{\beta_i \cdot \mathrm{RSS}_{ij}} - d_{ij})^2. \tag{5.4}$$

To find the solution of this unconstrained optimization problem, the trust region algorithm [32] (introduced in Section 2.6.1) is applied in our work, because it is robust and has strong global convergence properties. The red solid curve in Figure 5.3(b) indicates the NLR model to fit the RSS measurements and the ground truth ranges. Since the training experiments are only used to obtain the two parameters (α, β) instead of building a radio map, which needs to cover the whole testing environments, the training efforts for ranging are much lower than for fingerprinting.

5.2.3 Two-Stage Trilateration

After the ranging step, the propagation distance information is fed into a trilateration algorithm. A well designed trilateration algorithm should be robust to ranging errors. As introduced in Section 5.2.1, extracting power from direct path via CIR can mitigate the influence of multipath propagation in LOS condition but in NLOS condition the obtained RSS is less reliable and has larger ranging errors. Correspondingly, the ranging accuracy in NLOS condition is worse than in LOS conditions. To locate a target, a trilateration algorithm requires at least three ranging values and in a complex indoor environment these ranging values are normally mixed by ranges with and without LOS connections. The trilateration algorithm using LLS, which just simply treats all the ranging values equally, is prone to ranging errors. Weighting techniques can be adopted to mitigate the influence of ranging errors by assigning larger weights to more reliable ranges and smaller weights to less reliable ranges. WLS and CWLS (introduced in Section 2.6) are two commonly used trilateration algorithms by using weighting techniques. However, as introduced in Section 2.6, to achieve the optimal solution, the information of the ground truth ranges and the variances of each measurement is needed to set weights, which is unrealistic in real applications. In our work, we propose a new two-stage trilateration algorithm combining the WC (Weighted Centroid in Section 2.4.2) and CWLS (Section 2.6.4) algorithms to mitigate the remaining ranging errors, which does not require the ground truth ranges. In the remainder of this thesis, we refer to this two-stage trilateration algorithm as WC-CWLS.

In our proposed WC-CWLS, we first estimate an initial location based on a WC algorithm because of its simplicity and robustness to inaccurate ranging. For the WC algorithm, the location of the target is estimated as a weighted average of the coordinates of ANs as

$$(x', y') = \sum_{i=1}^{M} [w_i \cdot (x_i, y_i)], \tag{5.5}$$

where M is the number of ANs and (x_i, y_i) are the coordinates of the ith AN. Each weight w_i is inversely proportional to the range as

$$w_i = \frac{\frac{1}{d_i}}{\sum_{j=1}^{M} \frac{1}{d_j}}. \tag{5.6}$$

Based on the initial location (x', y'), we calculate the distance between the ith AN and the initial location as,

$$r_i = \sqrt{(x' - x_i)^2 + (y' - y_i)^2}. \tag{5.7}$$

At the second step, a CWLS is adopted to fine tune the estimated location. For the CWLS algorithm, the location of the target is estimated by solving a constrained optimization problem,

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} (\mathbf{G}\boldsymbol{\theta} - \mathbf{h})^T \mathbf{W} (\mathbf{G}\boldsymbol{\theta} - \mathbf{h}), \tag{5.8}$$

subject to

$$\mathbf{q}^T \boldsymbol{\theta} + \boldsymbol{\theta}^T \mathbf{P} \boldsymbol{\theta} = 0,$$

where $\boldsymbol{\theta} = [x,\ y,\ R]^T$, $R = \sqrt{x^2 + y^2}$,

$$\mathbf{h} = \tfrac{1}{2} \begin{pmatrix} x_1^2 + y_1^2 - d_1^2 \\ \vdots \\ x_M^2 + y_M^2 - d_M^2 \end{pmatrix}, \mathbf{G} = \begin{pmatrix} x_1 & y_1 & -0.5 \\ \vdots & \vdots & \vdots \\ x_M & y_M & -0.5 \end{pmatrix},$$

$$\mathbf{P} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \text{ and } \mathbf{q} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}.$$

Before solving the optimization problem in Equation (5.8), the weight matrix $\mathbf{W}$ needs to be properly set. The optimal solution as introduced in [37] requires ground truth range information and variances, which are unrealistic in a practical application. Therefore, in our work, we propose a suboptimal solution to set the weight matrix $\mathbf{W}$ based on the initial location estimated by WC on the first step. We set the weight matrix based on the error between the squared ranging outputs d_i and r_i in Equation (5.7) as

$$\varepsilon_i = r_i^2 - d_i^2,\ i = 1, 2, ..., M. \tag{5.9}$$

The covariance matrix of the disturbance is calculated as

$$\mathbf{\Psi} = \text{diag}([\varepsilon_1^2, \varepsilon_2^2, ..., \varepsilon_M^2]), \tag{5.10}$$

and the weighted matrix is obtained as

$$\mathbf{W} = \mathbf{\Psi}^{-1}. \tag{5.11}$$

After setting the weight matrix, the constrained optimal problem in Equation (5.8) is equivalent to minimize the Lagrangian equation,

$$\begin{aligned} \hat{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\text{argmin}}\, L(\boldsymbol{\theta}, \lambda) \\ &= \underset{\boldsymbol{\theta}}{\text{argmin}}((\mathbf{G}\boldsymbol{\theta} - \mathbf{h})^T \mathbf{W} (\mathbf{G}\boldsymbol{\theta} - \mathbf{h}) + \lambda(\mathbf{q}^T \boldsymbol{\theta} + \boldsymbol{\theta}^T \mathbf{p} \boldsymbol{\theta})), \end{aligned} \tag{5.12}$$

where λ is the Lagrange multiplier. To obtain the final estimation of the target location, we use the proposed method in [37] to solve the Equation (5.12) and to obtain the final target location (x, y) in $\hat{\boldsymbol{\theta}}$.

5.3 System Implementation

Our proposed positioning algorithms have been implemented in a software defined radio based passive positioning system for WiFi devices, which is another contribution in this work. Figure 5.4 indicates an overview of this passive positioning system. In this system, users just use their WiFi devices as usual, e.g., a laptop, which exchanges packets with a WiFi router (Access Point (AP)). The deployed WiFi sniffers are used to passively sniff the packets from WiFi users and a central server is used to locate the users. Figure 5.5 indicates the structure of this system. Basically, the system are divided into three main components: receiving hardware for WiFi signals, WiFi packet decoding, and positioning algorithms.

5.3.1 Receiving Hardware for WiFi Signals

As mentioned in Section 2.5.3, off-the-shelf network cards (IWL5300) with firmware [60] can not be adopted for a passive positioning system to extract the channel state information in ANs. Hence, to decode IEEE 802.11n uplink messages and extract channel state information, we adopt SDR techniques for ANs. In our work, the sniffing component is based on USRP N210 receivers [17].

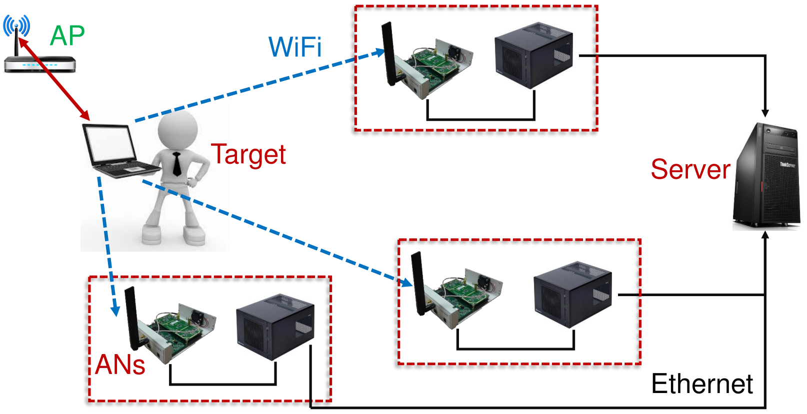


Figure 5.4: An Overview of the Passive Positioning System for IEEE 802.11n

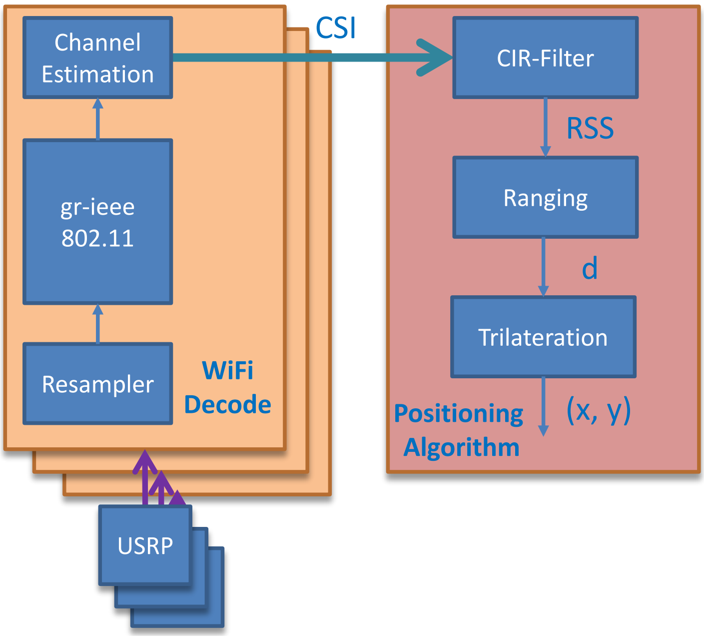
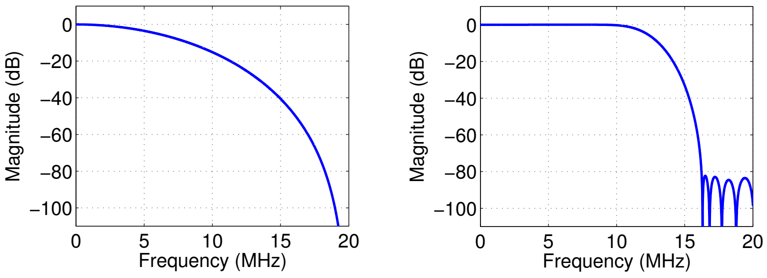


Figure 5.5: The Structure of Passive Positioning System for IEEE 802.11n

5.3.2 WiFi Packet Decoding

Each USRP device is connected to one individual desktop, in which GNU Radio software [5] is adopted for signal processing. WiFi packet decoding is mainly realized by the framework [28] for IEEE 802.11a/g/p decoding in GNU Radio (gr-ieee 802.11 block in Figure 5.5). We extract the long preambles from the decoded WiFi packets and design a channel estimation block based on MATLAB to estimate CSI in frequency domain. To meet our requirements of channel estimation and positioning, we modify the framework (gr-ieee 802.11) as follows.

First, in order to mitigate estimation errors of CSI, the distortion of the filters (DDC filters in FPGA as introduced in Section 2.3) in the USRP receivers need to be minimized in the frequency domain. To achieve this goal, instead of 20MHz sampling rate in the framework [28], we use a 25MHz sampling rate to keep the amplitude of the frequency response of the DDC filters in the USRP receivers constant in the target bandwidth. We work on the IEEE 802.11n standard with 20MHz bandwidth and therefore the amplitude of the frequency response of DDC filters in the baseband is required to be constant within 10MHz. In the mother-board of USRP, a Cascaded Integrator Comb (CIC) filter is implemented for DDC to convert the sampling rate from 100MHz to the required rate. In the CIC filter, the down-sampling rate is required to be integer multiplications of 4 to avoid the serious frequency roll-off (the steepness of the amplitude of frequency response). Figures 5.6(a) and 5.6(b) show the simulated baseband frequency response of the CIC filter used in the mother-board of USRP N210 with 20MHz and 25MHz sampling rates respectively. With a sampling rate of 20MHz, the filter has about 18dB attenuation at 10MHz compared to 0MHz. This will cause the distortion of incoming signals (e.g., pilot signals for channel estimation) in the frequency domain. With a sampling rate of 25MHz, the amplitude of frequency response of CIC filter is constant from 0MHz to 10MHz. Therefore, we use the sampling rate of 25MHz as the output from USRP and a resampler is adopted at the beginning of signal processing to convert the sampling rate from 25MHz to 20MHz.



(a) 20MHz Sampling Rate (b) 25MHz Sampling Rate

Figure 5.6: Baseband Frequency Response of CIC Filter with Different Sampling Rates

Second, we use Long Preambles (LP) in the decoded WiFi packets for channel estimation, which are normally used for packet synchronization and channel estimation in WiFi receivers [13]. In the framework [28], the long preambles are detected before demodulation and the packets are reconstructed after demodulation. Therefore, to map the long preambles to their corresponding packets, we pass the long preambles to the packet reconstruction module through the whole decoding procedure. To achieve this, we adopt the stream tags mechanism [5] from GNU Radio. In addition, for passive positioning, the server running positioning algorithms need to aggregate the RSS values of the packets from different ANs in the same time interval. Hence, the clock time in ANs are synchronized by GPS receivers with an accuracy of sub-microseconds. GPS timestamps that indicate the detection time of the packets are also attached by stream tags in GNU Radio. Consequently, the server aligns the packets from different ANs with GPS time and aggregate the RSS values in the same time interval from different ANs.

Third, long preambles are attached to packets and passed to MATLAB for channel estimation. We adopt block-type pilot channel estimation based on long preambles to estimate CSIs in 64 subcarriers. We adopt the Least Square (LS) estimator to estimate CSI in the frequency domain. CSI is estimated as,

$$\hat{\mathbf{H}} = \underset{\mathbf{H}}{\operatorname{argmin}} (\mathbf{Y} - \bar{\mathbf{X}}\mathbf{H})^H (\mathbf{Y} - \bar{\mathbf{X}}\mathbf{H}), \tag{5.13}$$

where $(\cdot)^H$ indicates the conjugate transpose operation, and $\bar{\mathbf{X}}$ and $\mathbf{Y}$ are the predefined and received long preambles in the frequency domain respectively. As derived in [112], the solution of the LS estimator is given as,

$$\hat{\mathbf{H}} = \bar{\mathbf{X}}^{-1}\mathbf{Y}. \tag{5.14}$$

5.3.3 Positioning Algorithms

The CSI information measured at each AN is passed to a central server to locate the target based on our proposed positioning algorithms in Section 5.2. All the positioning algorithms are implemented in MATLAB. As shown in Figure 5.5, the estimated CSI should be first converted to CIR and then extract the power from direct path as RSS. Then, RSS is converted to the propagation distance based on our proposed NLR ranging model. Finally, our proposed WC-CWLS is adopted to locate the target.

Because the system is passive, positioning algorithms are adopted based on the packets, which are sent from the terminal, e.g., a smart phone or laptop. Assume that the target person is holding a terminal e.g., a laptop, to watch online video or browse website. After some control messages exchanging between the terminal and WiFi router, e.g., RTS/CTS (Request To Send/ Clear To Send), the WiFi router sends data packets to the terminal. After receiving data packets from the WiFi router, the terminal sends back ACK packets. ANs in our system are deployed to capture these ACK packets for positioning. As mentioned in Section 2.2.3, ACKs in IEEE 802.11n consist of block ACK and legacy ACK. According to our experiments, legacy ACKs sent from the WiFi devices are much more than block ACKs in IEEE 802.11n. However, the head of a legacy ACK only contains the destination MAC address (the WiFi router) but no source MAC address (the terminal). Therefore, in our current system, legacy ACK can only be used in the condition that a single user (target terminal) connects to one WiFi router, where the destination MAC address (the WiFi router) is used to distinguish the target terminal from the other terminals connected to the other WiFi routers. To identify multiple users, our system stores the block ACKs and identifies the multiple users with the source MAC address in the block ACKs. However, depending on the configurations of the WiFi routers, the packet rates for block ACKs are very different. In the worst case that aggregate-MPDU frame aggregation mechanism is disabled, block ACK packets are not used. In addition, ANs can also capture the data packets from the WiFi access point, in which the MAC address of the user is included in the MAC header. Hence, in future work, each legacy ACK can be further matched to its corresponding data packet and consequently different users connected to the same access point can be distinguished by the legacy ACKs.

Since in this thesis we mainly focus on the accuracy and deployment efforts of the positioning algorithms, the positioning and tracking accuracy of this system is tested with legacy ACKs for a single user. Extension of the system to better support multi-user positioning and tracking is considered as one of the future work, which is discussed in Section 8.2.

5.4 Performance Evaluation

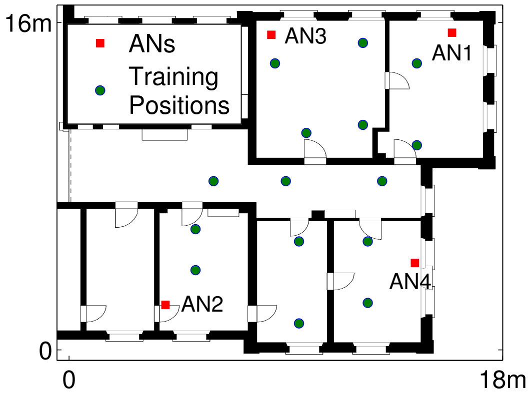
To evaluate the positioning accuracy of our proposed system, we have conducted a set of comprehensive experiments in a complex indoor environment. The system is deployed on the second floor of the INF building at University of Bern.

5.4.1 Measurement Setup

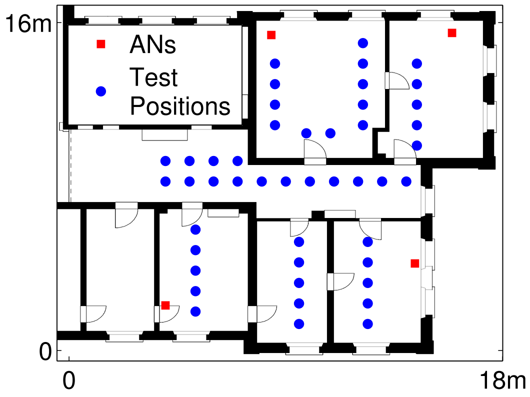
As shown in Figure 5.7, four USRP receivers have been deployed in our working area as ANs to monitor the packets from a laptop. In our experiments, the target for positioning is a Thinkpad T430 laptop with an Intel N6300AGN wireless card. The laptop was configured to continuously refresh a website to generate enough data traffic. In the following experiments, 10 legacy ACK packets are received in one second on average.

- First, some initial experiments for training were conducted as shown in Figure 5.7(a) to define the unknown parameters (α, β) in the NLR model. 15 training positions that are spread over the whole interesting area were selected to acquire the exponential factor in the LDPL model and (α, β) in the NLR model. These initial experiments were conducted during weekend when there are no moving people in the testing environment.

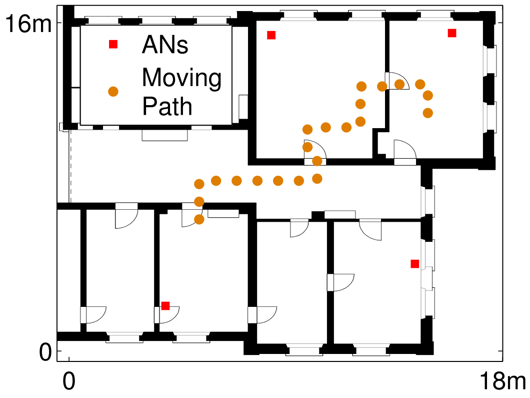
- Second, to evaluate the accuracy of our proposed positioning algorithms, a set of experiments have been conducted as shown in Figure 5.7(b). In these



(a) Initial Measurements



(b) Stationary Target Measurements



(c) Mobile Target Measurements

Figure 5.7: Measurement Setup

Table 5.1: Parameters for the NLR Model

	AN1	AN2	AN3	AN4
α	2.583	1.926	3.024	3.034
β	-0.044	-0.046	-0.037	-0.031

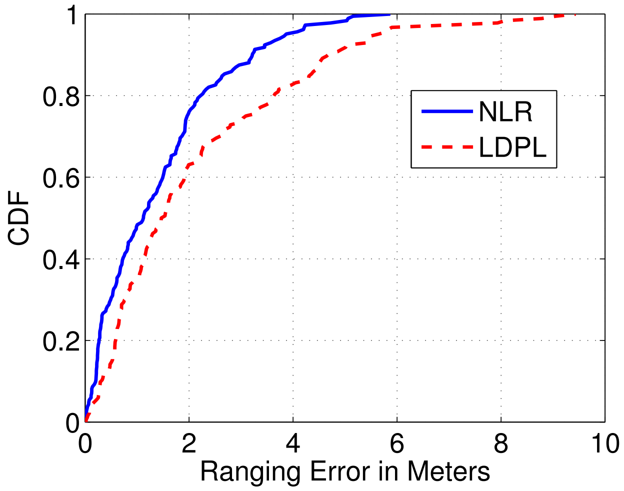


Figure 5.8: Ranging Errors

experiments, the laptop was stationary at 46 positions, which cover the aisles in the whole target area. At each position, the data collection duration was 30s and hence around 300 packets were collected for positioning. 4 ANs were deployed at the same locations as in the initial experiments. The experiments were conducted during working hours when people were moving in the offices.

- Third, additional experiments have been conducted to analyze the performance of the system for a mobile target, in which the laptop was held by a person and moved as the indicated trace in Figure 5.7(c). The speed of movement was around $0.86m/s$. The positioning algorithms run every second to estimate the position of the moving target and hence around 10 packets were collected for each location estimation along the moving trace.

