Time Synchronization and Localization in Wireless Networks

by

Khalid Almuzaini
B.Sc., King Saud University, 1998
M.Sc., University of Southern California, 2003

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical and Computer Engineering

Time Synchronization and Localization in Wireless Networks

by

Khalid Almuzaini
B.Sc., King Saud University, 1998
M.Sc., University of Southern California, 2003

Supervisory Committee

Dr. T. Aaron Gulliver, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Lin Cai, Departmental Member
(Department of Electrical and Computer Engineering)

Dr. Andrew Rowe, Outside Member
(Department of Mechanical Engineering)

Supervisory Committee

Dr. T. Aaron Gulliver, Supervisor
(Department of Electrical and Computer Engineering)

Dr. Lin Cai, Departmental Member
(Department of Electrical and Computer Engineering)

Dr. Andrew Rowe, Outside Member
(Department of Mechanical Engineering)

ABSTRACT

Localization is very important for self-organizing wireless networks. The localization process involves two main steps: ranging, i.e., estimating the distance between an unlocalized node and the anchor nodes within its range, and the localization algorithm to compute the location of the unlocalized nodes using the anchor coordinates and the estimated ranges. To be able to estimate the distance, the receiver needs to detect the arrival time of the received signals precisely. Thus, the first part of this research is related to time synchronization.

We propose two new symbol timing offset estimation (STO) algorithms that can detect the start of an orthogonal frequency division multiplexing (OFDM) symbol more accurately than others in a Rayleigh fading channel. OFDM is used to perform timing synchronization because it is incorporated in many current and future wireless systems such as 802.11, WiMAX, wireless USB, and WiMedia. The first

proposed algorithm uses a metric that is calculated recursively. Two estimation methods are considered: one using the average of the metric results, and the other using the median. The second approach uses a preamble designed to have a maximum timing metric for the correct location and very small values otherwise. These algorithms are shown to outperform recent algorithms in the literature.

In the second part of this dissertation we explore the second step of the localization problem. There are two kinds of localization: range-free and range-based. A new distributed range-free localization algorithm is proposed where every unlocalized node forms two sets of anchors. The first set contains one-hop anchors from the unlocalized node. The second set contains two-hop and three-hop anchors away from the unlocalized node. Each unlocalized node uses the intersections between the ranging radii of these anchors to estimate its position.

Four different range-based localization algorithms are proposed. These algorithms use techniques from data mining to process the intersection points between an unlocalized node and nearby anchors. The first proposed scheme is based on decision tree classification to select a group of intersection points. The second is based on the decision tree classification and K-means clustering algorithms applied to the selected intersection points by the decision trees. The third is based on decision tree classification and the density-based spatial clustering of applications with noise (DBSCAN) algorithm applied to the intersection points selected by decision trees. The last approach uses the density-based outlier detection (DBOD) algorithm. DBOD assigns density values to each point being used in the location estimation. The mean of these densities is calculated and those points having a density larger than the mean are kept as candidate points. These proposed approaches are shown to outperform recent algorithms in the literature.

Contents

List of Tables

List of Figures

List of Abbreviations

AOA	Angle of Arrival
CIR	Channel Impulse response
CP-OFDM	Cyclic Prefix Orthogonal Frequency Division Multiplexing
CM	Channel Model
CTF	Channel Transfer Function
CAZAC	Constant Amplitude and Zero Autocorrelation
CFO	Carrier Frequency Offset
DAA	Detect And Avoid
DS-UWB	Direct Sequence UWB
EIRP	Effective Isotropic Radiated Power
ETSI	European Telecommunications Standards Institute
FCC	Federal Communications Commission
GPS	Global Positioning Systems
HDR	High Data Rate
ICI	Inter-Carrier Interference
ISI	Inter-Symbol Interference
IEEE	Institute of Electrical and Electronics Engineers
LDR	Low Data Rate
LOS	Line Of Sight
MCM	Multi-Carrier Modulation
MB-OFDM	Multi Band Orthogonal Frequency Division Multiplexing
MPC	Multipath Component
MPF	Multipath Fading
NLOS	Non Line Of Sight
OFDM	Orthogonal Frequency Division Multiplexing
PDP	Power Delay Profile
PN	Pseudo Noise

PAN	Personal Area Network
SV model	Saleh-Valenzuela model
TOA	Time Of Arrival
TOF	Time Of Flight
UWB	Ultra Wideband
ZP-OFDM	Zero Padding Orthogonal Frequency Division Multiplexing

ACKNOWLEDGEMENTS

My unreserved praises and thankfulness are to Allah, the Most Compassionate, and the Most Merciful. He blessed me with his bounties.

I am heartily thankful to my supervisor, Dr. T. Aaron Gulliver, whose encouragement, guidance and support throughout this challenge enabled me to develop an understanding of the subject. He has also been a great support to me in my academic and personal life. He has patiently listened to what I faced during my years of study at UVic. I am really grateful to him and I cannot find the words to adequately describe what he has done for me. He never let me down at any moment. Instead, he convinced me that I can fight more to reach my goals no matter what obstacles lie in my way. Actually, I disappointed him many times, but he never looked at those moments, and only tried hard to keep me on track as much as he could. I can only say thank you from the bottom of my heart and I know it is not enough at all.

Also I would like to thank my mother and my brothers in Saudi Arabia. They kept calling, supporting me, and believing that I would finish my PhD in spite of the passing of my father, who promised to come to my convocation. Also I would like to thank my wife who is always inspiring, encouraging me in spite of the pain she bears after the loss of her parents in one year. We were there for each other throughout the difficult time of taking care of my little son during the very tough and painful start to his life.

I am grateful to the King Abdulaziz City for Science and Technology (KACST) for trusting me, understanding my situation, sympathizing with the struggles in my personal life, and supporting me by extending my scholarship until I could finish the PhD program. Also I would like to thank my colleagues at KACST for also being my confidants, especially to my friends Hani Alzaid and Faisal Alsalem.

I cannot forget my close friends here at UVic for their valuable and friendly discussion, support, and availability. Thank you Amro Altamimi, Ahamd Abdullah, Abdusslam Amer, Mohammed Almardy, Ahamd Morgan, Bassam Sayed, Mo-

hammed Algamal, Behzad Bahr-Hosseini, Carlos Quiroz-Perez, Abolfazel Ghassemi, and all the people who were real friends. Special thanks to Soltan Alharbi, Khalid Alfarraj, Eiad Algal, and Zabnan Aldossari. Those guys have supported me by being company of the best kind every week and by creating a lovely, friendly, and fun environment every time we met. Thank you all.

Khalid Almuzaini

DEDICATION

To the soul of my father, my mother, my wife, and my children.

Chapter 1

Introduction

Digital modulation is well established in wireless communication systems. It enables the use of advanced signal processing and coding techniques to improve transmission quality. In recent decades, great attention has been paid to improving mobile and indoor wireless networks. In parallel, the ever increasing demand for higher speeds in these networks has caused a move from narrowband to wideband systems [1].

Advanced wireless devices should be capable of locating and tracking a user or wireless device in a network precisely. This capability can enable many applications in wireless networks. These wireless devices should also survive in extreme environments with multipath fading and non-line-of-sight (NLOS) signal paths.

Two important design issues in developing such a system are precise location estimation and a flexible data rate to accommodate varied applications. For example, sending only location information to an access point in the network requires a low date rate. However, sending real-time audio or video from a terminal node to the access point requires a high data rate to achieve sufficient quality of service. Providing accurate location information in wireless networks will be an essential

feature in the future to introduce services such as secure routing, location-based routing and monitoring the coverage area of the network. The coverage area of a wireless network can be determined by knowing the location of each node and its transmission range. This knowledge also helps to deploy additional nodes in uncovered areas if needed, or replace dead nodes by new ones.

Providing accurate location information requires accurate ranging, and accurate ranging requires accurate time synchronization. Thus, the first step in constructing a localization system is accurately determining the arrival time of the signal at the receiver.

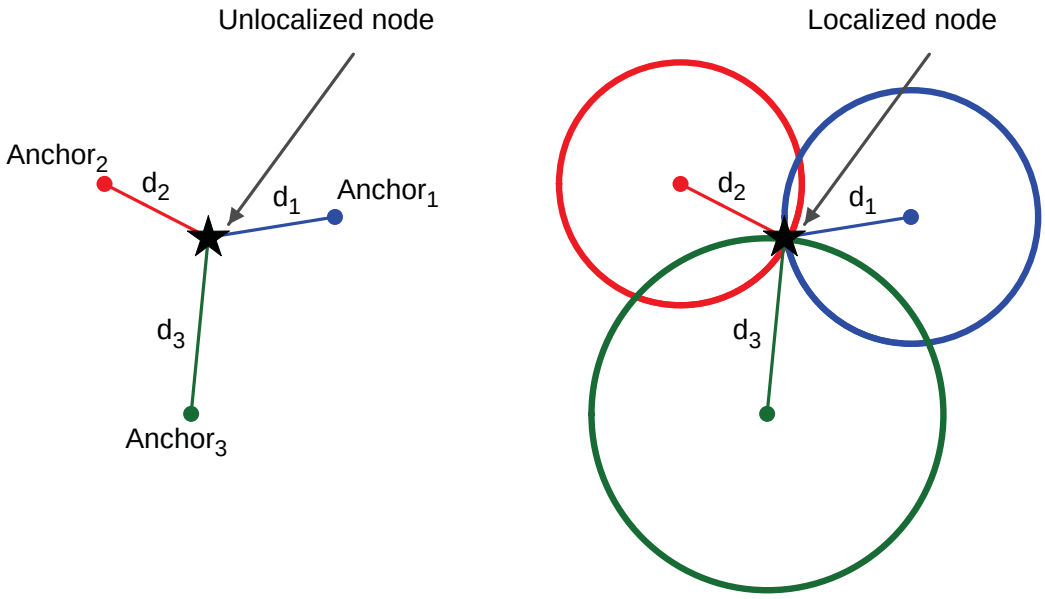
1.1 Time Synchronization and Localization

Time synchronization is the first step in determining the location of wireless nodes. Accurate synchronization leads to accurate distance estimation between the nodes. Localization means computing the coordinates of a node position. Navigation and tracking are two applications of localization. They act like dynamic localization where the localization process is repeated periodically.

The two steps of localization are ranging i.e., measuring the distance between an unlocalized node and anchor nodes, and localizing i.e., computing node coordinates based on distances estimates, as shown in Fig. 1.1. An unlocalized node is one whose location has not yet been determined. An anchor node knows its coordinates, for example they can be obtained via GPS. In Fig. 1.1a, three anchors are shown with one unlocalized node residing within their ranges. d_1, d_2, and d_3 represent the estimated distances between $Anchor_1$, $Anchor_2$, and $Anchor_3$, respectively and the unlocalized node, which is represented by a black star. In Fig. 1.1b, the circles represent the estimated distance for each anchor to the unlocalized node. If

all anchors are able to estimate the distance precisely, the three circles will intersect exactly on the unlocalized node and then the node location is known exactly.

Estimating the distance is required only in range-based localization schemes. In range-free localization the first step is not needed. Proximity sensing is used to determine if a node is in range of other nodes or not. In the literature, there are different names for *anchor* nodes. They are called *landmarks*, *beacons*, *references*, or *locators* nodes. The *unlocalized* nodes are called *agents*, *blindfolded*, *dump*, *unknown*, or *targets* nodes. The terms *anchor* nodes and *unlocalized* nodes are used throughout this dissertation.



(a) Ranging. (b) Localization.

Figure 1.1: The two steps of the localization process.

Localization in wireless networks has seen a lot of research in the last decade. One can ask, why not use the global positioning system (GPS) to locate each node? Implementing a GPS device in each node is a costly solution, especially if there are a large number of wireless nodes. In addition, GPS will not give accurate results

for indoor wireless networks. The goal is to determine the location of every node after the deployment process with the highest possible accuracy and at a low cost.

1.2 Motivation

Wireless networks are currently deployed in urban areas, as well as in indoor environments such as offices and homes. In these environments, the transmitted signal experiences multipath propagation before reaching the receiver. This means that multiple copies of the transmitted signal arrive with different delays and attenuations at the receiver. A signal that is based on a multiple carrier technique can survive very well in a multipath environment. With multiple carriers, the system bandwidth is divided into several parallel subbands. The most common implementation is based on the fast fourier transform (FFT) and is named orthogonal frequency division multiplexing (OFDM) [1]. FFT is an efficient algorithm to compute the discrete Fourier transform (DFT), and a DFT transforms the signal from a time domain representation to a frequency domain representation. In OFDM, a large number of closely-spaced orthogonal subcarriers are used to carry the data. This data is divided into several parallel data streams or channels, one for each subcarrier. Each subcarrier is modulated with a conventional modulation scheme such as quadrature amplitude modulation (QAM) or phase-shift keying (PSK) at a low symbol rate, maintaining the total data rate similar to conventional single-carrier modulation over the same bandwidth.

As mentioned earlier, adding location information for wireless nodes and the ability to locate any new node or wireless device in a wireless network is a great feature that has many applications. There are civil applications like monitoring patients in hospitals and tracking animals, as well as military applications like

tracking soldiers and vehicles in a battlefield.

The use of OFDM signals for time synchronization is based on its use in current standards like WLAN (802.11), WiMAX, wireless USB, and WiMedia and also its promise for future standards. This motivates research into localization using OFDM technology. In this dissertation solutions are developed that can be used in any OFDM-based system. More information about OFDM will be presented in Chapter 2.

1.3 Problem Statement

The first problem is time synchronization in OFDM systems. Fig. 1.2 shows the received OFDM signal via a multipath channel. In an OFDM system, the data bits at the transmitter are first converted to PSK or QAM symbols via mapping. These symbols are divided into blocks of length N, where N is the number of subcarriers in an OFDM symbol. Then each symbol vector $X = [X_0\ X_1\ X_2\ \cdots\ X_{N-1}]^T$ is converted to the corresponding OFDM symbol x via an IFFT to generate the time-domain OFDM signal which is expressed as:

$$x(n) = \sum_{k=0}^{N-1} X_k e^{j2\pi kn/N} \tag{1.1}$$

where X_k is the PSK or QAM symbol. The nth received sample has the form:

$$y(n) = \sum_{m=0}^{L-1} h(m)\, x(n-m) \tag{1.2}$$

where $h(m)$ is the channel impulse response (CIR) with memory or length L. The CIR is given by

$$h(t) = \sum_{l=0}^{L-1} \alpha_l \delta(t - \tau_l) \tag{1.3}$$

where α_l is the amplitude of the lth MPC and the τ_l is the delay of the lth MPC.

At the receiver, we have time and frequency offsets. The time offset is modelled as a delay ε in the received signal, and the frequency offset is modelled as a phase distortion v normalized to the frequency spacing between subcarriers. The received signal with time and frequency offsets plus AWGN noise $w(n)$ with variance σ_w^2 is

$$r(n) = y(n - \varepsilon) e^{j2\pi v n / N} + w(n) \tag{1.4}$$

In this dissertation we focus only on estimating ε, assuming the frequency offset v is estimated and compensated already.

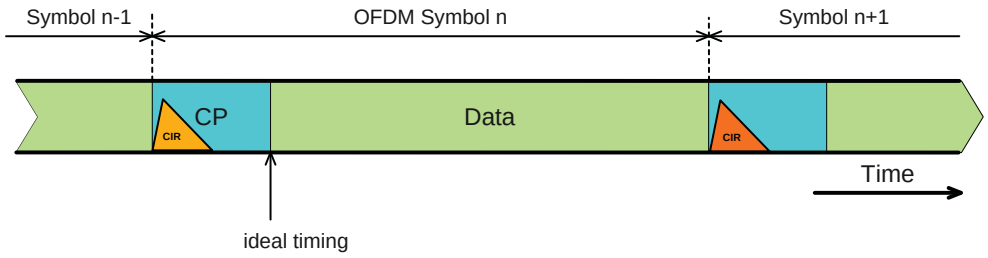


Figure 1.2: Ideal timing at the receiver.

The other problem considered is localization in wireless networks. The nodes in the network are divided into anchors that know their locations and unlocalized nodes that must determine their locations by communicating with anchors within range. Each unlocalized node in the wireless network calculates its location based on the estimated distances to at least three anchors within its range in the case of range-based schemes, or just knowing the maximum possible range of anchors in the case of range-free schemes. An unlocalized node must be in range of at least three anchors to be able to estimate the distance. The location of an unlocalized

node is given by $p\,(x,y)$, the anchors within its range are given by $a_i\,(x_i,y_i)$, and the estimated distances between the unlocalized node and each anchor is d_i as shown in Fig. 1.3. The goal is to estimate the location of the unlocalized node $p\,(\hat{x},\hat{y})$ based on the given information, which in this case is the three distance estimates and anchors locations.

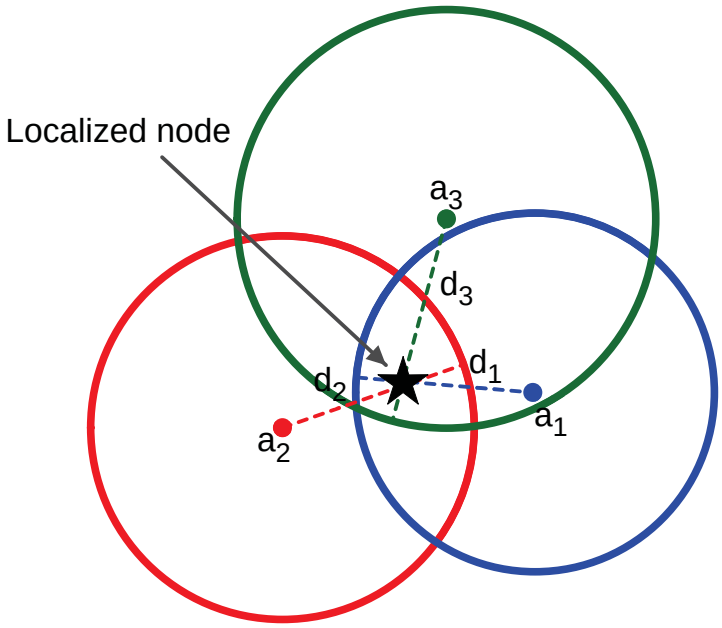


Figure 1.3: An illustration of the localization problem.

1.4 Contributions

We propose two algorithms to solve the OFDM time synchronization problem and five new algorithms to solve the localization problem. The first OFDM timing algorithm is based on an enhanced correlation method with a new metric. The proposed algorithm outperforms a recent one presented in the literature. The second approach uses a preamble designed to have a maximum timing metric for the correct location and very small values otherwise.

A distributed range-free localization algorithm that is based on proximity in-

formation from multi-hop anchor nodes is proposed. This algorithm outperforms a recently published algorithm. Finally, five different distributed range-based localization algorithms are proposed, namely variance-based localization using decision trees (VBLDT), distance-based localization using decision tree (DBLDT), K-means-based localization (LKmeans), localization based on density-based spatial clustering of applications with noise (LDBSCAN), and finally a density-based outlier technique localization algorithm (LDBOD). These algorithms were inspired by results in the data mining field. Data mining is a rich field with algorithms that can be used to solve many problems [2].

1.5 Outline

Chapter 2 provides an introduction to multicarrier modulation and its advantages and disadvantages. Current OFDM technologies and standards that use OFDM are presented. The time synchronization problem is presented by reviewing some current time synchronizing methods. The proposed algorithms are also introduced. Performance results are given for different wireless channels.

Chapter 3 provides a general introduction to the localization problem. It starts with a review of different ranging techniques and localization algorithms. Sources of ranging errors are discussed and localization applications are reviewed to motivate the problem.

Chapter 4 considers range-free localization. Several known algorithms are reviewed and the proposed distributed range-free localization scheme is presented with different performance measures. Performance results are presented to compare the proposed algorithm with some methods in the literature.

Chapter 5 introduces a new approach to solving the problem of localization. A

number of classification and clustering algorithms from data mining are combined together to accurately estimate the location of different nodes in the network. Performance results and comparisons with recent schemes are given.

Finally in **Chapter 6**, conclusions are presented along with future directions.

Chapter 2

Time Synchronization

Multicarrier technology in the form of orthogonal frequency-division multiplexing (OFDM) is widely recognized as one of the most promising schemes for next generation wireless networks [3]. This technique has already been adopted in many applications, including terrestrial digital video broadcasting (DVB-T) and some commercial wireless LANs. OFDM is used in the European digital broadcast radio system, as well as in wired systems such as asymmetric digital subscriber-lines (ADSL).

2.1 OFDM Basics

Frequency division multiplexing (FDM) divides the channel bandwidth into subchannels and transmits multiple relatively low rate signals by transmitting each signal on a separate carrier frequency. A guard band is left between the subcarriers to ensure that the signal of one subcarrier does not overlap with the signal of an adjacent subcarrier. However, the bandwidth is not used efficiently. In order to solve the bandwidth efficiency problem, orthogonal frequency division multiplexing was proposed. In this case, the different carriers are orthogonal to each other.

Each narrowband subchannel experiences almost flat fading as shown in Fig. 2.1.

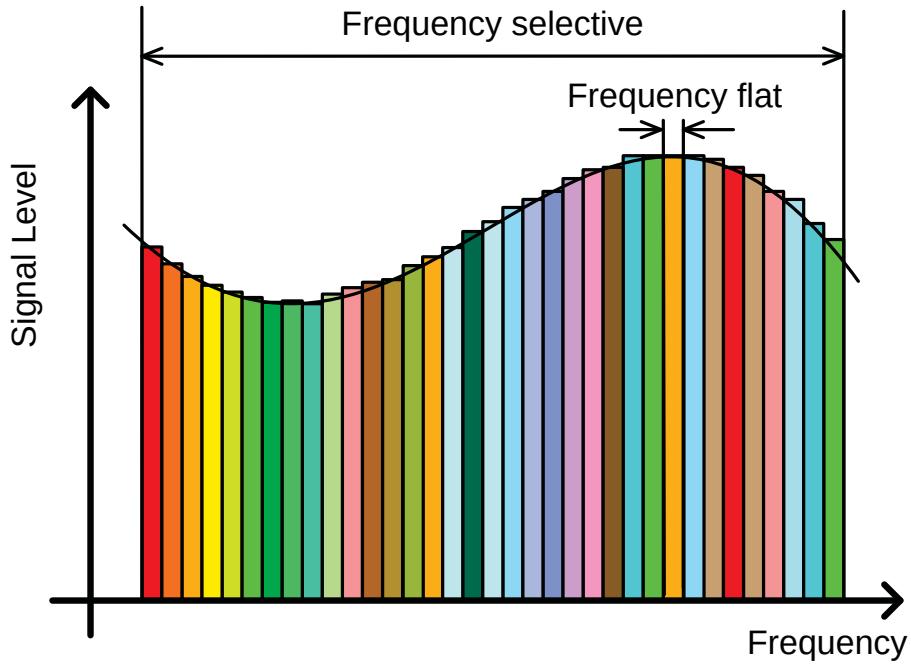


Figure 2.1: An illustration of frequency selective and flat channel.

The orthogonality property is apparent from Fig. 2.2, where at the the peak of one subcarrier all other carriers have zero amplitude [1]. OFDM is an efficient broadband multicarrier modulation method which offers superior performance and benefits over older, more traditional single carrier modulation methods. In single carrier communications, the data is modulated onto a single carrier frequency where the available bandwidth is occupied by each symbol. This can lead to inter-symbol-interference (ISI) between adjacent symbols in the case of a frequency selective channel. A spectrum example with three OFDM subcarriers is shown in Fig. 2.2. This shows that the spectra are partly overlapping, significantly increasing the spectral efficiency as compared to conventional FDM multicarriers systems. Fig. 2.3 shows the time and frequency representation of OFDM symbols

with guard interval. The same OFDM signal in Fig. 2.2 is depicted in the time-domain in Fig. 2.4 with the guard interval represented by a cyclic prefix that is just a copy of the end of the symbol appended to the front. The guard interval and its importance are explained in Section 2.2.

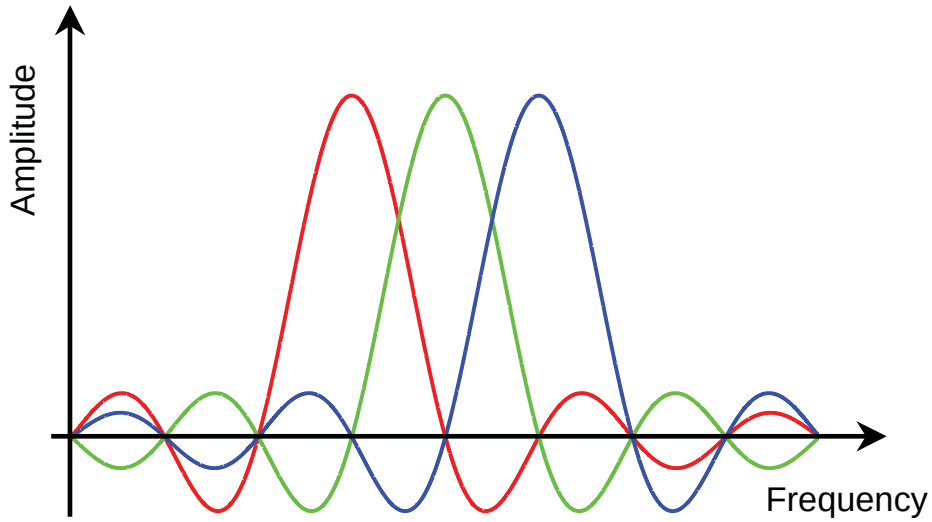


Figure 2.2: Spectrum of OFDM.

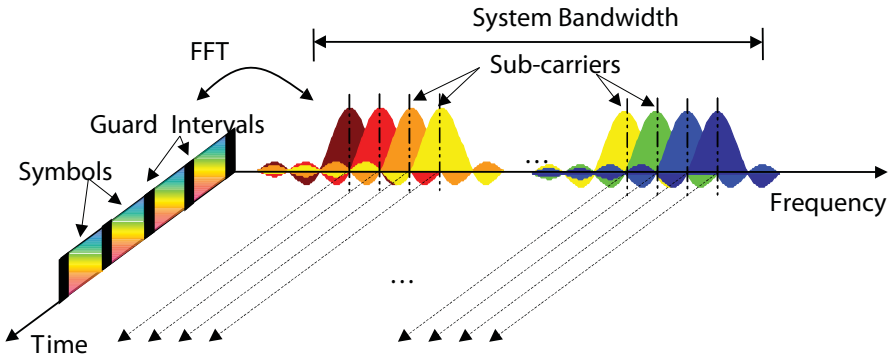


Figure 2.3: Time and frequency representation of OFDM symbols with a guard interval.

The basic OFDM processing at the transmitter and receiver is summarized in the block diagram of Fig. 2.5. First the bits are modulated and then sent to a

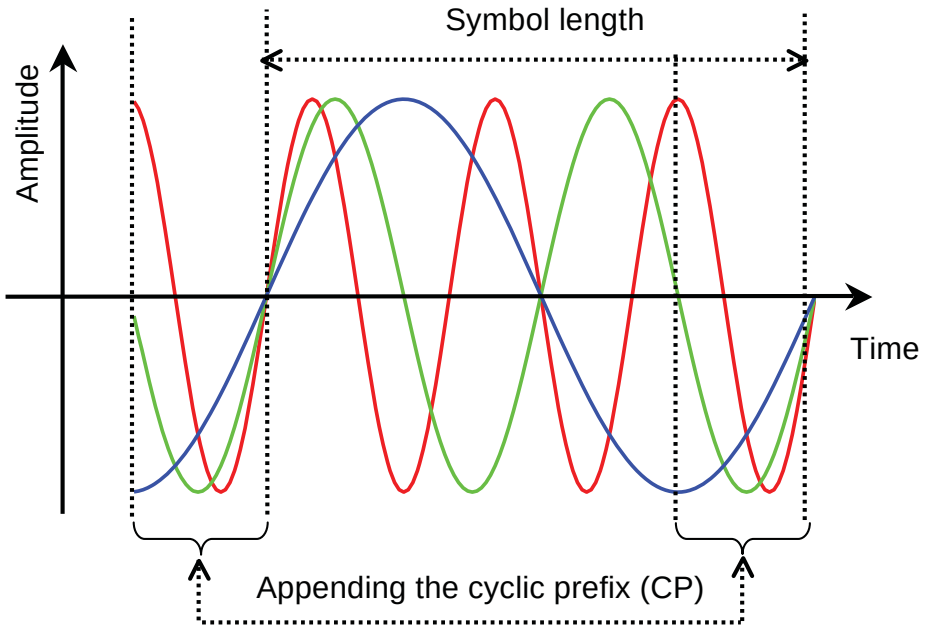


Figure 2.4: OFDM in the time domain with a cyclic prefix (CP).

serial to parallel block. Next the IFFT block converts the signal to the time domain. The resulting signal is converted to a serial stream, a guard interval is appended, converted to analogue, and finally sent through the channel. At the receiver, the same process is reversed.

2.2 Guard Interval

Two main signal degradations in OFDM systems are inter symbol interference (ISI) and inter carrier interference (ICI). Inter symbol interference (ISI) occurs when energy from one symbol spills over to the next symbol. This is usually caused by time dispersion in multipath channels when reflections of the previous symbol interfere with the current symbol. Inter carrier interference (ICI) occurs when the subcarriers lose their orthogonality, causing them to interfere with each other. This can

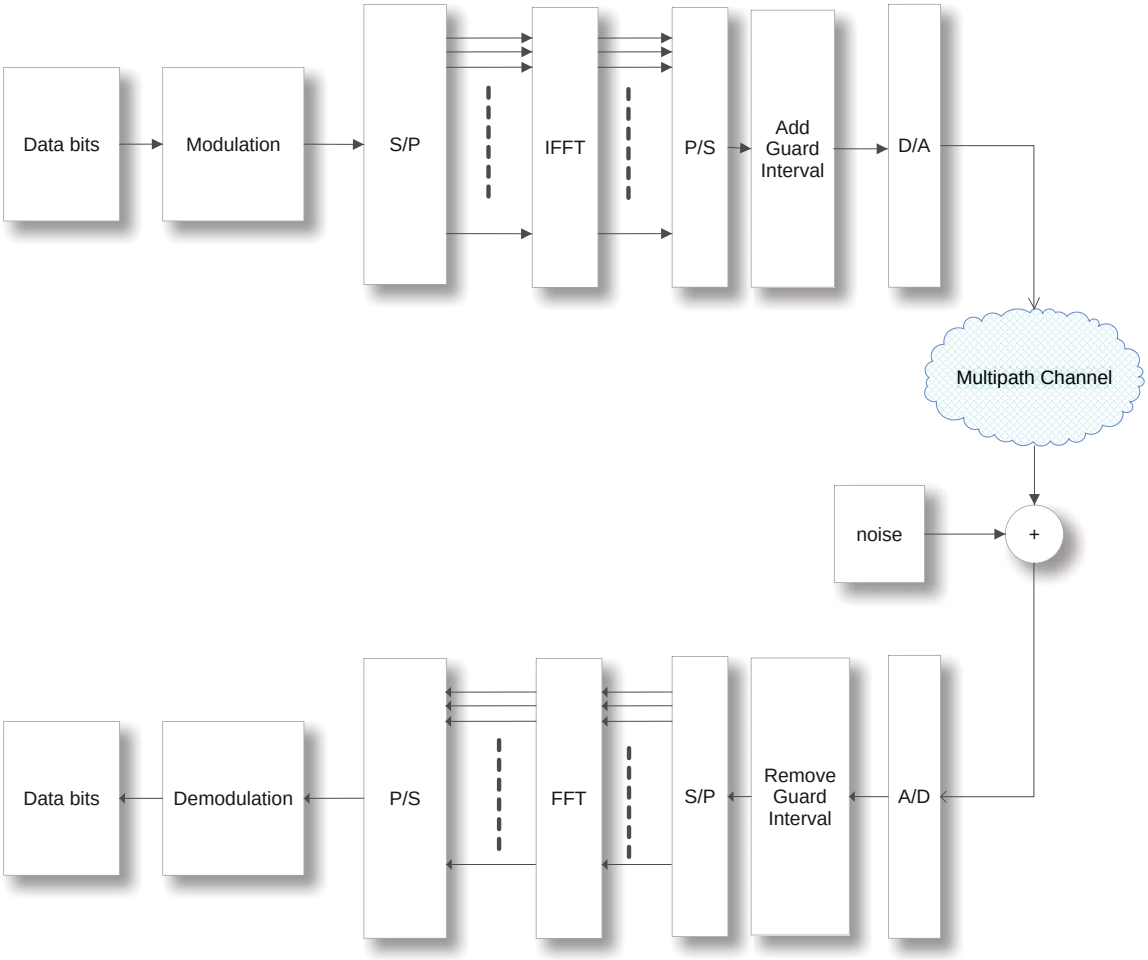


Figure 2.5: OFDM system block diagram.

arise due to doppler shifts and frequency and phase offsets.