5.4.2 Experiments with a Stationary Target

Ranging Accuracy

Accurate ranging is a preliminary step for accurate range-based positioning. Based on the trust region algorithm for the NLR model and the initial experiments, we obtain α and β for different ANs as shown in Table 5.1. Because ANs at different locations face different propagation channels, α and β in Table 5.1 are different for different ANs. In addition, we also obtain the LDPL model based on these

Table 5.2: Positioning Errors for Stationary Target

	LDPL Model			NLR Model		
Algorithms	Mean	SD	Max.	Mean	SD	Max.
WC-CWLS	2.6m	1.2m	5.7m	2.4m	1.3m	6.1m
LLS	3.8m	2.6m	11.3m	2.5m	1.6m	6.8m
WC	2.9m	1.6m	6.6m	3.1m	1.5m	6.8m

measurements for comparison.

We calculate the ranging errors to each AN at the 46 test positions based on the NLR and LDPL models. Figure 5.8 indicates CDFs of the ranging errors for both models. The NLR model improves the median ranging error by 27% (from $1.5m$ to $1.1m$) compared to the LDPL model. In addition, the overall improvement of the maximum ranging error based on the NLR model is about 36% (from $9.2m$ to $5.9m$). Therefore, we can conclude that the NLR model significantly outperforms the LDPL model for the ranging step.

The training stage for path loss model is still required when the system is deployed in a new scenario or when the surrounding environments change. However, compared to fingerprinting methods, which need large amount of training positions to cover the whole testing environment, the training efforts (15 training positions) for path loss model is much less and correspondingly the time consumed in the deployment of the system is much less.

Positioning Accuracy

After evaluating the ranging accuracy, we investigate the performance of our proposed WC-CWLS algorithm with both the LDPL and NLR models. LLS and WC, whose weights are the same as in Equation (5.6), are also evaluated for comparison.

LDPL Model: The performance of the WC-CWLS algorithm is first analyzed based on the LDPL model and compared to LLS and WC algorithms. Figure 5.9 presents CDFs of positioning errors based on the WC-CWLS, LLS and WC algorithms. Table 5.2 summarizes the mean, Standard Deviation (SD) and maximum values of the positioning errors.

Recall that the accuracy of ranging is low with the LDPL model. Under this condition, the LLS algorithm is sensitive to the ranging errors and thus has the worst performance with a mean error of $3.8m$. The WC and WC-CWLS algorithms are more robust against ranging errors. As introduced in Section 2.4.2, WC is more robust to ranging errors than LLS and hence achieves higher positioning accuracy. For WC-CWLS, the introduced weighting technique as in Equation (5.11) efficiently mitigates the influence of ranging errors on trilateration. Therefore, in these experiments with LDPL model for ranging, the WC-CWLS algorithm achieves the best performance. The mean error of WC-CWLS is $2.6m$, which is $0.3m$ lower than the WC algorithm and $1.2m$ lower than the LLS algorithm. Moreover, the maximum positioning error of the WC-CWLS algorithm is $5.7m$, which

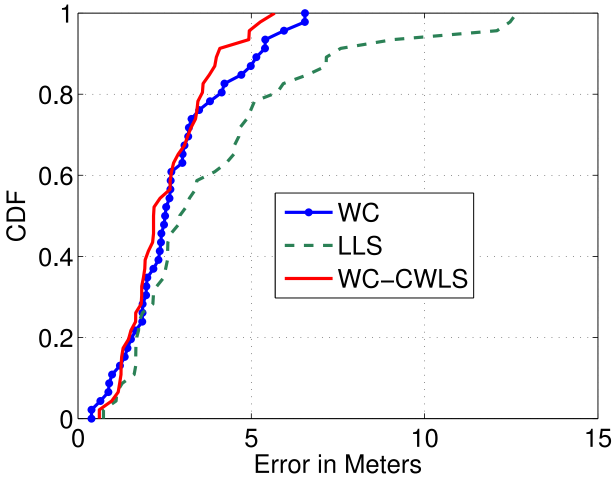


Figure 5.9: Positioning Errors with LDPL Model (Stationary)

is $0.9m$ lower than the WC algorithm and $5.6m$ lower than the LLS algorithm. CDFs of the errors in Figure 5.9 present a complete view on the performance of the three algorithms. We find that the WC-CWLS algorithm generally outperforms both the WC and LLS algorithms.

In addition, the standard deviation of the positioning errors based on the WC-CWLS algorithm is the lowest among these three algorithms ($1.2m$ compared to $2.6m$ for the LLS algorithm and $1.6m$ for the WC algorithm), which indicates that the performance of the WC-CWLS algorithm is more stable.

NLR Model: Figure 5.10 shows CDFs of the positioning errors for the three positioning algorithms with the proposed NLR model. The mean, standard deviation and maximum values of the positioning errors are summarized in Table 5.2.

Compared to the LDPL model, the NLR model improves the performance of trilateration algorithms, i.e., the WC-CWLS and LLS algorithms, due to higher ranging accuracy. The NLR model improves the mean error of WC-CWLS from $2.6m$ to $2.4m$ and LLS from $3.8m$ to $2.5m$. However, the performance of the WC algorithm becomes worse because it is a proximity algorithm and its performance does not highly depend on the ranging accuracy.

With the NLR model, the WC-CWLS algorithm achieves a mean error of $2.4m$, which outperforms both the LLS ($2.5m$) and WC ($3.1m$) algorithms. With a higher ranging accuracy, the performance of the LLS algorithm gets significantly improved and is comparable to the WC-CWLS algorithm. Based on the CDF curves in Figure 5.10, the LLS algorithm achieves quite similar performance as the WC-CWLS algorithm for small errors (lower than $3m$). However, the WC-CWLS algorithm is generally better than the LLS algorithm for errors larger than $3m$ because the WC-CWLS algorithm is more robust to ranging errors. For instance, the maximum error of the WC-CWLS algorithm is $6.1m$, which is $0.7m$ lower than the LLS algorithm. With a high ranging accuracy, the WC-CWLS algorithm significantly

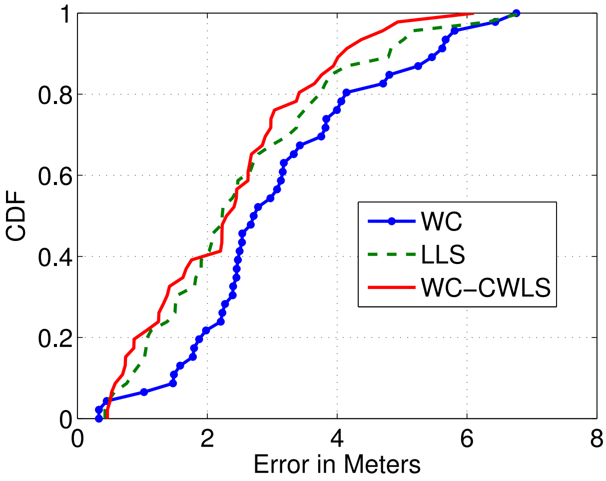


Figure 5.10: Positioning Errors with NLR Model (Stationary)

outperforms the WC algorithm.

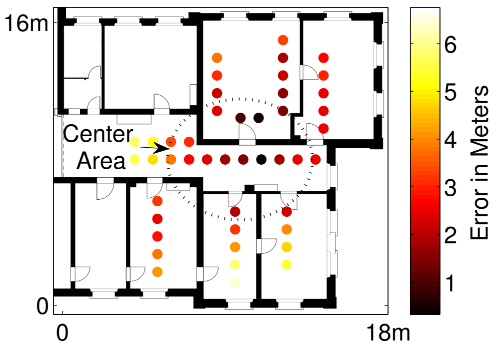
In addition, we analyze the distribution of positioning errors in different areas for all the three algorithms. As shown in Figure 5.11, we refer to the area in the center of the target area, i.e., the polygon with ANs as vertexes, as center area and the rest as surrounding areas. Figure 5.11 shows the distribution of positioning errors with the NLR model, in which the color gets lighter for larger positioning errors. First, we can find that the positioning errors of all the three algorithms in the center area are lower than the surrounding areas, especially for the WC algorithm in Figure 5.11(a). Second, for trilateration algorithms, i.e., the LLS and WC-CWLS algorithms, the error distributions are much more uniform than the WC algorithm in the whole area. Hence, trilateration algorithms with the NLR model achieve higher accuracy than the WC algorithm, especially for the surrounding areas.

Table 5.3: Positioning Errors for Mobile Target

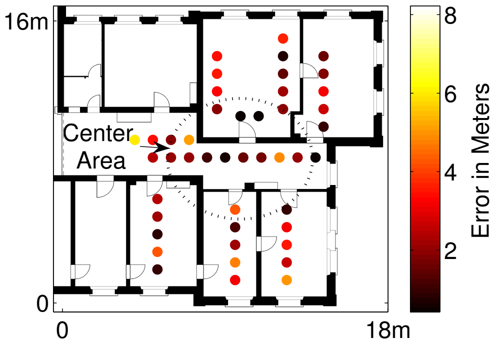
	LDPL Model			NLR Model		
Algorithms	Mean	SD	Max.	Mean	SD	Max.
WC-CWLS	2.0m	0.9m	3.7m	1.7m	0.8m	3m
LLS	2.4m	1.1m	4.7m	2.1m	1.6m	5.8m
WC	2.2m	1.1m	4.5m	1.8m	1.1m	4.5m

5.4.3 Experiments with a Mobile Target

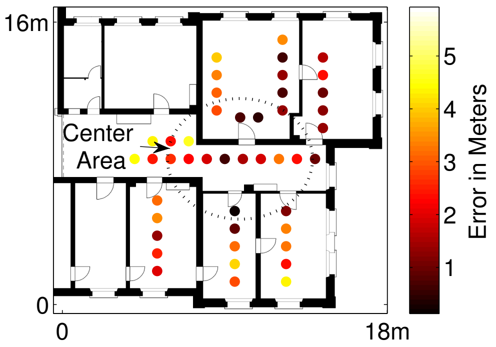
Figures 5.12 and 5.13 present CDFs of the positioning errors with the LDPL model and NLR model for a mobile target respectively. Table 5.3 summarizes the mean, standard deviation and maximum values of the positioning errors. The positioning accuracy with a mobile target is a superset of the positioning accuracy at each point of the trajectory. Thus, depending on the trajectory, the accuracy can be worse or



(a) WC



(b) LLS



(c) WC-CWLS

Figure 5.11: Positioning Error Distribution with NLR Model

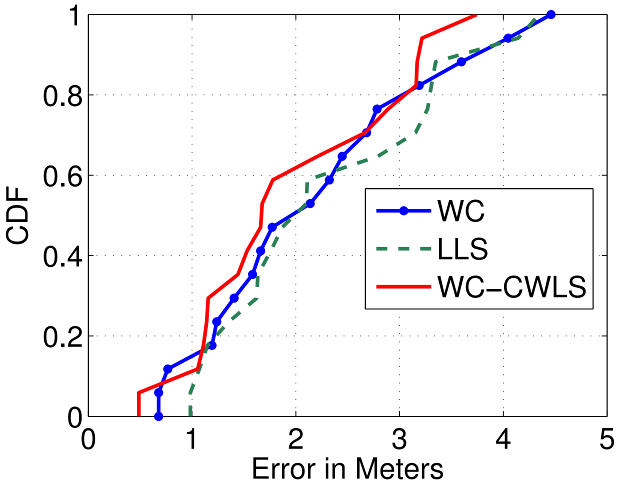


Figure 5.12: Positioning Errors with LDPL Model (Moving)

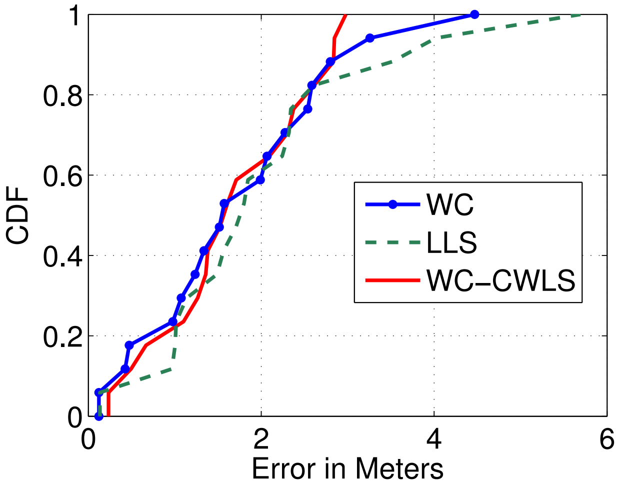


Figure 5.13: Positioning Errors with NLR Model (Moving)

better compared to the stationary target measurements. In our case, the positioning errors in the experiments with the mobile target are lower than in the experiments with the stationary target because most parts of the trajectory are located in or around the center area, where the positioning errors of all the test algorithms are low.

Based on these results, we find some similar observations as in the experiments with the stationary target. First, trilateration algorithms with the NLR model, i.e., the LLS and WC-CWLS algorithms, perform better than with the LDPL model. For example, the mean error of WC-CWLS with the NLR model is $1.7m$ that is $0.3m$ lower than the LDPL model. Second, WC-CWLS generally outperforms the WC and LLS algorithms with both the LDPL and NLR models. Compared to the experiments with the stationary target, the improvement is smaller because most parts of our testing trace as indicated in Figure 5.7(c) are located in or around the center area, where the other two algorithms achieves higher accuracy than in the surrounding areas. For example, with the NLR model, the mean error of the WC-CWLS algorithm is $1.7m$ that is $0.4m$ lower than the LLS algorithm and $0.1m$ lower than the WC algorithm. Finally, the WC-CWLS algorithm achieves $3.7m$ and $3m$ of the maximum errors with the LDPL and NLR models respectively, which outperforms both the WC ($4.5m$ with both the LDPL and NLR models) and LLS algorithms ($4.7m$ with the LDPL model and $5.8m$ with the NLR model).

5.5 Conclusions

In this work, we designed and implemented a software defined radio based passive positioning system for WiFi signals. In the system, software defined radio techniques are adopted to overhear the packets from the target devices and allow us to extract physical-layer channel information for positioning. We adopted channel impulse response to mitigate the influence of multipath propagation on the received signal strength. The new model based on nonlinear regression (the NLR model) proposed in this work to relate the measured RSS to propagation distance achieves significant higher accuracy than the commonly used LDPL and requires much less calibration efforts than fingerprinting. This accurate ranging method forms the basis for the positioning with high accuracy. Furthermore, our proposed WC-CWLS combining the Weight Centroid (WC) algorithm and Constrained Weighted Least Square (CWLS) algorithm is able to further mitigate the influence of remaining range errors on the positioning accuracy. Comprehensive experiments for a single user show that our proposed WC-CWLS algorithm is more robust to ranging errors than the LLS algorithm and achieves better positioning accuracy than the WC and LLS algorithms with both the NLR model and LDPL model. By combining NLR for ranging and WC-CWLS for positioning, we are able to achieve a mean accuracy of 2.4m for a stationary target.

Chapter 6

Passive Tracking of WiFi Devices with an Enhanced Particle Filter based on Channel Information

6.1 Introduction

In Chapter 5 we have introduced a passive positioning system with an enhanced trilateration algorithm (WC-CWLS). The enhanced trilateration algorithm achieves better performance compared to LLS. Although we have evaluated its performance with a mobile target in Chapter 5, WC-CWLS as a trilateration algorithm considers the positioning problem as a convex optimization problem and independently estimates location of the target at each time interval.

In this chapter, we extend the passive positioning system for WiFi signals (in Chapter 5) to support tracking a mobile target by designing an enhanced particle filter. To design passive positioning systems for mobile targets, one of the critical challenges is the limited information for positioning. Since ANs only overhear signals, the ranging schemes only rely on simple radio signal parameters measured at ANs, such as timing and power information. Converting this basic information into the dynamic locations of a mobile target involves a sequence of steps, each of which introduces errors.

In our work, we aim to provide a passive tracking system for WiFi users. The foremost goal of our work is to minimize the aforementioned errors in each step as much as possible to achieve high tracking accuracy. In addition to high tracking accuracy, we also aim to reduce the calibration efforts compared to fingerprinting, which is the dominating power-based positioning method. Additionally, we also consider the computation efforts of the positioning system.

To achieve the aforementioned goals, we design an enhanced particle filter exclusively relying on fine-grained power-based ranging, in which the initial calibration efforts are significantly less than fingerprinting. Our main scientific contributions are summarized as follows.

- First, we propose an enhanced particle filter for indoor tracking. In the proposed enhanced particle filter, we have three main scientific contributions. First, we propose to use a single coordinated turn model, which considers the angle variation of the moving direction in the movement state. The particle filter with the single coordinated turn model provides higher tracking accuracy than a commonly used constant velocity model. Additionally, it requires much less computation efforts than the multi-model particle filter. Second, we investigate the impact of ranging errors on the likelihood in the particle filter. By weighting the likelihoods for different ANs based on their ranging outputs, our particle filter mitigates the influence of ranging errors. Third, in a passive positioning system, speed information is normally unavailable for the tracking process because the system can not get the inertial sensor information from the target. In our system, we consider the moving speed limitation on the likelihood by filtering out the uncommonly large moving speed for people in indoor environments.

- Second, we use a similar method as in Chapter 5 to achieve high ranging accuracy by using CIR to obtain power from the direct path and a NLR model to calculate the ranges. In this work, considering the movement of the target, we smooth the sequentially measured fine-grained power by a Savitzky-Golay (S-G) filter [71], which considers the trend of power variation in the moving window, to further mitigate the multipath effects.
- Third, besides our proposed tracking mechanism, we also implement a collection of commonly used positioning mechanisms, i.e., Bootstrap Particle Filter (BPF), MM-BPF, Extended Kalman Filter (EKF), Trilateration with Maximum Likelihood (ML same as NLS in Section 2.6.1), Linear Least Square (LLS), and WC-CWLS. We provide an experimental evaluation and comparison of those positioning mechanisms along different moving paths in complex indoor environments.

In the remainder of the chapter, a general form of the range-based particle filter is introduced in Section 6.2. In this section, the problems in a passive tracking system are particularly stated. Our main contributions are introduced in Section 6.3, in which the proposed enhanced particle filter is described. The ranging mechanisms are presented in Section 6.4. Section 6.5 presents the implementation of the proposed algorithms in a passive SDR-based positioning system for WiFi signals. Section 6.6 presents the evaluation results in complex indoor environments. Finally, Section 6.7 concludes this chapter.

6.2 Range-only Particle Filter and Problem Statement

Based on the concepts of particle filters introduced in Section 2.8, we discuss a general form of particle filter exclusively relying on range information, i.e., range-only particle filter in this section. Then, we state the problems in this range-only particle filter.

6.2.1 Range-only Particle Filter

As introduced in Section 2.8, one of the most widely used and efficiently implementable particle filter is the bootstrap particle filter (BPF) [63], in which the importance density is chosen to be equal to the transition density as

$$q(\mathbf{x}_k|\mathbf{x}_{k-1},\mathbf{z}_k) = p(\mathbf{x}_k|\mathbf{x}_{k-1}). \tag{6.1}$$

Hence, the associated weights are calculated as

$$w_k^i \propto w_{k-1}^i \cdot p(\mathbf{z}_k|\mathbf{x}_k^i), \tag{6.2}$$

in which the associated weights are only determined by the likelihood function of $p(\mathbf{z}_k|\mathbf{x}_k^i)$.

An efficient and accurate derivation of the likelihood function $p(\mathbf{z}_k|\mathbf{x}_k^i)$ is important for accurate tracking by BPF. For range-based tracking, $\mathbf{z}_k$ comprises range information from different ANs, i.e., $\mathbf{z}_k = [d_1, d_2, \cdots, d_N]$, where d_j is the range between the target and the jth AN. Assuming that the range information from different ANs are independent from each other, a traditional likelihood is defined as

$$p(\mathbf{z}_k|\mathbf{x}_k^i) = \Pi_{j=1}^N p(d_j|\mathbf{x}_k^i). \tag{6.3}$$

In order to distinguish two likelihoods, we refer to $p(\mathbf{z}_k|\mathbf{x}_k^i)$ as the *whole likelihood* and $p(d_j|\mathbf{x}_k^i)$ as the *individual likelihood* in the remainder of this chapter.

In the remainder of this chapter, the bootstrap particle filter with the CV model as introduced in Section 2.8.4 and the likelihood as Equation (6.3) is referred to as the traditional BPF. BPF with multi-model as introduced in Section 2.8.4 is referred to as MM-BPF.

6.2.2 Problem Statement for Range-only Particle Filter

To achieve high tracking accuracy exclusively relying on power-based ranging, we will address the following three problems in BPF and MM-BPF particularly for passive indoor tracking. The details about these two filters (BPF and MM-BPF) have been introduced in Section 2.8.4. Recall that the state vector in both two filters are: $\mathbf{x} = [x, y, \hat{x}, \hat{y}]^T$ where (x, y) are the coordinates of the target and $(\hat{x}, \hat{y})$ are the velocity components on the x and y axes. In the dynamic equations of MM-BPF, a turning rate ω is introduced to model the a turn of the target.

- First, the velocity components $(\hat{x}, \hat{y})$ on the x and y axes are updated and treated independently in the CV model, which does not consider the relation between the two components. Actually, the two velocity components on the x and y axes can be related by the angle variation of the target's moving direction, especially when the target changes its moving direction. In MM-BPF, the velocity components $(\hat{x}, \hat{y})$ are related by the turning rate ω. However, MM-BPF requires accurate prediction of the regime $r(t)$ to correctly use the transition matrix **F**. This can be achieved based on accurate ranging, such as time-based ranging in an outdoor environment [88]. However, in indoor environments, the ranging accuracy based on power is still limited. Therefore, it is very challenging to accurately predict the regime $r(t)$ and the tracking accuracy is correspondingly low. Additionally, the transitional probability matrix with π_{mn} and the turning rate ω need to be predefined, which are normally difficult to know. Finally, in MM-BPF, in addition to the particles for the system states, another set of particles for estimating the regime $r(t)$ are required, which increases the computation efforts.

- Second, different from time-based positioning with specific signals, which benefits from high ranging accuracy, power-based positioning suffers from large ranging errors. Because of the large ranging errors, $\mathbf{z}_k$ is normally shifted from the real value, $\mathbf{z}'_k$, which makes the whole likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$ shifted from the real whole likelihood, $p(\mathbf{z}'_k|\mathbf{x}_k^i)$. Correspondingly, the associated weights are inaccurately updated, which results in inaccurate location estimation.

- Third, in BPF, only ranging information is considered in the likelihood estimation but velocity information is normally neglected due to lack of velocity information. However, due to the inaccurate ranging information and the lack of velocity observation information in the observation model to correct the predicted velocity from the system model, the predicted velocity in the state $\mathbf{x}_k$ can get very large, which is unusual for people moving in indoor environments and introduces large tracking errors.

6.3 An Enhanced Range-only Particle Filter

As introduced in Section 6.2, the accuracy of indoor tracking based on particle filter is deteriorated due to inaccurate likelihood and system models. In this section, we propose an enhanced particle filter to address the three problems in a traditional BPF and MM-BPF as mentioned in Section 6.2.2.

6.3.1 A Single Coordinated Turn Model

In MM-BPF, if we consider ω as positive for turning in anticlockwise direction and negative for turning in clockwise direction, then $\mathbf{F}_2 = \mathbf{F}_3$. Additionally, if ω approaches to 0, we get that

$$\lim_{\omega \to 0} \mathbf{F}_2 = \mathbf{F}_3 = \mathbf{F}_1 = \mathbf{F}_{\text{CV}}. \tag{6.4}$$

In this work, we propose to use a Single Coordinated Turn (SCT) model for range-only tracking based on power information, in which instead of estimating the moving models $\mathbf{F}(r(t))$ in MM-BPF, the turning rate ω is augmented in the system state, and the regime as in [88] is not required to switch between different models. The state vector is augmented with ω as

$$\mathbf{x}' = [x, y, \hat{x}, \hat{y}, \omega]^T. \tag{6.5}$$

Considering the relation between the two-dimensional moving speed vector $(\hat{x}, \hat{y})$ and ω, the SCT model and $\boldsymbol{\eta}$ in Equation (2.84) are defined as

$$\mathbf{F}_{\text{SCT}} = \begin{pmatrix} 1 & 0 & sin(\Delta T\omega)/\omega & (cos(\Delta T\omega)-1)/\omega & 0 \\ 0 & 1 & (1-cos(\Delta T\omega))/\omega & sin(\Delta T\omega)/\omega & 0 \\ 0 & 0 & cos(\Delta T\omega) & -sin(\Delta T\omega) & 0 \\ 0 & 0 & sin(\Delta T\omega) & cos(\Delta T\omega) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{6.6}$$

$$\boldsymbol{\eta}_{\text{SCT}} = \begin{pmatrix} \Delta T^2/2 & 0 & 0 \\ 0 & \Delta T^2/2 & 0 \\ \Delta T & 0 & 0 \\ 0 & \Delta T & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{6.7}$$

The noise vector $\mathbf{w}$ is a 3×1 i.i.d process noise vector $[n_x, n_y, n_\omega]$, in which n_x and n_y are the acceleration speeds on x and y axes, and n_ω is the noise of the turning rate. This augmented state method has been investigated for radar-based aircraft tracking [130] but indoor tracking faces different challenges, such as multipath and NLOS effects. To our knowledge, we are the first to investigate this method in indoor tracking with radio signals.

First, by introducing ω, the particle filter more smoothly tracks the targets compared to CV model, especially when the target suddenly changes its moving direction. Second, with the SCT model, we do not need to estimate the regime $r(t)$, which is challenging to be accurately estimated and correspondingly deteriorates the tracking accuracy. Third, with the SCT model, only particles for the state $\mathbf{x}'$ are used and therefore the computation efforts are expected to be lower than MM-BPF, which requires another set of particles to estimate the regime $r(t)$. In the remainder of this chapter, BPF with the single coordinated Turn (SCT) model is referred to as T-BPF.

6.3.2 Weighted Likelihood based on Ranging Information

As observed in Chapter 5, although the ranging accuracy has been improved by using some enhanced ranging methods with fine-grained power (channel information), there are still remaining ranging errors with a median error of $1.1m$ and maximum error of $5.9m$. These remaining ranging errors result in inaccurate tracking by adopting the traditional BPF. Therefore, we propose a modified BPF, whose performance is robust to the remaining ranging errors.

As mentioned in Section 6.2, range estimation is often biased and correspondingly the individual likelihoods $p(d_j|\mathbf{x}_k^i)$ from different ANs are often shifted from the real individual likelihoods $p(d'_j|\mathbf{x}_k^i)$, where d'_j is the ground truth propagation distance. Furthermore, depending on the locations of ANs, the ranges estimated by different ANs normally face different ranging errors. Especially in a complex indoor environment with mixed LOS and NLOS conditions, LOS and NLOS ranging are often substantially different. However, a traditional BPF just simply treats all the individual likelihoods from different ANs equally as in Equation (6.3). This oversimplification introduces large estimation errors, because in a multiplication form of the individual likelihoods as in Equation (6.3), the inaccurate individual likelihoods $p(d_j|\mathbf{x}_k^i)$ from certain ANs with large ranging errors will significantly influence the accuracy of the whole likelihood estimation $p(\mathbf{z}_k|\mathbf{x}_k^i)$.

Therefore, to mitigate the influence of large ranging errors on the estimation of the whole likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$, we propose to adopt a weighting technique on the whole likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$ estimation by suppressing the emphasis on the individual likelihoods $p(d_j|\mathbf{x}_k^i)$ with larger ranging errors and magnifying the contributions of the individual likelihoods with smaller ranging errors. To achieve this, we provide a Weighted-likelihood BPF (W-BPF) with exponential weights on each individual likelihood from different ANs as

$$p(\mathbf{z}_k|\mathbf{x}_k^i) = \Pi_{j=1}^{N} p(d_j|\mathbf{x}_k^i)^{m_j}, \tag{6.8}$$

where m_j is the *exponential weight* for the individual likelihood of the jth AN. To reduce the contribution of the individual likelihoods with large ranging errors, a direct way is to set weights m_j to indicate the error of each range. However, we can not measure the real ranging errors in practice, because it requires the ground truth location of the target.

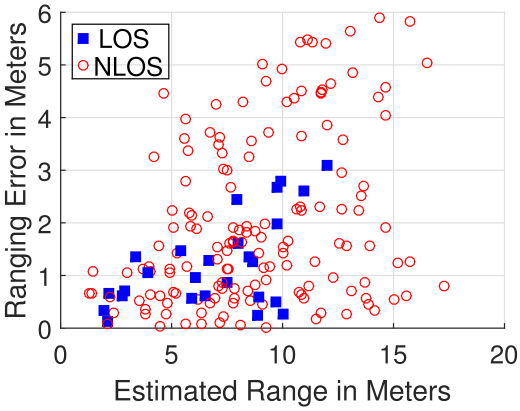


Figure 6.1: Ranging Error vs. Estimated Range

Therefore, we need to find another way to set a proper value for each exponential weight. Figure 6.1 indicates the relation between the ranging outputs and their corresponding ranging errors based on a set of preliminary measurements in our institute building, which provides a complex indoor environment under mixed LOS and NLOS conditions. Note that the ranging method is based on the channel information and NLR model. In general, we can find that the range errors increase with the estimated range values. Therefore, instead of relying on the ranging errors, we use the estimated ranging outputs to infer their corresponding errors and set the exponential weights to be inversely proportional to the estimated range outputs as

$$m_j = \frac{1/d_j}{\sum_{n=1}^{N} 1/d_n}, \tag{6.9}$$

which are normalized by $\sum_{j=1}^{N} m_j = 1$. With the proposed W-BPF method, we expect to mitigate the influence of ranging errors, especially for NLOS propagation, whose ranging errors are normally larger than for LOS conditions.

6.3.3 Moving Velocity Limitation on Likelihood

Since the state of the system, $\mathbf{x}_k$, includes the moving velocity of the target, we propose to further introduce a velocity related parameter γ_k^i to the whole likelihood estimation. The whole likelihood as in Equation (6.8) is defined as

$$p(\mathbf{z}_k|\mathbf{x}_k^i) = \gamma_k^i \cdot \Pi_{j=1}^{N} p(d_j|\mathbf{x}_k^i)^{m_j}. \tag{6.10}$$

As mentioned before, in an active positioning system, the velocity related parameter γ_k^i can be determined by the output of inertial sensors. For example, accelerometer sensors are used to estimate the absolute value of moving velocity. In passive positioning systems, this information is unavailable to the tracking system and hence the likelihood of velocity is typically ignored. Furthermore, because of inaccurate ranging, the location estimation between two sequential sampling intervals can be far away from each other, which results in large estimated moving velocity. However, this fast moving velocity is typically impossible in indoor environments, e.g., offices and shopping malls.

Therefore, in our work, instead of estimating the moving speed by some inertial sensors, we consider the limitation on the moving speed of people in an indoor environment, where walking is normally considered as the usual case. Some studies have been done to investigate the walking speed of people. As reported in [30], the maximum walking speed is limited to around $2.5m/s$. Therefore, we configure the velocity related parameter γ_k^i as,

$$\begin{cases} \gamma_k^i = 1, & 0 < |v_k^i| < 3m/s; \\ \gamma_k^i = 4 - |v_k^i|, & 3m/s < |v_k^i| < 4m/s; \\ \gamma_k^i = 0, & 4m/s < |v_k^i|, \end{cases} \tag{6.11}$$

$|v_k^i| = \sqrt{\hat{x}_k^{i2} + \hat{y}_k^{i2}}$ is the absolute value of the estimated velocity in each particle, where $(\hat{x}_k^i, \hat{y}_k^i)$ are the velocity components of the ith particle on x and y axes. In an office environment, people may sit at their working place or walk between offices. Therefore, we set the velocity related parameter γ_k^i as 1 when the moving velocity is smaller than $3m/s$, which is $0.5m/s$ larger than the maximum walking

Figure 6.2: Ranging Methods

speed in [30]. If the velocity is larger than $3m/s$ but smaller than $4m/s$, γ_k^i linearly decreases from 1 to 0. If the velocity is larger than $4m/s$ that does not frequently happen in an office environment, γ_k^i is set to 0.

Based on this velocity related parameter γ_k^i, the particles with uncommon moving velocity are filtered out and hence the estimated moving trace is smoothed. BPF only considering the velocity limited on the likelihood is referred to as V-BPF in the remainder of the chapter. BPF equipped with the SCT model (Equation (6.6)) and adopting the modified likelihood (Equation (6.10) including γ_k^i and exponential weights) is referred to as WVT-BPF.

6.4 Ranging

More accurate estimation of ranges is a prerequisite to improve the radio-based tracking accuracy. To achieve high ranging accuracy, we adopt the similar ranging method as introduced in Section 5.2, which uses CIR to extract the power from the direct path. Figure 6.2 shows the procedure of this ranging method, which comprises three steps. First, after converting CSI to CIR, RSS is obtained by extracting the strongest power in CIR. Second, RSS is smoothed by a S-G filter to further mitigate the influence of multipath propagation. Finally, the NLR (Non-Linear Regression) model is adopted to calculate the range information.

In contrast to locating a stationary target, the mobile target faces different multipath effects in different locations along his moving path, which results in large variation in the measured power. Typically, we can adopt a smoothing filter to smooth the measured power and mitigate the multipath effect. In this work, we propose to adopt a S-G filter to smooth the measured power. The S-G filter applies a moving window smoothing technique based on least squares polynomial fitting [32], which has the advantage of preserving the original shape and features of the signal, e.g., the trend of RSS changes in the moving window. We take the group of $2M + 1$ RSS samples centred at n, which is moving from 0 to the end of the samples. The RSS values are estimated as a polynomial with the coefficients $[a_0, a_1, ..., a_{N_p}]$,

$$RSS'_{\text{SG}}(n) = \sum_{i=0}^{N_p} a_i n^i. \tag{6.12}$$

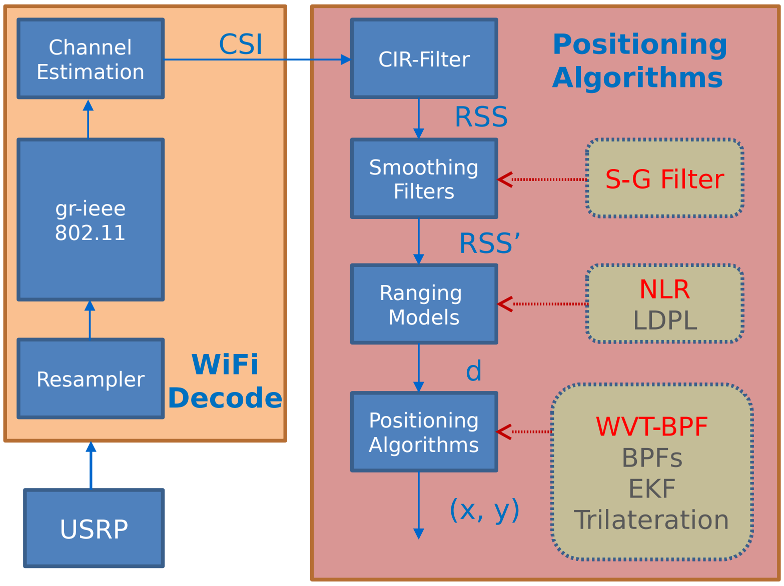


Figure 6.3: Testbed Structure

To obtain the coefficients, we minimize the mean-squared approximation error as

$$\underset{[a_0,a_1,\ldots,a_{N_p}]}{\operatorname{argmin}} \sum_{j=n-M}^{n+M} \left(\sum_{i=0}^{N_p} a_i j^i - RSS(j)\right)^2, \tag{6.13}$$

where N_p is the order of polynomial. $RSS'_{\mathrm{SG}}(n)$ is a smoothed version of the raw RSS values by the S-G filter.

6.5 System Implementation

We have extended our passive positioning system in Chapter 5 to support tracking the mobile WiFi target device by using our proposed particle filter. Figure 6.3 indicates the structure of this system. Basically, the system is divided into three main components: receiving hardware for WiFi signals, WiFi packet decoding, and positioning algorithms. The first two components, receiving hardware for WiFi signals and WiFi packets decoding, are the same as the passive positioning system in Chapter 5.

In this section, we mainly introduce the implementation of positioning algorithms in this passive tracking system for WiFi signals. As shown in Figure 6.3, positioning algorithms are designed in a central server, which runs MATLAB to analyze the moving path of the WiFi target. The positioning related algorithms are designed in four steps. First, CSI needs to be converted to CIR in time domain by IFFT, and the power (RSS) from the direct path is estimated based on Equation (5.2). Second, the RSS values are smoothed by the S-G filter, in which the window size is set to 5 and the order of the polynomial is 3. Third, the outputs of the

S-G filter are fed into the non-linear regression (NLR) model to calculate the range information from different ANs. Finally, the range information is the input to the WVT-BPF algorithm (Algorithm 1) to track the target.

Algorithm 1: WVT-BPF

```
1  Initialize filter
   (I) Initial particles: $\mathbf{x}_0^i = q(\mathbf{x}_0), i = 1, \dots, N_s$;
   (II) Initial weights: $w_0^i = \frac{1}{N_s}$;
2  Update particles: $\mathbf{x}_k^i = \mathbf{F}_{\text{SCT}} \cdot \mathbf{x}_{k-1}^i + \boldsymbol{\eta}_{\text{SCT}}\mathbf{w}$;
3  Calculate exponential weights: $m_j = \frac{1/d_j}{\sum_{n=1}^{N} 1/d_n}$;
4  Calculate the individual likelihood:
       $p(d_j|\mathbf{x}_k^i) = \frac{1}{\sigma_j\sqrt{2\pi}} e^{-\frac{[d_j-\sqrt{(x^i-x_j)^2+(y^i-y_j)^2}]^2}{2\sigma_j^2}}$;
5  Update unnormalized weights:
       $\hat{w}_k^i = \gamma_k^i \cdot \Pi_{j=1}^{N} p(d_j|\mathbf{x}_k^i)^{m_j}$;
6  Normalize weights: $w_k^i = \hat{w}_k^i / \sum_{n=1}^{N_s} \hat{w}_n^i$;
7  Calculate $N_{\text{eff}}$: $N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N_s}(w_k^i)^2}$;
8  if $N_{eff} < 0.5 * N_s$ then
9      Resample particles based on systematic resampling method;
10 Compute the estimated state: $\mathbf{x}_k = \sum_{i=1}^{N_s} w_k^i \mathbf{x}_k^i$;
11 Go back to step 2 for the next iteration.
```

Dynamic Equation: To passively track WiFi users, the proposed SCT model in Section 6.3.1 is adopted in WVT-BPF. Recall that the state vector in WVT-BPF includes the Cartesian coordinates of the target (x, y), the two-dimensional moving speed vector $(\hat{x}, \hat{y})$, and the turning rate of ω, i.e., $\mathbf{x}' = [x, y, \hat{x}, \hat{y}, \omega]^T$. For each iteration, the particles are updated based on $\mathbf{x}_k^i = \mathbf{F}_{\text{SCT}} \cdot \mathbf{x}_{k-1}^i + \boldsymbol{\eta}_{\text{SCT}}\mathbf{w}$.

Observation Equation: The measurement vector includes ranging information from different ANs as $\mathbf{z}_k = [d_1, d_2, \cdots, d_N]$. For each AN, the observation function is defined as:

$$d_j = \sqrt{(x - x_j)^2 + (y - y_j)^2} + u_j, \tag{6.14}$$

where (x_j, y_j) are the coordinates of the jth AN and u_j is the Gaussian noise of the jth AN with a variance of σ_j.

Weight Update and Location Estimation: Based on Equation (6.14), the individual likelihood for the jth AN is written as

$$p(d_j|\mathbf{x}_k^i) = \frac{1}{\sigma_j\sqrt{2\pi}} e^{-\frac{[d_j - \sqrt{(x^i - x_j)^2 + (y^i - y_j)^2}]^2}{2\sigma_j^2}}, \tag{6.15}$$

and the whole likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$ is calculated based on Equation (6.10), which considers the exponential weights m_j and the velocity related parameter γ_k^i. Finally, the associated weights are updated based on Equation (6.2) and the location of the target is estimated by calculating the weighted average of the particles as

$$\mathbf{x}_k = \sum_{i=1}^{N_s} w_k^i \mathbf{x}_k^i. \tag{6.16}$$

Resampling: With this weighted multiplication likelihood and speed limitation, the particle filter is prone to the sample degeneracy problem, which results in serious performance degradation. To deal with the sample degeneracy problem, resampling is typically adopted [63]. A suitable measure of degeneracy is the effective sample size $N_{\text{eff}} = 1/\sum_{i=1}^{N_s}(w_k^i)^2$. As soon as N_{eff} is smaller than $0.5N_s$, the degeneracy is considered to be serious and a suitable resampling method should be adopted. In our work, a systematic resampling method [63] is adopted in our work, because of its high accuracy and efficient implementation.