The guard interval (GI) acts as a guard space between successive OFDM symbols and, therefore, limits ISI, as long as the length of the GI is longer than the channel impulse response (CIR). The GI ensures orthogonality between the subcarriers by keeping the OFDM symbol periodic over the extended symbol duration, and therefore avoiding inter-carrier interference (ICI). Traditionally, the GI is the copy of the tail of each IFFT block, called the cyclic prefix (CP). Recently, zero-padding (ZP) has been proposed to replace the generally non-zero GI [4] [5]. Specifically, in each block of the so-called ZP-OFDM transmission, zero symbols with a length

greater than CIR are appended after the IFFT. If the number of zero symbols equals the CP length, then ZP-OFDM and CP-OFDM transmissions have the same spectral efficiency [4]. However, the receiver complexity is increased in the case of ZP-OFDM. Conversely, ZP-OFDM can perform well, even in the presence of deep frequency fading [6].

2.3 Wireless Systems using OFDM

The application of OFDM for wireless local-area-networks (WLANs) was first standardised in 1999 as IEEE 802.11a [7], often referred to as Wi-Fi, where it was used in the 5 GHz frequency band. A similar system was standardized within ETSI under the name of HiperLAN/2. Both systems specify a system with 20 MHz bandwidth, 64 subcarriers and a GI length of 16 samples (800 ns).

Recently, the application of OFDM has been proposed for ultra wideband (UWB) communications [8] under the IEEE 802.15.3a PAN framework. This proposal employs a signal bandwidth of 528 MHz, which is divided into 128 subcarriers, 100 of which are used for data transmission. The system is based on multiband OFDM, where consecutive symbols are transmitted in different frequency bands. Initial deployment is foreseen in the 3.1-4.9 GHz band, but extensions to bands up to 10 GHz are envisioned for the future. The proposal is based on QPSK modulation and a data rate varying from 53.3 Mbps up to 480 Mbps. In parallel to the standardisation by IEEE, a similar proposal was accepted in May 2005 as the wireless USB [9] specification. Vendors are now starting to deliver products based on this standard [1].

The use of OFDM has also been standardized for outdoor networks, for example under the IEEE 802.16 framework. This effort is focused on wireless metropolitan-

area-networks (WMANs). These systems [10] are often collectively referred to as WiMax. They operate in the 2-11 GHz band, providing speeds up to 75 Mbps. The bandwidth is flexible, varying from 1.5 - 20 MHz. The system provides OFDM and OFDMA modes with 256 and 2048 subcarriers respectively [1]. OFDMA is OFDM with multiple access to serve multiple users. It is covered briefly in Section 2.4.

Recently, an extension to the 802.16 standard was proposed for applications to mobile networks. This extension is proposed for the frequency bands below 6 GHz and employs a scalable OFDMA design. The bandwidth is flexible, but, at a bandwidth of 5 MHz, a maximum data rate of 15 Mbps can be achieved. This design is being standardized as IEEE 802.16e and a version is currently being deployed in Korea as Wireless Broadband (WiBro) [1].

2.4 OFDM Properties and Main Advantages

A frequency selective channel occurs when the transmitted signal experiences a multipath environment. Under such conditions, a given received symbol can be potentially corrupted by a number of previous symbols. This effect is commonly known as inter symbol interference (ISI), as mentioned before. To avoid such interference, the symbol duration, T_{symbol}, should be much larger than the delay spread τ_m (maximum amount of time between the first and last multipath signal at the receiver). This leads to poor efficiency in terms of transmission rate in the case of single carrier systems. In an OFDM system, N data symbols are transmitted on N different subcarriers to overcome the effects of frequency selective channels. In this way, the transmission rate remains the same but the system is now more robust to ISI.

The performance of OFDM in frequency selective fading can be improved by

the use of channel coding, yielding coded OFDM (COFDM). Here, the bits are encoded in codewords, which are spread over the different subcarriers using an interleaver. Since the codewords are spread over different carriers, the probability that an entire codeword is received on channels with a high bit error rate will be low, thus the resulting probability of error is also low [1].

Adaptive modulation can be used to improve the performance of OFDM systems in frequency selective fading. With adaptive modulation, the estimate of the channel state is used at the transmitter to determine the modulation [1]. This channel estimate relates to the SNR experienced at the receiver. In adaptive modulation, the subcarriers with high SNRs are assigned symbols of higher order modulation, and subcarriers with low SNRs are assigned symbols from a lower order modulation or even no symbols. The use of adaptive modulation is illustrated in Fig. 2.6. Here poor subcarriers are assigned no bits, subcarriers with moderate channels carry 1 bit/subcarrier (e.g., BPSK modulation), and subcarriers with the best channels are assigned 2 bits/subcarrier (e.g., QPSK modulation). In the case of conventional OFDM, all subcarriers would be assigned the same modulation.

OFDM can also be used as a multiple access technique. Orthogonal frequency division multiple access (OFDMA) allows multiple users to communicate simultaneously [1]. In OFDMA, the subcarriers are subdivided between the different users. The use of OFDMA is illustrated in Fig. 2.7, where the left block of carriers is assigned to user 1, the middle block to user 2 and the right block to user 3. Although the assignment is applied in blocks of carriers here, other patterns can be employed. For example, individual subcarriers throughout the spectrum can be assigned to a user, providing channel diversity for the different users [1].

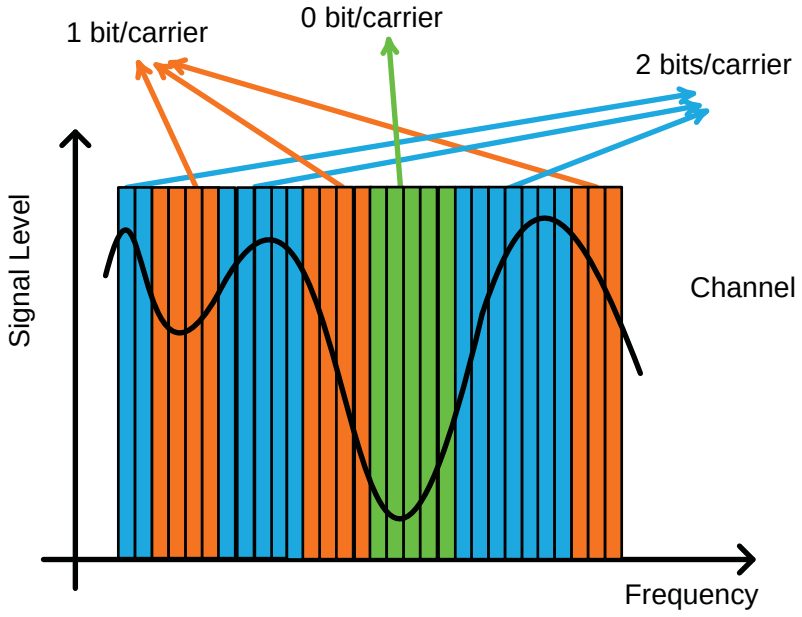


Figure 2.6: An illustration of adaptive modulation.

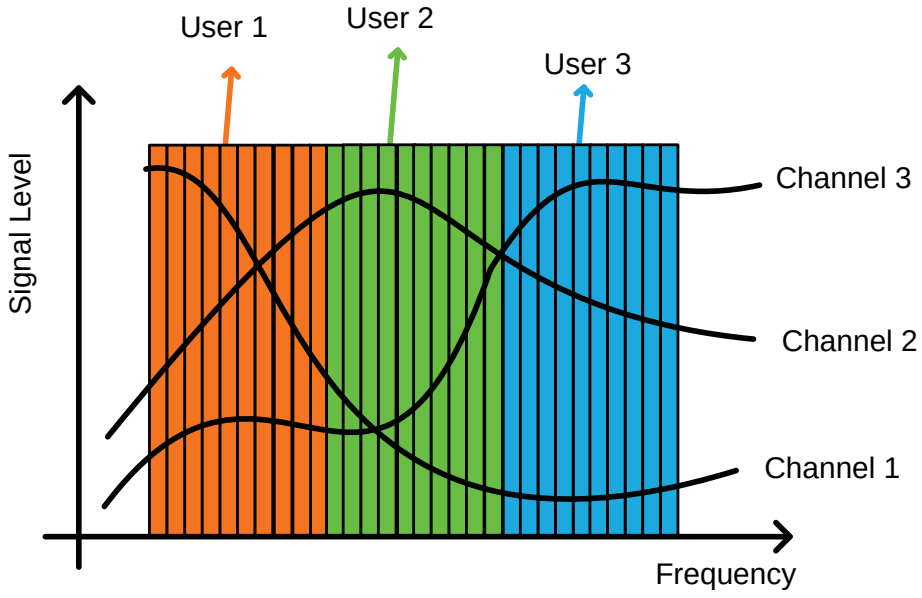


Figure 2.7: An illustration of the OFDMA technique.

2.5 Symbol Timing Offset (STO)

The goal in timing offset estimation is to find the start of the OFDM symbol, as shown in Fig. 2.8. Depending on the location of the estimated starting point of the OFDM symbol, the effect of STO might be different. Fig. 2.8 shows four different cases of timing offset, in which the estimated starting point is too early, little early, exact, or little later than the exact timing instance. In the first case, a negative timing error occurs with ISI. This is the case when the starting point of the OFDM symbol is estimated as prior to the end of the (lagged) channel response of the previous OFDM symbol, and thus, the symbol timing is too early to avoid the ISI. In this case, the orthogonality among subcarriers is destroyed by the ISI (from the previous symbol), and furthermore, ICI occurs.

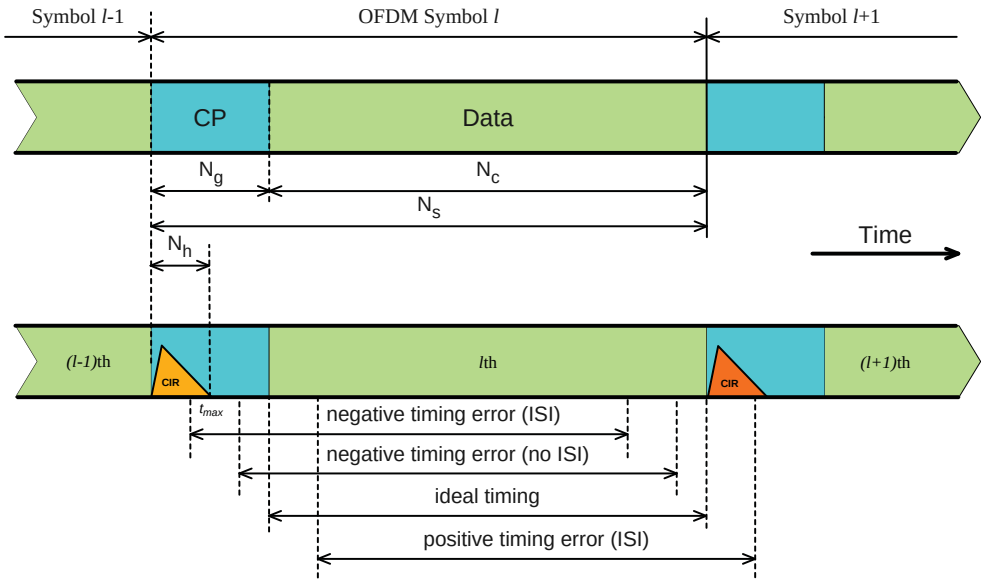


Figure 2.8: Timing errors in receiving OFDM symbols.

In the second case, a negative timing error occurs without ISI. This is the case when the estimated starting point of the OFDM symbol is before the exact point, yet after the end of the (lagged) channel response of the previous OFDM symbol. In this case, the lth symbol is not overlapped with the $(l-1)$th OFDM symbol,

that is, no ISI occurs.

In the third case, the estimated starting point of an OFDM symbol coincides with the exact timing, preserving the orthogonality of the subcarriers. In this case, the OFDM symbol can be perfectly recovered without any interference.

The final case is positive timing error with ISI. This occurs when the estimated starting point of the OFDM symbol is after the exact point, which means the symbol timing is a little later than the exact one. In this case, the signal within the FFT interval at the receiver consists of part of the current OFDM symbol and part of the next one.

2.5.1 Pilot-based STO Estimation

Pilots, such as pseudo-random sequences or null symbols, can be used to determine the start of an OFDM symbol. They are particularly useful for systems with low SNR, where synchronization might otherwise be difficult. Some single-carrier timing synchronization methods can be extended to OFDM systems [11].

The pilot symbols can be OFDM-based or non-OFDM-based. If OFDM-based pilot symbols are used, the frame size may need to be increased to lower the overhead. Otherwise, for shorter frames, non-OFDM-based pilot synchronization bursts may be more appropriate [12].

Another issue to consider when designing pilots for a system is the continuity and burstiness of the data. If the data is continuous, a null signal can be used to signal the start of a symbol. This is the method employed by the European DAB standard. If the data is bursty, a null signal is inappropriate [12].

2.5.2 Non-Pilot-based STO Estimation

Most non-pilot-based methods for timing offset estimation are based on the redundancy of the cyclic prefix. Even though many of these non-pilot-based methods are derived for an AWGN channel, with minor modification, they can also be used for dispersive channels [12].

Many algorithms use the periodicity of the correlation function of the time-domain OFDM symbol [13], [14], [15]. This periodicity can be exploited to find the start of an OFDM signal. How to use this periodicity varies from method to method. Some methods look only at the correlation function while others include the relative power of the samples as well [12].

There are three aspects of OFDM synchronization, namely subcarrier frequency offset (CFO) estimation, receiver sampling frequency offset (SFO) estimation and symbol timing offset (STO) estimation. The first one can be termed as frequency error and the last two can be grouped as timing error [16]. In this dissertation, only STO is studied.

Symbol timing relates to the problem of detecting the start of a symbol. The requirement for this is somewhat relaxed when a CP is used. The symbol timing result defines the FFT window; i.e. the set of samples used in the FFT calculation for each received OFDM symbol [16].

2.6 Proposed STO Estimation for ZP-OFDM Systems

As mentioned previously, synchronization is an essential component of any communication system. In [17], the authors use the received signal power to determine the symbol starting point. In [18], two symbols are employed to detect the OFDM symbol by doing cross-correlation, and then autocorrelation between the cross-correlated symbols. Um *et al.* [19] use a special preamble with three identical components. The authors of [17] [18] [19] synchronize the symbol using the strongest multipath component whether it is the first one or not. In [20], Li *et al.* use two sliding windows for synchronization by exploiting the energy levels in both windows. However, the mean square error (MSE) of their solution is very high compared to other approaches. In ranging applications, employing the strongest multipath component may result in a large error, especially in non-line-of-sight (NLOS) environments where the first multipath component (MPC) is not the strongest one.

In OFDM systems, inter-symbol interference (ISI) can be avoided by inserting a cyclic prefix (CP) in each OFDM symbol. This CP is just a copy of the end of the OFDM symbol. There is an alternative method to reducing ISI called zero padding, in which zeros are simply inserted at the end of each OFDM symbol. Zero padding has been proposed for multi-band OFDM ultra wideband (MB-OFDM-UWB) systems, and so it is employed here.

In an OFDM system, the data at the transmitter are first mapped to PSK or QAM symbols. These symbols are divided into blocks of length N, where N is the number of subcarriers in an OFDM symbol. Then an inverse fast Fourier transform (IFFT) is used to convert each symbol vector $X = [X_0\ X_1\ X_2\ \cdots\ X_{N-1}]^T$ into a time-

domain OFDM signal

$$x(n) = \sum_{k=0}^{N-1} X_k e^{j2\pi kn/N} \tag{2.1}$$

The nth sample of the received signal with perfect timing and frequency synchronization is given by

$$y(n) = \sum_{m=0}^{L-1} h(m)\, x(n-m) + w(n) \tag{2.2}$$

where $h(m)$ is the channel impulse response of length L. It is assumed that L is not longer than the cyclic prefix. $w(n)$ is additive white Gaussian noise (AWGN) with variance σ_w^2. At the receiver, there are typically timing and frequency offsets. The timing offset is a delay ε in the received signal.

2.6.1 Timing Estimators

In this section, four different timing estimators are presented. They will be compared with our proposed estimation technique.

In the Schmidl *et al.* method [21], a training symbol is employed as a preamble. It has two identical halves in the time domain given by

$$S_{sch} = [A_{N/2}\ A_{N/2}] \tag{2.3}$$

where $A_{N/2}$ represents a sequence of length $N/2$. The estimate of the starting point of the symbol is at the maximum of the following timing metric

$$M_{sch}(d) = \frac{|P_{sch}(d)|^2}{(R_{sch}(d))^2} \tag{2.4}$$

where d is the time index corresponding to the first sample in a window of length

N with

$$P_{sch}(d) = \sum_{m=0}^{N/2-1} r^*(d+m) \cdot r\,(d+m+N/2) \tag{2.5}$$

and

$$R_{sch}(d) = \sum_{m=0}^{N/2-1} |r\,(d+m+N/2)|^2 \tag{2.6}$$

In the Minn *et al.* method [22], the training symbol has four parts given by

$$S_{minn} = [B_{N/4}\, B_{N/4}\, -B_{N/4}\, -B_{N/4}] \tag{2.7}$$

where $B_{N/4}$ is a sequence of length $L = N/4$. They use (2.4), but with the following operands

$$P_{minn}(d) = \sum_{k=0}^{1} \sum_{m=0}^{N/4-1} r^*(d+kN/2+m) \cdot r(d+kN/2+m+N/4) \tag{2.8}$$

and

$$R_{minn}(d) = \sum_{k=0}^{1} \sum_{m=0}^{N/4-1} |r(d+kN/2+m+N/4)|^2 \tag{2.9}$$

where d is the time index corresponding to the first sample in a window of length N.

In the Park *et al.* method [23], the following training symbol structure is employed

$$S_{park} = [A_{N/4}\, B_{N/4}\, A^*_{N/4}\, B^*_{N/4}] \tag{2.10}$$

where $A_{N/4}$ is a sequence of length $N/4$, and $B_{N/4}$ is symmetric with $A_{N/4}$. $A^*_{N/4}$ and $B^*_{N/4}$ are the conjugates of $A_{N/4}$ and $B_{N/4}$ respectively. They use (2.4) with the following operands

$$P_{park}(d) = \sum_{k=0}^{N/2} r(d-k)\,.\,r(d+k) \tag{2.11}$$

and

$$R_{park}(d) = \sum_{k=0}^{N/2} |r(d+k)|^2 \tag{2.12}$$

where d is the time index corresponding to the first sample in a window of length N. This metric improves the timing peak compared to the other metrics.

Choi *et al.* [24] proposed the following preamble using zero padding instead of a cyclic prefix

$$S_{choi} = [C_{N/2}\, D^*_{N/2}] \tag{2.13}$$

where $C_{N/2}$ is a sequence of length $N/2$ sequence, and $D^*_{N/2}$ is the complex conjugate of $D_{N/2}$, which is a time reversed version of $C_{N/2}$. The operands in (2.4) are

$$P_{choi}(d) = \sum_{k=0}^{N/2-1} r(d-k) \cdot r(d+k+1) \tag{2.14}$$

and

$$R_{choi}(d) = \frac{1}{2} \sum_{k=0}^{N-1} |r(d+k-N/2|^2 \tag{2.15}$$

2.6.2 Proposed STO Estimation Algorithm

Two problems in symbol synchronization are noise and multipath fading. Using the correlation of the received signal with the conjugate of the training symbol can give good results at low SNRs in the presence of only noise. However, with multipath fading, the first signal component may not be the dominant one. Thus, one cannot just pick the maximum metric value as the timing estimate. In fact, picking the maximum can result in significant timing errors in a rich multipath environment. This is a common problem with the techniques in [21, 22, 23]. In [24] a threshold is employed. However, at low SNRs, determining a suitable threshold η is very difficult [24]. In our proposed algorithm, no threshold is required, and we do not assume the first multipath component is the dominant one. Thus, our

algorithm is more practical than previous techniques, and provides precise timing results even at low SNR values.

In low SNR environments, using a stored reference for synchronization can provide more accurate timing. Our approach to synchronization employs the following concepts

- Symbol boundary detection using an enhanced correlator.
- A new metric to determine the first signal component.
- Recursively using this metric to refine the estimate.

To detect the start of the OFDM symbol, a Zadoff-Chu constant amplitude and zero autocorrelation symbol (CAZAC) is used [25], defined as

$$s_{\gamma}^{(k)} = \begin{cases} e^{j2\pi k\left(q\gamma+\gamma^2/2\right)/\Gamma} & \Gamma \text{ even} \\ e^{j2\pi k(q\gamma+\gamma(\gamma+1)/2)/\Gamma} & \Gamma \text{ odd} \end{cases} \quad 0 \leq \gamma \leq N-1 \tag{2.16}$$

where q is any integer, and k is an integer relatively prime to Γ, which is the length of the preamble. Zadoff-Chu sequences have a low peak-to-average power ratio (PAPR) of 2 dB [25]. In this dissertation, we consider $q = 1$, $k = 1$, and $\Gamma = N$.

In our timing estimator, at the receiver we correlate the received signal r with a stored reference that is the conjugate of s. The conventional correlator is given by

$$P_{conv}(d) = \left| \sum_{i=0}^{N-1} s_i^* \cdot r(d+i) \right| \tag{2.17}$$

and its performance is shown in Fig. 2.9 for $N = 128$, SNR = 30 dB and an AWGN channel. To improve this correlator, we multiply the conventional correlator values

at each time index d by two received values $N-1$ samples apart giving

$$P_{enhanced}(d) = \left| \left(\sum_{i=0}^{N-1} s_i^* \cdot r(d+i) \right) \cdot r(d) \cdot r(d+N-1) \right| \tag{2.18}$$

This is called the enhanced correlator. When the sequences are aligned in time, the two values $N-1$ samples apart are not noise only values, hence the performance is improved. Fig. 2.10 presents the performance with the enhanced correlator, and shows that it provides a better estimate than the conventional correlator. In a mul-

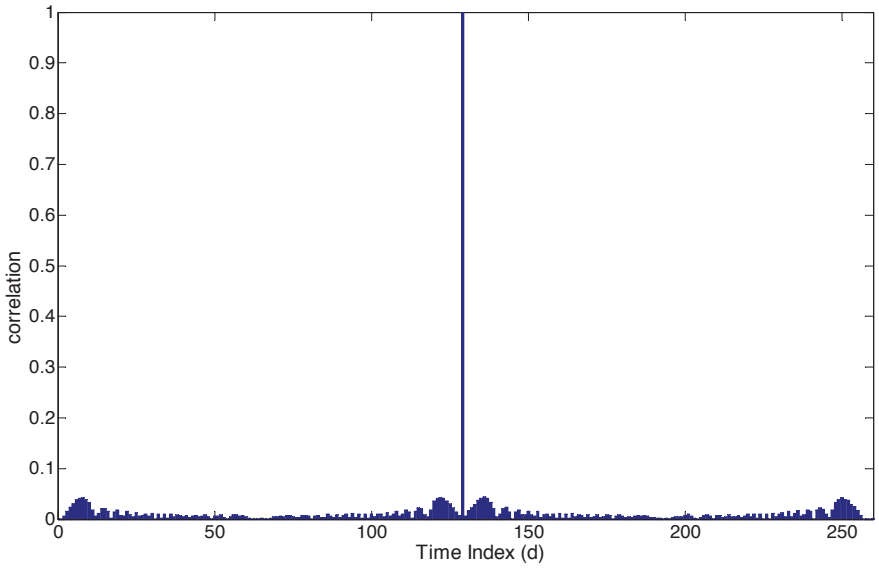


Figure 2.9: Conventional correlator output.

tipath fading environment, the correlator will provide several peaks depending on the number of MPCs. In an inter-symbol interference (ISI) free channel where the zero padding interval is greater than channel impulse response, the difference in time indices between the first MPC and the last MPC does not exceed the length of the zero padding interval. The correct synchronization point or the leading edge of an OFDM symbol is the first correlator peak.

The proposed algorithm searches a region of length equal to the channel im-

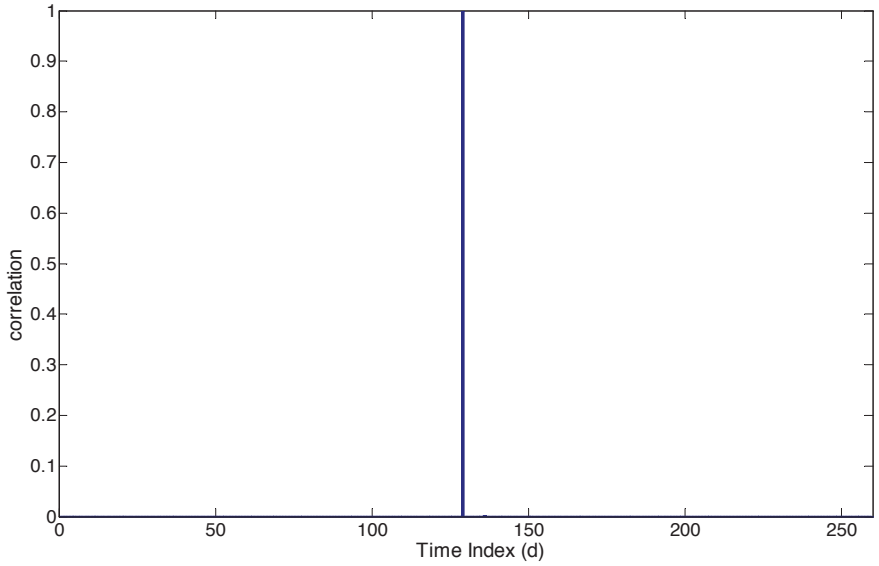


Figure 2.10: Enhanced correlator output.

pulse response length (CIR). The end of the search region is defined as where the maximum of the correlator output occurs, and is denoted as d^1_{max} given by

$$d^1_{max} = \underset{d}{\text{argmax}}\, P_{enhance}\,(d) \tag{2.19}$$

The feasible region is defined as the region before d^1_{max} and is equal to CIR length. In our algorithm, the recursive process ends when the maximum component goes outside the feasible region.

The new metric works recursively, starting from the values of $P_{enhanced}$. The first solution is d^1_{max}. Let the length of the feasible region be FR. The first iteration is given by

$$M^1\,(d) = P_{enhanced}\,(d) \left(\sum_{i=d}^{d+ZP-1} P_{enhanced}\,(i) \right)^2 \tag{2.20}$$

where $d \in [d^1_{max} - FR, d^1_{max}]$. The new value of d_{max} is

$$d^2_{max} = \underset{d}{\text{argmax}}\, M^1\,(d) \tag{2.21}$$

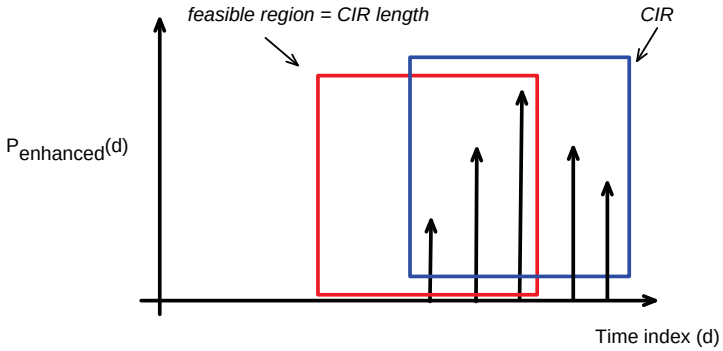
If d^2_{max} lies outside the feasible region, the algorithm stops, and the previous d_{max} is the synchronization estimate. Otherwise the algorithm continues with the second iteration using $M^1(d)$ as follows

$$M^2(d) = M^1(d) \left(\sum_{i=d}^{d+ZP-1} M^1(i) \right)^2 \tag{2.22}$$

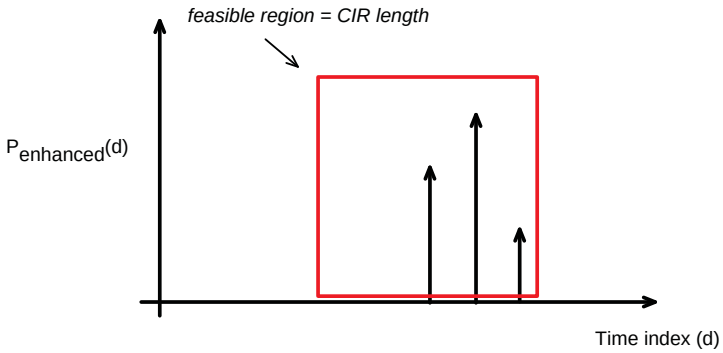
where $d \in [d^1_{max} - FR, d^2_{max}]$. Then d^3_{max} calculated. This process continues until d^j_{max} is outside the feasible region. The synchronization estimate is the last d_{max} inside the feasible region. Note that in the above process, d_{max} shifts to the left.

The OFDM symbol synchronization algorithm is shown in Fig. 2.11 and is summarized as follows:

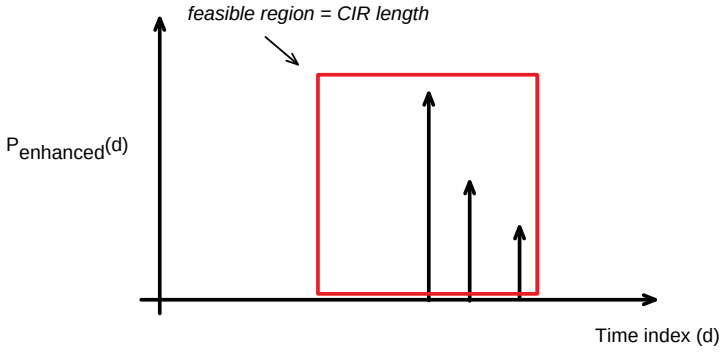
1. Correlate the received signal with the stored reference at the receiver.
2. Find the maximum correlator output.
3. The outputs are normalized by the largest signal component, and used as the initial values for the proposed metric.
4. Find the maximum of the metric, and normalize the metric values before starting the next iteration.
5. Recursively apply the metric until the maximum is outside the feasible region.
6. The OFDM synchronization estimate is given by the last maximum value before leaving the feasible region.



(a) Picking the dominant MPC.



(b) Intermediate step of applying the proposed metric.



(c) Picking the first MPC.

Figure 2.11: The steps of proposed ZP-OFDM synchronization algorithm.

2.6.3 Performance Results

To determine the performance of the new metric, we considered OFDM symbols of length $N = 128$ in two different UWB channels, CM1 and CM4 [26]. All channels are in discrete bins where the bin width depends on the application. This width can vary from one nanoseconds to a few milliseconds depending on the system. Also, in all simulations, a zero mean Gaussian noise is assumed which combines thermal noise and interference. Tables 2.1 and 2.2 shows the relative error for channels CM1 and CM4, respectively. Figs. 2.12 and 2.13 show the corresponding average MSE values. The MSE is the timing mean square error with units sample^2, as in [23, 24]. These figures show that as the SNR increases, the performance improves. The SNR is per data symbol. The number of MPCs also has a significant effect on performance.

Table 2.1: Percentage of errors in bins with channel CM1, $N = 128$

	Number of MPCs			
SNR (dB)	1	5	10	15
0	0.5437	4.6602	4.1086	4.5367
10	0	0.2734	0.3711	0.6328
20	0	0.0602	0.1133	0.2891
30	0	0.0195	0.0719	0.2125

Table 2.2: Percentage of errors in bins with channel CM4, $N = 128$

	Number of MPCs			
SNR (dB)	1	5	10	15
0	0.6969	3.7508	4.8914	4.9562
10	0	0.2820	0.4406	0.8156
20	0	0.0906	0.1359	0.5352
30	0	0.0219	0.0844	0.4773

The median of several synchronization outputs was also considered as a timing metric. Each symbol had the same number of MPC but with different fading.