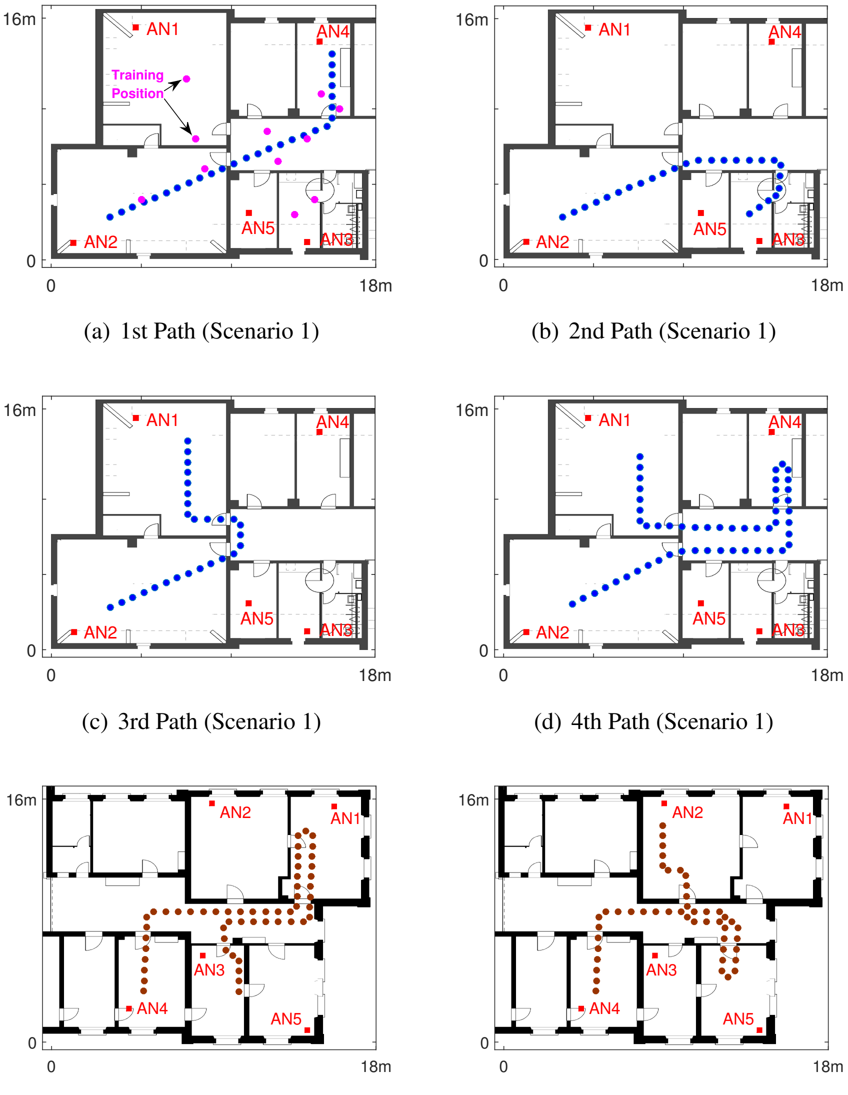
Algorithm 1 summarizes the procedure of WVT-BPF. Additionally, some commonly used positioning algorithms are also implemented in our system: a traditional BPF, a MM-BPF, extended Kalman filter (EKF), trilateration algorithms including ML, LLS and WC-CWLS, which is proposed in Chapter 5. Note that the traditional BPF and EKF [72] adopt the CV model.

6.6 Performance Evaluation

To evaluate the tracking accuracy of our proposed algorithms, we have conducted a set of comprehensive measurements in complex indoor environments.

6.6.1 Measurement Setup

Our proposed tracking system has been deployed in two scenarios including the third floor (referred to as scenario 1) and the second floor (referred to as scenario 2) in the INF building at the University of Bern. Five USRP receivers have been deployed as ANs to monitor the packets from a laptop as shown in Figure 6.4. A central server equipped with a 4-core i5 CPU (3.3GHz) is adopted to collect data from the five ANs and offline runs positioning algorithms for accuracy evaluation. In our measurements, the positioning target is a Thinkpad T430 laptop with an Intel N6300AGN wireless card. The laptop is configured to continuously refresh a website to generate enough data traffic.



(a) 1st Path (Scenario 1) (b) 2nd Path (Scenario 1)

(c) 3rd Path (Scenario 1) (d) 4th Path (Scenario 1)

(e) 5th Path (Scenario 2) (f) 5th Path (Scenario 2)

Figure 6.4: Tracking in Different Paths

	AN1	AN2	AN3	AN4	AN5
α	7.448	10.326	5.514	5.878	4.466
β	-0.03088	-0.0507	-0.04365	-0.05519	-0.05269

Table 6.1: (α, β) in NLR

We first conducted our experiments in scenario 1. Note that the evaluation results in Sections from 6.6.2 to 6.6.5 are obtained in scenario 1. In this scenario, 11 training positions (Figure 6.4(a)) that spread over the whole area of interest are selected to acquire (α, β) in the NLR model as shown in Table 6.1. The tracking experiments along four different moving paths (Figures 6.4(a)-6.4(d)) have been conducted to analyze the performance of the system for a mobile target (laptop), which is held by a moving person with a speed of $0.88m/s$. Along the moving traces, positioning algorithms run every second to estimate the location of the moving target, during which 10 packets in average are received in each AN and aggregated by averaging to report one RSS. The tracking accuracy is finally evaluated at 132 points along the four moving paths (blue circle points in Figures 6.4(a)-6.4(d)). To effectively measure the ground truth locations, the moving paths are predefined, in which the coordinates of all the turning points are measured. Then, the ground truth coordinates of the other locations along the paths are obtained by interpolation. We also need to know the time when the moving person passes each ground truth location to aggregate RSS of the received packets in the corresponding time interval. To achieve that, we firstly record the starting and end time of the movement and secondly keep the moving speed constant for the whole moving path. As mentioned in Section 5.3.2, all the received packets are timestamped and all the ANs are synchronized. Then, we map the received packets to each ground truth location based on the time when the person passes. To keep the constant moving speed, first, we divide the moving paths into individual constant segments with markers, whose length is the same as one step length. The moving person follows the markers to keep the constant step length. Then, the moving person follows tick sounds with a constant frequency from a sound tuner to keep constant step frequency.

Additionally, to test our system in different scenarios, we conducted another set of experiments in scenario 2 (Figures 6.4(e)-6.4(f)), with a different layout from scenario 1. The results are presented in Section 6.6.6.

6.6.2 Ranging Errors

Similar as the evaluation in Section 5.4 for positioning of a stationary target, we evaluate the ranging accuracy for a mobile target. We calculate the ranging errors to each AN for the 132 test positions along four moving paths in scenario 1 based on the NLR and LDPL models. Figure 6.5 indicates CDF of the ranging errors for both models. 90% of ranging errors with NLR are smaller than $3m$, which gets improved by 40% compared to LDPL (red dashed curve), in which the exponential

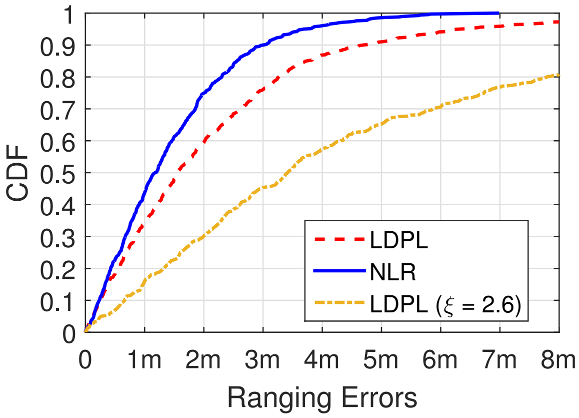


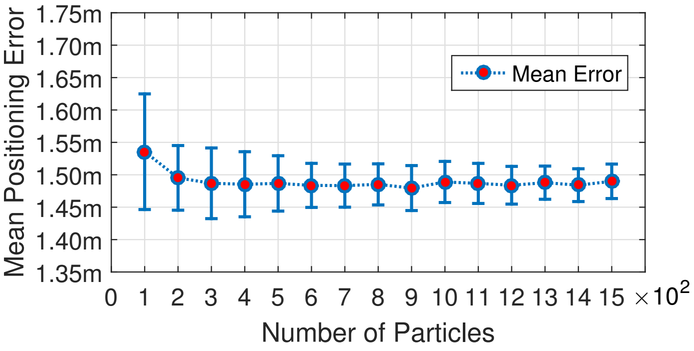
Figure 6.5: Ranging Errors

factor is also calculated for different ANs based on the 11 training positions. Additionally, both NLR and LDPL trained for different ANs significantly outperform a LDPL with a predefined value of 2.6, which is commonly used in indoor environments. Therefore, we get the same findings as in Section 5.4 that with small training effort (only 11 training positions) the NLR model achieves high ranging accuracy, which is a prerequisite for accurate positioning, and significantly outperforms the LDPL model for the ranging stage.

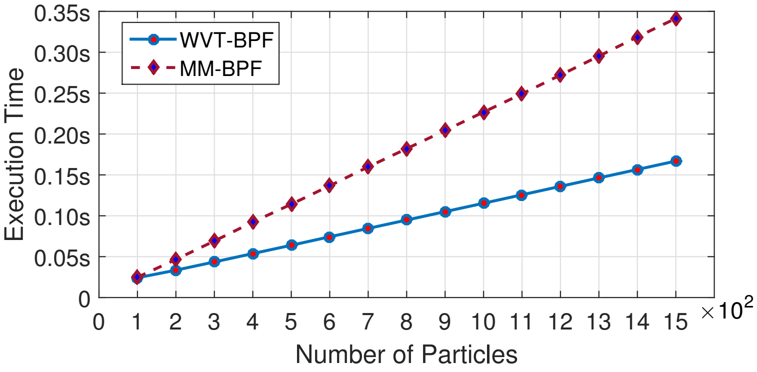
6.6.3 Design Parameters

Number of Particles

The number of particles is a critical influencing factor on the performance of particle filters. Theoretically, more particles improve the tracking accuracy but increase the computation efforts. To investigate the performance of our proposed WVT-BPF with different numbers of particles, we adapt the numbers of particles from 100 to 1500 at steps of 100. For each number of particles, we run WVT-BPF on the four paths 100 times and at each time we calculate the mean value of the positioning errors over the 132 positions (along four paths). Figure 6.6(a) indicates the mean values and standard deviations of positioning errors for each number of particles. In general, the mean and standard deviation of errors with WVT-BPF get smaller with the number of particles. However, the improvement gets very marginal when the particle numbers are larger than 1000. Figure 6.6(b) indicates the execution time of the particle filters in the central server. The execution time linearly increases with larger numbers of particles. In our work, in order to achieve high tracking accuracy and limit the computation effort, we set the particle number to 1000.



(a) Positioning Errors



(b) Execution Time

Figure 6.6: Performance with Different Numbers of Particles

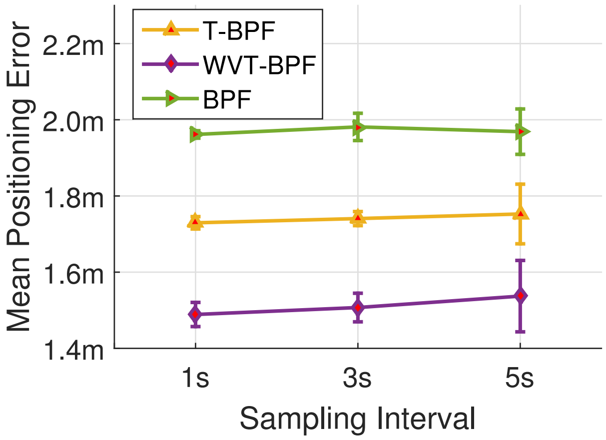


Figure 6.7: Different Sampling Intervals

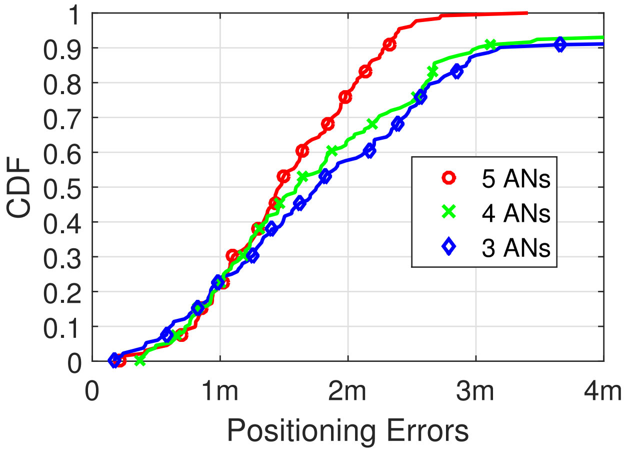


Figure 6.8: Different Numbers of ANs

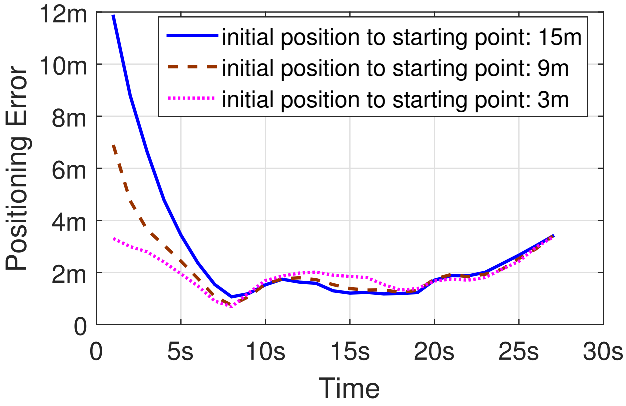


Figure 6.9: Different Initial Positions (Path 1)

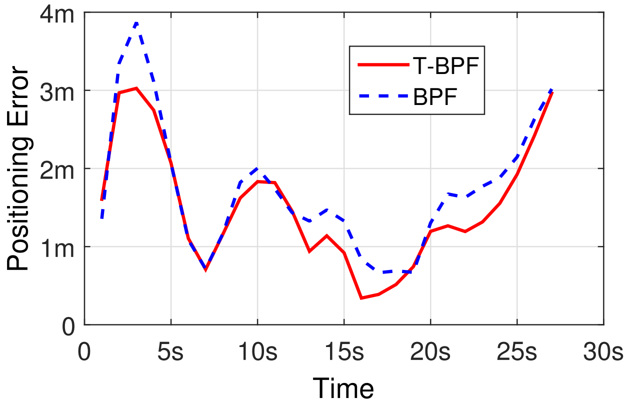


Figure 6.10: Positioning Errors (Path 1)

Sampling Interval

Sampling interval is another influencing factor on the performance of particle filters. Figure 6.7 indicates the mean and standard deviations of WVT-BPF with different sampling intervals. Generally, with larger sampling intervals, the standard deviations obviously increase, which means the performance of tracking is less stable. In the following evaluations, we set the sampling interval to 1s.

Number of ANs

The accuracy of range-based positioning algorithms depends a lot on the number of ANs. We run the WVT-BPF 100 times on each moving path and calculate the mean positioning error on each testing position. Then, CDF is calculated over the mean positioning errors at 132 testing positions along the 4 moving paths. This procedure is also adopted to calculate CDF of positioning errors for all the particle filters in the remainder of this thesis. Figure 6.8 indicates CDF of positioning errors with different numbers of ANs. It is clearly shown that the tracking accuracy gets improved if the number of ANs increases. Therefore, in the following analysis, we use all the five ANs in our deployment.

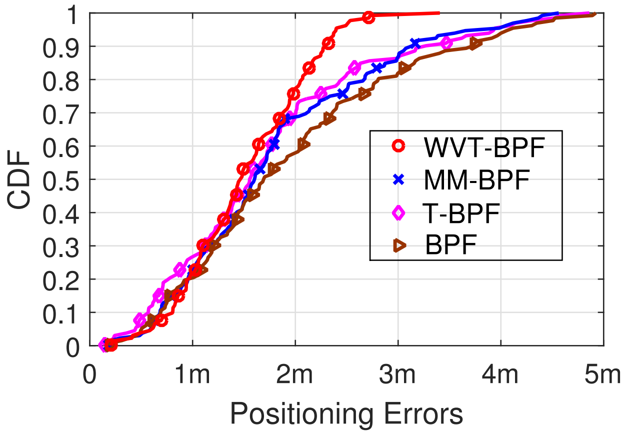
Initial Position

For tracking with Bayesian estimation, the initial position influences the tracking accuracy and convergence time. We adjust the initial position with distances to the starting point at 3m, 9m and 15m in path 1. Figure 6.9 indicates the positioning errors over time. After around 7s (7 iterations), the accuracy has converged.

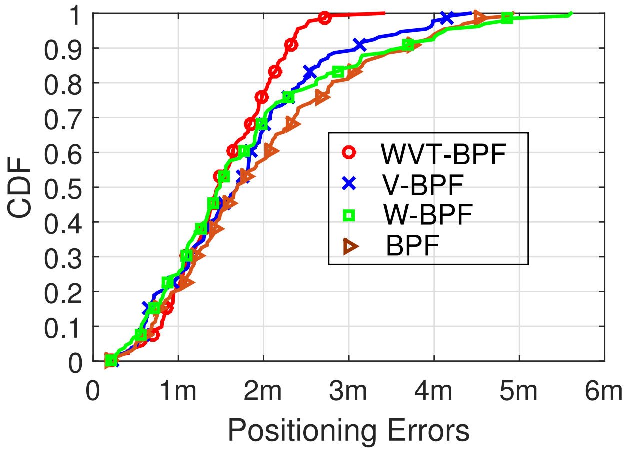
6.6.4 Performance Comparison with the NLR Model

Because ranging accuracy gets impressively improved under the NLR model, in this subsection, we analyze the performance of our proposed enhanced particle filter with the NLR model.

First, we compare the performance of T-BPF with the SCT model to MM-BPF and BPF with the CV model. As shown in Figure 6.11(a), both MM-BPF and T-BPF achieve higher positioning accuracy than BPF. Additionally, according to Figure 6.7, T-BPF achieves better positioning accuracy than BPF with different sampling intervals. We take path 1 as an example to better investigate the performance between T-BPF and BPF. Figure 6.10 indicates the positioning errors over time for T-BPF and BPF in path 1. Generally, after smoothing by introducing turning rate ω, the accuracy of T-BPF is higher than BPF, especially after 20s when there is a turn in the moving path. The performance of MM-BPF and T-BPF is quite similar because the regime in MM-BPF is still difficult to be correctly estimated under large ranging errors. However, as shown in Figure 6.6(b), considering the computation efforts, MM-BPF requires much longer execution time than T-BPF

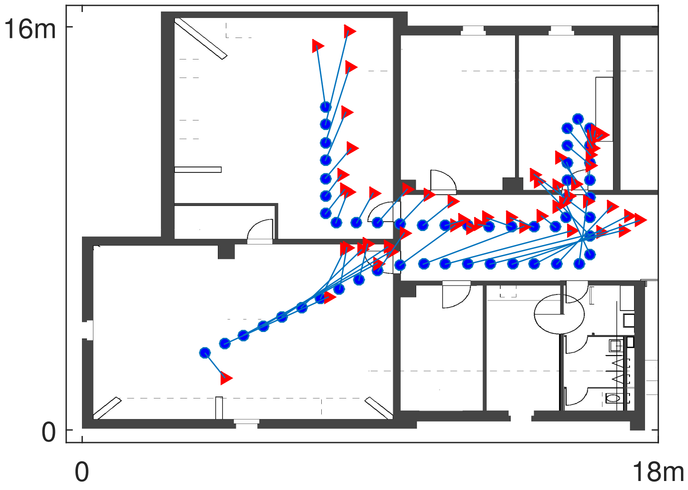


(a) Different Particle Filters 1

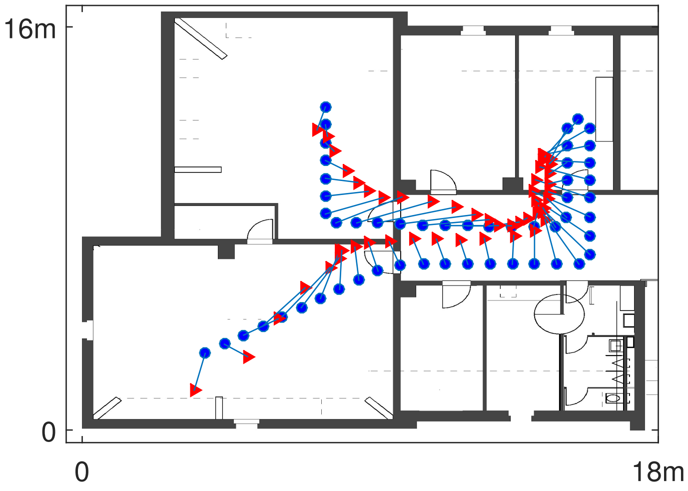


(b) Different Particle Filters 2

Figure 6.11: CDF of Positioning Errors for Different Particle Filters (Scenario 1)



(a) Estimated Path by BPF for Path 4



(b) Estimated Path by WVT-BPF for Path 4

Figure 6.12: Estimated Moving Path (Path 4) by BPF and WVT-BPF

		Path 1	Path 2	Path 3	Path 4
WVT-BPF	Mean	1.66m	1.68m	1.18m	1.45m
	STD	0.56m	0.39m	0.61m	0.58m
BPF	Mean	1.69m	1.77m	1.6m	2.01m
	STD	0.75m	0.69m	0.83m	1.35m

Table 6.2: Positioning Errors in Different Paths (STD: standard deviation)

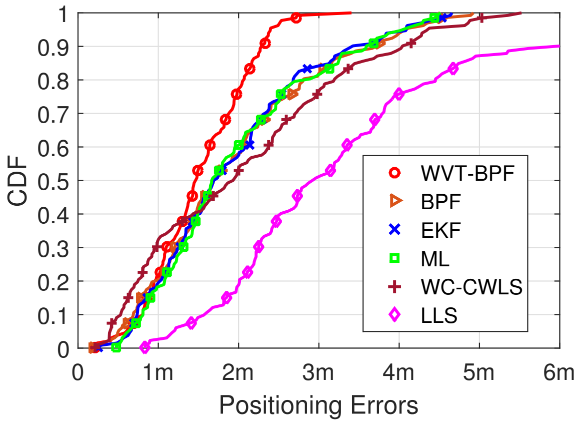
(almost the same execution time as WVT-BPF) because MM-BPF needs another set of particles for regime estimation.