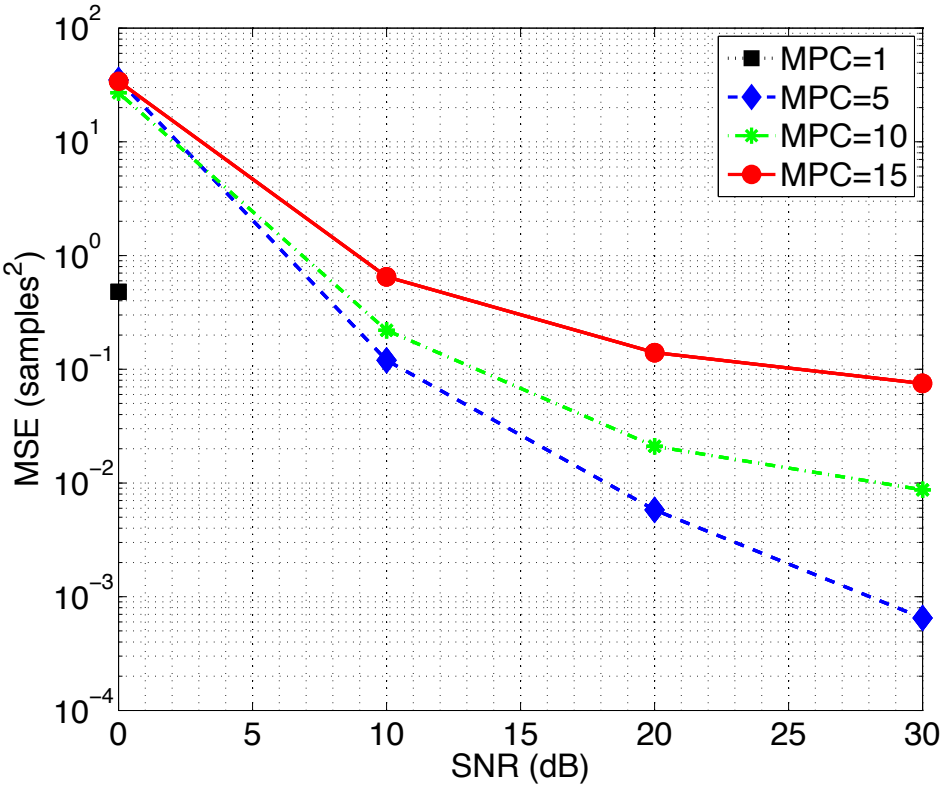


Figure 2.12: MSE in channel CM1 with different numbers of MPCs when $N = 128$.

Using the median of 1000 symbols, no errors occurred except at a low SNR of 0 dB and a significant number of MPCs, as shown in Table 2.3 and Table 2.4 for channels CM1 and CM4, respectively.

Table 2.3: Number of errors in bins using the median of 1000 symbols with channel CM1, $N = 128$

	Number of MPCs			
SNR (dB)	1	5	10	15
0	0	1	1	4
10	0	0	0	0
20	0	0	0	0
30	0	0	0	0

To compare our result with other techniques, we consider exponential and non-exponential Rayleigh fading channels with the same parameters as in [24] and

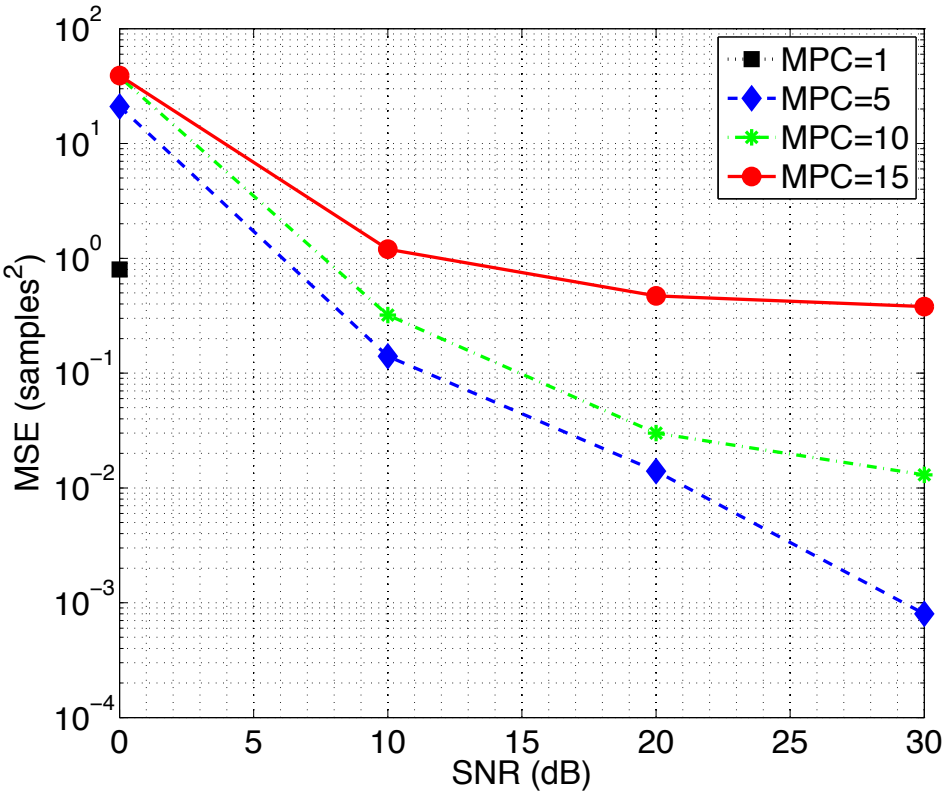


Figure 2.13: MSE in channel CM4 with different numbers of MPCs when $N = 128$.

Table 2.4: Number of errors in bins using the median of 1000 symbols with channel CM4, $N = 128$

	Number of MPCs			
SNR (dB)	1	5	10	15
0	0	1	4	4
10	0	0	0	0
20	0	0	0	0
30	0	0	0	0

in Table 2.5. Fig. 2.14 shows the MSE for the four previous algorithms and the two proposed algorithms, i.e., using the average and median in an exponential Rayleigh fading channel. Note that the average method has an MSE less than 10^{-6} for SNR $>$ 12 dB. However, using the median of the metric result of sending 10^3 instead of their average we obtain an MSE less than 10^{-6} for SNR $>$ 0 dB. We also

compared our algorithms with that of Choi *et al.* in a non-exponential Rayleigh fading channel using the same parameters as in [24]. These results are shown in Fig. 2.15, which clearly shows the superiority of our algorithm. The average number of iterations of the metric at each SNR value is shown in Table 2.6 for an exponential Rayleigh fading channel, and Table 2.7 for a non-exponential Rayleigh fading channel.

Table 2.5: Simulation parameters

Parameter	Value
Number of the subcarriers	2048
Number of FFT/IFFT points	2048
Guard interval length (samples)	256
Number of channel taps	17
Channel tap spacing	8
Ratio between first tap to last tap (dB)	20

Table 2.6: Average number of iterations for the exponential channel

SNR (dB)	Average number of iterations
0	2.1880
5	3.4280
10	5.6390
15	9.3230
20	16.4110
25	27.2430
30	41.9190

Table 2.7: Average number of iterations for the non-exponential channel

SNR (dB)	Average number of iterations
0	2.1860
10	4.6520
20	9.3710
30	18.8440

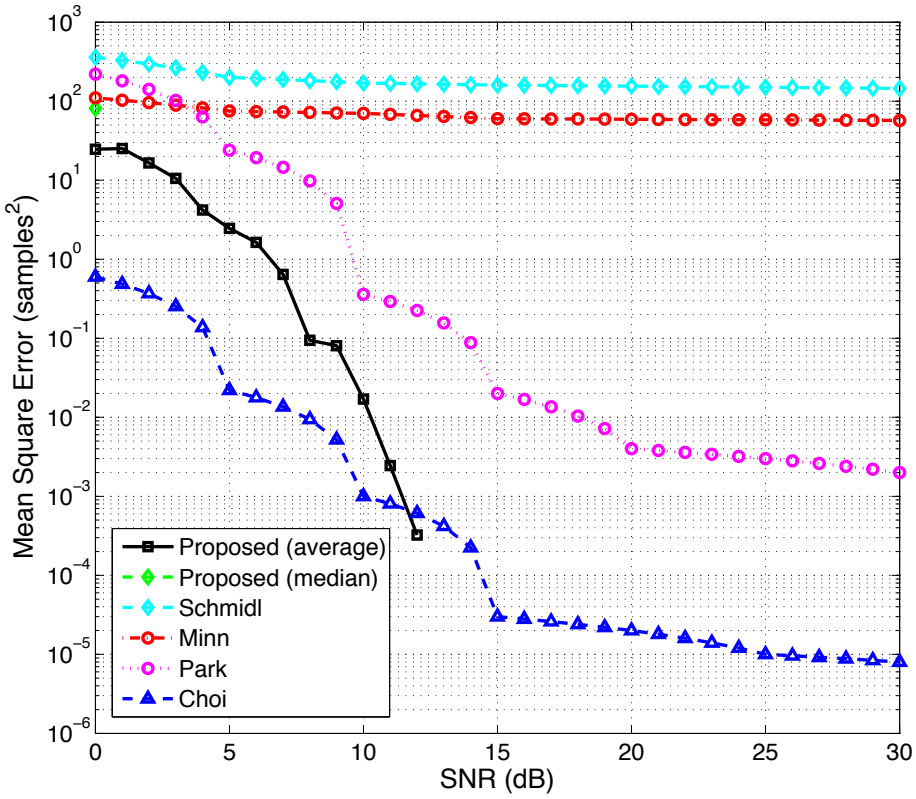


Figure 2.14: MSE of the two proposed algorithms compared with previous algorithms in an exponential Rayleigh channel.

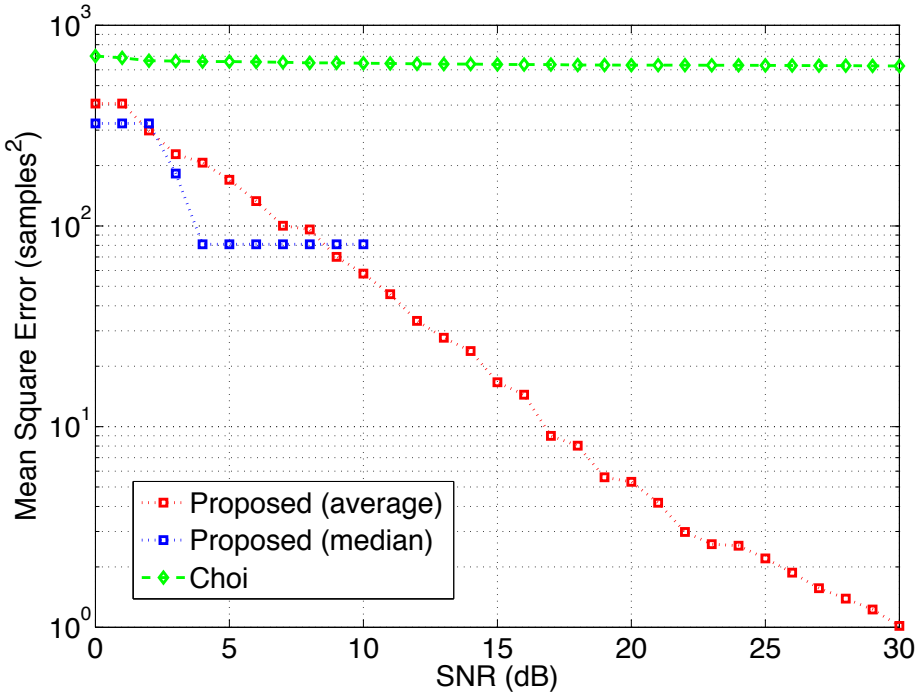


Figure 2.15: MSE of the two proposed algorithms compared with the algorithm in Choi *et al* in a non-exponential Rayleigh channel.

2.7 Proposed STO Estimation for CP-OFDM Systems

In this section, a new preamble-based timing method is presented and compared with the methods in [21], [22], and [23] because they are the best known techniques used in CP-OFDM systems. To make it easier for the reader, the three previous algorithms are presented again here. In Schmidl's *et al.* method [21], a training symbol is employed as a preamble. It has two identical halves in the time domain given by

$$S_{sch} = [A_{N/2}\ A_{N/2}] \tag{2.23}$$

and a correlation function given by

$$P_{sch}(d) = \sum_{m=0}^{N/2-1} r^*(d+m) \cdot r\,(d+m+N/2) \tag{2.24}$$

In Minn *et al.* [22], the training symbol has four parts given by

$$S_{minn} = [B_{N/4}\ B_{N/4}\ -B_{N/4}\ -B_{N/4}] \tag{2.25}$$

and a correlation function given by

$$P_{minn}(d) = \sum_{k=0}^{1} \sum_{m=0}^{N/4-1} r^*(d+kN/2+m) \cdot r(d+kN/2+m+N/4) \tag{2.26}$$

Park *et al.* [23] proposed the following training symbol structure

$$S_{park} = [A_{N/4}\ B_{N/4}\ A^*_{N/4}\ B^*_{N/4}] \tag{2.27}$$

and a correlation function given by

$$P_{park}(d) = \sum_{k=0}^{N/2} r(d-k) \cdot r(d+k) \tag{2.28}$$

For OFDM time synchronization, the preamble is constructed such that the effects of the channel are minimized and frequency offset estimation is simple [21]. However, the start of the preamble must be accurately estimated. This requires a preamble with a large value of the timing metric at the start of the symbol and small otherwise. At locations far from the correct starting point, the metric will be small because the sequence overlap is small, i.e., the correlation at the receiver is primarily with random values. Thus the challenge is to minimize the metric near the start of the OFDM symbol. This can be seen as making P small for incorrect timing positions.

2.7.1 Proposed STO Estimation Algorithm

The preamble sequence should provide a large difference in the metric between correct and incorrect timing locations. In Schmidl and Cox [21], the value at the correct location is $N/2$, but at one sample away, the metric is $N/2 - 1$. In Minn *et al.* [22], the maximum is the same, but at one sample away, the metric is only $N/4$. The technique by Park *et al.* [23] also has a maximum of $N/2$, while the value one sample away is very small. In the proposed scheme, the metric at the correct timing location has value N (which is the maximum achievable), while one sample away it is also very small.

To achieve a maximum of N, the new preamble has the following structure

$$S_{prop} = [A_{N/2}\, B_{N/2}] \tag{2.29}$$

where $A_{N/2}$ is a sequence of length $N/2$, and $B_{N/2}$ is the reverse of $A_{N/2}$. The

expressions

$$P_{prop}(d) = \sum_{k=0}^{N-1} r(d+k) \cdot r^*(d+N-1-k) \tag{2.30}$$

and

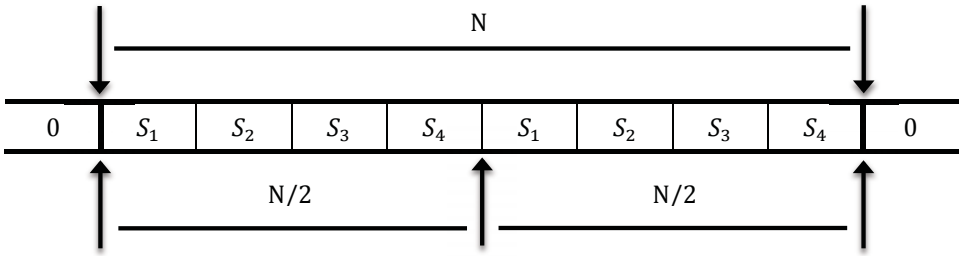
$$R_{prop}(d) = \sum_{k=0}^{N-1} |r(d+k)|^2 \tag{2.31}$$

are used in the metric (2.4), where d is the time index corresponding to the first element in a window of length N.

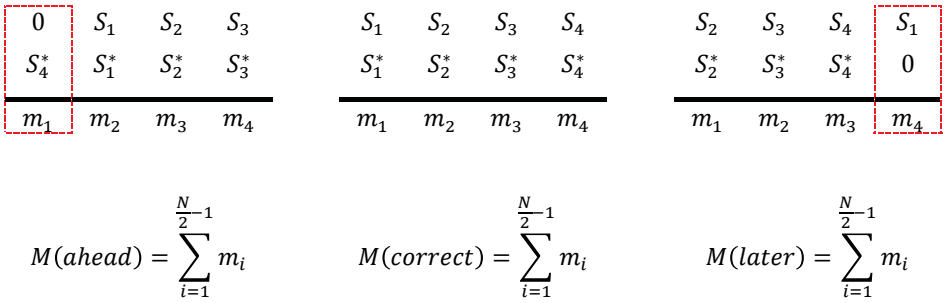
To compare between different preambles and how they differ in terms of metric values, an illustration of each preamble is shown in Figs. 2.16, 2.17, 2.18, and 2.19. In Fig. 2.16, the Schmidl *et al.* method preamble is shown in 2.16a and the metric values at three different locations: 1-sample before the correct timing, at exact timing, and 1-sample after the correct timing is shown in Fig. 2.16b. In Fig. 2.17, the preamble given in [22] is shown in 2.17a, and the metric values at three different locations: 1-sample before the correct timing, at exact timing, and 1-sample after the correct timing, are shown in Fig. 2.17b. In Fig. 2.18, the preamble given in [23] is shown in Fig. 2.18a and the metric values at three different locations: 1-sample before the correct timing, at exact timing, and 1-sample after the correct timing are shown in Fig. 2.18b. In Fig. 2.19, the preamble of the new proposed method is shown in Fig. 2.19a and the metric values at three different locations: 1-sample before the correct timing, at exact timing, and 1-sample after the correct timing, are shown in Fig. 2.19b, Fig. 2.19c, and Fig. 2.19d, respectively.

2.7.2 Performance Results

To determine the performance with the new preamble, a Rayleigh-fading channel model is considered with an exponential power delay profile. All channels are in digital bins where the bin width depends on the application and it varies



(a) Preamble structure.



(b) Metric values in different locations.

Figure 2.16: An illustration of method in Schmidl and Cox.

from one nanoseconds to few milliseconds depends on the application. Also, in all simulations a zero mean Gaussian noise is assumed to combine thermal noise and interference. The channel coefficients h_l are complex Gaussian random variables, i.e.

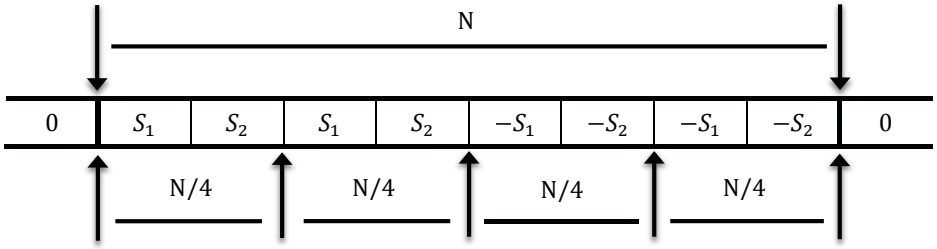
$$\mathrm{Re}\{h_l\}, \mathrm{Im}\{h_l\} \sim N\left(0, \sigma_l^2/2\right) \Rightarrow h_l \sim CN\left(0, \sigma_l^2\right)$$

The exponential power profile is given by

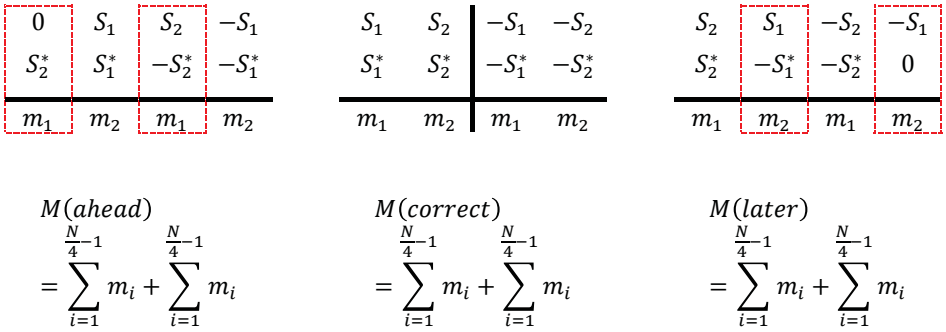
$$\frac{\sigma_l^2}{\sigma_0^2} = \exp\left(-l/\alpha\right), \quad l = 0, \ldots, L-1 \tag{2.32}$$

where L is the number of paths or taps, and α is the attenuation factor, which is chosen to be 0.96 so that the ratio between the first and last taps is 26 dB.

Symbol fading is assumed, i.e., the channel impulse response h is constant during an entire OFDM symbol, but changes randomly from one OFDM sym-



(a) Preamble structure.

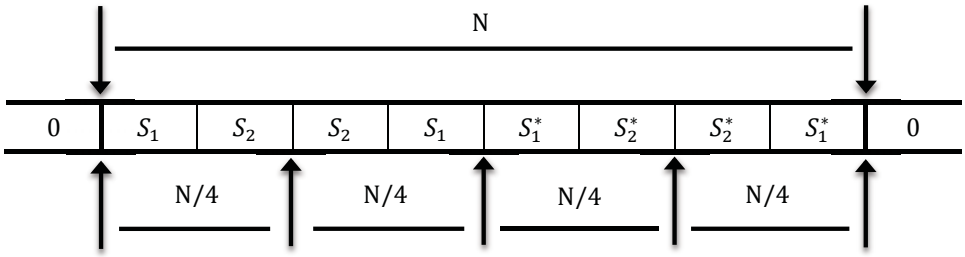


(b) Metric values in different locations.

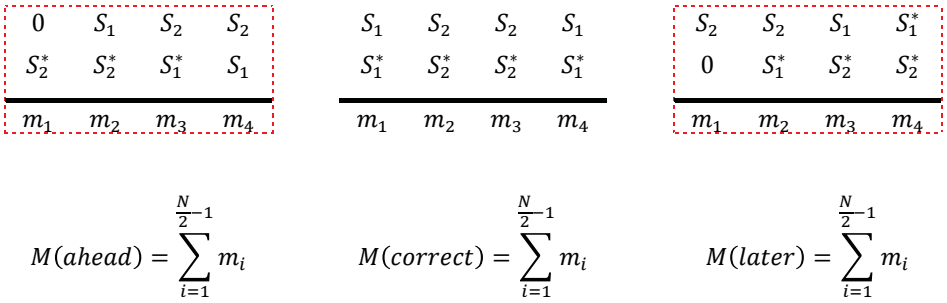
Figure 2.17: An illustration of method in Minn *et al*.

bol to the next. The noise samples are i.i.d. complex Gaussian random variables $\sim CN\left(0,\sigma_w^2\right)$. With BPSK modulation, the signal to noise ratio (SNR) is given by $E_b/N_0 = 1/\sigma_w^2$.

We consider a Rayleigh fading channel with $L = 8$ taps and a cyclic prefix of length $N_g = 32$. The search region to find the peak is assumed to be within N_g samples of the correct location, so the search window size is $2N_g$. Figs. 2.20 and 2.21 show the timing metrics with $N = 256$ and without noise. Both the method in [23] and the proposed preamble have a sharp peak at the correct timing location. However, Fig. 2.21 shows that the mean of the metric values around the peak ($N/2$ on each side) are lower with the proposed method. The mean using the technique in [23] is 0.0069, while that of the proposed method is 0.0029, which is an improvement of 57%.



(a) Preamble structure.



(b) Metric values in different locations.

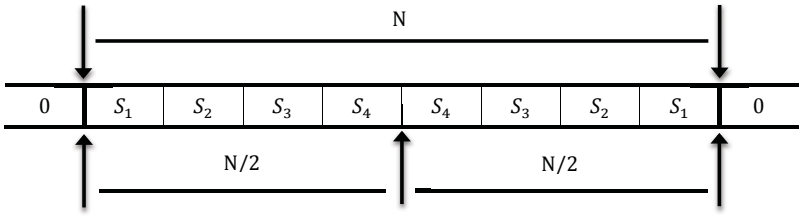
Figure 2.18: An illustration of method in Park *et al*.

Fig. 2.22 shows the mean square error (MSE) of the techniques in an AWGN channel for $k = 10^4$ symbols and $N = 128$. Fig. 2.23 shows the corresponding MSE in an exponential Rayleigh fading channel. The MSE is given by

$$MSE = \frac{(\varepsilon - \hat{\varepsilon})^2}{k} \tag{2.33}$$

where ε, $\hat{\varepsilon}$, and k are the correct timing delay, estimated timing delay, and number of symbols, respectively.

Figs. 2.22 and 2.23 show that the proposed technique has a smaller MSE compared to the other methods in both channels. This improvement can be inferred from the results given previously.



(a) Preamble structure of the proposed method.

0	S_1	S_2	S_3	S_4	S_4	S_3	S_2
S_2^*	S_3^*	S_4^*	S_4^*	S_3^*	S_2^*	S_1^*	0
m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8

$$M(ahead) = \sum_{i=1}^{N-1} m_i$$

(b) Metric value at one sample before correct timing.

S_1	S_2	S_3	S_4	S_4	S_3	S_2	S_1
S_1^*	S_2^*	S_3^*	S_4^*	S_4^*	S_3^*	S_2^*	S_1^*
m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8

$$M(correct) = \sum_{i=1}^{N-1} m_i$$

(c) Metric value at correct timing.

S_2	S_3	S_4	S_4	S_3	S_2	S_1	0
0	S_1^*	S_2^*	S_3^*	S_4^*	S_4^*	S_3^*	S_2^*
m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8

$$M(later) = \sum_{i=1}^{N-1} m_i$$

(d) Metric value at one sample after correct timing.

Figure 2.19: An illustration of the proposed method.

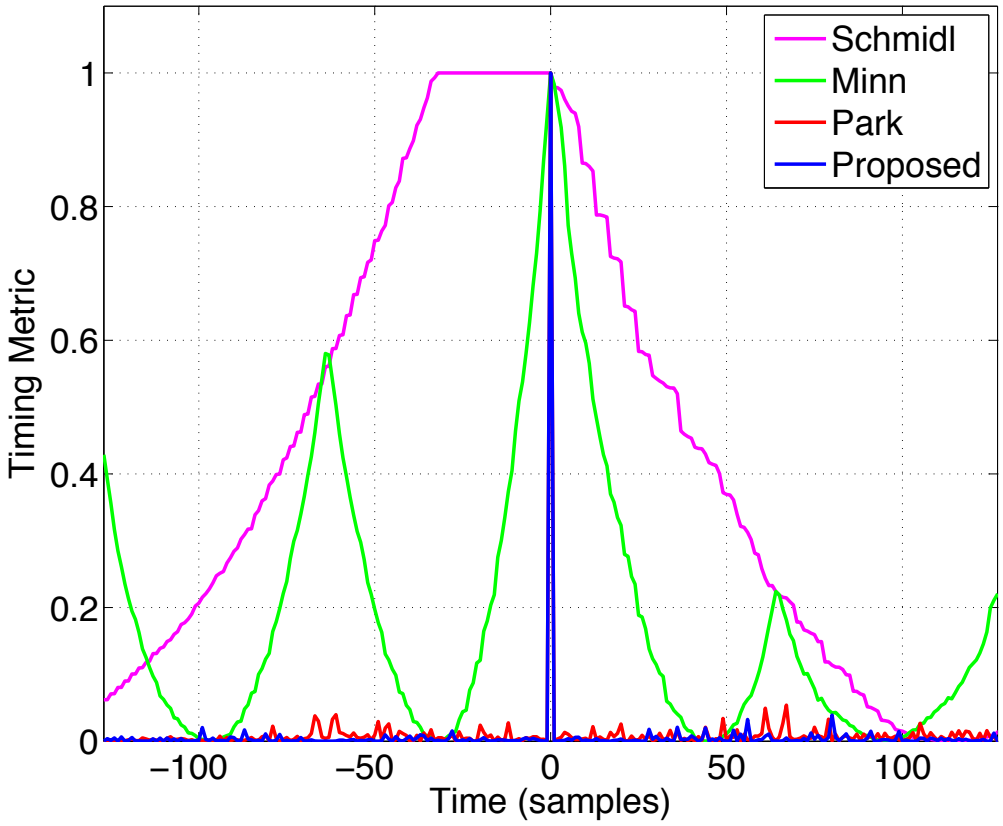


Figure 2.20: The timing metrics under ideal conditions.

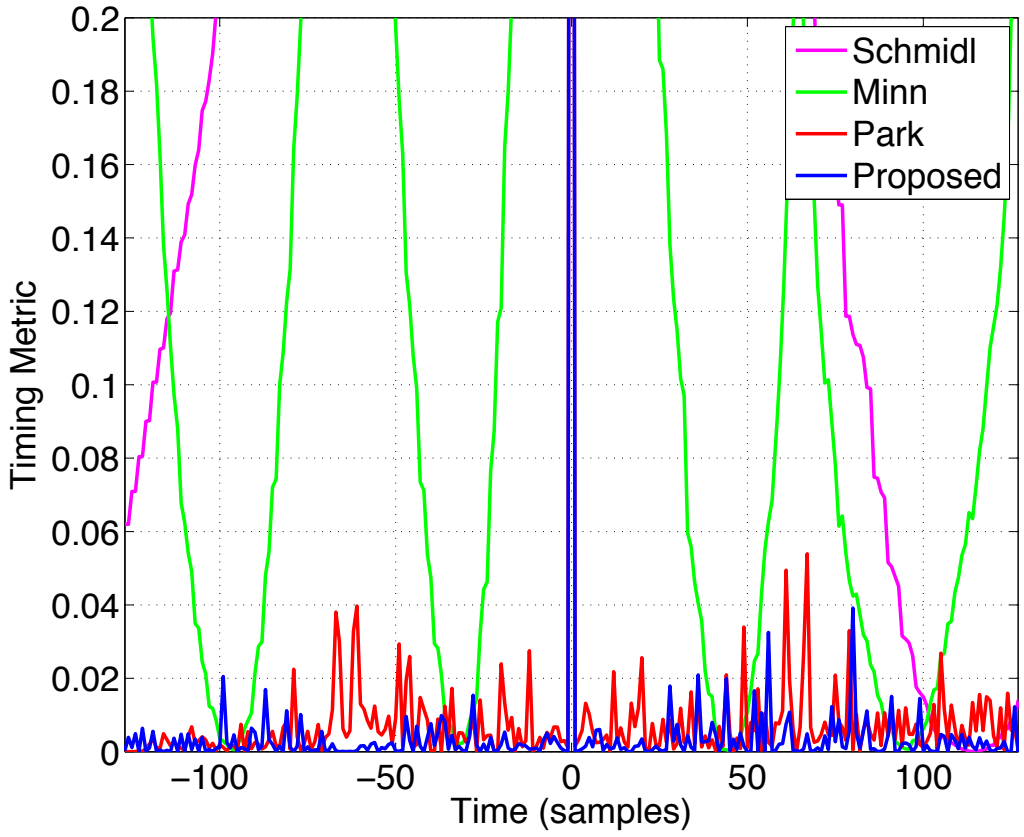


Figure 2.21: Closeup of the timing metrics under ideal conditions.

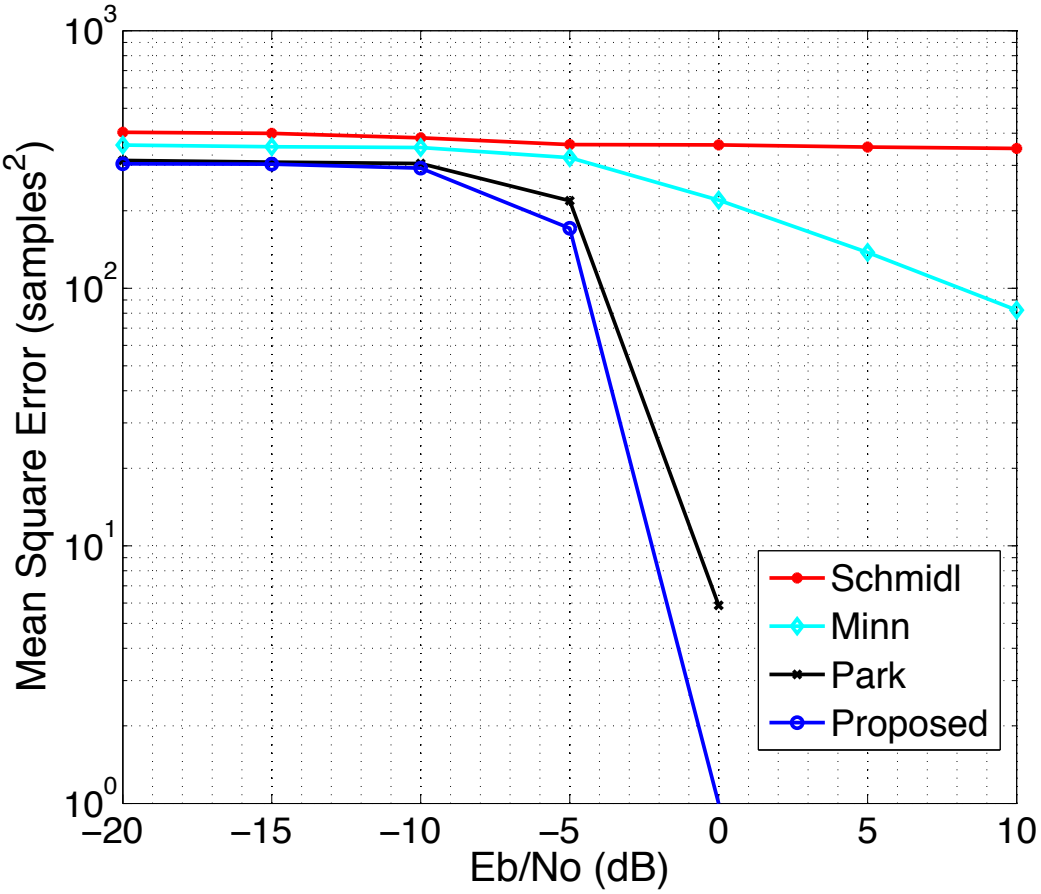


Figure 2.22: Mean square error in an AWGN channel.

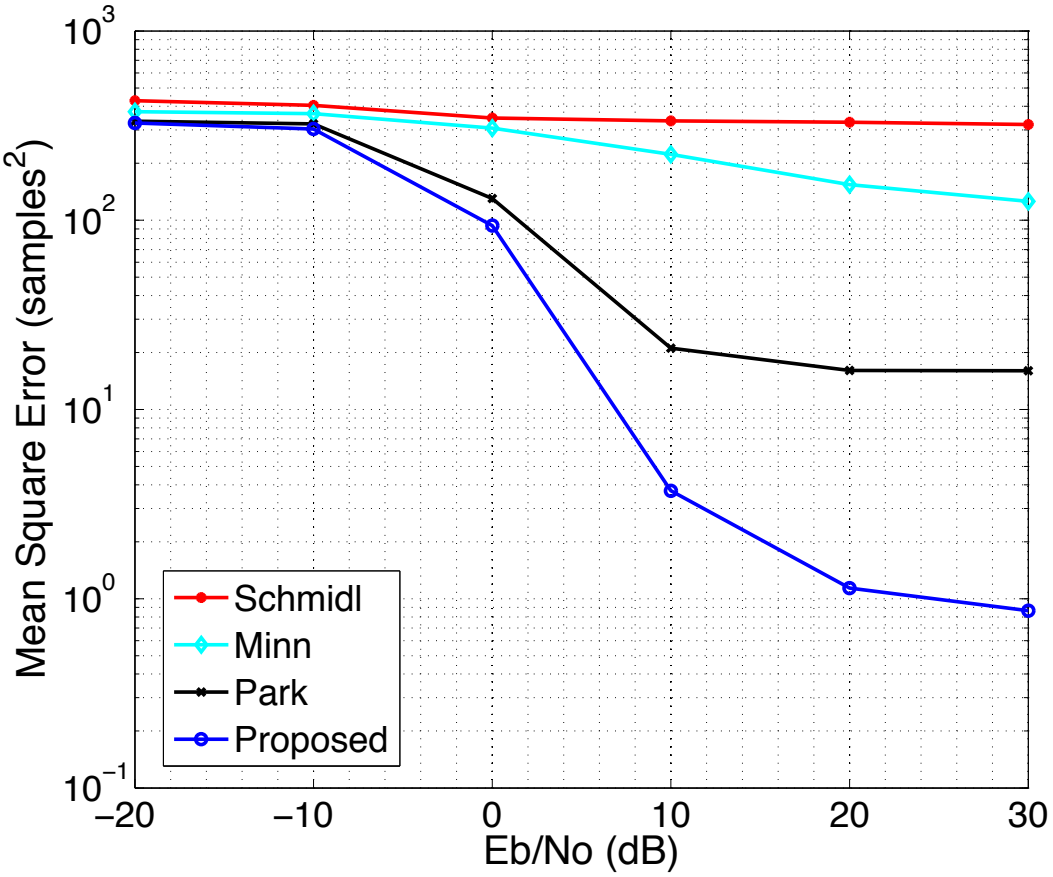


Figure 2.23: Mean square error in a Rayleigh fading channel.

Chapter 3

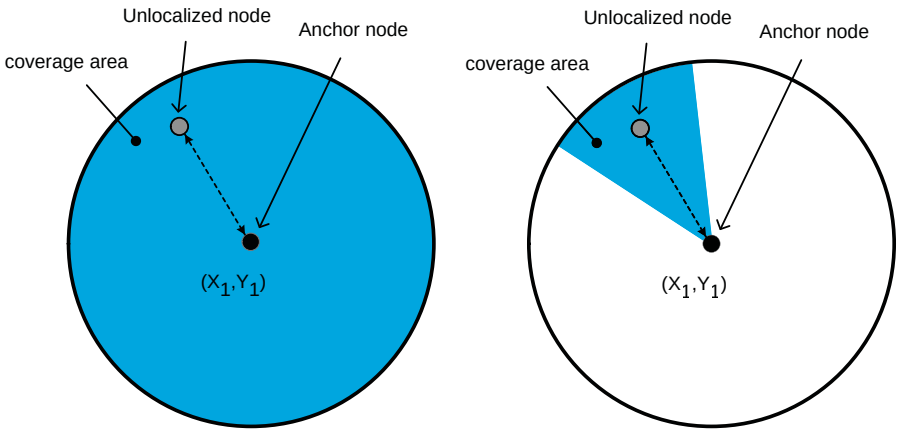
Localization

The importance of localization arises from many applications, including, for example, node addressing, evaluation of node density and coverage, energy map generation, geographic routing, and object tracking. These applications make localization a key technology for the development and operation of wireless networks. In some cases, wireless nodes might be deployed in inaccessible terrains, disaster relief operations or battlefields. In such cases the position of wireless nodes may not be predetermined. Thus, localization is required in order to provide position information to the nodes.

In general, localization can be divided into two types, range-based localization, and range-free localization. In range-based localization, the measurement of one or more location metrics, for example, angle of arrival (AOA), time of arrival (TOA), time difference of arrival (TDOA), or received signal strength (RSS), is needed to calculate the location of unlocalized nodes. Such a metric usually reflects the spatial relation of an unlocalized node relative to anchor nodes in the surrounding environment, where an anchor node denotes a point of known location. They are often measured by utilizing the physical fundamentals of radio, infrared or ultra-

sound signals, such as velocity or attenuation [27]. After the required metrics have been determined, the unlocalized node locations must be derived after taking into consideration the measurement results and the coordinates of the anchor nodes.

On the other hand, range-free localization needs only proximity sensing to calculate the location of an unlocalized node. The location of a unlocalized node is derived from the coordinates of the anchors that are within its range. Range-free algorithms do not provide as accurate an estimation of the unlocalized node location as range-based ones. However, if the wireless nodes use a directional antenna instead of an omnidirectional, more accurate results can be achieved. This is illustrated in Fig. 3.1. Fig. 3.1a shows a configuration with omnidirectional antennas, while Fig. 3.1b shows proximity sensing with a directional antenna.



(a) Omnidirectional antenna. (b) Directional antenna.

Figure 3.1: An illustration of proximity sensing.

3.1 Ranging Sensing Techniques

The most common location metrics are TOA, TDOA, AOA, and RSS. TOA and TDOA are time-based location metrics. The measurement of absolute TOAs of a known signal transmitted by the unlocalized node to N_a different anchors ($N_a \geqslant$

3) is employed by TOA-based algorithms, and requires that all the participating anchors and unlocalized node clocks are synchronized [28].

The TDOA-based algorithms measure the difference in the time of reception of signals at different anchors, without requiring a synchronization of all the participating anchors and unlocalized node. Actually, the uncertainty between the reference time of the anchors and that of the unlocalized node can be removed by means of a differential calculation. Therefore, only the anchors involved in the location estimation process must be tightly synchronized.

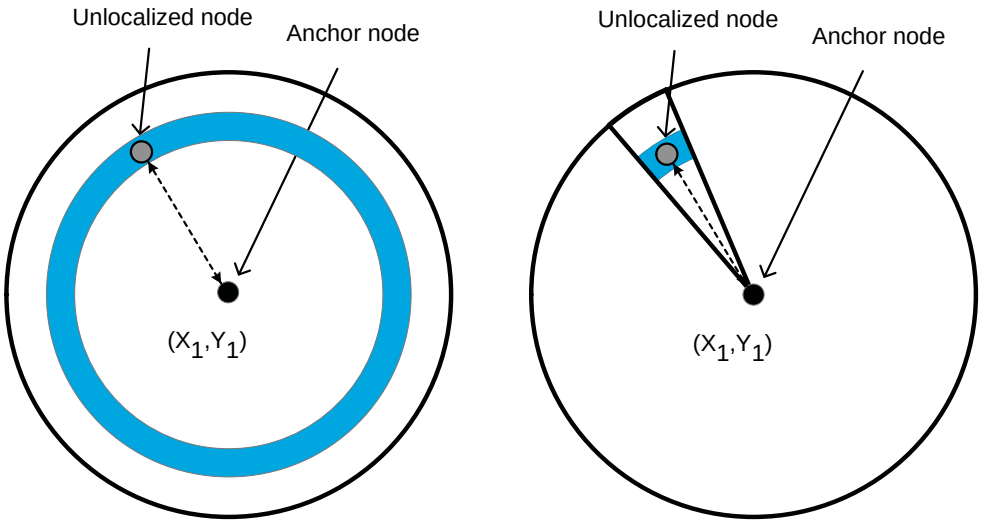
RSS-based localization algorithms measure the power of a known signal sent by the unlocalized node to N_a different anchors $(N_a \geqslant 3)$. If the transmit power is known, the distance between an anchor and the unlocalized node can be estimated using the received power level and a mathematical model for the path loss attenuation. By using at least three anchors to resolve ambiguities, the unlocalized node location can be estimated. However, RSS-based measurements are in general very inaccurate compared to time measurements.

In AOA-based algorithms, the location metrics are the angles between the unlocalized node and a number of anchors. Angulation is also termed angle of arrival (AOA) or direction of arrival (DOA). The implementation of the AOA requires complex receiving antenna arrays and proper direction of arrival (DOA) estimation algorithms [27].

Localization can also be realized by pattern matching. The main principle here is to observe the site (or scene) where localization is to be applied and to draw conclusions about the position of a unlocalized node from these observations. It is classified as optical and non-optical pattern matching [27]. In the optical version, also known as scene analysis, visual images of a site are generated by a camera, which are then compared with each other. In non-optical pattern matching, other

physical quantities are taken into account. A popular method is to detect a position from the propagation characteristics of radio signals, which the unlocalized node's terminal experiences at a certain position on a site. In this case, pattern matching is also known as fingerprinting [27].

Finally, it is possible to implement any combination of the localization methods presented earlier and consider several different location metrics. Therefore, an obvious method is to combine ranging and angulation, which provides an intersection of the circle and the direction of the radio beam (see Fig. 3.2) [27].



(a) Proximity sensing and range estimate

(b) Proximity sensing, range and angle estimates

Figure 3.2: Hybrid approach.

In spite of the estimation accuracy of different approaches for node localization, the geometry of the anchors may lead to large estimation errors as we will see in Chapter 5.

3.2 Sources of Errors

Poor accuracy is a result of errors of the measured location metrics, which have their origin in a number of error sources. Inaccurate and unstable clocks of positioning equipment directly lead to errors in the measured TOA and TDOA. Although mechanisms of clock synchronization and mathematical models for eliminating offsets and drifts are used, there always remains a certain error potential [27].

Non line of sight (NLOS) is an important source of error. If a LOS exists between an unlocalized node and an anchor, it can be assumed that the first arriving signal has travelled on the LOS path, and subsequent signals arrive on reflected paths. However, in some cases, a line of sight does not exist and the signals arrive on reflected paths, which may lead to considerable time delays and subsequent range errors.

Multipath propagation is one of the main sources of error, especially when the localization process is applied in an indoor environment. As a result of multipath propagation, different copies of a signal may overlay at the receiver and distort the signal amplitude and phase.

A bad geometry of anchors involved in localization may result in significant range or angle errors. Generally, a bad geometry is when the anchors are arranged along a line, such as along highways, or if they are arranged very close to each other. Finally, errors result from inaccurate location of anchors as well.

3.3 Localization Applications

To illustrate the potential of localization systems, we will now provide a partial list of applications that will be enhanced using wireless location information.

Mobile advertising Location-specific advertising and marketing will benefit once the location information becomes available. For example, stores will be able to track customer locations and attract them by displaying customized coupons on customers wireless devices [29].

Asset tracking (indoor/outdoor) Wireless location technology can also assist in advanced public safety applications, such as locating and retrieving lost children, patients, or pets. In addition, wireless location technology can be used to track personnel/assets in a hospital or a manufacturing site to provide more efficient management of assets and personnel. One could also consider applications such as smart and interactive tour guides, smart shopping guides that direct shoppers based on their location in a store, and traffic controls in parking structures that guide cars to free parking slots. Department stores, enterprises, hospitals, manufacturing sites, malls, museums, and campuses are some of the potential end users to benefit from this technology [29].

Fleet management Many fleet operators, such as police forces, emergency vehicles, and other services like shuttle and taxi companies, can make use of the wireless location technology to track and operate their vehicles in an efficient manner to minimize response times. In addition, a large number of drivers on roads and highways carry cellular phones while driving. Wireless location technology can help track these phones, thus transforming them into sources of real-time traffic information that can be used to enhance transportation safety [29].

Location-based wireless access security New location based wireless security schemes can be developed to heighten wireless network security and avoid the interception of digital information. By using location information, only people in specific physical areas could be allowed access to files or databases through a WLAN [29].

Location sensitive billing Using the location information of wireless users, wire-

less service providers can offer variable rate plans or services that are based on the user location [29].

Chapter 4

Range-free Localization

Many existing systems and protocols attempt to solve the problem of determining a nodes location within its environment. The approaches taken to solve this localization problem differ in the assumptions that they make about their respective network and device capabilities. These include assumptions about device hardware, signal propagation models, timing and energy requirements, the nature of the environment (indoor vs. outdoor), anchor density, time synchronization of devices, communication costs, error requirements, and device mobility.

Depending on the behaviour and requirements of protocols using location information, varying granularities of error may be appropriate from system to system. Also, there are many different aspects to the localization problem, such as when localization should be performed and how frequently. Upon network startup, all nodes must be initially localized. However, this should be updated periodically, especially if there are mobile nodes in the network.

Acknowledging that the cost of hardware required by range-based solutions may be inappropriate in relation to the required location precision, researchers have sought alternate range-free solutions to the localization problem. Range-free

algorithms use connectivity information to estimate node locations, including CPE [30], centroid [31], and APIT [32]. In [31], a heterogeneous network containing powerful nodes with established location information is considered. Anchors send their location to neighbours that keep an account of all received locations. Using this proximity information, a simple centroid algorithm is applied to estimate the listening nodes location.

An alternate solution, DV-HOP [33] assumes a heterogeneous network consisting of sensing nodes and anchors. Instead of single hop broadcasts, anchors flood their location throughout the network maintaining a running hop-count at each node along the way. Nodes calculate their position based on the received anchor locations, the hop-count from the corresponding anchor, and the average-distance per hop, which is a value obtained through anchor communication.

When the number of anchor nodes is low, they cannot cover the entire wireless network. This means that not all unlocalized nodes can hear the signals from the anchor nodes. In this case, the localized nodes or settled nodes can participate in the localization process by considering them as anchors. This is called cooperative localization.

If high location estimation is required we should consider range-based algorithms that estimate the distance between them using location metrics such as time of arrival (TOA) [34], time difference of arrival (TDOA) [35], received signal strength (RSS) [36], or angle of arrival (AOA) [37].

The algorithm proposed here is based on finding the centroid of the area where the unlocalized node may reside by considering anchors around it. Actually each unlocalized node employs two sets of anchors in their vicinity. The first set contains one-hop anchors from the unlocalized nodes. The second set contains two-hop and three-hop anchors from the unlocalized nodes. Each unlocalized node

uses the intersections between the first set of anchors and the second set anchors to estimate its position.

4.1 Related Work

Range-free localization algorithms are usually much simpler than range-based algorithms, but they provide only coarse resolution because they employ proximity sensing only. This work is motivated by Sheu *et al.* [38], which presents a distributed range-free localization scheme (DRLS). The proposed technique is compared with DRLS. In [38], the authors compare their algorithm with two other range-free algorithms, centroid [31] and the convex position estimation (CPE) [30], and show that DRLS is better than both of them. The first is the simplest and is based on calculating the centroid of the anchors that the unlocalizaed node can hear, so the estimated unlocalized node location is obtained by averaging the coordinates of these anchors. In CPE algorithm, each unlocalized node defines an area called the estimate rectangle (ER) [30]. This area is formed from the intersection of the anchors around an unlocalized node by considering a square area around each anchor. The center of the ER is the estimated unlocalized position.

With DRLS, an estimated rectangle (ER) is first defined as with CPE [30], and the ER is divided into small cells. A grid scan algorithm is then employed to calculate an initial estimate of the unlocalized node location. In the scan, each small cell starts with a weight equal to zero. This weight is increased by one if the center of the cell is within range of a given anchor. Thus, if the center of a certain cell is within range of three anchors, the weight of that cell is three. At the end of grid scan algorithm, the initial estimate of a unlocalized node's location is the average of the centers of the highest weight cells [38].

Another grid scan approach is presented in [39]. In this case, each unlocalized node represents its one- and two-hop neighbourhood by a grid partitioned into small cells. This is more complex than partitioning only the ER, which is typically a much smaller area. In [38], a vector-based refining process is used if any two or three hop anchors from the unlocalized node intersect with the intersection between anchors one hop from the unlocalized node.

The proposed algorithm differs from DRLS as follows:

- Two- and three-hop anchors from the unlocalized node (anchors in the second set), are used in the estimation process even if their range does not intersect with the area between the one-hop anchors around the unlocalized node, i.e. area ER. In [38], they are considered only if this intersection occurs.

- The proposed algorithm has only one step (there is no refining process), whereas DRLS has two steps. Each unlocalized node considers all available information to enhance the estimate.

- The computational cost of the proposed algorithm is less than for DRLS. As mentioned in [38], the grid scan algorithm is time consuming because in the first step, the center of each cell has to be checked to see whether it is within range of all other one-hop anchors from the unlocalized node. In the second step, a vector based refinement is done, and these calculations are similar to those of the proposed algorithm.

4.2 The Proposed Range-free Localization Algorithm

Considering only one-hop anchors means that some available information is not employed in the estimation process. In Fig. 4.1, the unlocalized node can hear

only one anchor, but there is another anchor that can be heard by the first one. In this case, we can consider the effect of this second anchor by using the intersection between the anchors to reduce the estimation area of the unlocalized node. Note that in this example, the centroid of the new estimation area is a better estimate of the unlocalizaed location.

The proposed algorithm uses all anchors that the unlocalized node can hear, and also the anchors that are two- or three-hop away from the unlocalized node. These anchors are classified into two sets. The first set contains one-hop anchors away from the unlocalized node. The second set contains two-hop and three-hop anchors from the unlocalized node. Previous algorithms, such as the centroid algorithm [31], consider only the set of anchors that are one hop from the unlocalized node. If the node density is low, this will provide a very coarse estimate.

The presence of anchors in the second set allows the anchors in the first set to better estimate the unlocalized node as shown in Fig. 4.2. In this figure, the blue dots are the one-hop (first set) anchors, and the red dots are the second set of anchors. The red arrows represent the effect of the second set on the one-hop anchors, with the arrow heads showing the virtual locations of these anchors. The calculation of the new (virtual) first set of anchors is described in Section 4.3 and is given by (4.7). The steps of the new algorithm are given below.
For each anchor node:

- Receive information from the anchors (IDs and locations) within two hops via two-hop flooding.
- Broadcast the received anchor information to the unlocalized nodes that are within one hop.

For each unlocalized node:

- Each unlocalized node forms two sets of anchors. The first set contains one-hop anchors, and the second set contains two- and three-hop anchors from the unlocalized node.
- Each unlocalized node calculates a set of virtual coordinates for the anchors in the first set based on the second set of anchors.
- If an anchor in the first set is affected by more than one anchor from the second set, the unlocalized node averages the positions obtained from using each anchor in the second set.
- The unlocalized node processes the new virtual coordinates of the first set by calculating the intersection of the linear line of positions (LLOPs) formed by the intersection of their anchor areas [40].
- The estimated location of the unlocalized node is the average of the points in the previous step.
- The localized nodes broadcast their new location and ID to the one-hop unlocalized nodes.
- The unlocalized nodes estimate their locations based on the anchors and localized nodes within three hops using the steps above.
- The algorithm stops when there are no more unlocalized nodes that can be localized, i.e., if all unlocalized nodes in the network have been localized, or if the remaining unlocalized nodes cannot be heard by any other nodes in the network, i.e., isolated unlocalized nodes when the network is not fully connected.

4.3 Analysis of the Proposed Algorithm

The proposed algorithm constructs the centroids of partial circles, as shown in Fig. 4.3. Assume that an anchor centered at p_1 is a member of the first set, and an anchor centered at p_2 is a member of the second set. We first find the centroid of area B_1. Let A_1 and A_2 be the areas of the circles centered at anchors p_1 and p_2 respectively, and A the area of their intersection. This area is given by [41]

$$A = r_1^2 \cos^{-1}\left(\frac{d^2 + r_1^2 - r_2^2}{2dr_1}\right) + r_2^2 \cos^{-1}\left(\frac{d^2 + r_2^2 - r_1^2}{2dr_2}\right) - \frac{1}{2}\sqrt{(-d + r_1 + r_2)(d + r_1 - r_2)(d - r_1 + r_2)(d + r_1 + r_2)} \tag{4.1}$$

where r_1 and r_2 are the ranges of the first and second anchors respectively, and d is the distance between them, which is given by

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{4.2}$$

After finding the intersection area between the two anchors, its center must be found. This is just the intersection of two lines. The first line connects the two centers, and the second line connects the intersection points between the two circles. The coordinates (x, y) of these intersection points, p_3 and p_4, are given by [42]

$$x = \frac{x_2 + x_1}{2} + \frac{(x_2 - x_1)(r_1^2 - r_2^2)}{2d^2} \pm \frac{y_2 - y_1}{2d^2}\sqrt{\left((r_1 + r_2)^2 - d^2\right)\left(d^2 - (r_2 - r_1)^2\right)} \tag{4.3}$$

and

$$y = \frac{y_2 + y_1}{2} + \frac{(y_2 - y_1)\left(r_1^2 - r_2^2\right)}{2d^2} \mp \frac{x_2 - x_1}{2d^2}\sqrt{\left((r_1 + r_2)^2 - d^2\right)\left(d^2 - (r_2 - r_1)^2\right)} \tag{4.4}$$

where the p_3 x-coordinate corresponds to the plus sign in (5.5), and the y-coordinate corresponds to the minus sign in (5.6). The two anchors are located at $p_1 = (x_1, y_1)$ and $p_2 = (x_1, y_1)$.

Now we have four points, p_1 and p_2 (the two anchors), and p_3 and p_4 (the intersections). The intersection of the two lines shown in Fig. 4.4 is point c given by [42]

$$x_c = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & x_1 - x_2 \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & x_3 - x_4 \end{vmatrix}}{\begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}} \tag{4.5}$$

$$y_c = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & y_1 - y_2 \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & y_3 - y_4 \end{vmatrix}}{\begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}} \tag{4.6}$$

Finally, the new centroid of, say, area B_1 is calculated as the average of the positions, p_i, and weighted by the areas, a_i [43]

$$R = \frac{\sum a_i p_i}{\sum a_i} \tag{4.7}$$

According to Fig. 4.3, the center of area B_1 is obtained by using the points

$$x_{B_1} = \frac{x_1 A_1 - x_c A}{A_1 - A} \tag{4.8}$$

and

$$y_{B_1} = \frac{y_1 A_1 - y_c A}{A_1 - A} \tag{4.9}$$

where $p_1 = (x_1, y_1)$ and $c = (x_c, y_c)$. A_1 represents the area covered by the first node, and A represents the intersection area between the two nodes. Note that $B_1 = A_1 - (A_1 \cap A_2) = A_1 - A$.

Equations (4.8) and (4.9) are used to find the positions of the virtual anchors for each intersection between a first set anchor and a second set anchor. For the example in Fig. 4.2, this calculation must be repeated four times. Then, the virtual

anchor positions are averaged to give the one-hop anchor locations to be used in the unlocalized node estimation. In the example, there are two estimates for the right hand one-hop anchor as it has an area overlapping that of two second set anchors. Averaging these two positions give us one virtual anchor position. Applying the proposed algorithm on the network shown in Fig. 4.2a, we obtain three virtual anchors as shown in Fig. 4.2b. Next, we calculate two points for each LLOP using (5.5) and (5.6). An LLOP is associated with every intersection of two virtual anchor areas. Lastly, we determine the intersection of the LLOP pairs using (4.5) and (4.6). The average of these intersection points is the estimated location of the unlocalized node.

4.4 Performance Results

In this section, the performance of the proposed algorithm is compared with that using DRLS [38]. The simulation parameters are: number of nodes $n = 200$; range $r = 10$ meters; deployment area $A = 10r \times 10r$ meters^2. All nodes are assumed to have the same communication range. Performance is averaged over 1000 trials for each anchor ratio, i.e., 10^3 different random node deployments. As mentioned previously, the proposed algorithm stops when no more unlocalized nodes can be localized, and all remaining unlocalized nodes are left unlocalized. The mean error for each anchor ratio is defined as

$$mean\ error = \frac{\sum_{I} \sum_{i=1}^{t} \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2}}{tIr} \tag{4.10}$$

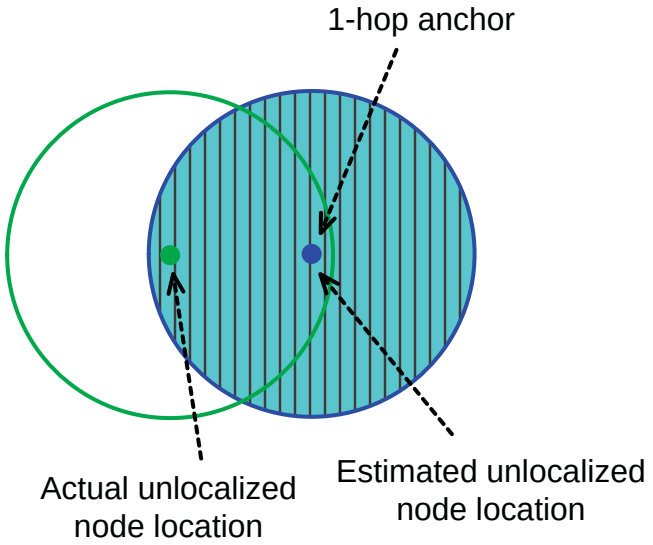
where t is the number of unlocalized nodes, I is the number of trials, $(\hat{x}, \hat{y})$ is the estimated unlocalized node position, and (x, y) is the actual position. Note that the

mean error is calculated relative to the range r.

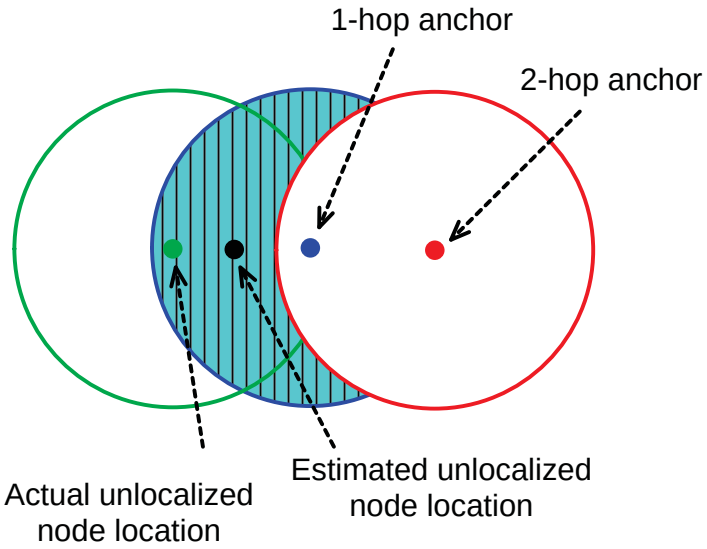
Fig. 4.5 shows the effect of the anchor ratio on the mean error of the proposed and DRLS algorithms. This shows that the proposed solution is better than DRLS at lower anchor ratios, which is typical in many applications, for example, when the number of anchors is low due to the cost of installing anchor nodes (which require GPS or manual determination of their location). Even with a high anchor ratio, the proposed algorithm has similar performance to that with DRLS. Because it has a lower computational cost than DRLS, the proposed algorithm is better overall.

Fig. 4.6 shows the performance of one trial with the proposed algorithm when 50% of the nodes are anchors. Note that there are some unlocalized nodes that are not localized because they cannot communicate with any anchor or localized node. The error distribution in the network closely follows the anchor distribution.

In areas with a high anchor density, the localization error is low and *vice versa*. This can be seen clearly in Fig. 4.7 which shows the relation between mean localization error and anchors density (only anchors are presented in this figure as blue dots). In Fig. 4.7 a high mean localization error occurred in areas when the anchors density is low and small mean error when the anchors density is high.

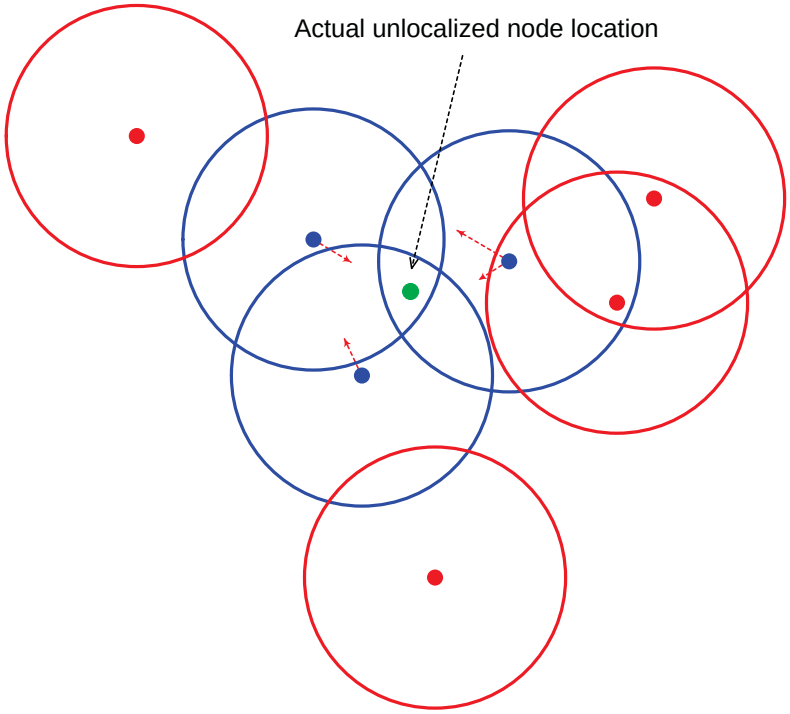


(a) 1-hop anchor effect on the unlocalized node estimation.