Second, as indicated in Figure 6.11(b), after introducing the exponential weights to different individual likelihoods, W-BPF mitigates the influence of ranging errors and correspondingly improves the positioning accuracy compared to the traditional BPF. Then, by filtering out the unreasonable particles with very large moving velocity in V-BPF, the estimated moving traces are smoothed and the positioning accuracy gets improved compared to BPF. Our proposed WVT-BPF combines T-BPF, V-BPF and W-BPF, which improve the tracking accuracy in three different aspects, so the tracking accuracy gets significantly higher than BPF. For example, 90% of tracking errors with WVT-BPF are lower than $2.3m$, which is 38% more accurate than traditional BPF ($3.7m$). Table 6.2 lists the mean and standard deviation of positioning errors in the 4 paths for BPF and WVT-BPF. We find that in all the 4 paths, the mean tracking accuracy of WVT-BPF is higher than BPF and the performance is more stable according to the lower standard deviation. Figure 6.12 shows an example of the estimated paths respectively by a traditional BPF and WVT-BPF for the fourth moving path. It is obvious that our proposed WVT-BPF tracks the moving target with a much higher accuracy and the estimated moving path is more smooth.

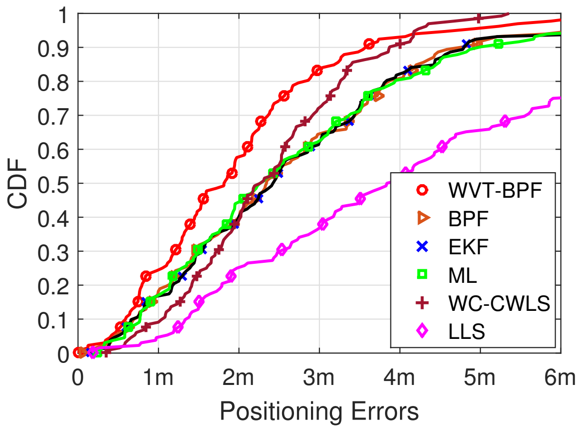
Third, we compare the tracking accuracy of WVT-BPF to some other positioning algorithms i.e. EKF and trilateration algorithms in Figure 6.13(a). Generally, our proposed WVT-BPF significantly outperforms the other positioning algorithms. BPF, EKF and ML-based trilateration achieve very similar performance, which are slightly better than WC-CWLS. The performance of LLS is the worst.

6.6.5 Positioning Accuracy with the LDPL Model

We further evaluate the performance of our proposed enhanced particle filter under large ranging errors (with the LDPL model) and compare it to the other algorithms. As shown in Figure 6.13(b), the performance of WVT-BPF with the LDPL model deteriorates by 61% for the 90% positioning accuracy (from $2.3m$ to $3.7m$) compared to the NLR model. However, WVT-BPF still significantly outperforms the other positioning algorithms. Similar to the evaluation results in Section 5.4, WC-CWLS, whose performance does not get significantly worse than with NLR model, is robust to ranging errors and outperforms EKF, BPF and ML-based trilateration. However, our proposed WVT-BPF still outperforms WC-CWLS. For example, the



(a) With NLR Model



(b) With LDPL Model

Figure 6.13: CDF of Positioning Errors for Different Positioning Algorithms (Scenario 1)

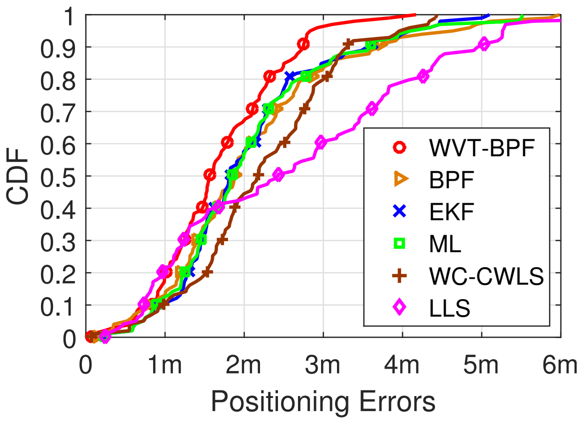


Figure 6.14: CDF of Positioning Errors in Scenario 2

median error of our proposed WVT-BPF achieves $1.9m$, which is $0.2m$ better than WC-CWLS. In addition, by introducing the enhanced mechanisms in the particle filter, our proposed WVT-BPF is much better than BPF in the case of large ranging errors. The 90% positioning accuracy is around $1.1m$ better than BPF $(4.8m)$. With the LDPL model, LLS is significantly worse than the other positioning algorithms because it is very sensitive to ranging errors.

6.6.6 Positioning Accuracy in Different Environments

To further investigate our proposed system in different environments, we conduct our experiments in scenario 2. Figure 6.14 indicates CDF of tracking errors for different positioning algorithms with the NLR model. We find that the evaluation results for different positioning algorithms are consistent with the evaluation in scenario 1. Our proposed WVT-BPF outperforms the other commonly used positioning algorithms.

6.7 Conclusions

In this chapter, we extended the passive positioning system for WiFi signals, which is introduced in Chapter 5, to support tracking mobile users with a high accuracy. In this system, an enhanced particle filter (WVT-BPF) exclusively relying on power-based ranging with low calibration effort is proposed to achieve high tracking accuracy. Our proposed WVT-BPF integrates three improvements including weighted likelihood, velocity limitation on likelihood and a single coordinated turn model. Each of the individual improvement improves the tracking accuracy compared to the traditional BPF. Additionally, T-BPF and WVT-BPF with the SCT model requires much less computation efforts than MM-BPF. By integrating all these improvements, our proposed WVT-BPF outperforms the traditional BPF, EKF, and trilateration algorithms. By combining WVT-BPF with the enhanced ranging methods proposed in Chapter 5, our system can passively track the WiFi target with high accuracy in complex indoor environments.

Chapter 7

Indoor Tracking by Fusing Inertial Sensor and Channel Information

7.1 Introduction

In Chapter 6, we have introduced our solutions for a passive indoor tracking system for WiFi users. Because of the lack of inertial sensor information, we can only introduce a speed limitation factor to prevent unusual moving speeds. In this chapter, we investigate an active indoor tracking systems to combine radio parameters, i.e., CSI from the received WiFi packets, and inertial sensor information.

With the development of smart phones, PDR (Pedestrian Dead Reckoning) systems can leverage inertial sensors, e.g., accelerometer, magnetometer, and gyroscope, to estimate the relative movement of the target by detecting steps, estimating stride length and heading orientation. By integrating the estimated relative locations at sequential time intervals, PDR systems can track the target. Because of integration, small positioning errors resulting from the noise in low cost IMUs (Inertial Measurement Unit) can be magnified [64].

In contrast to PDR, radio-based positioning relies on the measured radio parameters, e.g., power and time, to estimate the absolute locations of targets in a coordinate system instead of integrating the relative locations. As mentioned in Chapters 5 and 6, range-based positioning is less labour intensive than fingerprinting and achieves high positioning accuracy by adopting some enhanced ranging methods. Therefore, in this chapter, we continue on range-based positioning methods.

PDR and range-based methods are complementary because PDR can provide information about the relative movement between sequential time intervals, i.e., velocity, which is missing in range-based methods. Additionally, the absolute location information provided by range-based methods can also be used to mitigate the accumulative errors in PDR.

In this work, we investigate how to accurately track a WiFi target using an enhanced particle filter to fuse the velocity information estimated by inertial sensors and highly accurate range information by some enhanced ranging methods. Our main scientific contributions are summarized as follows.

- We propose an enhanced particle filter to fuse the CIR-based ranging and velocity information. The two observation parameters, i.e., ranges and velocity, are fused in the observation likelihood function. To achieve high ranging accuracy, some enhanced CIR-based ranging methods, which are proposed in Chapter 5, are adopted in our proposed particle filter. Additionally, we adopt the spatial diversity between different antennas to mitigate the multipath effect in the ranging step of this work. To mitigate the influence of the ranging errors, the aforementioned weighted likelihood method proposed in Section 6.3.2 is adopted. Furthermore, we propose an efficient method to estimate the velocity of the mobile target using the timestamped values from the accelerometer and compass sensors in a smart phone.

- We implement a network-based positioning system, which runs our proposed tracking algorithms in a central server. Compared to terminal-based positioning system, a network-based positioning system is able to run algorithms with high complexity, e.g., particle filters. In our system, all ANs are implemented on cheap commercial devices and able to collect inertial sensor and CSI information from the received WiFi packets.

- We evaluate our system in a complex indoor environment along three different moving paths. Our proposed tracking method can achieve $1.3m$ for mean accuracy and $2.2m$ for 90% accuracy, which is more accurate and stable than PDR and range-based positioning methods.

In the remainder of this chapter, our main contributions are introduced in Section 7.2, in which the proposed enhanced particle filter is described. The ranging and velocity estimation mechanisms are presented in Section 7.3. Section 7.4 presents the implementation of the proposed algorithms in a network-based indoor tracking system. Section 7.5 presents the evaluation results in a complex indoor environment. Finally, Section 7.6 concludes this chapter.

7.2 An Enhanced Particle Filter with Data Fusion and Weighted Likelihood (FW-PF)

As introduced in Section 7.1, tracking methods by using power-based ranging and PDR are complementary. Hence, we propose an enhanced particle filter to fuse velocity and range information to provide a tracking method with high accuracy and stability in this section.

In this work, a Constant Velocity (CV) model as introduced in Section 2.8.4 is used. Recall that the state vector is defined as

$$\mathbf{x} = [x, y, \hat{x}, \hat{y}]^T, \tag{7.1}$$

where (x, y) are the Cartesian coordinates of the target and $(\hat{x}, \hat{y})$ is a two-dimensional moving speed vector. Under the CV model, the dynamic equation is written as

$$\mathbf{x}_k = \mathbf{F}_{\text{CV}} \cdot \mathbf{x}_{k-1} + \boldsymbol{\eta}_{\text{CV}} \cdot \mathbf{w}, \tag{7.2}$$

where $\mathbf{F}_{\text{CV}}$ and $\boldsymbol{\eta}_{\text{CV}}$ are introduced in Section 2.8.4.

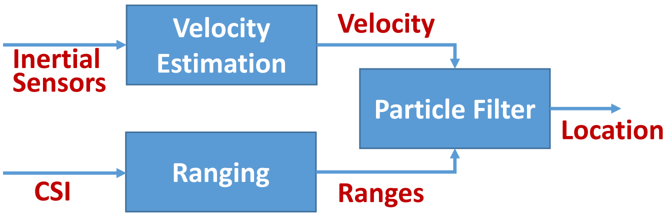


Figure 7.1: Data Fusion via a Particle Filter

7.2.1 Observation Model for Data Fusion

After updating particles based on dynamic equation as Equation (7.2), the associated weight w_k^i is updated from the weight at the previous moment w_{k-1}^i based on the likelihood of the observations conditioned on each particle $p(\mathbf{z}_k|\mathbf{x}_k^i)$ (Equation (6.2)). In this work, the observation vector obtained at each time interval contains an estimation of ranges to different ANs and velocity of the mobile target. Subsequently, the measurement vector is given as $\mathbf{z}_k = [\mathbf{d}_k, \mathbf{v}_k]$, where $\mathbf{d}_k$ includes ranges to N different ANs and $\mathbf{v}_k$ is the velocity information from the inertial sensors.

To fuse the range information $\mathbf{d}_k$ and velocity information $\mathbf{v}_k$, we can reasonably assume that the velocity information $\mathbf{v}_k$ is independent from ranges because the range information depends on the location of target but velocity does not. Hence, the likelihood $p(\mathbf{z}_k|\mathbf{x}_k^i)$ is written as

$$p(\mathbf{z}_k|\mathbf{x}_k^i) = p(\mathbf{d}_k|\mathbf{x}_k^i) \cdot p(\mathbf{v}_k|\mathbf{x}_k^i). \tag{7.3}$$

In order to distinguish different likelihoods, we refer to $p(\mathbf{z}_k|\mathbf{x}_k^i)$ as the overall likelihood, $p(\mathbf{d}_k|\mathbf{x}_k^i)$ as the ranging likelihood, and $p(\mathbf{v}_k|\mathbf{x}_k^i)$ as the velocity likelihood.

With the likelihood in Equation (7.3), the associated weight w_k^i is updated by considering both range and velocity observations. On one hand, the particles at the absolute positions (x^i, y^i), which have low probabilities to observe the measured ranges $\mathbf{d}_k$, are assigned small associated weights to suppress their contributions to the state estimation. On the other hand, the particles with velocities $(\hat{x}^i, \hat{y}^i)$, which have low probabilities to observe the measured velocity $\mathbf{v}_k$, are also assigned small associated weights, especially for some particles with unusual large moving speeds in indoor environments. This allows smoothing the estimated moving paths. Because ranges based on CIR achieve high accuracy according to our analysis in Chapters 5 and 6, range-only methods can achieve high positioning accuracy. Therefore, with this method, the positioning accuracy is more relying on the ranges estimated by CIR and the estimated velocity is only used to limit the weights of the particles, which is different from a typical method to update the particles based on the estimated movement state from the inertial sensor in the dynamic equation [64].

Velocity Likelihood

As we work on a two-dimensional tracking system, the measured velocity information $\mathbf{v}_k$ is a vector with two components $\hat{x}$ and $\hat{y}$, which are measured from inertial sensors. Assuming that these two components are independent from each other, the velocity likelihood $p(\mathbf{v}_k|\mathbf{x}_k^i)$ can be written as

$$p(\mathbf{v}_k|\mathbf{x}_k^i) = p(\hat{x}_k|\mathbf{x}_k^i) \cdot p(\hat{y}_k|\mathbf{x}_k^i). \tag{7.4}$$

Additionally, these two velocity components are independent from the coordinate components (x, y) in each particle. Hence we obtain that $p(\hat{x}_k|\mathbf{x}_k^i) = p(\hat{x}_k|\hat{x}_k^i)$ and $p(\hat{y}_k|\mathbf{y}_k^i) = p(\hat{y}_k|\hat{y}_k^i)$. The estimation of each velocity component is assumed to follow a Gaussian distribution. Equation (7.4) is rewritten as

$$\begin{aligned} p(\mathbf{v}_k|\mathbf{x}_k^i) &= p(\hat{x}_k|\hat{x}_k^i) \cdot p(\hat{y}_k|\hat{y}_k^i) \\ &= \frac{1}{\sigma_v\sqrt{2\pi}}\text{exp}[-\frac{(\hat{x}_k - \hat{x}_k^i)^2 + (\hat{y}_k - \hat{y}_k^i)^2}{2\sigma_v^2}], \end{aligned} \tag{7.5}$$

where σ_v is the variance of velocity estimation.

Ranging Likelihood

Besides velocity information, range information is another observation input. In this work, we use the weighted likelihood method, which is proposed in Section 6.3.2, because it can mitigate the influence of ranging errors. The ranging likelihood is written as

$$p(\mathbf{d}_k|\mathbf{x}_k^i) = \Pi_{j=1}^N p(d_j|\mathbf{x}_k^i)^{m_j}, \tag{7.6}$$

where d_j is the estimated range to the ith AN at the kth moment and $p(d_j|\mathbf{x}_k^i)$ is the individual likelihood. For the exponential weights, we still use the weights as in Section 6.3.2:

$$m_j = \frac{1/d_j}{\sum_{n=1}^N 1/d_n}. \tag{7.7}$$

Because the range information exclusively depends on the location of the target, the observation function for range is defined as:

$$d_j = \sqrt{(x - x_j)^2 + (y - y_j)^2} + u_j, \tag{7.8}$$

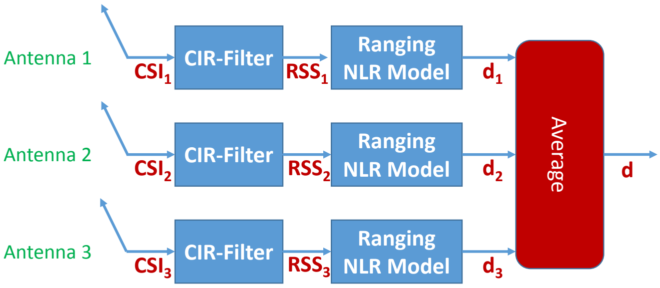


Figure 7.2: Ranging via CSI and MIMO

where (x_j, y_j) are the coordinates of the jth AN and u_j is a Gaussian noise with a variance of σ_j. Each individual likelihood is written as

$$p(d_j|\mathbf{x}_k^i) = \frac{1}{\sigma_j\sqrt{2\pi}} e^{-\frac{[d_j - \sqrt{(x^i - x_j)^2 + (y^i - y_j)^2}]^2}{2\sigma_j^2}} . \tag{7.9}$$

7.3 Range and Velocity Estimation

This section introduces how to estimate the two observation parameters (ranges and velocity) in our proposed particle filter.

7.3.1 Range Estimation using CIR

More accurate estimation of ranges is a prerequisite to improve the radio-based tracking accuracy. To achieve high ranging accuracy, we adopt the same method as in Chapter 5, which uses channel information to extract the power from the direct path. As shown in Figure 7.2, first, CSI in frequency domain is converted to CIR in time domain by IFFT and the power from the direct path is obtained by extracting the strongest power in CIR. Second, the proposed NLR (Non-Linear Regression) model in Chapter 5 is adopted to calculate the range information from the measured power.

Additionally, most recent WiFi standards (IEEE 802.11n/ac standards) support MIMO (Multiple Input and Multiple Output), which introduces spatial diversity. Multiple antennas separated by certain distances normally face different multipath effects. Therefore, we exploit multiple antennas to smooth and mitigate the multipath effects as shown in Figure 7.2. In our work, we estimate the range information based on the aforementioned procedures on each antenna and then calculate the average range from all the antennas in one AN as one input range information to the particle filter.

7.3.2 Velocity Estimation using Inertial Sensors

Velocity is another observation input in our proposed particle filter. In our work, the velocity of the mobile target is estimated by analyzing the timestamped values of

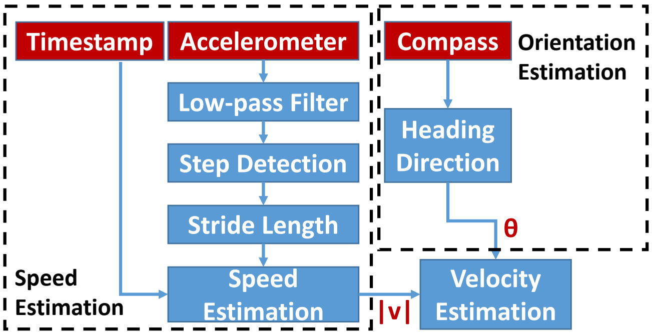


Figure 7.3: Velocity Estimation

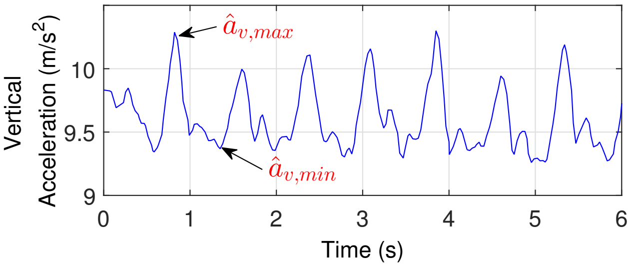


Figure 7.4: Vertical Acceleration

inertial measurement units in a smart phone. To estimate the two-dimensional velocity, which is a vector value with two components on x and y axes in a Cartesian coordinate system, the heading orientation θ and speed $|v|$, which is the absolute value of velocity, are estimated based on compass and accelerometer respectively.

Speed Estimation

As shown in Figure 7.3, first, the raw values from the accelerometer are smoothed through a low pass filter using Equation (7.10) to mitigate the influence of noise and dynamic pushes.

$$\hat{a}_{v,i} = (1-\beta)a_{v,i} + \beta(\hat{a}_{v,i-1}), \tag{7.10}$$

where $a_{v,i}$ is the raw vertical acceleration and β is a constant value ranging from 0 to 1 (0.9 in our work).

Second, during walking, every step generates one peak and dip in the measured vertical acceleration $\hat{a}_{v,i}$ as shown in Figure 7.4. Therefore, we detect the dips and peaks from $\hat{a}_{v,i}$ as steps.

Third, Equation (7.11) is used to estimate stride length [66].

$$l = K(\hat{a}_{v,max} - \hat{a}_{v,min})^{1/4}, \tag{7.11}$$

where l is stride length, $\hat{a}_{v,max}$ and $\hat{a}_{v,min}$ are the peak and dip values of $\hat{a}_v$ on each stride respectively, and K is a coefficient calibrated for individuals.

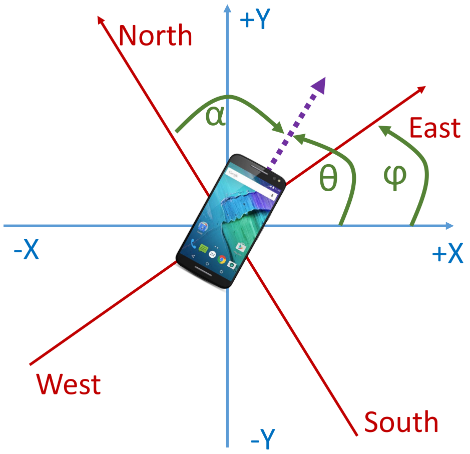


Figure 7.5: Angle Relations

Fourth, because all the accelerometer values are timestamped in the smart phone, we calculate the time interval for each stride ΔT and the speed is calculated as

$$|\mathbf{v}| = \frac{l}{\Delta T}. \tag{7.12}$$

Orientation Estimation

To estimate the heading orientation, we adopt the compass [99] in smart phones, which derives its data from the accelerometer and magnetometer. As shown in Figure 7.5, the compass reports a value called azimuth α, which is the clockwise angle from the north. α is an angle in the global coordinate system and we should map this angle to the local coordinate system. Normally, there is an angle rotation between the global coordinate system and the local coordination system. As shown in Figure 7.5, the counter-clockwise angle from +x in the local coordinate system to the east in the global coordinate system is φ. After obtaining α, we calibrate α to the counter-clockwise angle from +x in our local coordinate system as

$$\theta = (90^\circ - \alpha) + \varphi, \tag{7.13}$$

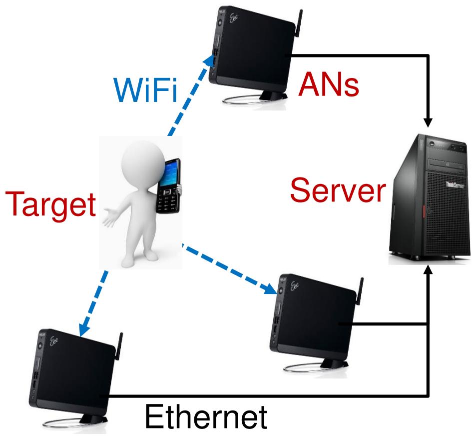
where $(90^\circ - \alpha)$ is to rotate the azimuth α to the counter-clockwise angle from the east.

Velocity Estimation

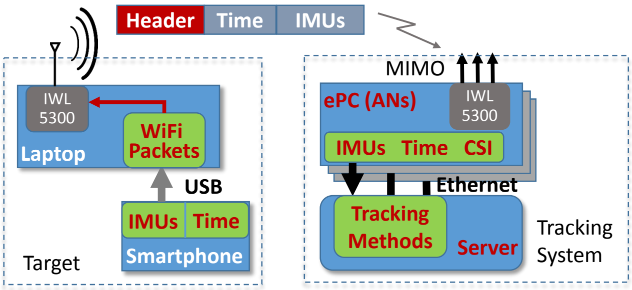
After estimating the speed and heading orientation of the mobile target, we get the velocity as

$$\mathbf{v} = [|\mathbf{v}|\cos(\theta), |\mathbf{v}|\sin(\theta)], \tag{7.14}$$

where $|\mathbf{v}|\cos(\theta)$ and $|\mathbf{v}|\sin(\theta)$ are the x and y components of moving velocity respectively.



(a) Overview of the System



(b) Implementation of the System

Figure 7.6: Network-based Tracking System

7.4 Implementation of Tracking Algorithms in a Network-based System

We have implemented a network-based indoor tracking system, in which our proposed tracking algorithms are running in a central server. Different from the passive systems, which have been introduced in Chapter 5 and 6, this system requires the target to actively participate in the positioning system by sending the information of inertial sensors and therefore it is an active positioning system. Figure 7.6(a) presents the overview of this system, which comprises three main components: target, ANs, and server. Figure 7.6(b) shows the implementation details of each component. The main idea behind this system is that by integrating the inertial sensor information (IMUs in Figure 7.6(b)) in the payload of WiFi packets broadcast from the target, the server can read these IMU information from the received packets, extract the CSI information from commercial WiFi cards (IWL5300) in ANs, and finally track the target with these two types of information.

7.4.1 Mobile Target

The mobile target needs to 1) inject the timestamped IMU information from the smart phone into the payload of the WiFi packets and 2) broadcast these packets using monitoring mode with an IEEE 802.11n High Throughput (HT) rate, which is required by IWL5300 at the receivers (ANs) to extract the CSI information [60].

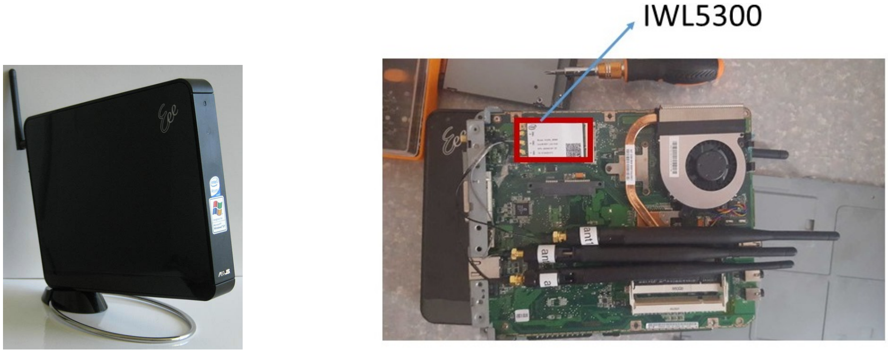
Because most of the WiFi chips in smart phones (including vendors like Apple, Samsung, Nokia, and HTC) do not support monitoring mode, it is impossible to extract CSI information at ANs directly from the received packets transmitted by a smart phone. Therefore, in our system, a smart phone has to transfer the timestamped IMU values to a laptop (via USB), which then transmits the WiFi packets using its on-board IWL5300 WiFi card and with hardcoded MAC address of $00:16:ea:12:34:56$ (required for the IWL5300 card [60]). In the smart phone, the sampling rate of the compass and accelerometer is 100Hz. As soon as the smart phone reads a pair of values from compass and accelerometer, it forwards these values together with their timestamps to the laptop over a USB cable by a Java application. The laptop prepares the WiFi packet, whose payload includes the values of compass, accelerometer and their timestamps, and then broadcasts it over the IWL5300 WiFi card using monitoring mode. The WiFi packet rate is also 100Hz.

7.4.2 Anchor Nodes

Anchor nodes are distributed over the area of interest to capture the packets from the target. To reduce the cost, we adopt ASUS EeeBox PCs (ePC) as ANs (Figure 7.7(a)). Each ePC only costs around 100$. First, we replace the original WiFi card in each ePC by an IWL5300 card (Figure 7.7(b)), which is configured in monitoring mode. Second, after receiving a WiFi packet, each ePC reads the timestamp and IMU information from the payload and extract CSI information. Because the IWL5300 card supports three antennas, we read CSI from all the three antennas. Finally, all these information from all ANs are forwarded to the central server over Ethernet by TCP sockets.

7.4.3 Server

A desktop PC equipped with a 4-core 3.30GHz i5 CPU is used as the server to collect the information from ANs and run offline tracking algorithms to analyze the moving trace of the target based on MATLAB. For the tracking algorithms, we first estimate the range and velocity information based on the algorithms introduced in subsections 7.3.1 and 7.3.2 respectively. Since we can get CSI from three antennas in one AN, we calculate the mean value of the estimated ranges from these three antennas as the input range to the particle filter from this AN. Finally, the range and velocity information is fused in our proposed particle filter (FW-PF) to track the target. Algorithm 2 indicates the procedures of FW-PF. Besides our proposed FW-PF, we implement the traditional range-only particle filter (R-PF is the same

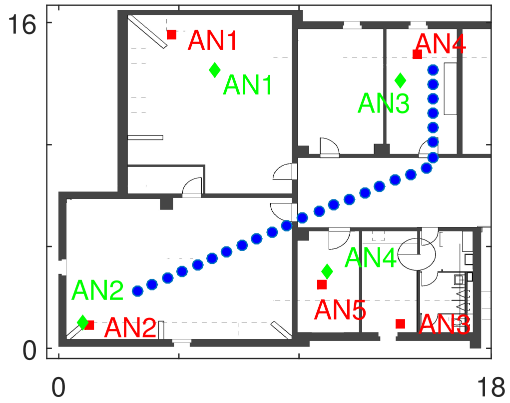


(a) ASUS ePC (b) Replace WiFi Card by IWL5300

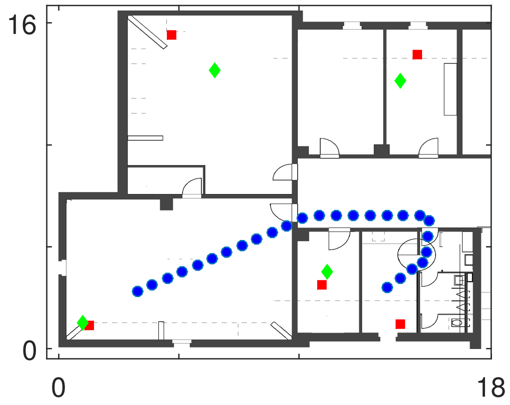
Figure 7.7: ePC as Anchor Nodes

Algorithm 2: FW-PF

```
1  Initialize filter
   (I) Initial particles: $\mathbf{x}_0^i = q(\mathbf{x}_0), i = 1, \ldots, N_s$;
   (II) Initial weights: $w_0^i = \frac{1}{N_s}$;
2  Update particles: $\mathbf{x}_k^i = \mathbf{F}_{\text{CV}} \cdot \mathbf{x}_{k-1}^i + \boldsymbol{\eta}_{\text{CV}}\mathbf{w}$;
3  Calculate exponential weights: $m_j = \frac{1/d_j}{\sum_{n=1}^{N} 1/d_n}$;
4  Calculate the individual likelihood:
       $$p(d_j|\mathbf{x}_k^i) = \frac{1}{\sigma_j\sqrt{2\pi}} e^{-\frac{[d_j - \sqrt{(x^i - x_j)^2 + (y^i - y_j)^2}]^2}{2\sigma_j^2}};$$
5  Calculate the velocity likelihood:
       $$p(\mathbf{v}_k|\mathbf{x}_k^i) = \frac{1}{\sigma_v\sqrt{2\pi}} \text{exp}[-\frac{(\hat{x}_k - \hat{x}_k^i)^2 + (\hat{y}_k - \hat{y}_k^i)^2}{2\sigma_v^2}].$$
6  Update unnormalized weights:
       $$\hat{w}_k^i = p(\mathbf{v}_k|\mathbf{x}_k^i) \cdot \Pi_{j=1}^{N} p(d_j|\mathbf{x}_k^i)^{m_j};$$
7  Normalize weights: $w_k^i = \hat{w}_k^i / \sum_{n=1}^{N_s} \hat{w}_n^i$;
8  Calculate $N_{\text{eff}}$: $N_{\text{eff}} = \frac{1}{\sum_{i=1}^{N_s} (w_k^i)^2}$;
9  if $N_{\mathit{eff}} < 0.5N_s$ then
10     Resample particles based on systematic resampling method;
11 Compute the estimated state: $\mathbf{x}_k = \sum_{i=1}^{N_s} w_k^i \mathbf{x}_k^i$;
12 Go back to step 2 for the next iteration.
```



(a) The First Path



(b) The Second Path

(c) The Third Path

Figure 7.8: Tracking in Different Paths
(Diamond Points: ANs in Scenario 1; Rectangular Points: ANs in Scenario 2; Circle Points: Ground Truth Positions)

as the traditional BPF in Chapter 6) and PDR algorithms in the central server for comparison.

7.5 Performance Evaluation

To evaluate the tracking accuracy of our proposed system, we have conducted a set of comprehensive measurements in a complex indoor environment.

7.5.1 Measurement Setup

We have evaluated our system in two scenarios on the third floor of the INF building at University of Bern. Four ANs were deployed in the first scenario (green and diamond points) and five ANs in the second scenario (red and rectangular points) as shown in Figure 7.8. In each scenario, the target (laptop and smartphone) was held by a person moving along three different paths (Figure 7.8) and experiments along each path were repeated five times. The moving speed was around $0.9m/s$ for scenario 1 and $0.6m/s$ for scenario 2. Along these moving paths, the point accuracy, which is the error from the estimated position to the ground truth position, is calculated every second. Three algorithms are evaluated along these moving paths, i.e., PDR (Pedestrian Dead Reckoning), R-PF (Ranging-only Particle Filter), FW-PF (our proposed Particle Filter with data Fusion and Weighted likelihood).

7.5.2 Experiment Results

Figure 7.9 shows CDF of positioning errors for the three algorithms in scenario 1 (4 ANs) and scenario 2 (5 ANs). Since the performance of PDR is not related to the number of ANs, the CDF curve of PDR positioning errors summarizes all the experiments in both scenarios. Table 7.1 summarizes the mean error, standard deviation and 90% accuracy. Based on these results, we find the following observations.

Table 7.1: Mean Errors and Standard Deviation

Tracking Methods	Mean Error	Standard Deviation	90% Accuracy
FW-PF (5ANs)	1.3m	0.7m	2.2m
FW-PF (4ANs)	1.6m	0.9m	2.8m
R-PF (5ANs)	1.7m	1.5m	3m
R-PF (4ANs)	1.8m	1.0m	3m
PDR	1.6m	2.5m	4m

First, our proposed FW-PF achieves higher accuracy and more stable performance compared to PDR. It is commonly known that PDR is prone to accumulated errors because it estimates the location of the target by integrating the relative locations of the previous moments. Because of the accumulative errors, it is very accurate at the beginning of the moving paths by using PDR but the positioning

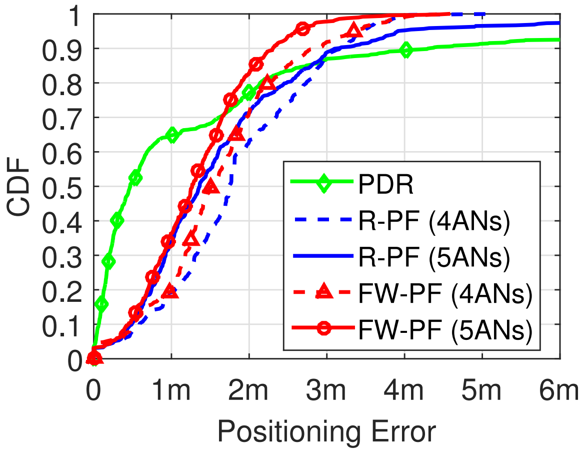


Figure 7.9: CDF of Positioning Errors

error increases along the moving paths. Therefore, for 50% accuracy, PDR can achieve around $0.5m$ but the accuracy severely deteriorates to around $4m$ considering 90% accuracy. In our proposed FW-PF, besides the moving velocity, which provides the relative moving information between two sequential time intervals, the range information is considered in the likelihood function, which provides additional information to calculate the absolute position in the local coordinate system. By considering the range information, our proposed FW-PF is more robust to accumulative errors and it achieves around $2.2m$ for 90% accuracy, which outperforms PDR by 45%. The mean accuracy is $1.3m$, which is 19% higher than PDR. Additionally, FW-PF is more stable than PDR because the standard deviation of FW-PF is $0.7m$, which is 72% smaller than PDR.

Second, our proposed FW-PF outperforms R-PF considering accuracy and stability. For ranging only particle filter (R-PF), the velocity information is not considered in the likelihood function and the corresponding associated weight update. Therefore, some particles with unusual large moving speeds could be assigned large values of associated weights. For our proposed FW-PF, the estimated velocity based on inertial sensors is considered in the likelihood function. The particles with large shift velocity components from the estimated velocity are assigned small values of associated weights. Hence, their contributions to the final estimation are suppressed. By considering the exponential weights on the ranges from different ANs, the influence of ranging errors on the likelihood function is further mitigated. Therefore, our proposed FW-PF outperforms R-PF by around $0.8m$ for the 90% accuracy with 5 ANs and $0.2m$ with 4 ANs. Furthermore, the standard deviation of FW-PF is smaller than of R-PF in both scenarios, which means that the performance of FW-PF is more stable and estimated moving paths are more smooth compared to R-PF.

Finally, by increasing the number of ANs, FW-PF can integrate more range values in the likelihood function and has larger opportunity to have line-of-sight connection to one certain AN, which has higher ranging accuracy than NLOS. Therefore, the performance of FW-PF gets improved by increasing the number of ANs. FW-PF with 5 ANs outperforms 4 ANs by 21% for the 90% accuracy and 19% for the mean accuracy.

7.6 Conclusions

In this work, we proposed a network-based indoor tracking system, which fuses the range and velocity information by an enhanced particle filter. The proposed particle filter fuses the range information and velocity information in the likelihood function of the observation parameters. First, by considering range information in the likelihood, it is able to filter out the particles with low probability to observe the estimated ranges from the channel information. Second, by considering velocity information, the particles with low probability to observe the estimated velocity from inertial sensors are further filtered out. This allows us to track a mobile target with higher accuracy and stability than PDR and range-only particle filter. This is verified in our evaluation in a complex indoor environment. Additionally, to further mitigate multipath propagation, we exploit MIMO techniques on the previously proposed ranging method based on CSI (Chapter 5). Finally, our evaluation results show that the mean tracking accuracy achieves $1.3m$ and 90% accuracy is $2.2m$.

Chapter 8

Conclusions and Outlook

We first summarize the contributions of this thesis in the order of their occurrence in the thesis in Section 8.1. Our first contribution is time-based indoor positioning for narrow-band signals (GSM and IEEE 802.15.4 signals). The second contribution is a passive indoor positioning system mainly for stationary WiFi targets using fine-grained power information (channel information). The third contribution is to extend this passive indoor positioning system to support tracking mobile WiFi targets using an enhanced particle filter. The fourth contribution is an active (network-based) indoor tracking system for mobile WiFi targets by fusing the inertial sensor and channel information with an enhanced particle filter. Afterwards, we briefly discuss some possible future work in the field of indoor positioning in Section 8.2.

8.1 Main Contributions

In this thesis, we mainly work on radio-based indoor positioning systems and algorithms with fine-grained physical-layer information including physical-layer timestamps and channel information. For time-based indoor positioning, we provided solutions to compensate imperfect synchronization and design high-resolution timestamps for narrow-band signals including GSM and IEEE 802.15.4 signals, which were introduced in Part I (Chapters 3 and 4). In Part II (Chapters 5, 6, and 7), we mainly introduced our positioning and tracking methods relying on fine-grained power (channel information). In this part, we contributed to enhanced solutions for ranging, locating stationary targets and tracking mobile targets with IEEE 802.11 signals. Additionally, we also contributed to hybrid positioning systems by fusing the inertial sensor information and channel information for accurate tracking. The aforementioned contributions are supported by theoretical analysis and experiments in real-world deployments. Parts of our proposed algorithms have been transferred to our Swiss industry partner (DFRC AG) in the Eurostar project (In3DGuide). They adopt our proposed physical-layer timestamps in their prototype system to analyze the activities of GSM users.

Our contributions on time-based positioning are described in Chapter 3 and

4. In Chapter 3, we investigated two challenges of time-based positioning: timestamps and synchronization. First, we proposed to adopt timing error information from time recovery methods to improve the resolution of timestamps, which is a prerequisite for time-based indoor positioning. By analyzing the extracted timing error information from the physical layer, we are able to improve the resolution of timestamps (sub-sample timestamps) for GSM signals from microseconds to nanoseconds. We further adopted this timing error information to evaluate the clock offset and skew between GPS synchronized ANs. We derived the relation between the normalized timing error information and clock offset. Then, we proposed a two-step method including 1) smoothing noisy clock offsets by a S-G filter and 2) differentiation of the filtered clock offsets to calculate the clock skew. We evaluated our proposed methods in a GSM-like testbed based on SDR techniques. The main findings for synchronization are that (1) the maximum clock offset is up to $171ns$ but (2) the clock skews are smaller than $1.37 \cdot 10^{-9}$ for GPS synchronized receivers. Therefore, we conclude that GPS synchronization introduces a large error for TDOA measurements, which needs to be compensated to achieve high positioning accuracy.

In Chapter 4, we proposed to combine DTDOA with GPS synchronization to eliminate the influence of imperfect synchronization, based on the findings for GPS synchronization in Chapter 3. We theoretically investigated the influence factors on the DTDOA measurements and compared DTDOA to TDOA. Then, based on the measured DTDOA values, we proposed DTDOA-based LLS and fingerprinting algorithms to locate target devices. Our algorithms were implemented in the GSM-like testbed and a passive positioning system for IEEE 802.15.4 signals, which adopts SDR-based ANs to passively decode IEEE 802.15.4 packets and extract sub-sample timestamps. Based on these two systems, we evaluated our proposed time-based positioning methods for narrow-band signals in indoor and outdoor environments. Through experiments, we demonstrated that DTDOA-based ranging and LLS significantly improve ranging and positioning accuracy, compared to TDOA-based ranging and LLS. In outdoor environments, DTDOA-based ranging achieves around 10m accuracy for a GSM-like signal with only 250KHz bandwidth. However, in a complex indoor environment, multipath and NLOS propagation still introduce large errors for the estimated timestamps with IEEE 802.15.4 signals. We further evaluated the DTDOA-based fingerprinting and demonstrated that in the narrow-band system DTDOA-based fingerprinting significantly outperforms LLS and is able to locate the target in an indoor environment with a mean error of 3m. DTDOA-based and RSS-based fingerprinting algorithms achieve different performance in different areas. RSS-based fingerprinting achieves higher accuracy when there is one AN with LOS connection to the target. However, DTDOA-based fingerprinting outperforms RSS-based fingerprinting in the area, where there is no AN with LOS connection to the target.