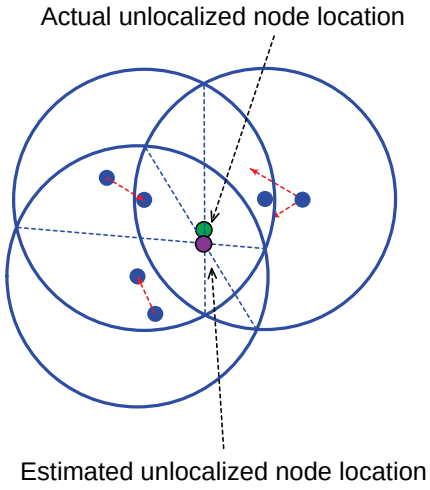


(b) 2-hop anchor effect on the unlocalized node estimation.

Figure 4.1: The effect of the second set of anchors on the estimated unlocalized node location.



(a) Proximity sensing between unlocalized and anchor nodes.



(b) Creating of new virtual anchors by the unlocalized node.

Figure 4.2: An example of the proposed location estimation algorithm.

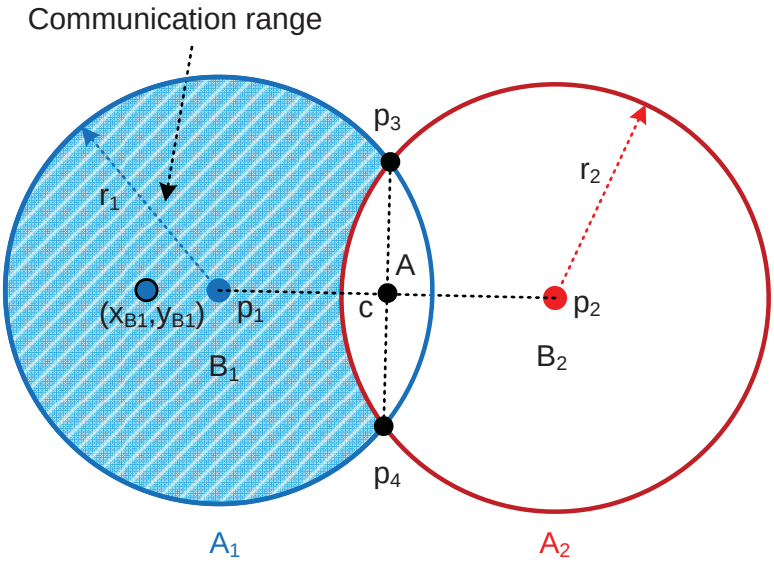


Figure 4.3: The area of intersection between two anchors.

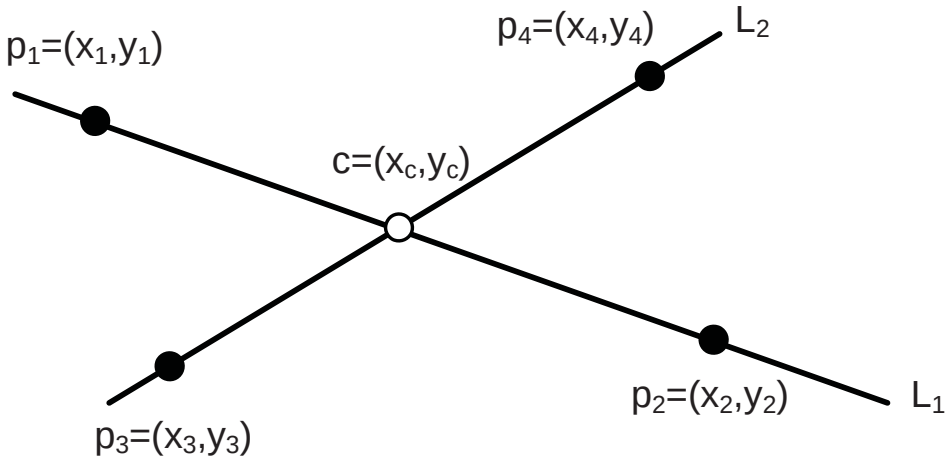


Figure 4.4: The intersection of two lines.

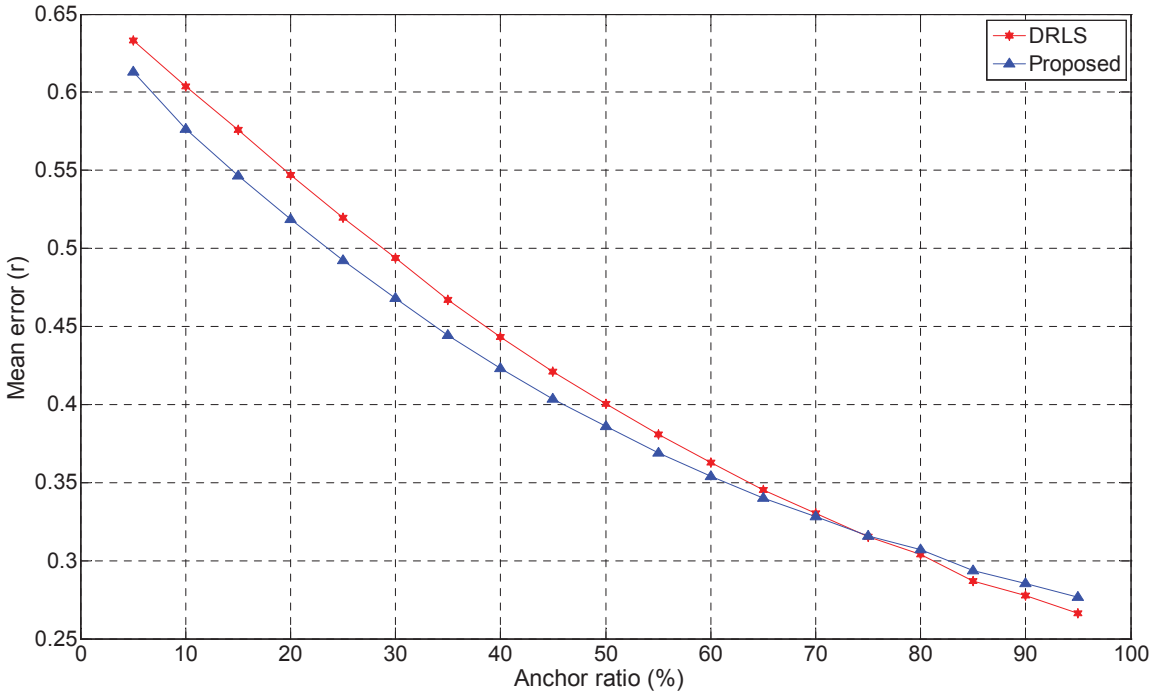


Figure 4.5: Anchor ratio versus mean localization error.

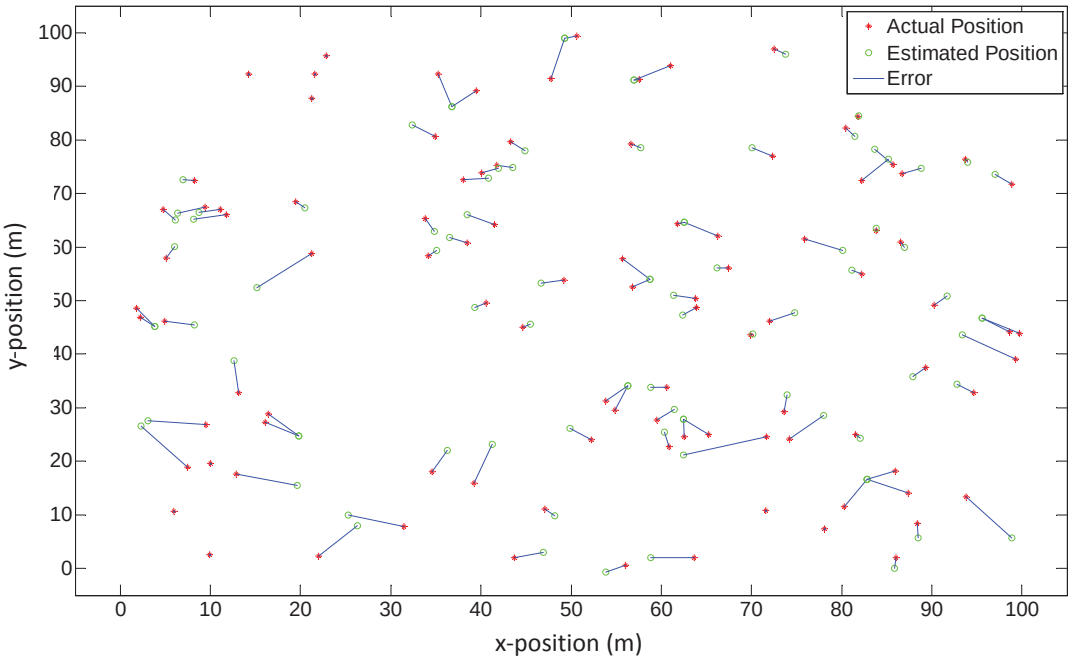


Figure 4.6: Actual and estimated position of all nodes in the network with 50% anchor nodes.

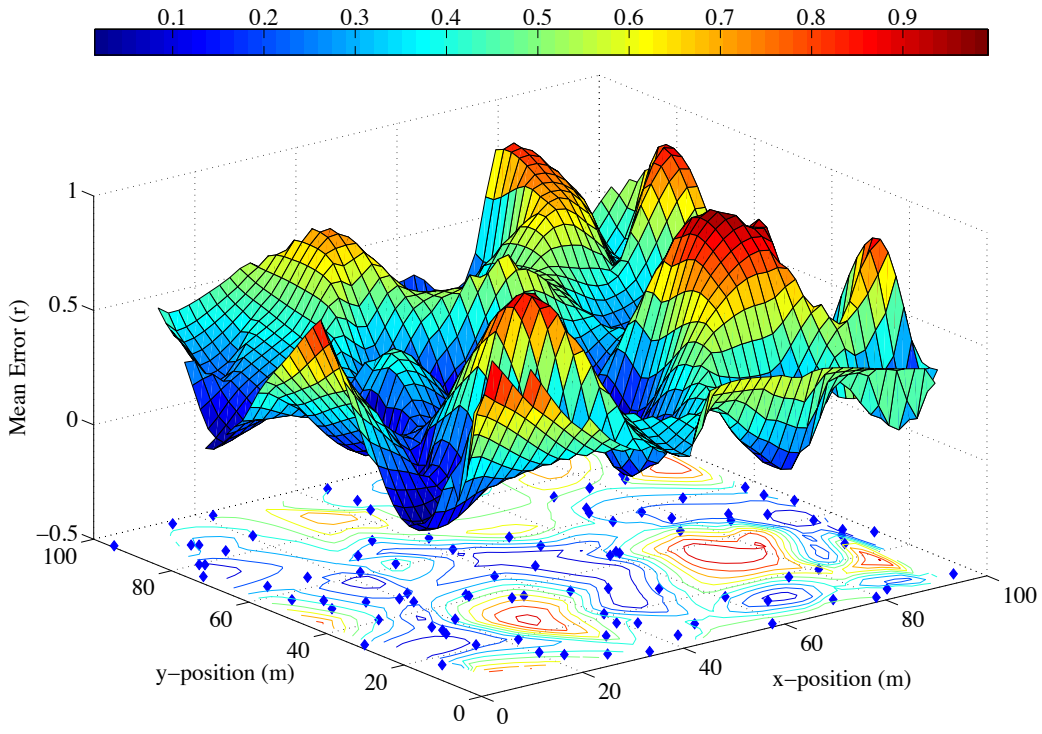


Figure 4.7: Localization error versus position.

Chapter 5

Range-based Localization

Range-based Localization methods are more accurate than range-free ones. Its accuracy is related to distance estimation accuracy. Range-based algorithms estimate the distance between nodes using location metrics such as time of arrival (TOA) [34], time difference of arrival (TDOA) [35], received signal strength (RSS) [36], or angle of arrival (AOA) [37].

Time of arrival (TOA) technology is commonly used as a means of obtaining range information via signal propagation time. The most basic localization system to use TOA techniques is GPS [44]. The time difference of arrival (TDOA) technique for ranging has been widely proposed as a first step towards localization algorithm. TDOA is used in many systems such as [45] [46], and [47]. To augment and complement TDOA and TOA technologies, an angle of arrival (AOA) technique has been proposed that allows nodes to estimate and map relative angles between neighbors [37]. Received signal strength based localization systems (RSS) technology such as RADAR [48] has been proposed for hardware constrained systems. In wireless environment problems, like multipath fading, background interference, and irregular signal propagation characteristics make range estimates

inaccurate using RSS.

Position accuracy is not constant across the area of coverage, and poor geometry of the unlocalized nodes relative to the anchor nodes can lead to high geometric dilution of precision (GDOP). GDOP is commonly used to assess localization accuracy, where a larger GDOP implies poorer accuracy.

5.1 Dilution of Precision

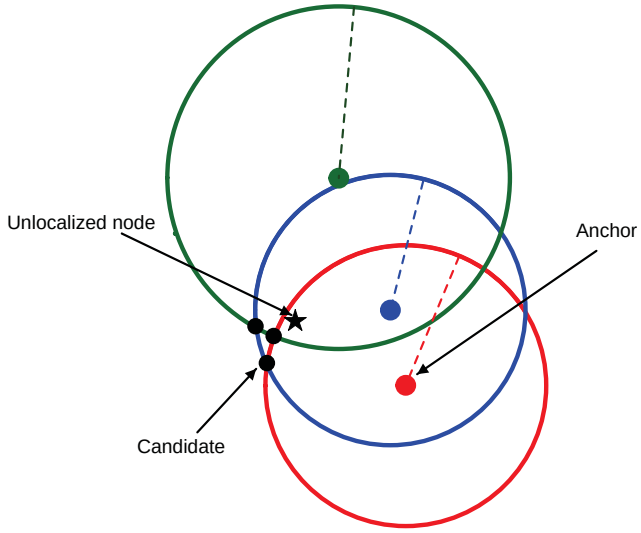
Dilution of precision is a metric that describing how well an anchor nodes geometry is for localization. The distance measurements used to compute the node coordinates always contain some error. These measurement errors result in errors in the computed node coordinates. The magnitude of the final error depends on both the measurement errors and the geometry of the structure induced by the nodes. The contribution due to geometry is called the geometric dilution of precision (GDOP). GDOP is used extensively in the GPS community as a measure of localization performance [49]. The distribution of the anchors around an unlocalized node can have a good or poor GDOP, as shown in Fig. 5.1.

Another version of GDOP is the generalized geometry of dilution precision GGDOP. GGDOP depends on the geometry of the anchors around an unlocalized node and the accuracy of the range measurements. GGDOP is defined as [50]

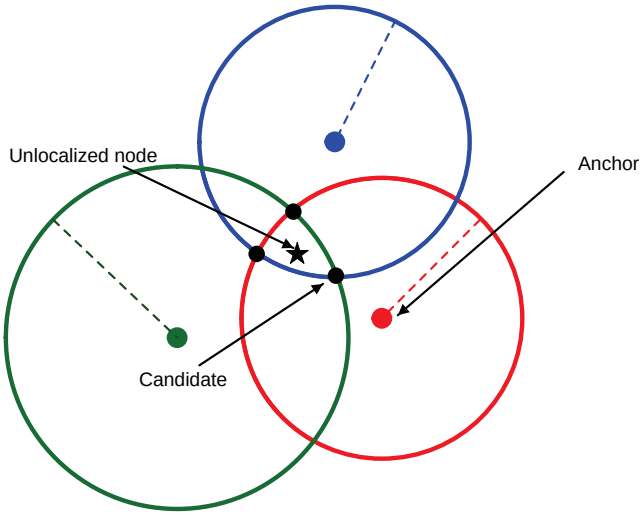
$$\Gamma_m = \frac{\psi_m}{\gamma_m^2} \tag{5.1}$$

where

$$\gamma_m = \sum_{i=1}^{m} \frac{1}{\sigma_i^2} \tag{5.2}$$



(a) Nodes with poor GDOP.



(b) Nodes with good GDOP.

Figure 5.1: Node locations with poor and good GDOP.

and

$$\psi_m = \sum_{i=1}^{m} \sum_{j=1, j>i}^{m} \frac{\sin^2\left(\alpha_i - \alpha_j\right)}{\sigma_i^2 \sigma_j^2} \tag{5.3}$$

The angle α_i is the orientation of the ith anchor or localized node relative to the node whose location is being estimated, σ_i^2 is the variance of the distance estimate, and m is the total number of anchors and localized nodes around this node, as

shown in Fig. 5.2. As the GGDOP increases, the localization error decreases. In [50], it was shown that

$$\forall \left\{\sigma_i^2\right\}, \{\alpha_i\}, 0 \leq \Gamma_m \leq {}^1/_4 \tag{5.4}$$

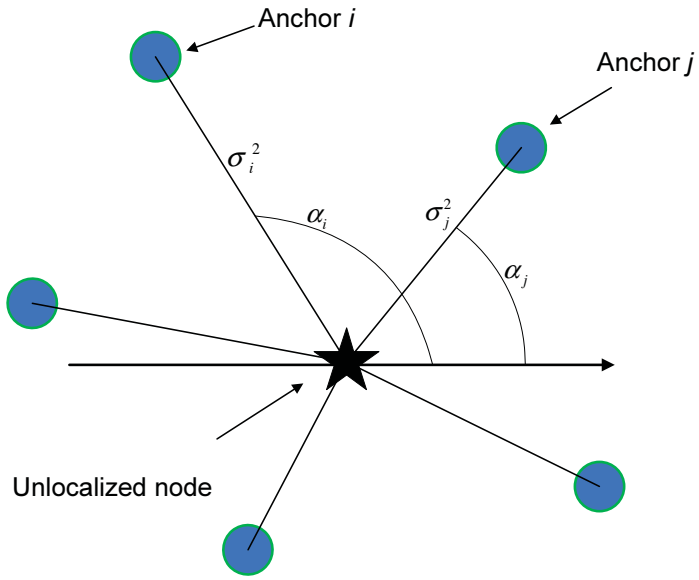


Figure 5.2: An unlocalized node with multiple anchors within its range.

In the following sections, four different range-based algorithms are proposed and compared with recent localization algorithms as well. All of them are based on techniques from data mining field.

We use the linear least square (LLS) [51] and the weighted linear least squares singular value decomposition (WLS-SVD) algorithm [52] as a benchmark. In [52], the WLS-SVD algorithm is compared with a maximum likelihood (ML) algorithm [53], multidimensional scaling (MDS) [54], and the best linear unbiased estimator approach based on least square calibration (BLUE-LSC) [55]. According to [52], WLS-SVD performs better than the three other algorithms.

5.2 Localization using Decision Tree Classification Algorithm

The proposed approach differs from conventional solutions to the localization problem in wireless networks. Typically, the locations of the anchors within range and the estimated distances between the unlocalized node and these anchors are used to directly estimate its location. Instead, we use a multi-step process. An approach from data mining, called decision tree classification, is used to retain the best (candidate) points, and these are averaged to get the estimated location of the unlocalized node.

5.2.1 Decision Tree Classification Algorithm

Classification assigns an object to one or more categories. Decision trees are a simple and widely used classification technique [2]. Moving from one level to another in a decision tree requires a test condition to decide which branch to follow. This process is continued until a leaf node is reached. In the proposed algorithm, we use both the shortest distance and the minimum variance between nodes as the test.

5.2.2 The Proposed Algorithm

The first step in localization is to obtain the distance estimates for the unlocalized nodes from the anchor and localized nodes that are within range. These estimates provide the radii for circles around the nodes. The intersection of these circles for an unlocalized node is the next step towards calculating the final location estimate. The key is to choose candidate intersection points that are closest to each other. In the ideal case, the circles intersect on the unlocalized node. For example, when we

have three anchors, three intersection points lie on the unlocalized node, while the other three do not. However, this event is unlikely in practical situations. In Fig. 5.3, the intersection points of the circles around nodes p_1 and p_2 are denoted as p_{12} and p_{21}, and their coordinates are given by [42]

$$x = \frac{x_2 + x_1}{2} + \frac{(x_2 - x_1)\left(r_1^2 - r_2^2\right)}{2d^2} \pm \frac{y_2 - y_1}{2d^2}\sqrt{\left((r_1 + r_2)^2 - d^2\right)\left(d^2 - (r_2 - r_1)^2\right)} \tag{5.5}$$

and

$$y = \frac{y_2 + y_1}{2} + \frac{(y_2 - y_1)\left(r_1^2 - r_2^2\right)}{2d^2} \mp \frac{x_2 - x_1}{2d^2}\sqrt{\left((r_1 + r_2)^2 - d^2\right)\left(d^2 - (r_2 - r_1)^2\right)} \tag{5.6}$$

where the p_{12} x-coordinate corresponds to the plus sign in (5.5), and the corresponding y-coordinate corresponds to the minus sign in (5.6). The distance between the two points $p_1 = (x_1, y_1)$ and $p_2 = (x_1, y_1)$ is given by

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{5.7}$$

Each unlocalized node estimates the distance from each anchor or localized node that it can receive a signal from. A node can estimate its position only if it hears from three or more of these nodes. The intersection between all estimates around the unlocalized node produce a set of intersection points. If we have m

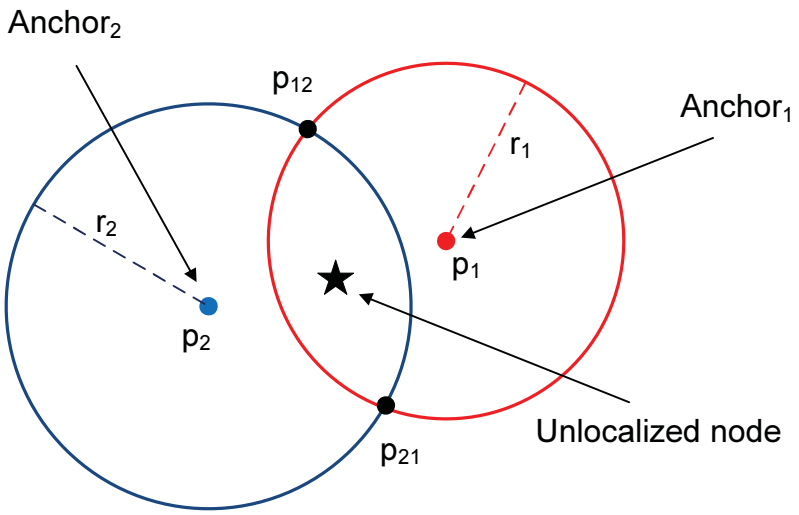


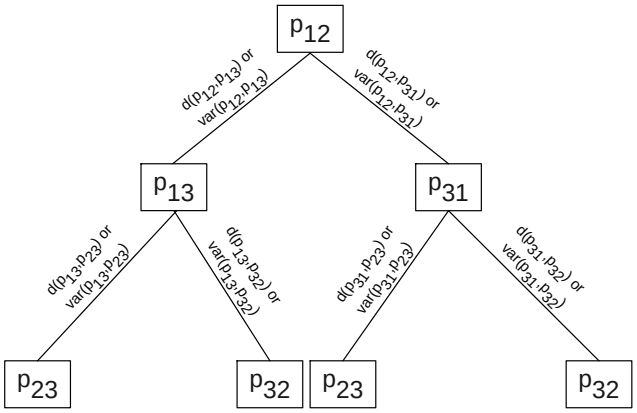
Figure 5.3: Intersection of the distance estimates for two anchors.

anchor and/or localized nodes, then they form g groups where

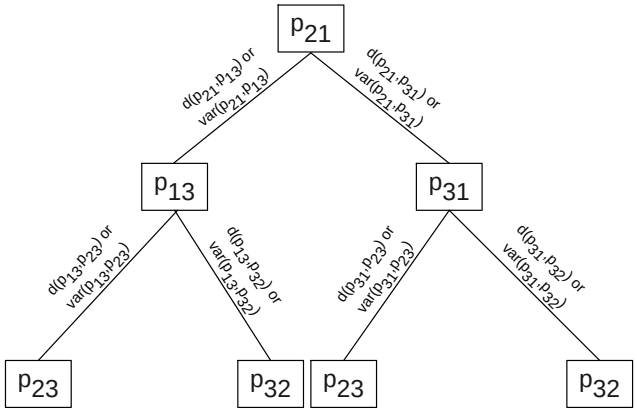
$$g = \begin{pmatrix} m \\ 2 \end{pmatrix} = \frac{m!}{2!\,(m-2)!} \tag{5.8}$$

Each group consists of two points as a result of the intersection between two anchors and/or localized nodes distance estimates, as shown in Fig. 5.3.

The final step in estimating the location of an unlocalized node is constructing two decision trees to select the candidate intersection points. The root or starting point in both trees can be chosen arbitrarily, but they should belong to the same group. In Fig. 5.4, the root points are p_{12} and p_{21}, and they are produced by the intersection of the Anchor$_1$ and Anchor$_2$ range estimates. The subtrees in this figure correspond to three anchor and/or localized nodes around the unlocalized node. To traverse from the root points down to the leaf points, we choose the branches with the smallest metrics, using either distances or variances. We then have Distance Based Localization using Decision Trees (DBLDT) and Variance Based Localization using Decision Trees (VBLDT).



(a) Subtree 1



(b) Subtree 2

Figure 5.4: Decision tree for three anchor nodes divided into two subtrees.

If all anchor and/or localized nodes intersect, the maximum number of points is $2g$, and we need to select g candidate points, one from each group. In some cases, a distance estimate is obtained by measuring the received signal strength (RSS), which may be short due to the non-line-of-sight (NLOS) effect, so not all circles intersect. However, the intersection points can still be calculated, but they will be complex numbers. If this occurs, we consider the real part as the intersection point in subsequent calculations.

DBLDT uses the Euclidean distance between two points as the decision tree metric. The path with the minimum distance is followed in each subtree. Suppose that in Fig. 5.4a the selected path has points p_{12}, p_{13}, and p_{32}, and in Fig. 5.4b the

selected path has points p_{21}, p_{31}, and p_{23}. Then we have two sets: a and b, each have g points, one from each subtree. The set with the minimum total Euclidean is chosen, and the average of the points in this set is the estimated location of the unlocalized node.

Fig. 5.5 illustrates the steps of the proposed algorithm for four anchor nodes. After the four distance estimates are determined, the $2g = 12$ intersection points are found, and then 6 candidate intersection points are selected. Note that because two pairs of circles do not intersect, there are only 10 points in Fig. 5.5a (since the real parts of the complex numbers are the same). The six candidate points are averaged to obtain the location estimate.

VBLDT uses the variance between two points as the decision tree metric. The variances σ_x^2 and σ_y^2 are calculated as

$$\sigma_x^2 = \frac{\sum (x - \mu_x)^2}{2} \text{ and } \sigma_y^2 = \frac{\sum (y - \mu_y)^2}{2}$$

where the sum is over the two points and μ_x and μ_y are the corresponding means. The path with the minimum metric, Ω, is followed in each subtree. We end up with two sets: a and b, each have g points, one from each subtree. The variances for the first set $\sigma_{x_a}^2$ and $\sigma_{y_a}^2$ are calculated as

$$\sigma_{x_a}^2 = \frac{\sum (x_a - \mu_{x_a})^2}{g} \text{ and } \sigma_{y_a}^2 = \frac{\sum (y_a - \mu_{y_a})^2}{g}$$

giving

$$\Omega_a = \sigma_{x_a}^2 + \sigma_{y_a}^2$$

Similarly for the second set of points, we calculate

$$\Omega_b = \sigma_{x_b}^2 + \sigma_{y_b}^2$$

Finally, the estimated location of the unlocalized node is the average of the points with the minimum $\Omega_i, i = a, b$.

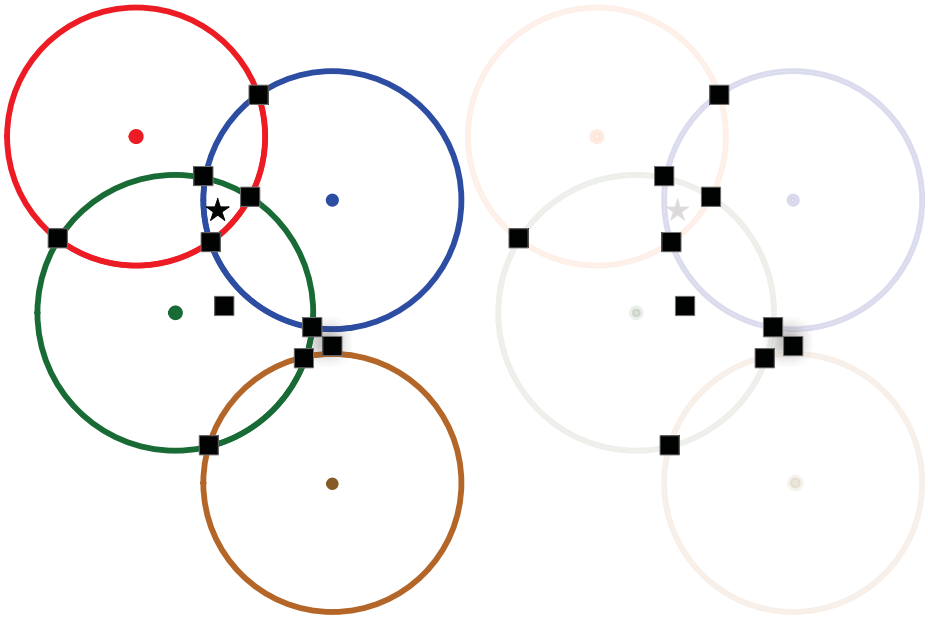
5.2.3 Performance Results for DBLDT and VBLDT

Here, the proposed algorithms DBLDT and VBLDT are compared with the WLS-SVD [52] and LLS [51] algorithms via simulation. We first consider distance variance to measure the accuracy of both techniques. 100 nodes are deployed, 50% of which are randomly chosen as anchors. The deployment area is $A = 100 \times 100$ meters^2, and the range is $r = 10$ meters. The distance error has a Gaussian distribution with variance that is a percentage of the actual distance. As a performance measure, we use the mean error, which is defined as

$$\text{mean error} = \frac{\sum_{t} \sum_{i=1}^{u} \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2}}{tu} \tag{5.9}$$

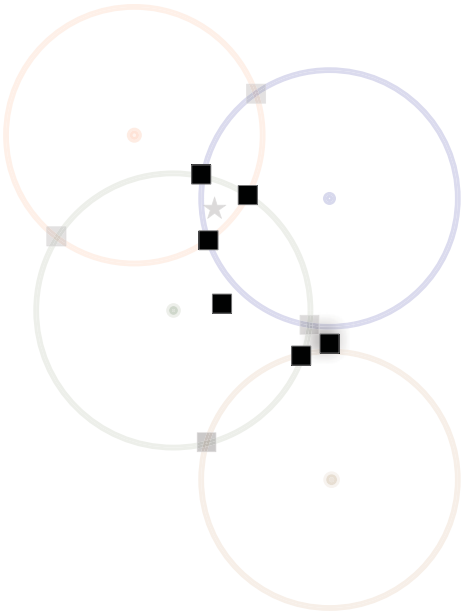
where u is the number of unlocalized nodes, t is the number of trials, $(\hat{x}, \hat{y})$ is the estimated unlocalized node position, and (x, y) is the actual position. The results were averaged over 10^5 trials. Localized nodes were used with the anchors to localize those unlocalized nodes that were not in the range of three anchors or more in the previous iterations.

The localization process ends when all nodes are localized or all remaining unlocalized nodes are isolated, i.e., not in the range of any anchor or localized nodes. Fig. 5.6 shows the mean error, where both proposed algorithms outperform LLS



(a) Step 1: Distance estimates for an unlocalized node from four anchors.

(b) Step 2: The intersection points.



(c) Step 3: The candidate intersection points.

Figure 5.5: The proposed algorithm with four anchor nodes.

and WLS-SVD, and DBLDT outperforms VBLDT. Also, the rate of change of the error is lower with the proposed algorithms.