Our contributions on positioning with channel information are described in Chapter 5, 6, and 7. In Chapter 5, we designed and implemented a SDR-based passive positioning system for WiFi signals and proposed a set of novel algorithms to locate stationary targets by using channel information. SDR techniques are adopted to overhear the packets from the target devices and allow us to extract physical-layer channel information for positioning. We adopted channel impulse response to mitigate multipath propagation and get the fine-grained power (RSS). Then, we provided a new model based on nonlinear regression (the NLR model) to estimate the propagation distance from the measured RSS. A new trilateration algorithm combining the Weight Centroid (WC) algorithm and Constrained Weighted Least Square (CWLS) algorithm, i.e., the WC-CWLS algorithm, was designed and implemented in the system to mitigate the influence of ranging errors. Comprehensive experiments for a single user show that the proposed NLR model for ranging significantly outperforms the LDPL model and hence trilateration algorithms based on the NLR model achieve better accuracy. The proposed WC-CWLS algorithm is more robust to ranging errors than the LLS algorithm and achieves better positioning accuracy than the WC and LLS algorithms with both the NLR and LDPL model. By combining NLR for ranging and WC-CWLS for positioning, we achieve a mean accuracy of $2.4m$.

In Chapter 6, we extended the passive positioning system for WiFi targets in Chapter 5 to better support tracking mobile users by designing an enhanced particle filter. In this system, the enhanced ranging methods, i.e., channel information and nonlinear regression for the ranging model (proposed in Chapter 5), are utilized to achieve highly accurate ranging. An enhanced particle filter (WVT-BPF) exclusively relying on power-based ranging with low calibration efforts is further proposed to achieve high tracking accuracy. WVT-BPF integrates three main novel contributions including weighted likelihood, velocity limitation on likelihood and a single coordinated turn model. Each of the individual contributions improves the tracking accuracy compared to the traditional BPF. By integrating all these improvements, WVT-BPF outperforms the traditional BPF, EKF, and trilateration algorithms. Additionally, the computation efforts of WVT-BPF are much lower than MM-BPF. By combining WVT-BPF with the enhanced ranging methods, our system passively tracks the WiFi target with an accuracy of $1.5m$ for 50% and $2.3m$ for 90%.

In Chapter 7, we proposed an active (network-based) indoor tracking system for WiFi targets, which fuses the range and velocity information by an enhanced particle filter. Velocity information is estimated by an efficient method based on the timestamped values from accelerometer and compass. The range information is basically estimated based on the enhanced ranging method relying on channel information as proposed in Chapter 5 but we extend this ranging method by utilizing MIMO techniques in IEEE 802.11n to further mitigate the influence of multipath. The enhanced particle filter (FW-PF) is adopted to fuse these two types of information in the likelihood function and is equipped with the weighting technique as proposed in Chapter 6 to mitigate the influence of ranging errors. The system was implemented by using some cheap commercial devices for ANs, which are able to extract inertial sensor and CSI information from the received WiFi packets. We evaluated our system in a complex indoor environment. Evaluation results indicate that our proposed FW-PF is more accurate and stable than pedestrian dead reckoning and range-only particle filters. The mean accuracy achieves $1.3m$ and 90% accuracy is $2.2m$.

8.2 Outlook

In this thesis, we contributed to multiple positioning techniques including time-based indoor positioning, fine-grained power (channel information) based indoor positioning and tracking, and hybrid approach by fusing inertial sensor and channel information. All these positioning methods rely on physical-layer information of radio signals. Hence, physical-layer properties of radio signals, e.g., modulation schemes, bandwidth, and operating frequency, play an important role on the positioning accuracy. Some of these physical-layer information including timestamps and channel information for positioning is limited by the current wireless standards. With the development of new wireless communication systems, we believe that the accuracy of positioning with these fine-grained physical-layer information can be further improved.

For time-based indoor positioning, we worked on narrow-band signals including GSM and IEEE 802.15.4 signals. Due to limited bandwidth, their timestamps are vulnerable to multipath effects. The latest standard for WiFi signals, IEEE 802.11ac [18], supports up to 160 MHz bandwidth. Therefore, thanks to this wide bandwith, accurate timestamps, which are more robust to multipath propagation, can be designed. Similar as time-based positioning, one of the limitations for multipath mitigation using channel information is the limited bandwidth of signals. The extracted CIR from IEEE 802.11n signals with 20 MHz can only distinguish a cluster of propagation paths with a resolution of 50ns other than individual multipath channels. Therefore, the latest standard IEEE 802.11ac would be able to provide a more fine-grained CIR information to better mitigate multipath effects, which would allow distinguishing a cluster of channels with a resolution of 6.25ns. To achieve accurate timestamps and channel estimation, the current commercial WiFi cards for IEEE 802.11ac do not support physical layer signal processing and hence SDR techniques would be interesting for decoding the signals and extracting the physical-layer information including timestamps and channel information.

Additionally, intensive research is currently conducted in the area of the 5th generation (5G) cellular networks [19], especially for the air interfaces. Although up to now there is still no standard for the bandwidth of 5G, it is clear that the bandwidth would be significantly higher than the current 4G network with 20MHz to support the expected maximum data rate of 10 Gbps. Therefore, these physical layer information including timestamps and channel information would be also interesting for positioning in 5G networks, which consider locations of users as an important information for handover, beamforming, and location based services. Therefore, our positioning methods with fine-grained physical-layer information are also very interesting to be adopted in future 5G networks.

Inertial sensors in modern smart phones play an important role in indoor tracking. In our work, we provided a hybrid approach to combine inertial sensors with radio signals. To improve tracking accuracy, additional information in the tracking environment can be used to constrain the moving path. For example the floor plan can be used to constrain the movement of the particles in the particle filter to further filter out some particles with low observation likelihood. SLAM (simultaneous localization and mapping) techniques [45], which are commonly used in robot navigation, are interesting to be investigated in the area of indoor tracking for smart phones. A robot adopts SLAM techniques to construct or update a map of an unknown environment while it simultaneously keeps tracking of its location within the map. In indoor tracking for smart phones, the map can be a radio map of RSS or channel information from some access points with unknown locations. By using inertial sensors, the smart phone can track its moving trajectories and simultaneously record the radio informations and construct the radio map. The radio map can be used for the target to improve the tracking accuracy when it comes back to the same locations and can also be used for other users when they come to the same locations.

Furthermore, our proposed positioning or tracking systems focus on two dimensional scenarios and are evaluated in indoor testing areas with a size of $18m \times 16m$. First, it would be very valuable to extend our systems to support three dimensional tracking and positioning. This would require our systems to distinguish users on different floors. We believe that inertial sensors are a powerful tool for identifying different floors. For example, we can track the users on different floors by some activity recognition algorithms to identify different activities of users, such as climbing up or down stairs. Second, it would be very interesting to test the algorithms and systems in a larger area with larger number of deployed ANs. In our passive positioning and tracking system for WiFi signals (in Chapter 5 and 6), the user is always connected to one WiFi router. In a large scale deployment, one of the problems is that the user may roam among different WiFi routers. In such case, we need to find a solution to identify the user roaming among different WiFi routers. One of the solutions is that the source MAC address in the block ACKs can be used to identify the user roaming among different WiFi routers, because the source MAC address will not change when the user roams among different WiFi routers. Then, the problem to use block ACKs is that the packet rate depends on the configurations of the WiFi routers, which are unavailable to change for a passive positioning system. As mentioned in Section 5.3.3, one future work to identify different users is to map each legacy ACK to its corresponding data packet and consequently distinguish different users by the legacy ACKs. Same as the block ACKs, the MAC address of the user in the data packets does not change when the user roams among different WiFi routers. Therefore, the user can be identified when he roams among different WiFi routers. Compared to the solution with block ACKs, the packet rate of legacy ACKs is much higher, which can improve the positioning accuracy.

Finally, although our IEEE 802.15.4 and WiFi positioning systems are able to distinguish multiple users by their node IDs or MAC addresses, our work focuses on improving the positioning accuracy and reducing calibration efforts of our positioning algorithms with a single user. For positioning and tracking multiple users especially for real-time positioning, response time and scalability are important issues. Large amount of users will definitely increase the computation efforts and response time at the server side. If the computation capability in the server is not powerful enough, we can only decrease the computation efforts for a single user to support more users in real time. As mentioned in Chapter 6, we can reduce the number of particles to reduce the computation efforts and execution time for tracking a single user and hence increase the number of supported users. However, this will reduce the tracking accuracy. With fast development of cloud techniques, cloud servers provide a higher computation capability and more flexible way to allocate resources. Therefore, we can move our proposed positioning algorithms to a cloud infrastructure, which can manage the cloud resources in an on-demand fashion to better support multi-user positioning with short response time and high accuracy.

Chapter 9

List of Acronyms

ADC	Analog-to-Digital Converter
AN	Anchor Node
AP	Access Point
ARM	Advanced RISC Machine
BPF	Bootstrap Particle Filter
BPSK	Binary Phase Shift Keying
CDF	Cumulative Distribution Function
CIC	Cascaded Integrator Comb
CIR	Channel Impulse Response
CSI	Channel State Information
CV	Constant Velocity
CWLS	Constrained Weighted Least Square
DC	Direct Current
DDC	Digital Down-Converter
DSSS	Direct Sequence Spread Spectrum
DSP	Digital Signal Processor
DTDOA	Differential Time Difference Of Arrival
DUC	Digital Up-Converter
DVB	Digital Video Broadcasting

EKF	Extended Kalman Filter
FCF	Frame Control Field
FCS	Frame Check Sequence
FFT	Fast Fourier Transform
FIFO	First Input First Output
FM	Frequency Modulation
FPGA	Field-Programmable Gate Array
FW-PF	Particle Filter with data Fusion and Weighted likelihood
GI	Guard Interval
GMSK	Gaussian Minimum Shift Keying
GPP	General Purpose Processor
GPS	Global Positioning System
GPSDO	GPS-Disciplined Oscillator
GPU	Graphics Processing Unit
GSM	Global System for Mobile Communication
HMM	Hidden Markov Model
HT	High Throughput
IF	Intermediate Frequency
IFFT	Inverse Fast Fourier Transform
IMU	Inertial Measurement Units
KDE	Kernel Destiny Estimation
KNN	K Nearest Neighbours
LDPL	Log-Distance Path Loss
LKF	Linear Kalman Filter
LLS	Linear Least Square
LOS	Line Of Sight
LS	Least Square

LTE	Long Term Evolution
LTS	Long Training Sequence
MAC	Media Access Control
SCT	Single Coordinated Turn Model
MIMO	Multiple Input and Multiple Output
ML	Maximum Likelihood
MM-BPF	Multi-Model BPF
MMSE	Minimum Mean Square Error
MPDU	MAC Protocol Data Unit
MSK	Minimum Shift Keying
NLOS	Non-Line Of Sight
NLR	Non-Linear Regression
NRZ	Non-Return to Zero
NLS	Non-linear Least Square
NTP	Network Time Protocol
OCXO	Oven-Controlled Crystal Oscillator
OFDM	Orthogonal Frequency Division Multiplexing
O-QPSK	Offset-Quadrature Phase Shift Keying
PDF	Probability Density Function
PDR	Pedestrian Dead Reckoning
PF	Particle Filter
PHY	Physical Layer
PLL	Phase Lock Loop
PPDU	Physical Protocol Data Unit
P/S	Parallel to Serial
PTP	Precision Time Protocol
RF	Radio Frequency

RN	Reference Node
R-PF	Range-only Particle Filter
RSCG	Reference and System Clock Generation
RSSI	Received Signal Strength Indicator
SD	Standard Deviation
SDR	Software Defined Radio
SFD	Start of Frame Delimiter
SLAM	Simultaneous Localization And Mapping
SIS	Sequential Importance Sampling
SNR	Signal to Noise Ratio
STS	Short Training Sequence
S-G Filter	Savitzky-Golay Filter
S/P	Serial to Parallel
T-BPF	BPF with modified Coordinated Turn
TDOA	Time Difference Of Arrival
TED	Timing Error Detection
TOA	Time Of Arrival
TOF	Time Of Flight
U-TDOA	Uplink Time Difference Of Arrival
UHD	USRP Hardware Driver
UMTS	Universal Mobile Telecommunication System
UWB	Ultra Wide Band
V-BPF	BPF with Velocity limitation
WAF	Wall Attenuation Factor
W-BPF	BPF with Weighted likelihood
WC	Weighted Centroid
WLAN	Wireless Local Area Network

WLS	Weighted Least Square
WKNN	Weighted K Nearest Neighbours
WVT-BPF	BPF with Weighted likelihood, Velocity limitation and modified coordinated Turn
3GPP	3rd Generation Partnership Project

Bibliography

[1] "2.4 GHz IEEE 802.15.4 / Zigbee-ready RF Transceiver," Texas Instruments, Tech. Rep.

[2] 3rd Generation Partnership Project (3GPP). [Online]. Available: http://www.3gpp.org/

[3] Ettus Research Website for SBX. [Online]. Available: http://www.ettus.com/product/details/SBX

[4] Ettus Research Website for WBX. [Online]. Available: http://www.ettus.com/product/details/WBX

[5] Gnu Radio Website. [Online]. Available: http://gnuradio.org/redmine/wiki/gnuradio

[6] WARP Project Website. [Online]. Available: http://warpproject.org/

[7] "3GPP Release 98," *3rd Generation Partnership Project*, 1999.

[8] "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11b-1999*, 1999.

[9] "Wireless LAN Medium Access control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11a-1999*, 1999.

[10] "3GPP Release 99," *3rd Generation Partnership Project*, 2000.

[11] "Wireless lan medium access control (mac) and physical layer (phy) specifications," *IEEE Std 802.11g-2003*, 2003.

[12] "3GPP Release 8," *3rd Generation Partnership Project*, 2008.

[13] "IEEE Standard for Information Technology– Local and Metropolitan Area Networks– Specific Requirements– Part 11: Wireless LAN Medium Access Control (MAC)and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput," *IEEE Std 802.11n-2009*, Oct 2009.

[14] "Spartan-3A DSP FPGA Family Data Sheet," 2010.

[15] “GPSDO Kit Instructions and Data Sheet,” 2012.

[16] “USRP E110/E100 Data Sheet,” 2012.

[17] “USRP N200/N210 Data Sheet,” 2012.

[18] “IEEE Standard for Information Technology– Telecommunications and Information Exchange Between Systemslocal and Metropolitan Area Networks– Specific Requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very high Throughput for Operation in Bands Below 6 GHz.” *IEEE Std 802.11ac-2013*, pp. 1–425, Dec 2013.

[19] “Understanding 5G: Perspectives on Future Technological Advancements in Mobile,” GSMA Intelligence, Tech. Rep., 2014.

[20] V. Abhayawardhana, I. Wassell, D. Crosby, M. Sellars, and M. Brown, “Comparison of Empirical Propagation Path Loss Models for Fixed Wireless Access Systems,” in *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, vol. 1, May 2005, pp. 73–77.

[21] R. Akl, D. Tummala, and X. Li, “Indoor Propagation Modeling At 2.4 Ghz For IEEE 802.11 Networks,” in *Wireless Networks and Emerging Technologies*, 2006.

[22] M. S. Arulampalam, B. Ristic, N. Gordon, and T. Mansell, “Bearings-only Tracking of Manoeuvring Targets Using Particle Filters,” *EURASIP J. Appl. Signal Process.*, vol. 2004, pp. 2351–2365, Jan. 2004.

[23] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking,” *Signal Processing, IEEE Transactions on*, vol. 50, no. 2, pp. 174–188, Feb 2002.

[24] P. Bahl and V. Padmanabhan, “RADAR: an in-Building RF-based user Location and Tracking System,” in *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 2, 2000, pp. 775–784 vol.2.

[25] P. Barralon, N. Vuillerme, and N. Noury, “Walk Detection With a Kinematic Sensor: Frequency and Wavelet Comparison,” in *Engineering in Medicine and Biology Society, 2006. EMBS ’06. 28th Annual International Conference of the IEEE*, Aug 2006, pp. 1711–1714.

[26] S. Beauregard, “A Helmet-Mounted Pedestrian Dead Reckoning System,” in *Applied Wearable Computing (IFAWC), 2006 3rd International Forum on*, March 2006, pp. 1–11.

[27] U. Birkel and M. Weber, “Indoor Localization with UMTS Compared to WLAN,” in *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, Nov 2012, pp. 1–6.

[28] B. Bloessl, M. Segata, C. Sommer, and F. Dressler, “Decoding IEEE 802.11a/G/P OFDM in Software Using GNU Radio,” in *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, ser. MobiCom ’13, 2013, pp. 159–162.

[29] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, “Weighted Centroid Localization in Zigbee-based Sensor Networks,” in *Intelligent Signal Processing, 2007. WISP 2007. IEEE International Symposium on*, Oct 2007, pp. 1–6.

[30] R. W. Bohannon, “Comfortable and Maximum Walking Speed of Adults Aged 20 to 79 Years: Reference Values and Determinants,” *Age and Ageing*, vol. 26, no. 1, pp. 15–19, 1997.

[31] A. Brajdic and R. Harle, “Walk Detection and Step Counting on Unconstrained Smartphones,” in *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp ’13. New York, NY, USA: ACM, 2013, pp. 225–234.

[32] R. H. Byrd, R. B. Schnabel, and G. A. Shultz, “A Trust Region Algorithm for Nonlinearly Constrained Optimization,” in *SIAM Journal on Numerical Analysis*, vol. 24, 1987, pp. 1152–1170.

[33] Y. Chapre, A. Ignjatovic, A. Seneviratne, and S. Jha, “CSI-MIMO: Indoor Wi-Fi Fingerprinting System,” in *Local Computer Networks (LCN), 2014 IEEE 39th Conference on*, Sept 2014, pp. 202–209.

[34] D. Chen, M. Nixon, and A. Mok, *WirelessHART: Real-Time Mesh Network for Industrial Automation*, 1st ed. Springer Publishing Company, Incorporated, 2010.

[35] P.-C. Chen, “A non-line-of-sight Error Mitigation Algorithm in Location Estimation,” in *Wireless Communications and Networking Conference, 1999. WCNC. 1999 IEEE*, 1999, pp. 316–320 vol.1.

[36] X. Cheng, A. Thaeler, G. Xue, and D. Chen, “TPS: a Time-Based Positioning Scheme for Outdoor Wireless Sensor Networks,” in *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, vol. 4, March 2004, pp. 2685–2696 vol.4.

[37] K. W. Cheung, H. C. So, W.-K. Ma, and Y. T. Chan, “A Constrained Least Squares Approach to Mobile Positioning: Algorithms and Optimality,” *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 150–150, Jan. 2006.

[38] V. Chironi, M. Pasca, S. D Amico, A. Leone, and P. Siciliano, "IR-UWB for Ambient Assisted Living Applications," vol. 11, pp. 209–218, 2015.

[39] Y. S. Cho, *MIMO-OFDM Wireless Communications with MATLAB*. IEEE Press, 2010.

[40] L. Choong, "Multi-Channel IEEE 802.15.4 Packet Capture Using Software Defined Radio," 2009.

[41] O. Costilla-Reyes and K. Namuduri, "Dynamic Wi-Fi Fingerprinting Indoor Positioning System," in *Indoor Positioning and Indoor Navigation (IPIN), 2014 International Conference on*, Oct 2014, pp. 271–280.

[42] A. Dammann and S. Kaiser, "Standard Conformable Antenna Diversity Techniques for OFDM and Its Application to the DVB-T System," in *Global Telecommunications Conference, 2001. GLOBECOM '01. IEEE*, vol. 5, 2001, pp. 3100–3105 vol.5.

[43] J. Del Peral-Rosado, J. Lopez-Salcedo, G. Seco-Granados, F. Zanier, and M. Crisci, "Achievable Localization Accuracy of the Positioning Reference Signal of 3GPP LTE," in *Localization and GNSS (ICL-GNSS), 2012 International Conference on*, June 2012, pp. 1–6.

[44] B. Denis, J. Keignart, and N. Daniele, "Impact of NLOS Propagation Upon Ranging Precision in UWB Systems," in *Ultra Wideband Systems and Technologies, 2003 IEEE Conference on*, Nov 2003, pp. 379–383.

[45] M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem," *Robotics and Automation, IEEE Transactions on*, vol. 17, no. 3, pp. 229–241, Jun 2001.

[46] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors," in *Local Computer Networks, 2004. 29th Annual IEEE International Conference on*, Nov 2004, pp. 455–462.

[47] V. Elamaran, R. Vaishnavi, A. Rozario, S. Joseph, and A. Cherian, "CIC for Decimation and Interpolation using Xilinx System Generator," in *Communications and Signal Processing (ICCSP), 2013 International Conference on*, April 2013, pp. 622–626.

[48] P. Enge and P. Misra, "Special Issue on Global Positioning System," *Proceedings of the IEEE*, vol. 87, no. 1, pp. 3–15, Jan 1999.

[49] V. Erceg, L. Greenstein, S. Tjandra, S. Parkoff, A. Gupta, B. Kulic, A. Julius, and R. Bianchi, "An Empirically based Path Loss Model for Wireless Channels in Suburban Environments," *Selected Areas in Communications, IEEE Journal on*, vol. 17, no. 7, pp. 1205–1211, Jul 1999.

[50] F. Evennou and F. Marx, “Advanced Integration of WIFI and Inertial Navigation Systems for Indoor Mobile Positioning,” *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 164–164, Jan. 2006.

[51] D. B. Faria, “Modeling Signal Attenuation in IEEE 802.11 Wireless LANs - Vol. 1,” Standford University, Tech. Rep., 2005.

[52] S. Ganeriwal, R. Kumar, and M. B. Srivastava, “Timing-sync Protocol for Sensor Networks,” in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, 2003.

[53] C. Gao and R. Harle, “Sequence-Based Magnetic Loop Closures for Automated Signal Surveying,” in *Indoor Positioning and Indoor Navigation (IPIN), 2015 International Conference on*, October 2015, pp. 1–12.

[54] F. Ge, C. Chiang, Y. Gottlieb, and R. Chadha, “GNU Radio-Based Digital Communications: Computational Analysis of a GMSK Transceiver,” in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, Dec 2011, pp. 1–6.

[55] S. Gezici, I. Guvenc, and Z. Sahinoglu, “On the Performance of Linear Least-Squares Estimation in Wireless Positioning Systems,” in *Communications, 2008. ICC ’08. IEEE International Conference on*, May 2008, pp. 4203–4208.

[56] S. Gezici, Z. Tian, G. Giannakis, H. Kobayashi, A. Molisch, H. Poor, and Z. Sahinoglu, “Localization Via Ultra-Wideband Radios: a Look at Positioning Aspects for Future Sensor Networks,” *Signal Processing Magazine, IEEE*, vol. 22, no. 4, pp. 70–84, July 2005.

[57] E. Goldoni, A. Savioli, M. Risi, and P. Gamba, “Experimental Analysis of RSSI-Based Indoor Localization with IEEE 802.15.4,” in *Wireless Conference (EW), 2010 European*, April 2010, pp. 71–77.

[58] P. Goyal, V. Ribeiro, H. Saran, and A. Kumar, “Strap-down Pedestrian Dead-Reckoning System,” in *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, Sept 2011, pp. 1–7.

[59] I. Güvenç, C.-C. Chong, F. Watanabe, and H. Inamura, “NLOS Identification and Weighted Least-Squares Localization for UWB Systems Using Multipath Channel Statistics,” *EURASIP J. Adv. Signal Process*, vol. 2008, Jan. 2008.

[60] D. Halperin, W. Hu, A. Sheth, and D. Wetherall, “Tool Release: Gathering 802.11N Traces with Channel State Information,” *SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 1, pp. 53–53, Jan. 2011.

[61] D.-M. Han and J.-H. Lim, “Smart Home Energy Management System using IEEE 802.15.4 and Zigbee,” *Consumer Electronics, IEEE Transactions on*, vol. 56, no. 3, pp. 1403–1410, Aug 2010.

[62] S. Hara and R. Prasad, “Overview of Multicarrier CDMA,” *Communications Magazine, IEEE*, vol. 35, no. 12, pp. 126–133, Dec 1997.

[63] A. J. Haug, *Bayesian Estimation and Tracking : A Practical Guide*. Hoboken, N.J.: Wiley, 2012.

[64] F. Hong, Y. Zhang, Z. Zhang, M. Wei, Y. Feng, and Z. Guo, “WaP: Indoor Localization and Tracking using WiFi-Assisted Particle Filter,” in *Local Computer Networks (LCN), 2014 IEEE 39th Conference on*, Sept 2014, pp. 210–217.

[65] A. Jimenez, F. Seco, C. Prieto, and J. Guevara, “A Comparison of Pedestrian Dead-Reckoning Algorithms using a Low-Cost MEMS IMU,” in *Intelligent Signal Processing, 2009. WISP 2009. IEEE International Symposium on*, Aug 2009, pp. 37–42.

[66] N. Kakiuchi and S. Kamijo, “Pedestrian Dead Reckoning for Mobile Phones through Walking and Running Mode Recognition,” in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 261–267.

[67] J. Kim, S. Hyeon, and S. Choi, “Implementation of an SDR System using Graphics Processing Unit,” *Communications Magazine, IEEE*, vol. 48, no. 3, pp. 156–162, March 2010.

[68] J. Kim, H. Jang, D. Hwang, and C. Park, “A Step, Stride and Heading Determination for the Pedestrian Navigation System,” *Journal of Global Positioning Systems*, pp. 273–279, 2004.

[69] L. Koski, T. Perala, and R. Piche, “Indoor Positioning using WLAN Coverage Area Estimates,” in *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on*, Sept 2010, pp. 1–7.

[70] S. Krishnan, P. Sharma, Z. Guoping, and O. H. Woon, “A UWB Based Localization System for Indoor Robot Navigation,” in *Ultra-Wideband, 2007. ICUWB 2007. IEEE International Conference on*, Sept 2007, pp. 77–82.

[71] S. Krishnan and C. Seelamantula, “On the Selection of Optimum Savitzky-Golay Filters,” *Signal Processing, IEEE Transactions on*, vol. 61, no. 2, pp. 380–391, Jan 2013.

[72] A. Kurt, “Indoor Tracking with Kalman Filters Using Rss-Based Ranging,” Bachelor’s Thesis, University of Bern, Switzerland, 2015.

[73] J. D. Laster, "Robust GMSK Demodulation Using Demodulator Diversity and BER Estimation," Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, 1997.

[74] K. Lee and J. Eidson, "IEEE-1588 Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," in *In 34 th Annual Precise Time and Time Interval (PTTI) Meeting*, 2002, pp. 98–105.

[75] X. Li, J. Wang, A. Olesk, N. Knight, and W. Ding, "Indoor Positioning within a Single Camera and 3D Maps," in *Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 2010*, 2010, pp. 1–9.

[76] Z. Li, D. B. Acuna, Z. Zhao, J. L. Carrera, and T. Braun, "Fine-grained Indoor Tracking by Fusing Inertial Sensor and Physical Layer Information in WLANs," in *Communications (ICC), 2016 IEEE International Conference on*, 2016.

[77] Z. Li and T. Braun, "Passively Track WiFi Users with an Enhanced Particle Filter using Power-based Ranging," University of Bern, Tech. Rep., 2015.

[78] Z. Li, T. Braun, and D. Dimitrova, "A Passive WiFi Source Localization System Based on Fine-Grained Power-Based Trilateration," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015 IEEE 16th International Symposium on a*, June 2015, pp. 1–9.

[79] ——, "A Time-Based Passive Source Localization System for Narrow-Band Signal," in *Communications (ICC), 2015 IEEE International Conference on*, June 2015, pp. 4599–4605.

[80] ——, "Methodology for GPS Synchronization Evaluation with High Accuracy," in *Vehicular Technology Conference (VTC Spring), 2015 IEEE 81st*, May 2015, pp. 1–6.

[81] Z. Li, D. Dimitrova, T. Braun, and D. Rosario, "Highly Accurate Evaluation of GPS S]ynchronization for TDOA Localization, year=2013, pages=1-3, issn=2156-9711, month=Nov,," in *Wireless Days (WD), 2013 IFIP*.

[82] Z. Li, D. Dimitrova, D. Raluy, and T. Braun, "TDOA for Narrow-band Signal with Low Sampling Rate and Imperfect Synchronization," in *Wireless and Mobile Networking Conference (WMNC), 2014 7th IFIP*, May 2014, pp. 1–8.

[83] Y. Lin, H. Lee, M. Woh, Y. Harel, S. Mahlke, T. Mudge, C. Chakrabarti, and K. Flautner, "SODA: A High-Performance DSP Architecture for Software-Defined Radio," *Micro, IEEE*, vol. 27, no. 1, pp. 114–123, Jan 2007.

[84] J. Link, P. Smith, N. Viol, and K. Wehrle, “Foot Path: Accurate Map-Based Indoor Navigation using Smartphones,” in *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, Sept 2011, pp. 1–8.

[85] M. Mahfouz, C. Zhang, B. Merkl, M. Kuhn, and A. Fathy, “Investigation of High-Accuracy Indoor 3-D Positioning Using UWB Technology,” *Microwave Theory and Techniques, IEEE Transactions on*, vol. 56, no. 6, pp. 1316–1330, June 2008.

[86] S. Marano, W. Gifford, H. Wymeersch, and M. Win, “NLOS Identification and Mitigation for Localization Based on UWB Experimental Data,” *Selected Areas in Communications, IEEE Journal on*, vol. 28, no. 7, pp. 1026–1035, September 2010.

[87] A. Marcaletti, M. Rea, D. Giustiniano, V. Lenders, and A. Fakhreddine, “Filtering Noisy 802.11 Time-of-Flight Ranging Measurements,” in *Proceedings of the 10th ACM International on Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT ’14. New York, NY, USA: ACM, 2014, pp. 13–20.

[88] E. Mazomenos, J. Reeve, N. White, and A. Brown, “A Tracking System for Wireless Embedded Nodes Using Time-of-Flight Ranging,” *Mobile Computing, IEEE Transactions on*, vol. 12, no. 12, pp. 2373–2385, Dec 2013.

[89] H. Meyr, M. Moeneclaey, and S. Fechtel, *Digital Communication Receivers: Synchronization, Channel Estimation, and Signal Processing*, 1997.

[90] D. Mills, “Network Time Protocol (Version 3) Specification, Implementation,” United States, 1992.

[91] P. Mirowski, T. K. Ho, S. Yi, and M. Macdonald, “SignalSLAM: Simultaneous Localization and Mapping with Mixed WiFi, Bluetooth, LTE and Magnetic Signals,” in *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, Oct 2013, pp. 1–10.

[92] P. Misra and P. Enge, *Global Positioning System: Signals, Measurements, and Performance*, 2nd ed. Ganga-Jamuna Press, Lincoln MA, 2006.

[93] S. Moon, P. Skelly, and D. Towsley, “Estimation and Removal of Clock Skew from Network Delay Measurements,” in *INFOCOM ’99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 1, Mar 1999, pp. 227–234 vol.1.

[94] J. J. Moré and D. C. Sorensen, “Computing a Trust Region Step,” vol. 4, pp. 553–572, 1983.

[95] K. Mueller and M. Muller, “Timing Recovery in Digital Synchronous Data Receivers,” *Communications, IEEE Transactions on*, vol. 24, no. 5, pp. 516–531, May 1976.

[96] K. Murota and K. Hirade, “GMSK Modulation for Digital Mobile Radio Telephony,” *Communications, IEEE Transactions on*, vol. 29, no. 7, pp. 1044–1050, Jul 1981.

[97] A. Nagy, R. Exel, P. Loschmidt, and G. Gaderer, “Time-based Localisation in Unsynchronized Wireless LAN for Industrial Automation Systems,” in *Emerging Technologies Factory Automation (ETFA), 2011 IEEE 16th Conference on*, 2011, pp. 1–8.

[98] J. Nocedal and S. Wright, *Numerical Optimization*, ser. Springer Series in Operations Research and Financial Engineering. Springer New York, 2006. [Online]. Available: https://books.google.ch/books?id=eNlPAAAAMAAJ

[99] T. Ozyagcilar, “Implementing a Tilt Compensated eCompass using Accelerometer and Magnetometer Sensors,” Freescale Semiconductor, Rev. 3, Tech. Rep., 2012.

[100] S. Petersen and S. Carlsen, “WirelessHART Versus ISA100.11a: The Format War Hits the Factory Floor,” *Industrial Electronics Magazine, IEEE*, vol. 5, no. 4, pp. 23–34, Dec 2011.

[101] M. Pichler, S. Schwarzer, A. Stelzer, and M. Vossiek, “Multi-Channel Distance Measurement With IEEE 802.15.4 (ZigBee) Devices,” *Selected Topics in Signal Processing, IEEE Journal of*, vol. 3, no. 5, pp. 845–859, Oct 2009.

[102] J. Polastre, R. Szewczyk, and D. Culler, “Telos: Enabling Ultra-Low Power Wireless Research,” in *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, April 2005, pp. 364–369.

[103] J. G. Proakis and M. Salehi, *Communication Systems Engineering*, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, August 2001.

[104] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, “Zee: Zero-effort Crowdsourcing for Indoor Localization,” in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, ser. Mobicom ’12. New York, NY, USA: ACM, 2012, pp. 293–304.

[105] C. Randell, C. Djiallis, and H. Muller, “Personal Position Measurement using Dead Reckoning,” in *Wearable Computers, 2003. Proceedings. Seventh IEEE International Symposium on*, Oct 2003, pp. 166–173.

[106] A. Rolando and E. Amoruso, “An Ubiquitous Positioning System Based on IEEE 802.15.4 Radio Signals,” in *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, Oct 2013, pp. 1–10.

[107] R. Schafer, “What Is a Savitzky-Golay Filter? [Lecture Notes],” *Signal Processing Magazine, IEEE*, vol. 28, no. 4, pp. 111–117, July 2011.

[108] J. Schmid, F. Beutler, B. Noack, U. D. Hanebeck, and K. D. Müller-Glaser, "An Experimental Evaluation of Position Estimation Methods for Person Localization in Wireless Sensor Networks," in *Proceedings of the 8th European Conference on Wireless Sensor Networks*, ser. EWSN'11, 2011, pp. 147 – 162.

[109] T. Schmid, "GNU Radio 802.15.4 En- and Decoding," Tech. Rep., 2006.

[110] Z. Shah, R. Malaney, X. Wei, and K. Tai, "Experimental Deployment of Particle Filters in WiFi Networks," in *Communications, 2007. ICC '07. IEEE International Conference on*, June 2007, pp. 4692–4697.

[111] S. Shearman and J. Kimery, "Software Defined Radio Prototyping Platforms Enable a Flexible Approach to Design [Application Notes]," *Microwave Magazine, IEEE*, vol. 13, no. 5, pp. 76–80, July 2012.

[112] Y. Shen and E. F. Martinez, "Channel Estimation in OFDM Systems," Freescale Semiconductor, Tech. Rep., January 2006.

[113] Y. Shu, C. Bo, G. Shen, C. Zhao, L. Li, and F. Zhao, "Magicol: Indoor Localization Using Pervasive Magnetic Field and Opportunistic WiFi Sensing," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 7, pp. 1443–1457, July 2015.

[114] J. Smith and J. Abel, "Closed-Form Least-Squares Source Location Estimation from Range-Difference Measurements," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 35, no. 12, pp. 1661–1669, Dec 1987.

[115] M. Stolz, "An LTE Signal Analyser," Master's Thesis, University of Bern, Switzerland, 2015.

[116] TechnoCom, "True Position Indoor Location Test Bed Report," Tech. Rep., 2013.

[117] Z. Tian, X. Liu, M. Zhou, and K. Xu, "Mobility Tracking by Fingerprint-Based KNN/PF Approach in Cellular Networks," in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, April 2013, pp. 4570–4575.

[118] J. R.-I. E. T.-R. D. Vargas-Rosales, C. Mass-Sanchez and A. Espinoza-Ruiz, "Performance Evaluation of Localization Algorithms for WSNs," *International Journal of Distributed Sensor Networks*, vol. 2015, no. 14, 2015.

[119] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "GSM Indoor Localization," *Pervasive Mob. Comput.*, vol. 3, no. 6, pp. 698–720, Dec. 2007.

[120] C. Wang, T. Jiang, and Q. Zhang, *ZigBee Network Protocols and Applications*. Boston, MA, USA: Auerbach Publications, 2014.

[121] H. Weinberg, "Using the ADXL202 in Pedometer and Personal Navigation Applications," Analog Devices, Tech. Rep., 2002.

[122] F. Winkler, E. Fischer, E. Grass, and P. Langendorfer, "An Indoor Localization System Based on DTDOA for Different Wireless LAN Systems," in *3rd Workshop on Positioning, Navigation and Communication*, 2006.

[123] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. Ni, "FILA: Fine-grained Indoor Localization," in *INFOCOM, 2012 Proceedings IEEE*, March 2012, pp. 2210–2218.

[124] J. Xiao, K. Wu, Y. Yi, and L. Ni, "FIFS: Fine-Grained Indoor Fingerprinting System," in *Computer Communications and Networks (ICCCN), 2012 21st International Conference on*, July 2012, pp. 1–7.

[125] J. Yang and Y. Chen, "Indoor Localization Using Improved RSS-Based Lateration Methods," in *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, Nov 2009, pp. 1–6.

[126] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: Indoor Localization via Channel Response," *ACM Comput. Surv.*, vol. 46, no. 2, pp. 25:1–25:32, Dec. 2013.

[127] J. yong Yoon, J.-W. Kim, W.-H. Lee, and D.-S. Eom, "A Tdoa-based Localization using Precise Time-Synchronization," in *Advanced Communication Technology (ICACT), 2012 14th International Conference on*, Feb 2012, pp. 1266–1271.

[128] A. Yongacoglu, D. Makrakis, and K. Feher, "Differential Detection of GMSK using Decision Feedback," *Communications, IEEE Transactions on*, vol. 36, no. 6, pp. 641–649, Jun 1988.

[129] M. Youssef and A. Agrawala, "The Horus WLAN Location Determination System," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '05. New York, NY, USA: ACM, 2005, pp. 205–218.

[130] X. Yuan, F. Lian, and C. Han, "Models and Algorithms for Tracking Target with Coordinated Turn Motion," *Mathematical Problems in Engineering*, 2014.

[131] S. Zhou and J. Pollard, "Position Measurement using Bluetooth," *Consumer Electronics, IEEE Transactions on*, vol. 52, no. 2, pp. 555–558, May 2006.

[132] Z. Zhou, Z. Yang, C. Wu, W. Sun, and Y. Liu, "LiFi: Line-Of-Sight Identification with WiFi," in *INFOCOM, 2014 Proceedings IEEE*, April 2014, pp. 2688–2696.

[133] L. Zimmermann, A. Goetz, G. Fischer, and R. Weigel, "GSM Mobile Phone Localization using Time Difference of Arrival and Angle of Arrival Estimation," in *Systems, Signals and Devices (SSD), 2012 9th International Multi-Conference on*, March 2012, pp. 1–7.

[134] D. Zito, D. Pepe, B. Neri, D. De Rossi, A. Lanata, A. Tognetti, and E. Scilingo, "Wearable System-on-a-Chip UWB Radar for Health Care and Its Application to the Safety Improvement of Emergency Operators," in *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, Aug 2007, pp. 2651–2654.

[135] U. Zysset, "Improved Power-based Lateration Via Wall Attenuation Factor," Bachelor's Thesis, University of Bern, Switzerland, 2015.

List of Publications

Refereed Papers (Journals, Conferences, Workshops)

- **Z. Li**, D.B. Acuña, Z. Zhao, J.L. Carrera, and T. Braun, ”Fine-grained Indoor Tracking by Fusing Inertial Sensor and Physical Layer Information in WLANs”, in *IEEE International Conference on Communications (ICC)*, Kuala Lumpur, Malaysia, May 2016.

- **Z. Li**, T. Braun, and D.C. Dimitrova, ”A Passive WiFi Source Localization System based on Fine-grained Power-based Trilateration”, in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Boston, USA, June 2015.

- **Z. Li**, T. Braun, and D.C. Dimitrova, ”A Time-based Passive Source Localization System for Narrow-band Signal”, in *IEEE International Conference on Communications (ICC)*, London, UK, June 2015.

- **Z. Li**, T. Braun, and D.C. Dimitrova, ”Methodology for GPS Synchronization Evaluation with High Accuracy”, in *IEEE 81st Vehicular Technology Conference (VTC)*, Glasgow, Scotland, May 2015.

- **Z. Li**, D.C. Dimitrova, D.H. Raluy and T. Braun, ”TDOA for Narrow-band Signal with Low Sampling Rate and Imperfect Synchronization”, in *7th IFIP Wireless and Mobile Networking (WMNC)*, Coimbra, Portugal, May 2014. (Best Paper Award)

- **Z. Li**, D.C. Dimitrova, and T. Braun, ”Highly accurate evaluation of GPS synchronization for TDOA localization”, in *IFIP Wireless Days (WD)*, Valencia, Spain, November 2013.

- **Z. Li**, A. Hossmann-Picu, and T. Braun, ”Range-based Weighted-likelihood Particle Filter for RSS-based Indoor Tracking”, in *1st KuVS workshop for Expert Talk on Localization*, Lübeck Germany, April 2015.

- **Z. Li**, and T. Braun, ”A Passive Source Localization System for IEEE 802.15.4 Signal”, in *The 2nd Conference on Networked Systems (NetSys) (Demo Session)*, Cottbus, Germany, March 2015.

- **Z. Li**, D.C. Dimitrova, and T. Braun, ”TDOA-Based Localization System with Narrow-band Signals”, in *PhD forum in the 1st Conference on Networked Systems (NetSys)*, Stuttgart, Germany, March 2013.

- A. Hossmann-Picu, **Z. Li**, etc., ”Synergistic User $< -- >$ Context Analytics”, in *7th International Conference covering topics in ICT Innovations*, Ohrid, R. Macedonia, October 2015.

Paper Under Review

- **Z. Li**, and T. Braun, ”Passively Track WiFi Users with an Enhanced Particle Filter using Power-based Ranging”, University of Bern, Technical Report, INF-15-005, 2015.

Declaration of consent

on the basis of Article 28 para. 2 of the RSL05 phil.-nat.

Name/First Name: Li, Zan

Matriculation Number: 11-117-108

Study program: Computer Science

Bachelor ☐ Master ☐ Dissertation ☑

Title of the thesis: Fine-Grained Indoor Positioning and Tracking Systems

Supervisor: Prof. Dr. Torsten Braun

I declare herewith that this thesis is my own work and that I have not used any sources other than those stated. I have indicated the adoption of quotations as well as thoughts taken from other authors as such in the thesis. I am aware that the Senate pursuant to Article 36 para. 1 lit. r of the University Act of 5 September, 1996 is authorised to revoke the title awarded on the basis of this thesis. I allow herewith inspection in this thesis.

Bern, 07 March 2016

Place/Date

Zan Li

Signature