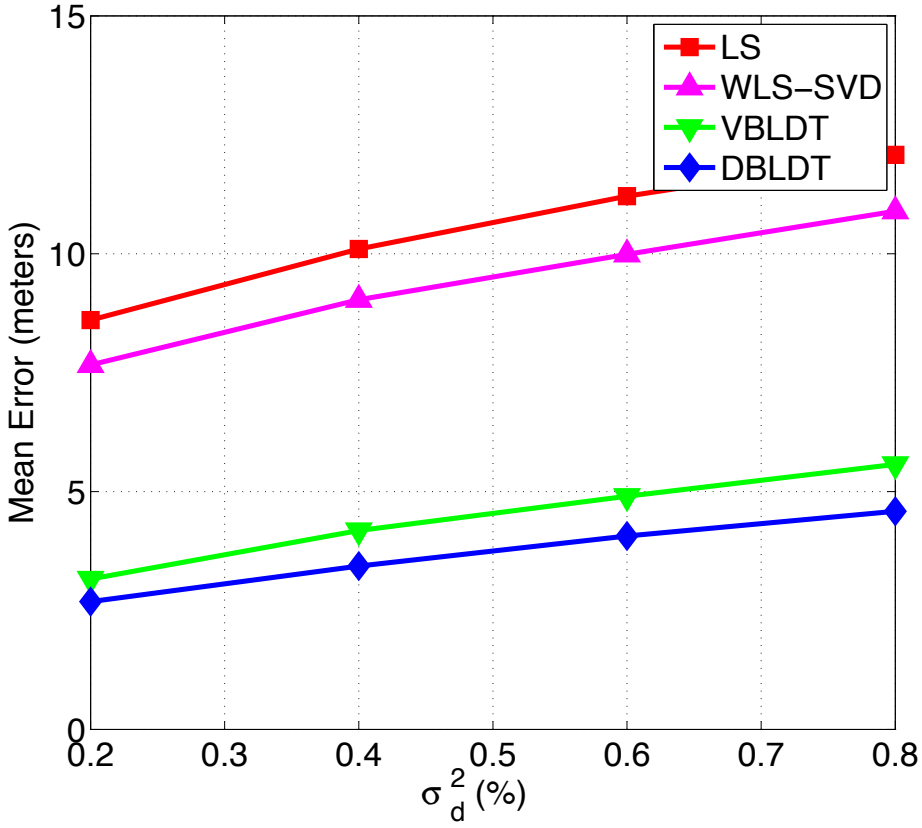


Figure 5.6: Mean error versus distance variance.

Next, all algorithms are compared considering the transmission range of the wireless nodes. The deployment area, the number of nodes, and the number of anchors are the same as before, but the transmission range varies from 10 to 50 meters. The distance error variance is fixed at 10% of the actual distance between nodes. The results were again averaged over 10^5 trials. Fig. 5.7 shows that the proposed algorithms perform better than LLS and WLS-SVD at low transmission ranges, where the unlocalized nodes can be reached by a small number of nodes. However, at high transmission ranges, the DBLDT algorithm is the best while VBLDT is the worst. The other algorithms have similar performance.

The anchor ratio is one of the most important factors affecting localization accuracy. Thus, we next compare the algorithms with a varying percentage of anchor

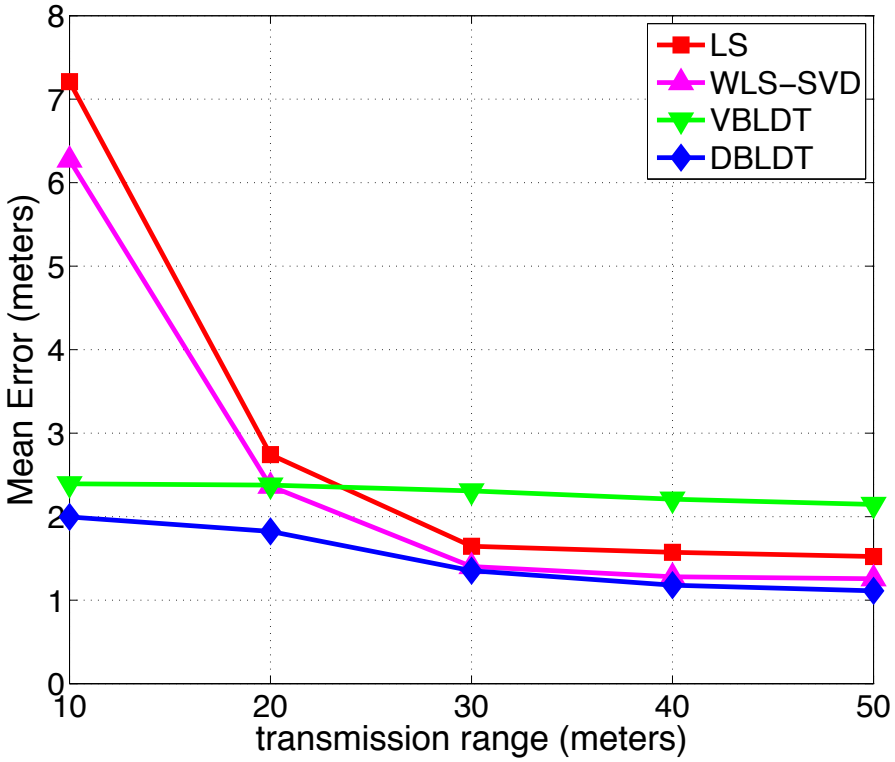


Figure 5.7: Mean error versus transmission range.

nodes. In this case, we are more interested in the performance when the anchor ratio is small because, in a practical system, the number of anchor nodes will be much less than the number of unlocalized nodes. The deployment area and the number of nodes are the same as before but the anchor ratio varies from 20% to 80%. The transmission range is fixed at 10 meters and the distance error variance is fixed at 10% of the actual distance. The results are again averaged over 10^5 trials. Fig. 5.8 shows that the proposed algorithms again outperform both the LLS and WLS-SVD algorithms, particularly at low anchor ratios.

At this point, we consider the effect of the node geometry on performance, with GGDOP used as the geometry measure. Three anchor nodes are deployed on a circle with a fourth unlocalized node in the center. The transmission range

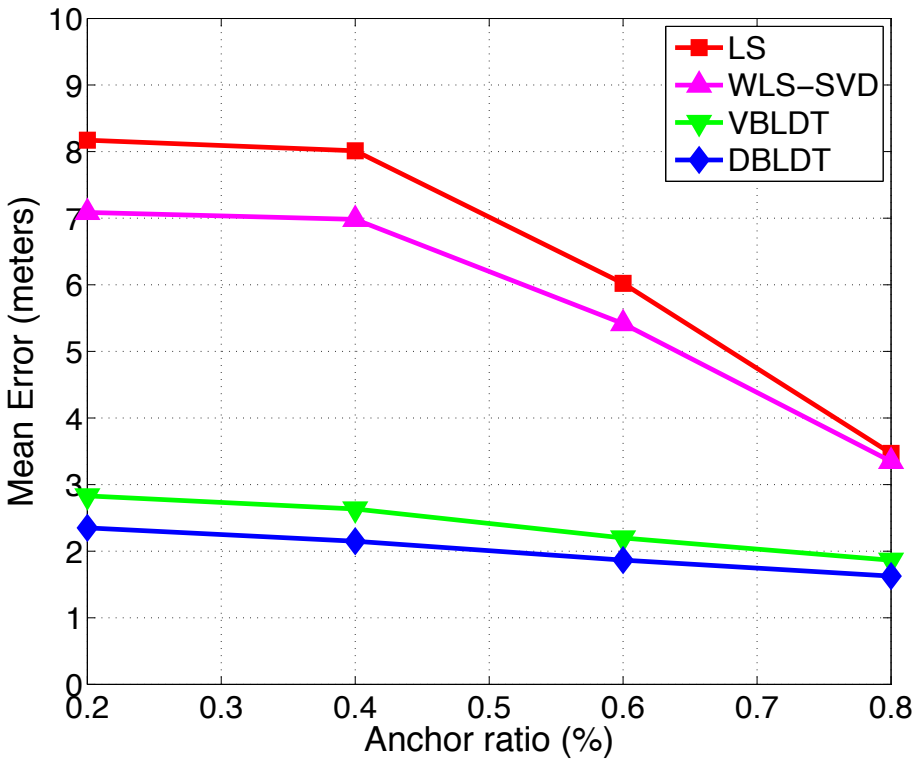


Figure 5.8: Mean error versus anchor ratio.

is set to 30 meters to ensure that the unlocalized node is within range of the three anchors, and the distance error variance is set at 10%. The anchors a_1, a_2, and a_3 are distributed around the unlocalized node u by changing the angles $\measuredangle a_1 u a_2$ and $\measuredangle a_2 u a_3$ from 1° to 101° (with both angles the same). The results are averaged over 10^5 trials, and are shown for GGDOP in Fig. 5.9 and for angle between anchors in Fig. 5.10. Both figures show that the proposed algorithms perform better, particularly with poor geometry, i.e., low GGDOP or small angles. At small angles, we have poor geometry. As they get larger, we approach the ideal case in which the anchors are distributed uniformly on the circle. Note that at an angle of 101°, all four algorithms perform similarly.

Finally, we evaluated the algorithms with different anchor ratios and distance error variances. Fig. 5.11 shows the resulting mean error surfaces. DBLDT clearly

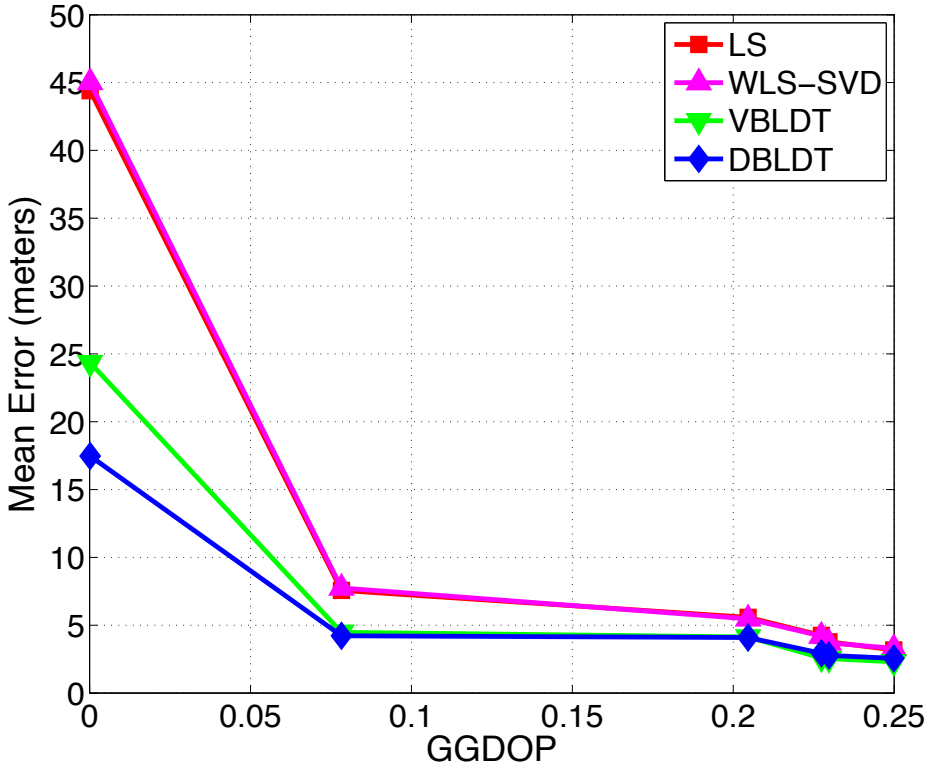


Figure 5.9: Mean error versus GGDOP.

outperforms the other algorithms, followed by VBLDT, which outperforms WLS-SVD and LLS at high distance error variances (90%) and low anchor ratios (10%), which is a very practical but challenging environment. The performance is similar at high anchor ratios (90%) and low distance error variances (10%), which is close to the ideal case.

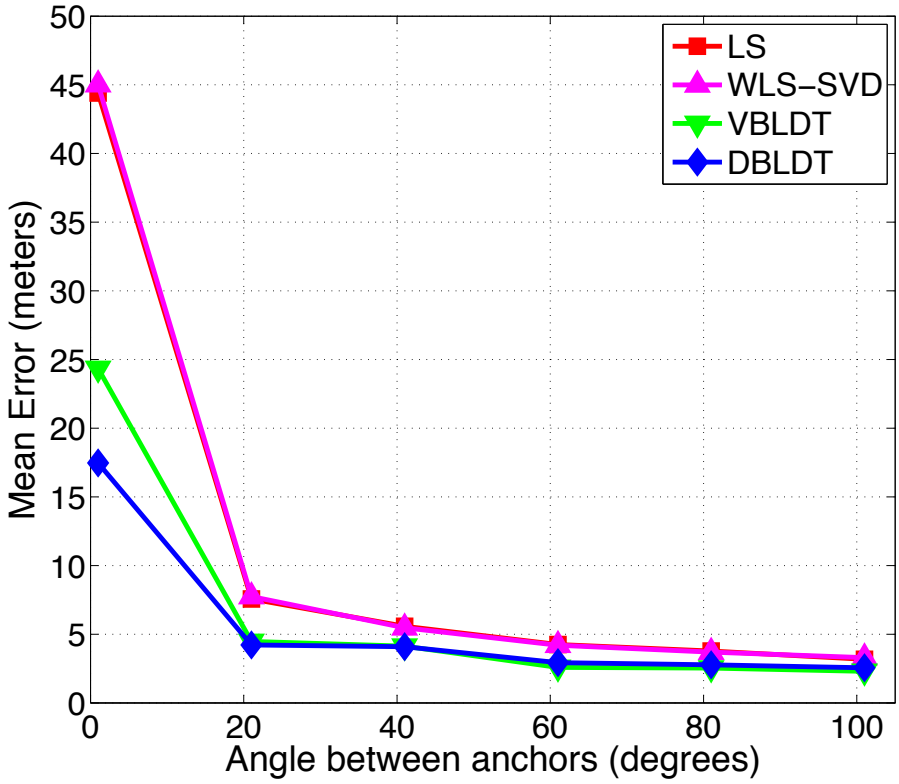
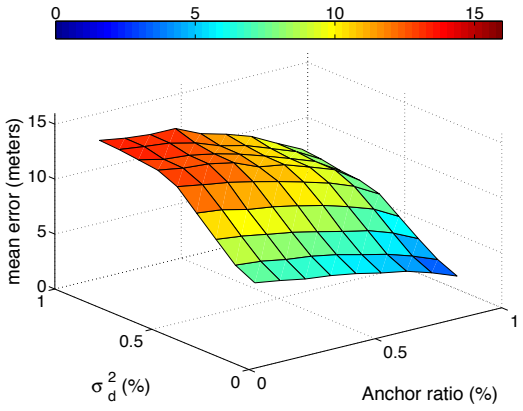
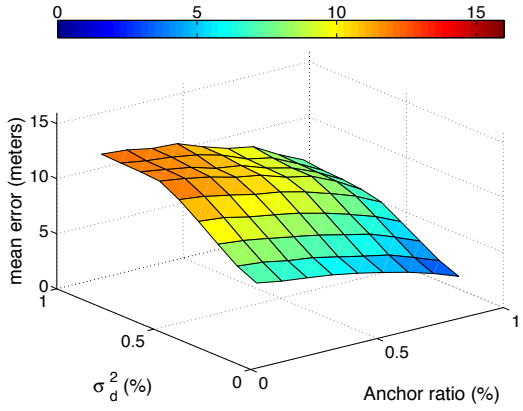


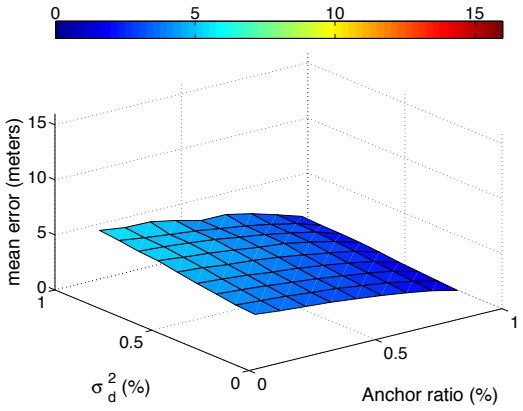
Figure 5.10: Mean error versus angle between anchors.



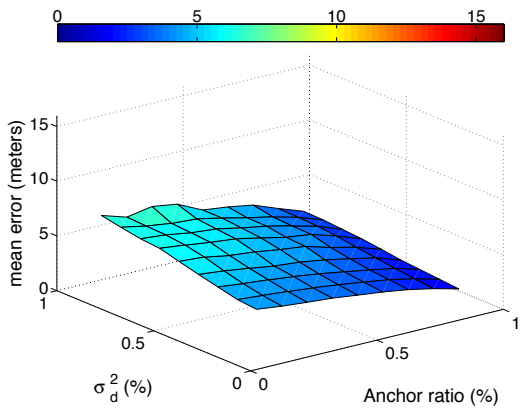
(a) Mean error surface for the LLS algorithm.



(b) Mean error surface for the WLS-SVD algorithm.



(c) Mean error surface for the DBLDT algorithm.



(d) Mean error surface for the VBLDT algorithm.

Figure 5.11: Mean error surfaces.

5.3 Localization using the K-means Clustering Algorithm

The proposed approach (LKmeans) differs from conventional solutions to the localization problem in wireless networks. Two approaches from data mining, called decision tree classification and K-means clustering, are used to retain the best (candidate) points, and these are averaged to get the estimated location of the unlocalized node.

5.3.1 K-means Clustering Algorithms

K-means defines a cluster or class in terms of a centroid, which is the mean of a group of nodes [2]. The K-means clustering algorithm is a simple algorithm. The algorithm starts by selecting K points randomly as centroids, then it tests all other nodes by calculating the distance between each point and each K initial centroid. Each point is assigned to be a member of the cluster it is close to. In the seconds iteration, the centroids are updated by taking the average of all cluster members as new centroids, then we assign all the nodes again. This process is repeated until no point change clusters. The K-means algorithm is shown in Alg. 1. Actually, K-means has difficulty detecting clusters that have non-spherical shapes or widely different sizes or densities [2]. However, in the scenario described in this dissertation, K-means does a good job because practically the number of inner intersection points is small, as it depends on how many anchors within a range of an unlocalized node in a randomly deployed wireless network. On the other hand, K-means accuracy can be boosted by carefully selecting the initial centroids at the first iteration. In the proposed algorithm the densities of all inner intersection points are calculated, and the points that have the minimum and maximum density are se-

lected as initial centroids, i.e $K = 2$ where K is the number of the wanted clusters. Our aim is to cluster the points that are close to the actual position of the unlocalized node, and to cluster the far points, which are known also as outliers, together. Q is the number of wanted nearest neighbours in the K-nearest neighbours (KNN) algorithm (to differentiate between the other K in the K-means algorithm), and the density is given by [2], which is the inverse of the mean distance to the KNN of point p

$$density\,(p,Q) = \left(\frac{\sum_{y \in N(p,Q)} d\,(p,y)}{|N\,(p,Q)|}\right)^{-1} \tag{5.10}$$

where $N\,(p,Q)$ is the set containing the Q-nearest neighbours of point p, $|N\,(p,Q)|$ is the size of that set, and y is a nearest neighbour.

Algorithm 1 Basic K-means algorithm

1: Select K points as initial centroids.
2: **repeat**
3: Form K clusters by assigning each point to its closest centroid.
4: Recompute the centroid of each cluster by averaging all the points of each cluster.
5: **until** Centroids do not change

5.3.2 LKmeans Localization Algorithm

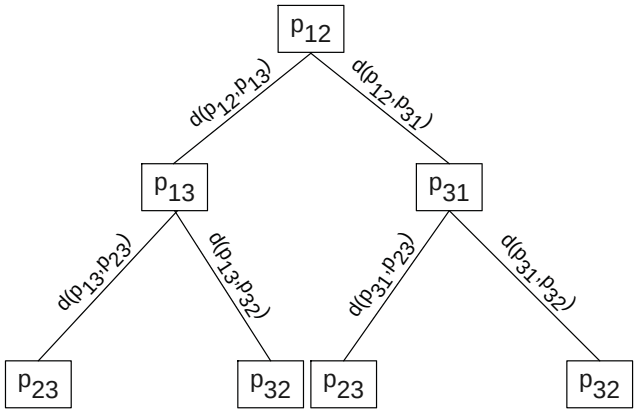
The first step in localization is to obtain the distance estimates for the unlocalized nodes from the anchor and localized nodes that are within range. These estimates provide the radii for circles around the nodes. The intersection of these circles for an unlocalized node is the next step towards calculating the final location estimate. The key is to choose candidate intersection points that are closest to each other. In the ideal case, the circles intersect on the unlocalized node. For example, when we have three anchors, three intersection points lie on the unlocalized node while the

other three do not. However, this event is unlikely in practical situations.

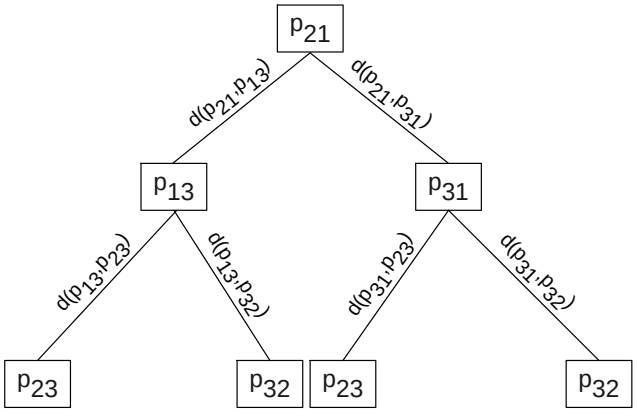
Each unlocalized node estimates the distance from each anchor or localized node that it can receive a signal from. This node can estimate its position only if it hears from three or more of these nodes. The intersections between all estimates around the unlocalized node produce a set of intersection points. If we have m anchor and/or localized nodes, then they form g groups where g is given by (5.8). Each group consists of two points as a result of the intersection between two anchor and/or localized node estimates, as shown in Fig. 5.3 in Section 5.2.

The third step in estimating the location of an unlocalized node is constructing two decision trees to select the inner intersection points. The root or starting point in both trees can be chosen arbitrarily, but they should belong to the same group. In Fig. 5.12, the root points are p_{12} and p_{21}, and they are produced by the intersection of the Anchor_1 and Anchor_2 range estimates. The subtrees in this figure correspond to three anchor and/or localized nodes around the unlocalized node. To traverse from the root points down to the leaf points, we choose the branches with the shortest distance. If all anchor and/or localized nodes intersect, the maximum number of points is $2g$, and we need to select g inner points, one from each group. The Euclidean distance between two points is used as the decision tree metric. The path with the minimum distance is followed in each subtree. Suppose that in Fig. 5.12a the selected path has points p_{12}, p_{13}, and p_{32}, and in Fig. 5.12b the selected path has points p_{21}, p_{31}, and p_{23}. Then we have two sets each consists of g points, one from each subtree. The set with the minimum total distance is chosen as inner intersection points, which is given by:

$$D_s = \sum_{i=1}^{g} \sum_{j=1, j>i}^{g} d\left(p_i, p_j\right) \tag{5.11}$$



(a) Subtree 1



(b) Subtree 2

Figure 5.12: Decision tree for three anchor nodes divided into two subtrees.

$$\mathbf{c} = \arg\min_{s} (D_s) \tag{5.12}$$

where s is the number of the set and $\mathbf{c}$ is the selected set of inner points.

When, for example, a distance estimate is obtained by measuring the received signal strength (RSS), it may be short due to the non-line-of-sight (NLOS) effect, and not all circles will intersect. However, the intersection points can still be calculated, although they will be complex numbers. If this occurs, we consider the real part as the intersection point in subsequent calculations.

Fig. 5.13 illustrates the steps of the proposed algorithm for four anchor nodes.

After the four distance estimates are determined, the $2g = 12$ intersection points are found, and then 6 inner intersection points are selected. Note that because two pairs of circles do not intersect, there are only 10 points in Fig. 5.13a (since the real parts of the complex numbers are the same).

Now, we apply K-means clustering with $K = 2$ to divide the inner points into dense points and outliers as a fourth step. To calculate the density of each inner point, we set $Q = g - 1$, i.e., each point considers its Q nearest neighbours in calculating its density. Then, we select the cluster with the maximum number of points. If more than one cluster has the same number of points, we pick the one with the lower cluster sum of squared error (SSE) [2].

$$cluster\,SSE = \sum_{p \in C_i} d\,(c_i, p)^2 \tag{5.13}$$

where d is the distance, p is an inner intersection point, C_i is the ith cluster, c_i is the centroid of cluster C_i, and K is the number of clusters.

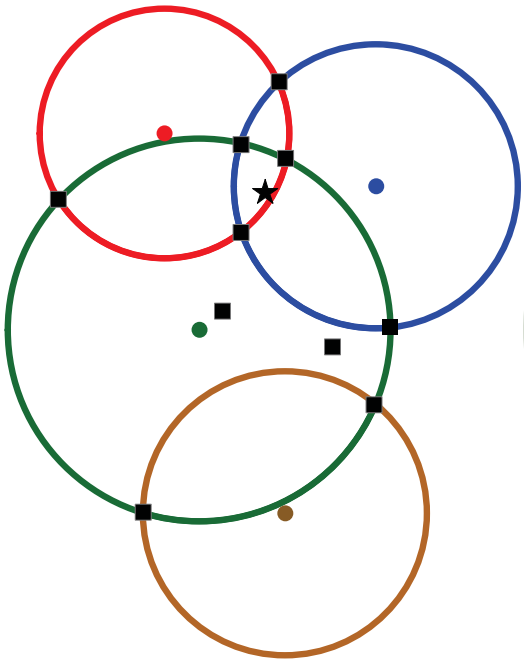
If the selected cluster is $\mathbf{p}$, where $\mathbf{p} = \{p_1, p_2, \ldots, p_q\}$ and q is the number of points inside that cluster, then estimated location of the unlocalized node is the average of that cluster with the candidate points.

$$\hat{\mathbf{u}} = \frac{\sum_{i=1}^{q} p_i}{q} \tag{5.14}$$

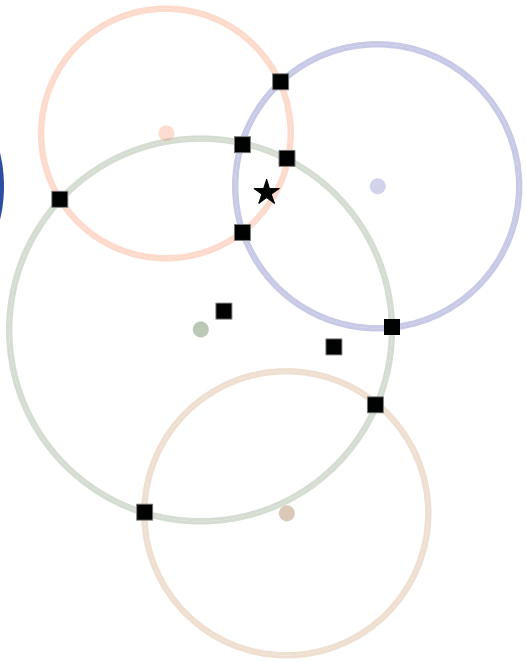
where $\mathbf{p} \subset \mathbf{c}$, and $\hat{\mathbf{u}}$ is estimated location.

5.3.3 Performance Results for LKmeans Algorithm

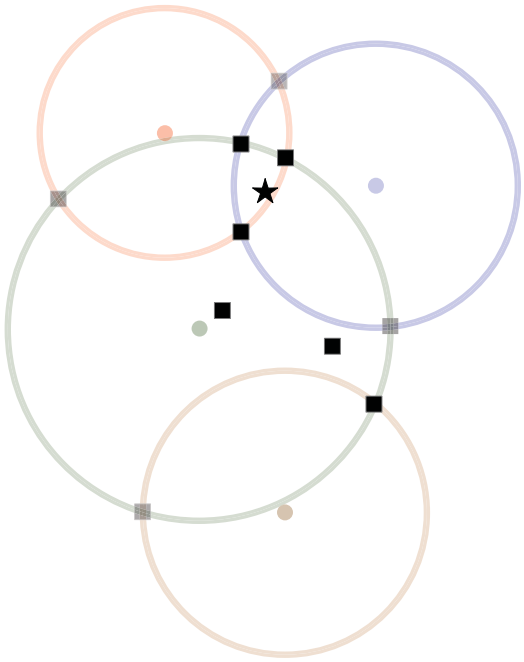
In this section, the proposed algorithm, LKmeans, is compared with the WLS-SVD [52] algorithm via simulation. We first consider distance variance to measure the



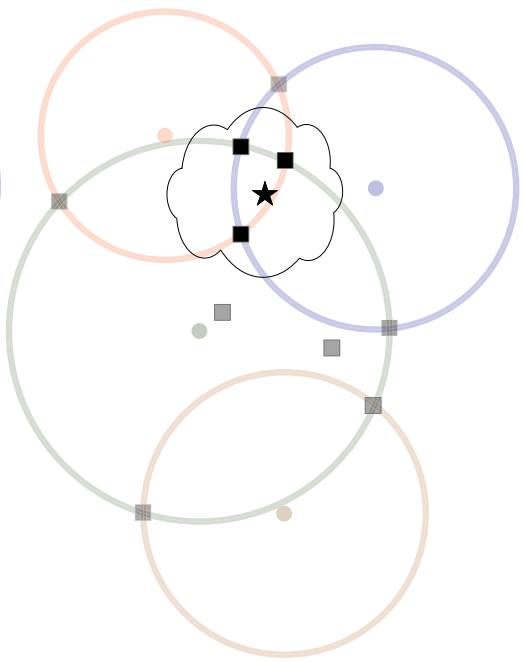
(a) Step 1: Distance estimates for an unlocalized node from four anchors.



(b) Step 2: The intersection points.



(c) Step 3: The inner intersection points.



(d) Step 4: The candidate intersection points.

Figure 5.13: The proposed algorithm with four anchor nodes.

accuracy of both techniques. 100 nodes are deployed, 50% of them are randomly chosen as anchors. The deployment area is $A = 100 \times 100$ meters^2, and the range $r = 10$ meter. The distance error has a Gaussian distribution with variance, which is a percentage of the actual distance. As a performance measure, we use the mean error, which is defined in (5.9) The results were averaged over 10^5 trials. Localized nodes were used with the anchors to localize those unlocalized nodes that were not in the range of any anchors in the previous iterations. The localization process ends when all nodes are localized or all remaining unlocalized nodes are isolated, i.e., not in the range of any anchor or localized nodes. Fig. 5.14 shows the mean error of the proposed algorithm outperform LLS and WLS-SVD algorithms. Also, the rate of change of the error is lower with the proposed algorithms.

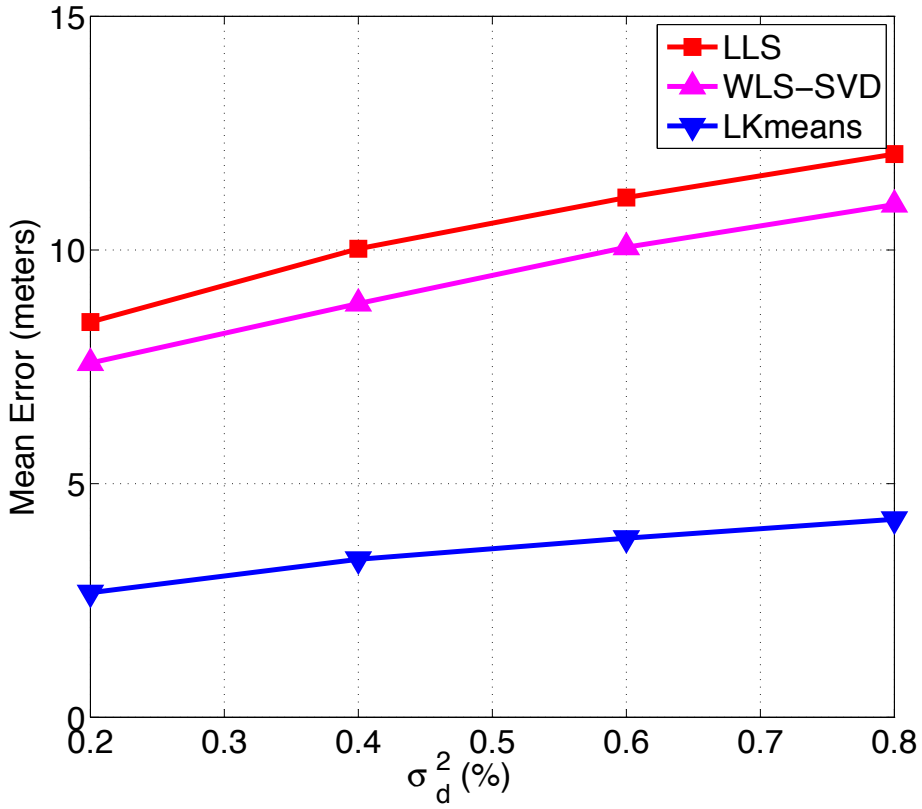


Figure 5.14: Mean error versus distance variance.

Next, all algorithms are compared considering the transmission range of the wireless nodes. The deployment area, the number of nodes, and the number of anchors are the same as before, but the transmission range varies from 10 to 50 meters. The distance error variance is fixed at 10% of the actual distance between nodes. The results were again averaged over 10^5 trials. Fig. 5.15 shows that the proposed algorithms perform better than LLS and WLS-SVD at low transmission ranges, where the unlocalized nodes can be reached by a small number of nodes and at high transmission range as well.

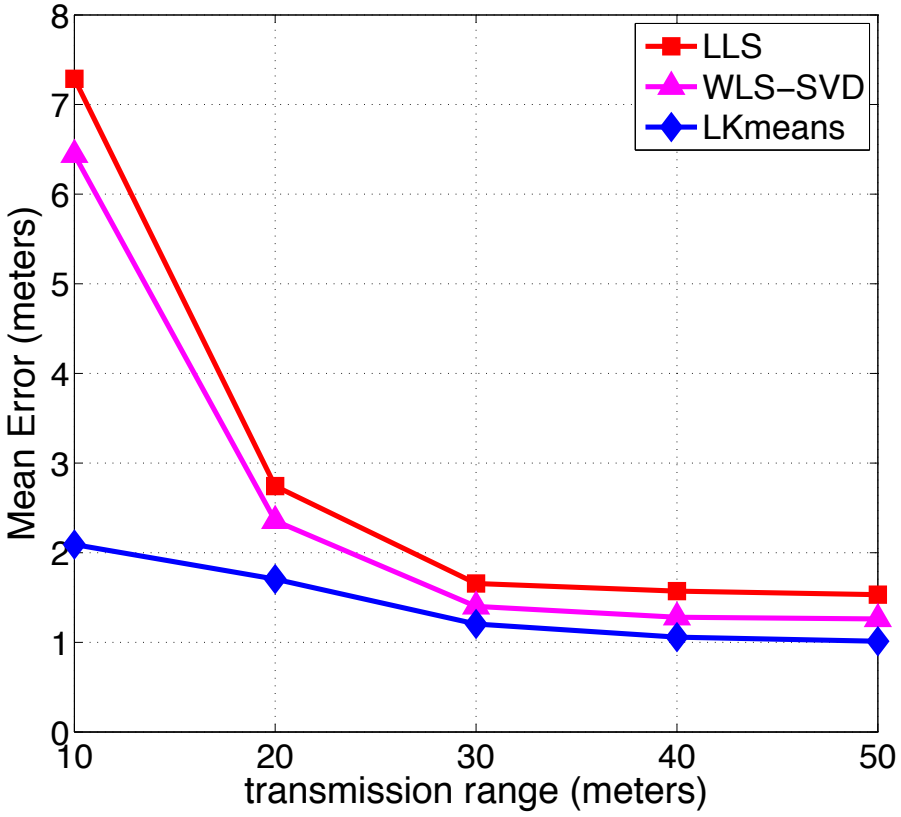


Figure 5.15: Mean error versus transmission range.

The anchor ratio is one of the most important factors affecting localization accuracy. Thus, we next compare the algorithms with a varying percentage of anchor nodes. In this case, we are more interested in the performance when the anchor ra-

tio is small because the number of anchor nodes will be much less than the number of unlocalized nodes in a practical system. The deployment area and the number of nodes are the same as before but the anchor ratio varies from 20% to 80%. The transmission range is fixed at 10 meters and the distance error variance is fixed at 10% of the actual distance. The results are again averaged over 10^5 trials. Fig. 5.16 shows that the proposed algorithm again outperforms both the LLS and WLS-SVD algorithms, particularly at low anchor ratios.

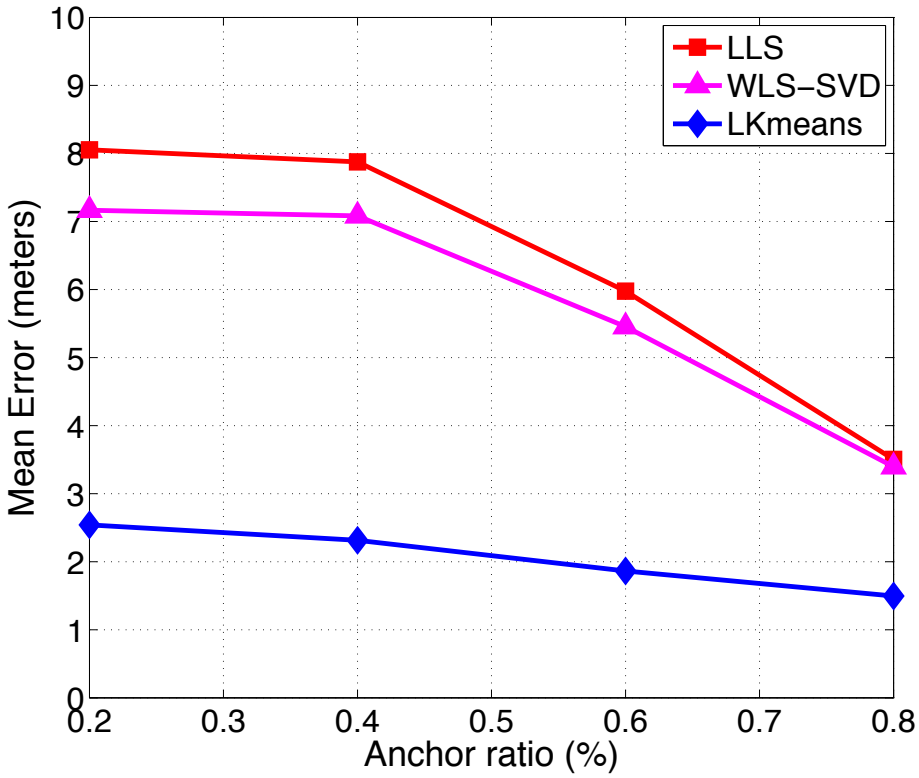


Figure 5.16: Mean error versus anchors ratio.

Next, we consider the effect of the node geometry on performance, with GGDOP used as the geometry measure. Three anchor nodes are deployed on a circle with a fourth unlocalized node in the center. The transmission range is set to 30 meters to ensure that the unlocalized node is within range of the three anchors, and

the distance error variance is set at 10%. The anchors a_1, a_2, and a_3 are distributed around the unlocalized node u by changing the angles $\measuredangle a_1ua_2$ and $\measuredangle a_2ua_3$ from 1° to 101° (with both angles the same). The results are averaged over 10^5 trials, and are shown in Fig. 5.17 for GGDOP and in Fig. 5.18 for angle between anchors. Both figures show that the proposed algorithm perform better, particularly with poor geometry, i.e., low GGDOP or small angles. LLS and WLS-SVD algorithms perform similarly at all different angles and GGDOP values because we deployed only four nodes in the field. At small angles, we have poor geometry; as they get larger, we approach the ideal case in which the anchors are distributed uniformly on the circle. Note that at an angle of 101°, all three algorithms perform similarly.

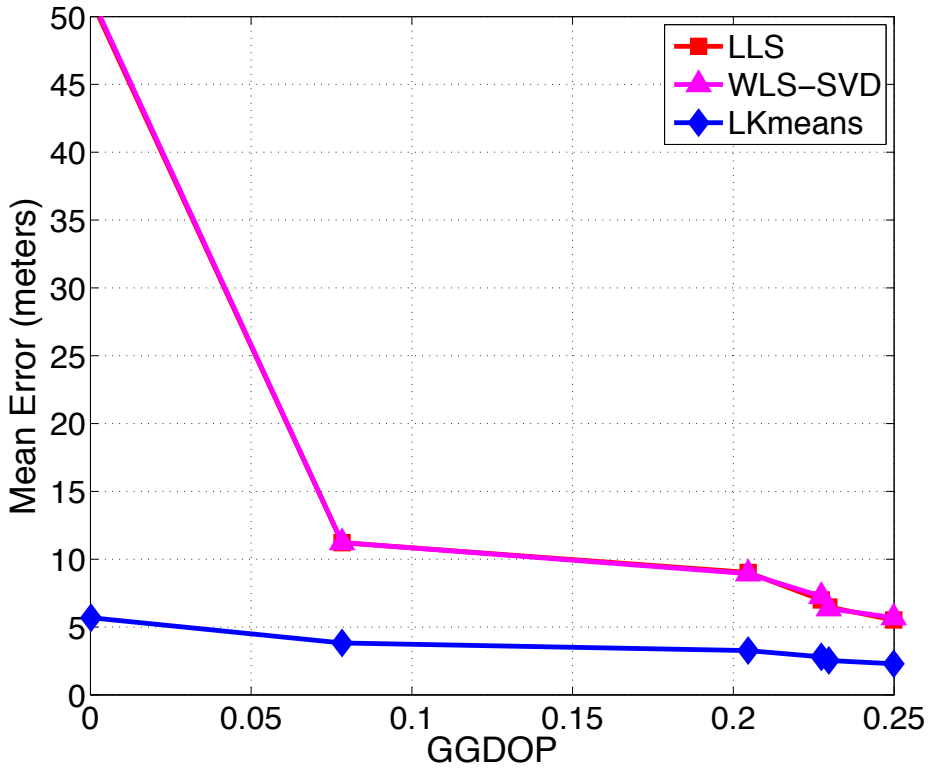


Figure 5.17: Mean error versus GGDOP.

Finally, we evaluated the algorithms with different anchor ratios and distance

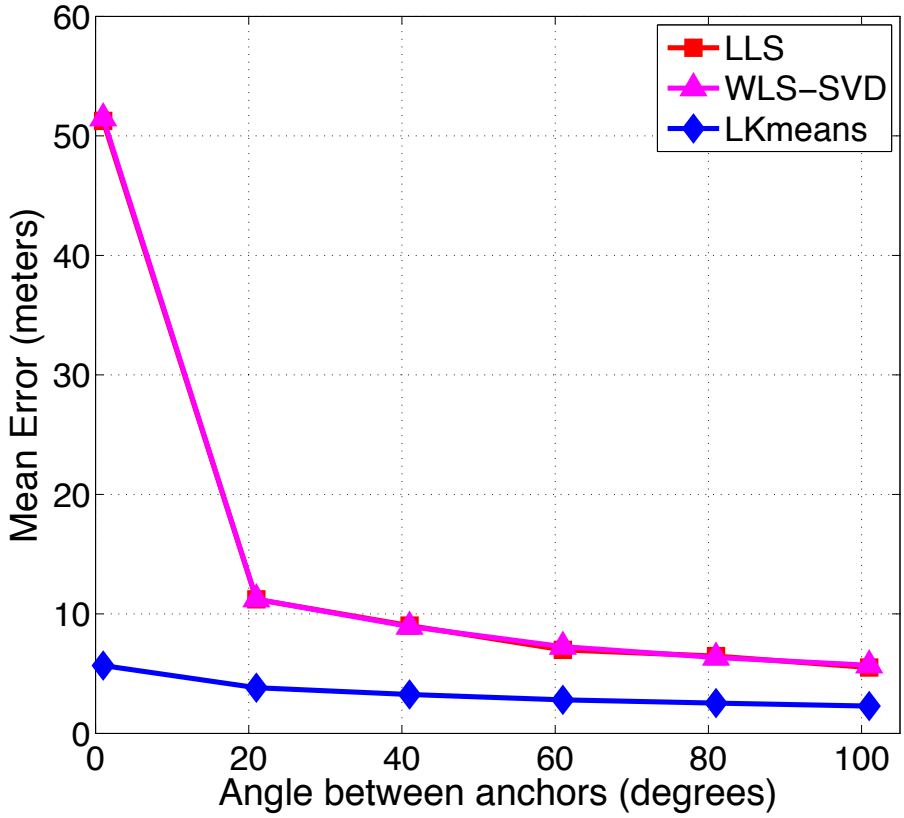
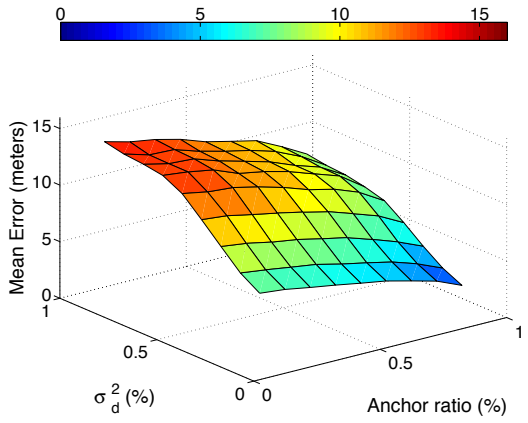
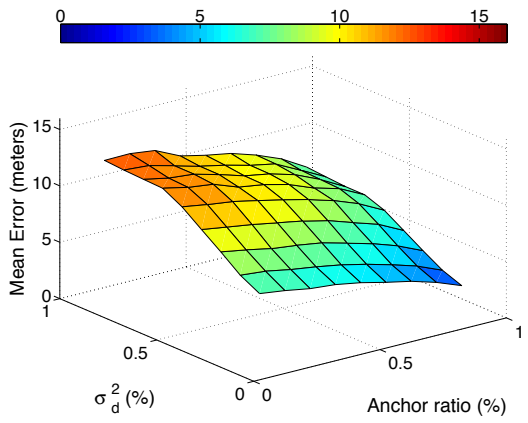


Figure 5.18: Mean error versus angle between anchors.

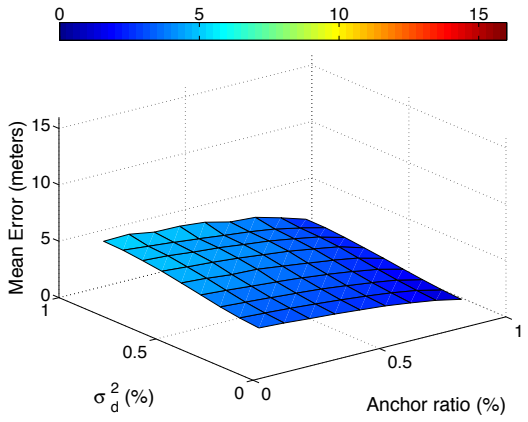
error variances. Fig. 5.19 shows the resulting mean error surfaces. Clearly LK-means outperform the other algorithms at high distance error variances (90%) and low anchor ratios (10%), which is a very practical but challenging environment. The performance is similar at high anchor ratios (90%) and low distance error variances (10%), which is close to the ideal case.



(a) Mean error surface for the LLS algorithm.



(b) Mean error surface for the WLS-SVD algorithm.



(c) Mean error surface for the LKmeans algorithm.

Figure 5.19: Mean error surfaces.

5.4 Localization using the DBSCAN Clustering Algorithm

Two approaches from data mining, called decision tree classification and DBSCAN [56] algorithm, are used to retain the best (candidate) points; and these are averaged to get the estimated location of the unlocalized node. In this proposed algorithm, using a specific density-based clustering algorithm, like DBSCAN that acts as a new key for solving the localization problem, instead of using a regular non-density-based clustering technique.

5.4.1 DBSCAN Clustering Algorithms

DBSCAN finds groups or regions of different densities, which are separated by low density regions [2]. DBSCAN classifies all the available points as core points, border points, and noise points. It has two parameters, a distance parameter *Eps* and threshold *MinPts*. Any two core points close to each other within a distance *Eps* are grouped in the same cluster. Also, any border point that is close to a core point, i.e. within a distance *Eps* is put in the same cluster as the core point. Finally, noise points are discarded. DBSCAN algorithm is given in Algorithm 2 [2].

Algorithm 2 Basic DBSCAN algorithm

1: Label all points as core, border, or noise points.
2: Discard noise points.
3: Group all core points that are within Eps of each other.
4: Make each group of connected core points into a separate cluster.
5: Assign each border point to one of the clusters of its associated core points.

Just to compare between K-means in the previous Section 5.3 and DBSCAN, 200 nodes are deployed in a field of 150×100 $meters^2$ to form two clusters as shown in Fig. 5.20. The results of K-means and DBSCAN clustering techniques are shown

in Fig. 5.21 and Fig. 5.22 respectively. It is clearly that the DBSCAN determined the two clusters more accurately than K-means clustering techniques.

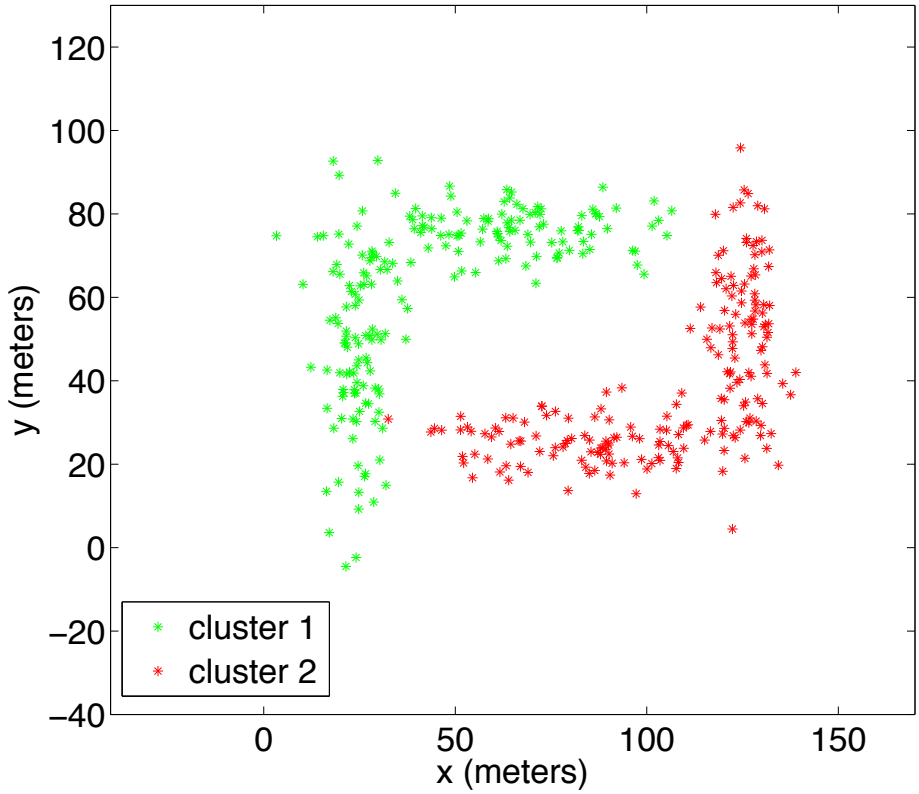


Figure 5.20: 200 nodes representing two clusters.

5.4.2 LDBSCAN Localization Algorithm

The first step in localization is to obtain the distance estimates for the unlocalized nodes from the anchor and localized nodes that are within range. These estimates provide the radii for circles around the nodes. The intersection of these circles for an unlocalized node is the next step towards calculating the final location estimate. The key is to choose the candidate intersection points that are closest to each other. In the ideal case, the circles intersect on the node.

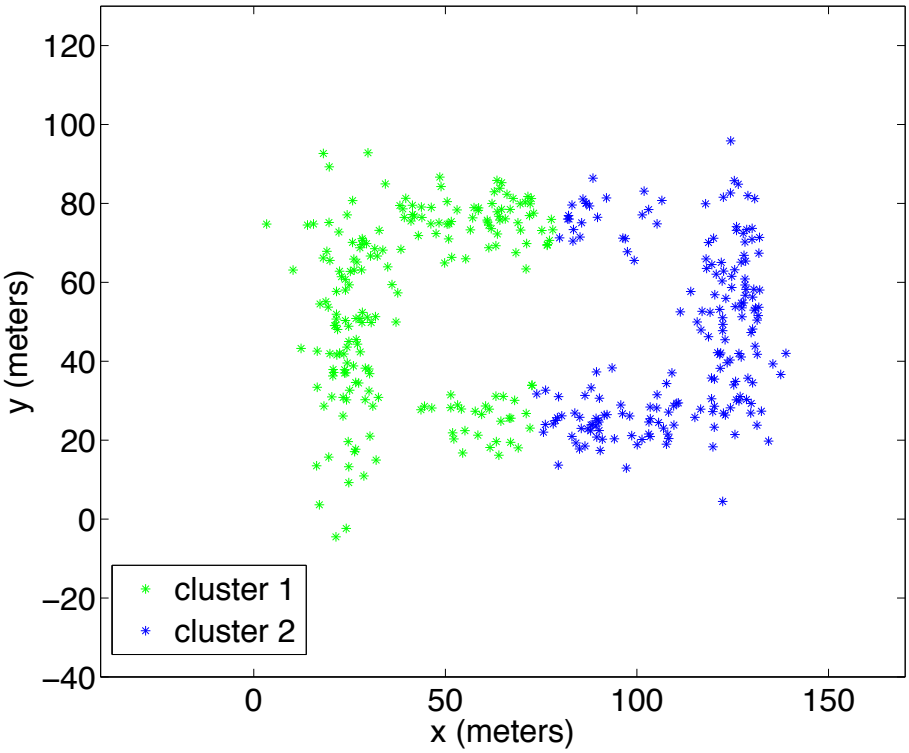


Figure 5.21: The result of applying the K-means clustering algorithm.

Each unlocalized node estimates the distance from each anchor or localized node that it can receive a signal from. This node can estimate its position only if it hears from three or more of these nodes. The intersection between all estimates around the unlocalized node produce a set of intersection points. If we have m anchor and/or localized nodes, then they form g groups where g is given by (5.8) Each group consists of two points as a result of the intersection between two anchor and/or localized node estimates, as shown in Fig. 5.3 in Section 5.2.

The third step in estimating the location of an unlocalized node is constructing two decision trees to select the inner intersection points, as shown in Fig. 5.12 in Section 5.3. The root or starting point in both trees can be chosen arbitrarily, but they should belong to the same pair of nodes. In Fig. 5.12 the root points are p_{12}

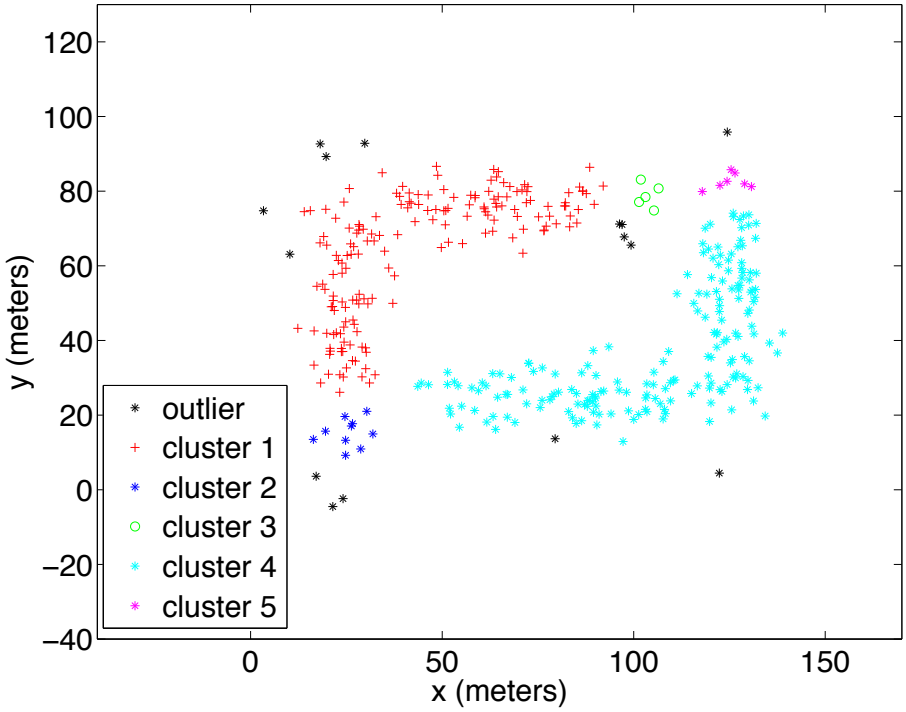


Figure 5.22: The result of applying the DBSCAN clustering algorithm.

and p_{21}, and they are produced by the intersection of the Anchor$_1$ and Anchor$_2$ range estimates. The subtrees in this figure correspond to three anchor and/or localized nodes around the unlocalized node. To traverse from the root down to the leaf points, we choose the branches with the smallest distance. If all anchor and/or localized nodes intersect, the maximum number of points is $2g$, and we need to select g inner points, one from each pair. The Euclidean distance between two points is used as the decision tree metric. The path with the minimum distance is followed in each subtree. Suppose that in Fig. 5.12a the selected path has points p_{12}, p_{13}, and p_{32}, and in Fig. 5.12b the selected path has points p_{21}, p_{31}, and p_{23}. Then, we have two sets each consisting of g points, one from each subtree. The set with the minimum total distance is chosen as the set of inner intersection points.

Fig. 5.23 illustrates the steps of the proposed algorithm for four anchor nodes.

After the four distance estimates are determined, the $2g = 12$ intersection points are computed, and 6 inner intersection points are selected. Note that because two pairs of circles do not intersect, there are only 10 points in Fig. 5.23a (since the real parts of the complex numbers are the same). The six inner intersection points are shown in Fig. 5.23c.

Now, we apply the DBSCAN algorithm to divide the inner intersection points into clusters and outliers. We select the cluster with the greatest number of points as the best estimates. The estimated location of the unlocalized node is the average of the points in this cluster (the candidate points). The two parameters used with DBSCAN are $MinPts = g - 1$ and

$$Eps = \frac{\sum_{i=1}^{2g} \sum_{j=1, j>i}^{2g} d\left(p_i, p_j\right)}{g(2g-1)} \tag{5.15}$$

where p_i and p_j are intersection points. These parameters are used to classify the points as core, border or noise. The distance to the $MinPts$ nearest neighbours is calculated for each point and this is compared with Eps. If this distance is less than Eps, it is classified as a core point, otherwise it is a border or noise point. For non-core points, if none of the points within distance Eps is a core point then it is noise, otherwise it is border point. All core and border points within Eps of each other are grouped into a cluster. The cluster with the greatest number of points is chosen, and the others are discarded.

If the selected cluster is $\mathbf{v} = \{v_1, v_2, \ldots, v_q\}$, then the estimated location of the unlocalized node is:

$$\hat{\mathbf{u}} = \frac{\sum_{i=1}^{q} v_i}{q} \tag{5.16}$$

where the v_i are the candidate points.

5.4.3 Performance Results for LDBSCAN

In this section, the proposed algorithm LDBSCAN is with the WLS-SVD [52] and LLS [51] algorithms via simulation. We first consider distance variance to measure the accuracy of both techniques. 100 nodes are deployed, 50% of which are randomly chosen anchors. The deployment area is $A = 100 \times 100$ meters^2, and the range is $r = 10$ meters. The distance error has a Gaussian distribution with variance that is a percentage of the actual distance. As a performance measure, we use the mean error, which is defined in (5.9) The results were averaged over 10^5 trials. Localized nodes were used with the anchors to localize those unlocalized nodes that were not in the range of any anchors in the previous iterations.

The localization process ends when all nodes are localized or all remaining unlocalized nodes are isolated, i.e., not in the range of any anchor or localized nodes. Fig. 5.24 shows the mean error with both proposed algorithms outperforming LLS and WLS-SVD. Also, the rate of change of the error is lower with the proposed algorithms.

Next, all algorithms are compared considering the transmission range of the wireless nodes. The deployment area, the number of nodes, and the number of anchors are the same as before but the transmission range varies from 10 to 50 meters. The distance error variance is fixed at 10% of the actual distance between nodes. The results were again averaged over 10^5 trials. Fig. 5.25 shows that the proposed algorithm perform better than LLS and WLS-SVD at low transmission ranges, where the unlocalized nodes can be reached by a small number of nodes. However, at high transmission ranges, all algorithms have similar performance.

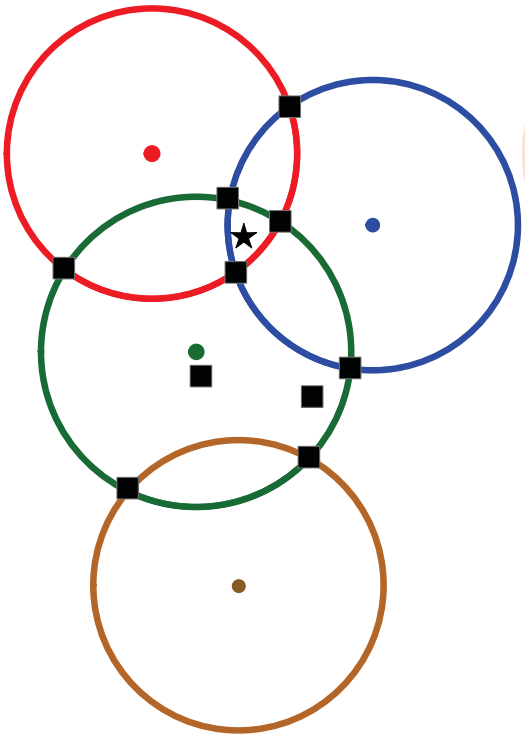
The anchor ratio is one of the most important factors affecting localization accuracy. Thus, we next compare the algorithms with a varying percentage of anchor nodes. In this case, we are more interested in the performance when the anchor ra-

tio is small because the number of anchor nodes will be much less than the number of unlocalized nodes in a practical system. The deployment area and the number of nodes are the same as before but the anchor ratio varies from 20% to 80%. The transmission range is fixed at 10 meters and the distance error variance is fixed at 10% of the actual distance. The results are again averaged over 10^5 trials. Fig. 5.26 shows that the proposed algorithm again outperforms both the LLS and WLS-SVD algorithms, particularly at low anchor ratios.

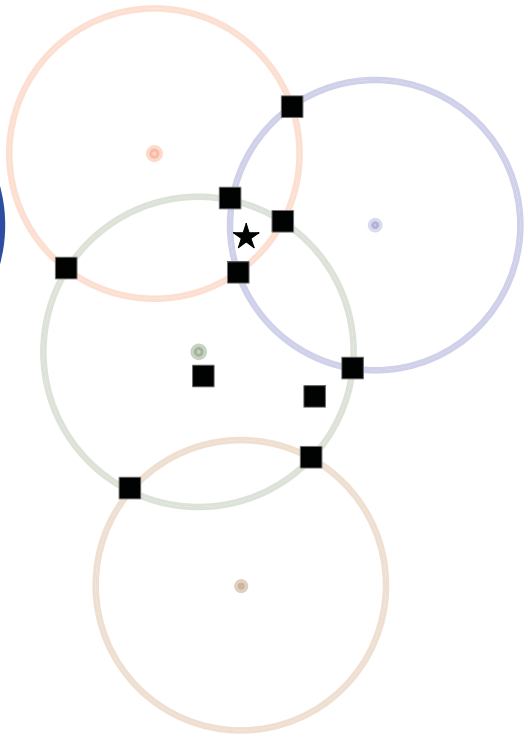
Next, we consider the effect of the node geometry on performance, with GGDOP used as the geometry measure. Three anchor nodes are deployed on a circle with a fourth unlocalized node in the center. The transmission range is set to 30 meters to ensure that the unlocalized node is within range of the three anchors, and the distance error variance is set at 10%. The anchors a_1, a_2, and a_3 are distributed around the unlocalized node u by changing the angles $\measuredangle a_1 u a_2$ and $\measuredangle a_2 u a_3$ from 1° to 101° (with both angles the same). The results are averaged over 10^5 trials, and are shown in Fig. 5.27 for GGDOP and in Fig. 5.28 for angle between anchors. Both figures show that the proposed algorithm performs better, particularly with poor geometry, i.e., low GGDOP or small angles. LLS and WLS-SVD algorithms perform similarly at all different angles and GGDOP values because we deployed only four nodes in the field. At small angles we have poor geometry, as they get large we approach the ideal case in which the anchors are distributed uniformly on the circle. Note that at an angle of 101°, all three algorithms perform similarly.

Finally, we evaluated the algorithms with different anchor ratios and distance error variances. Fig. 5.29 shows the resulting mean error surfaces. Clearly LDB-SCAN outperforms the other algorithms at high distance error variances (90%) and low anchor ratios (10%), which is a very practical but challenging environment. The performance is similar at high anchor ratios (90%) and low distance

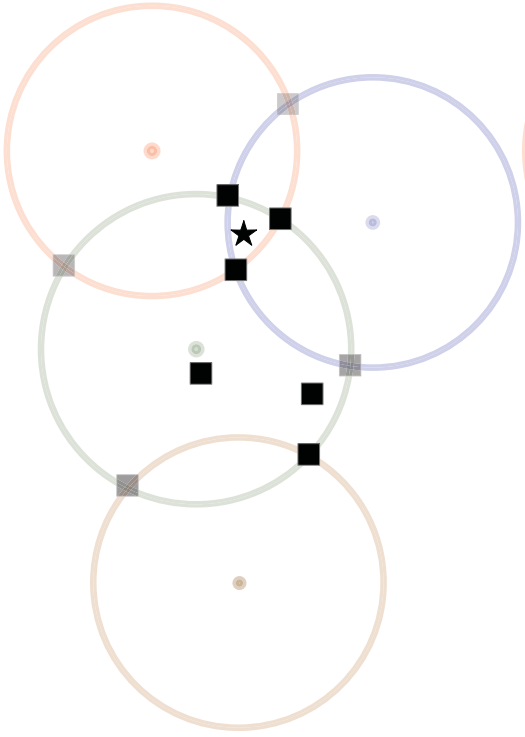
error variances (10%), which is close to the ideal case.



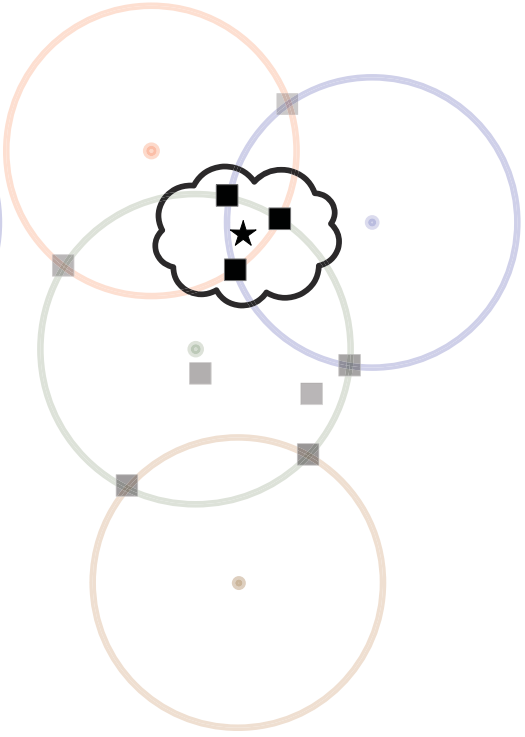
(a) Step 1: Distance estimates for an unlocalized node from four anchors.



(b) Step 2: The intersection points.



(c) Step 3: The inner intersection points.



(d) Step 4: The candidate intersection points.

Figure 5.23: The proposed algorithm with four anchor nodes.

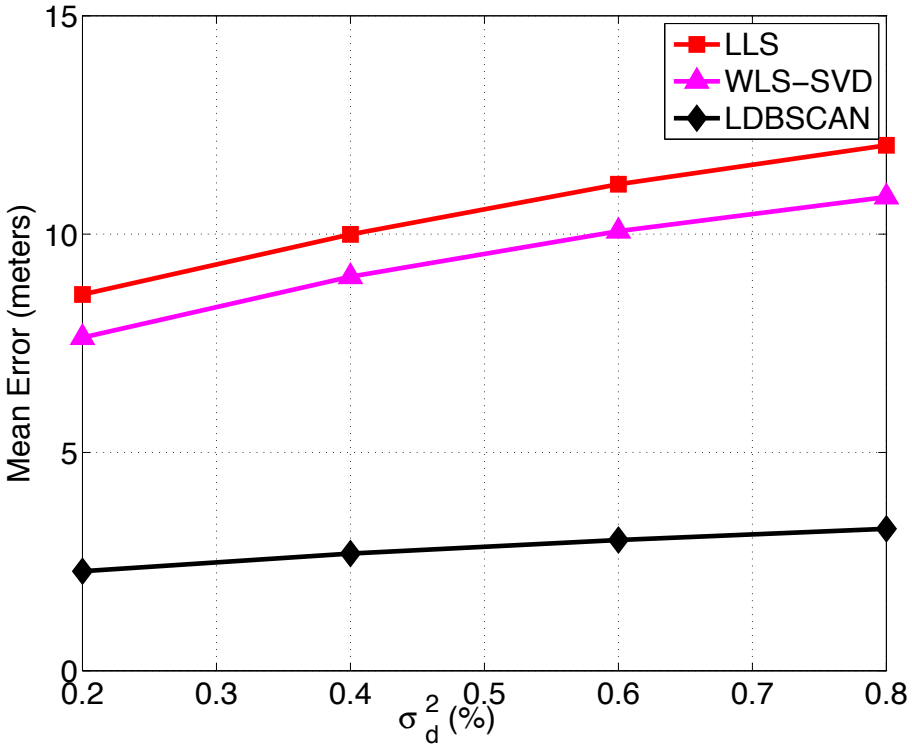


Figure 5.24: Mean error versus distance variance.

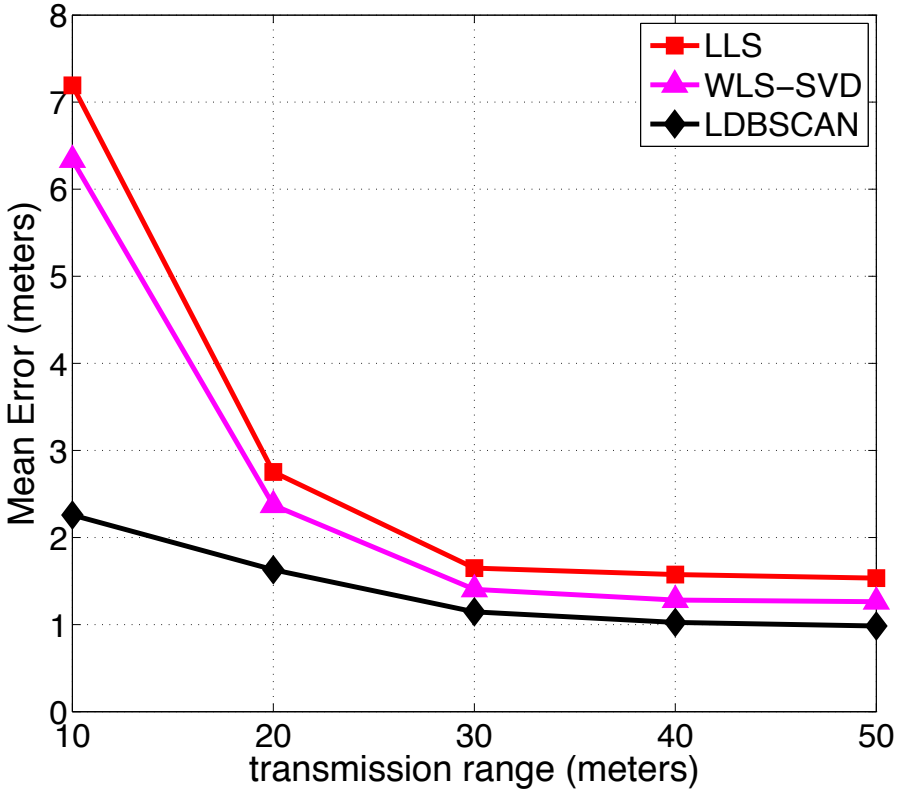


Figure 5.25: Mean error versus transmission range.

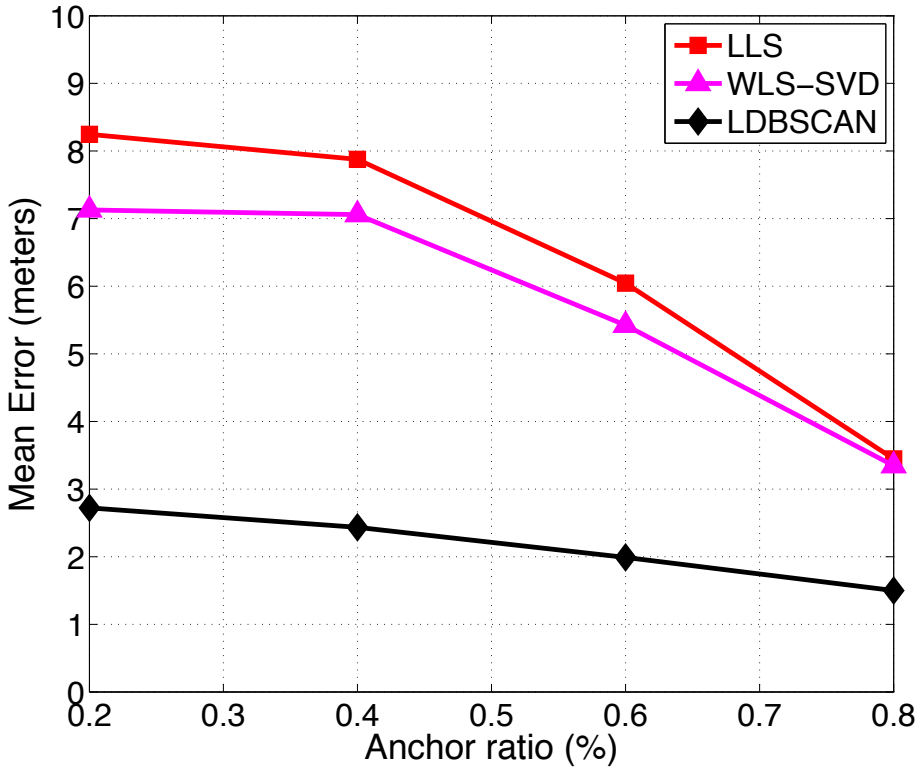


Figure 5.26: Mean error versus anchors ratio.

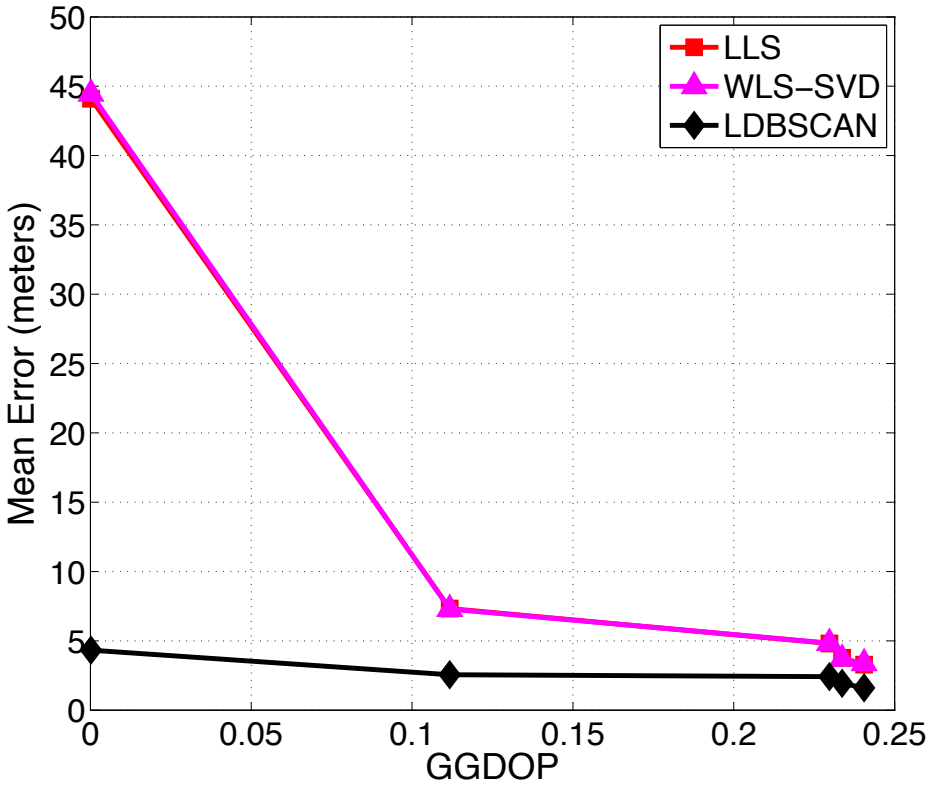


Figure 5.27: Mean error versus GGDOP.

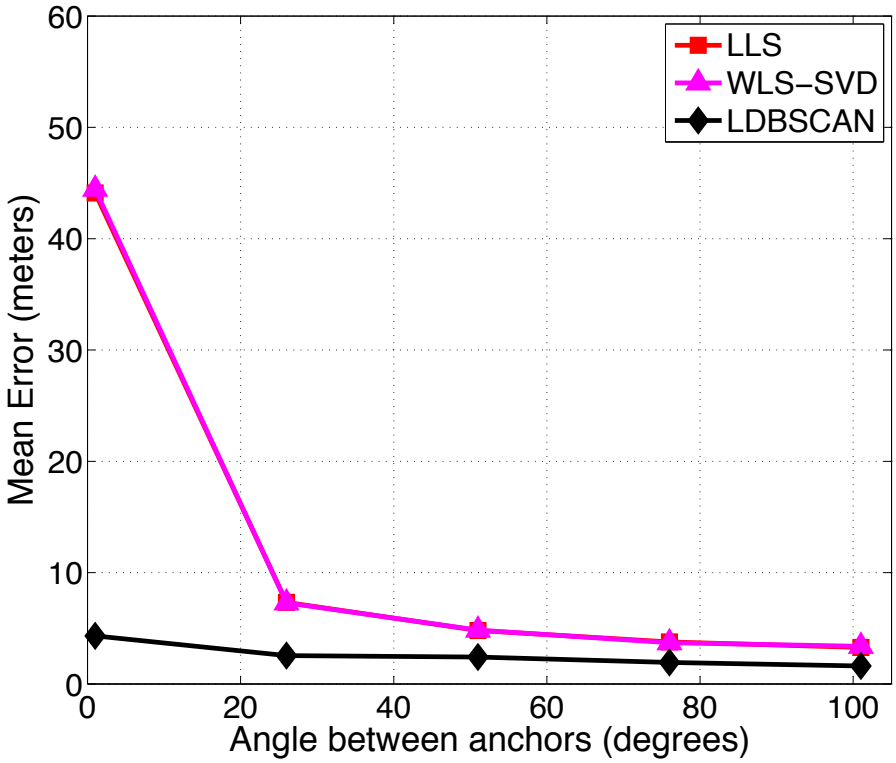
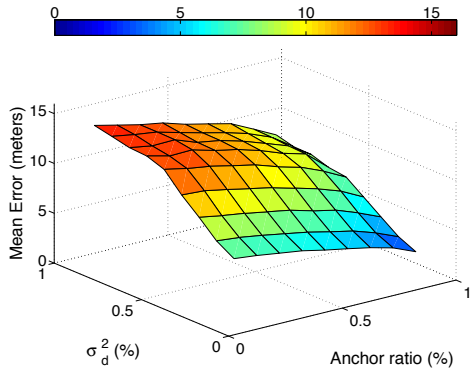
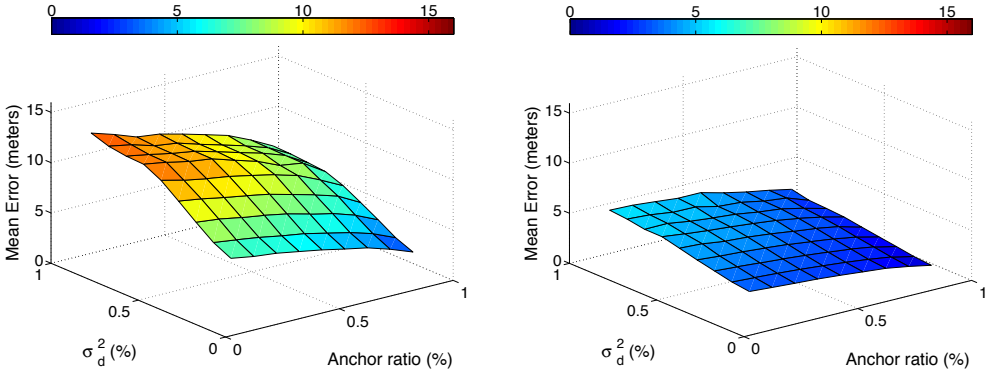


Figure 5.28: Mean error versus angle between anchors.



(a) Mean error surface for the LLS algorithm.



(b) Mean error surface for the WLS-SVD algorithm.

(c) Mean error surface for the LDBSCAN algorithm.

Figure 5.29: Mean error surfaces.

5.5 Localization using the Outlier Detection Algorithm

An approach from data mining called DBOD [2], which is based on K-nearest neighbour (KNN) [2], is used to retain the best (candidate) points, and these are averaged to get the estimated location of the unlocalized node. In this proposed method, we can see clearly that processing the intersection points based on their density saves one extra step which is classification as inner or outer points in LK-means and LDBSCAN. In proposed methods DBLDT, VBLDT, only classification via decision trees is used to classify the points as inner and outer points. In LK-means and LDBSCAN, both classification is used to choose the inner points, and clustering is used to select the candidate points.

5.5.1 Density-based Outlier Detection

The density-based outlier detection algorithm is one of the algorithms that are used in anomaly detection. The outlier score is just the inverse of the density score of a point. The density is given by [2], which is the inverse of the mean distance to the K-nearest neighbours of point p

$$density\,(p,K) = \left(\frac{\sum_{y \in N(p,K)} d\,(p,y)}{|N\,(p,K)|} \right)^{-1} \tag{5.17}$$

where $N\,(p,K)$ is the set containing the K-nearest neighbours of point p, $|N\,(p,K)|$ is the size of that set, and y is a nearest neighbour. Density-based outlier detection algorithm is given in Algorithm 3 [2].

5.5.2 LDBOD Localization Algorithm

The first step in localization is to obtain the distance estimates for the unlocalized nodes from the anchor and localized nodes that are within range. These estimates

Algorithm 3 Density-based outlier detection algorithm

1: K is the number of nearest neighbours
2: for all points p.
3: Determine $N(p,K)$ for p.
4: Determine the density of p.
5: outlier score is the inverse of the density.
6: end for

provide the radii for circles around the nodes. The intersection of these circles for an unlocalized node is the next step towards calculating the final location estimate. The key is to choose the candidate intersection points that are the closest to each other.

Each unlocalized node estimates the distance from each anchor or localized node that it can receive a signal from. This node can estimate its position only if it hears from three or more of these nodes. The intersection between all estimates around the unlocalized node produce a set of intersection points. If we have m anchor and/or localized nodes, then they form g groups where g is given by (5.8). Each group consists of two points as a result of the intersection between two anchor and/or localized node estimates. The total number of intersection points, if all anchor and/or localized nodes intersect, is $2g$. We need to select only a subset of those $2g$ points to average them and the result will be the location estimate.

The third step for towards finding an unlocalized node's location is calculating the density of each intersection point. Each intersection point find the K-nearest neighbours where $K = g - 1$ then calculate its density according to equation (5.17). Then, the points with higher density than the mean value are selected as candidates.

Fig. 5.30 illustrates the steps of the proposed algorithm for four anchor nodes. After the four distance estimates are determined, the $2g = 12$ intersection points are found. Note that because two pairs of circles do not intersect, there are only 10 points in Fig. 5.30a (since the real parts of the complex numbers are the same). Then the average density for all intersection points is calculated as:

$$D = \frac{\sum_{i=1}^{2g} density\,(p_i, K)}{2g} \tag{5.18}$$

Finally, all the points with *density* more than the average D, are selected as candidates points. If the selected candidate points form a cluster $\mathbf{p}$, where $\mathbf{p} = \{p_1, p_2, \ldots, p_q\}$ and q is the number of points inside that cluster, then the estimated location of the unlocalized node is the average of that cluster with the candidate points.

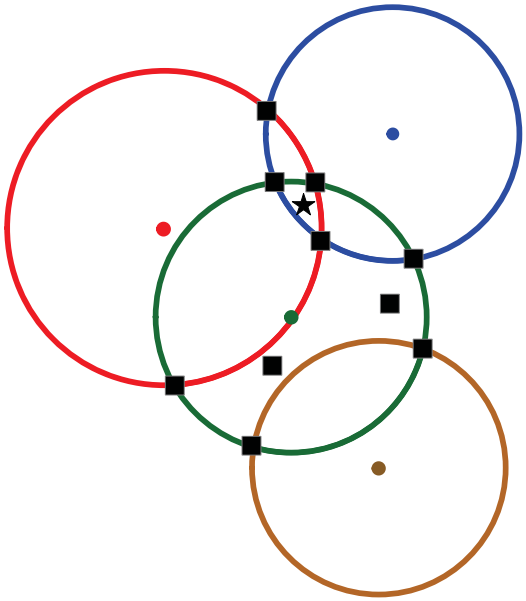
$$\hat{\mathbf{u}} = \frac{\sum_{i=1}^{q} p_i}{q} \tag{5.19}$$

5.5.3 Performance Results for LDBOD

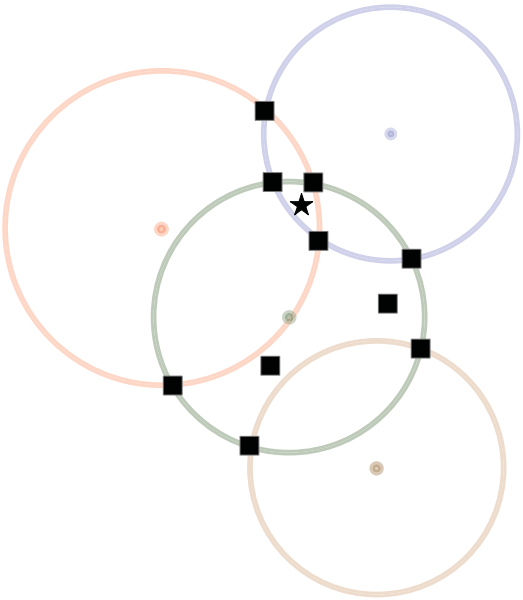
In this section, the proposed algorithm LDBOT is compared with the WLS-SVD [52] and LLS [51] algorithms via simulation. We first consider distance to measure the accuracy of both techniques. 100 nodes are deployed, 50% of which are randomly chosen anchors. The deployment area is $A = 100 \times 100$ meters^2, and the range is $r = 10$ meters. The distance error has a Gaussian distribution with variance σ_d^2, which is a percentage of the actual distance. As a performance measure, we use the mean error, which is defined in (5.9) The results were averaged over 10^5 trials. Localized nodes were used with the anchors to localize those unlocalized nodes that were not within range of a sufficient number of anchor and localized nodes in the previous iterations.

The localization process ends when all nodes are localized or all remaining unlocalized nodes are isolated, i.e., not within range of three or more anchor or localized nodes. The mean error with the proposed algorithm outperforms that with the LLS and WLS-SVD algorithms is shown in Fig. 5.31. Note that the rate of change of the error is also lower with LDBOD.

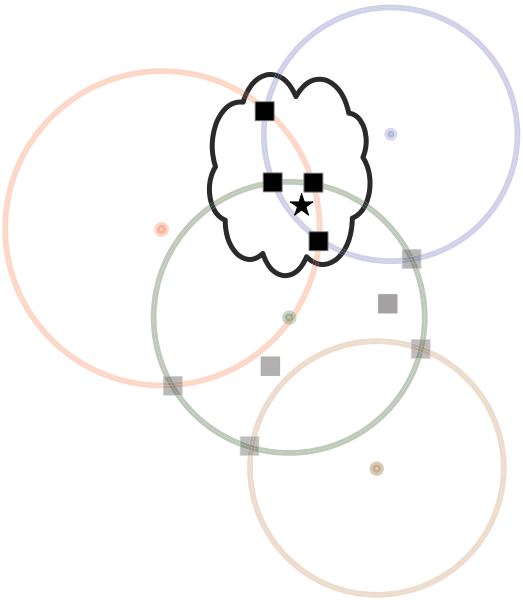
Next, all algorithms are compared considering the transmission range of the



(a) Step 1: Distance estimates for an unlocalized node from four anchors.



(b) Step 2: The intersection points.



(c) Step 3: The candidate intersection points.

Figure 5.30: The proposed algorithm with four anchor nodes.

wireless nodes. The deployment area, the number of nodes, and the number of anchors are the same as before, but the transmission range varies from 10 to 50 meters. The distance error variance is fixed at 10% of the actual distance between

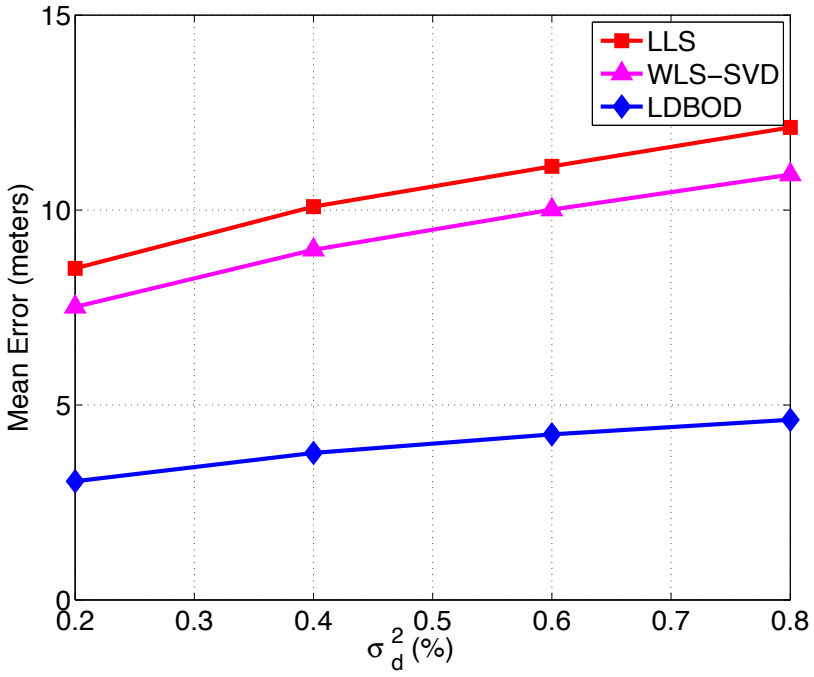


Figure 5.31: Mean error versus distance variance.

nodes. The results were again averaged over 10^5 trials. Fig. 5.32 shows that the proposed algorithm performs better than the LLS and WLS-SVD algorithms at low transmission ranges, in which case the unlocalized nodes are typically within range of a small number of nodes. However, at high transmission ranges all algorithms have similar performance.

We want to study the effect of adding localized nodes into picture to help those unlocalized nodes which have not enough number of anchors around them.

If we have N nodes which are deployed in an area A, and the transmission range of any node is r, then the probability of a node being within transmission range of other node is

$$\Pr = \frac{\pi r^2}{A}$$

The probability of a node having degree δ, i.e., δ anchor nodes within its range

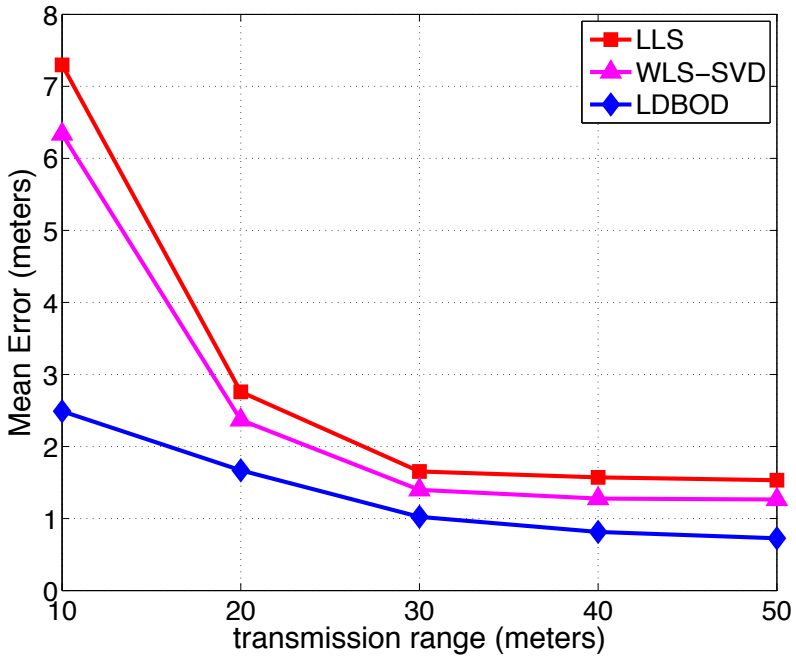


Figure 5.32: Mean error versus transmission range.

is given by

$$p(\delta) = \frac{\lambda^{\delta}}{\delta!} e^{-\lambda}$$

where

$$\lambda = N \cdot Pr$$

and N is the number of anchor nodes.

The probability of a node having n or more anchor nodes within its range is

$$P(\delta \geq n) = 1 - \sum_{i=0}^{n-1} p(i)$$

The probability of an unlocalized node having three or more anchors around it based on a given transmission range is shown in Fig. 5.33. We can see clearly that

using localized nodes in the localization process improves performance. When the range exceeds 25 meters, all the unlocalized nodes can be localized.

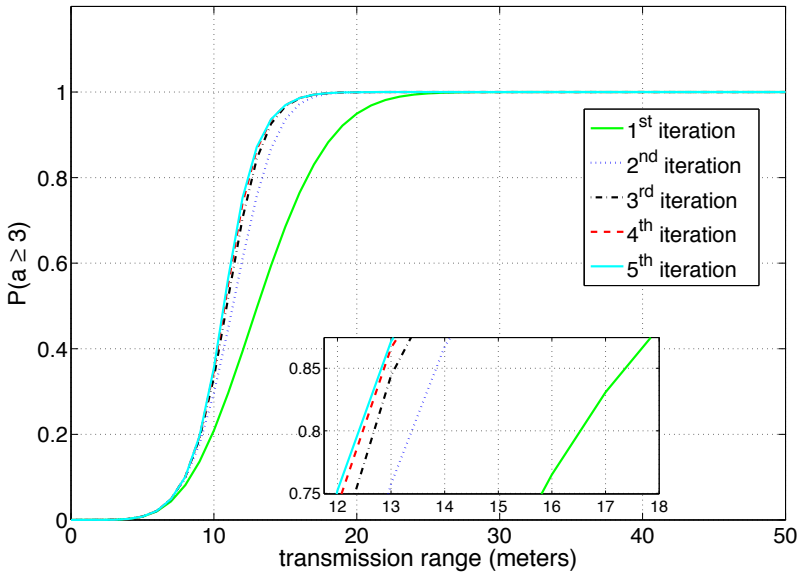


Figure 5.33: Probability of node localizability based on transmission range.

The anchor ratio is one of the most important factors affecting localization accuracy. Thus, we next compare the algorithms with a varying percentage of anchor nodes. In this case, we are more interested in the performance when the anchor ratio is small because the number of anchor nodes will be much lower than the number of unlocalized nodes in a practical system. The deployment area and the number of nodes are the same as before but the anchor ratio varies from 20% to 80%. The transmission range is fixed at 10 meters and the distance error variance is fixed at 10% of the actual distance. The results are again averaged over 10^5 trials. Fig. 5.34 shows that the proposed algorithm again outperforms both the LLS and WLS-SVD algorithms, particularly at low anchor ratios.

Also, the probability for of an unlocalized node having three or more anchors around it based on a given anchor ratio is shown in Fig. 5.35. The probability of

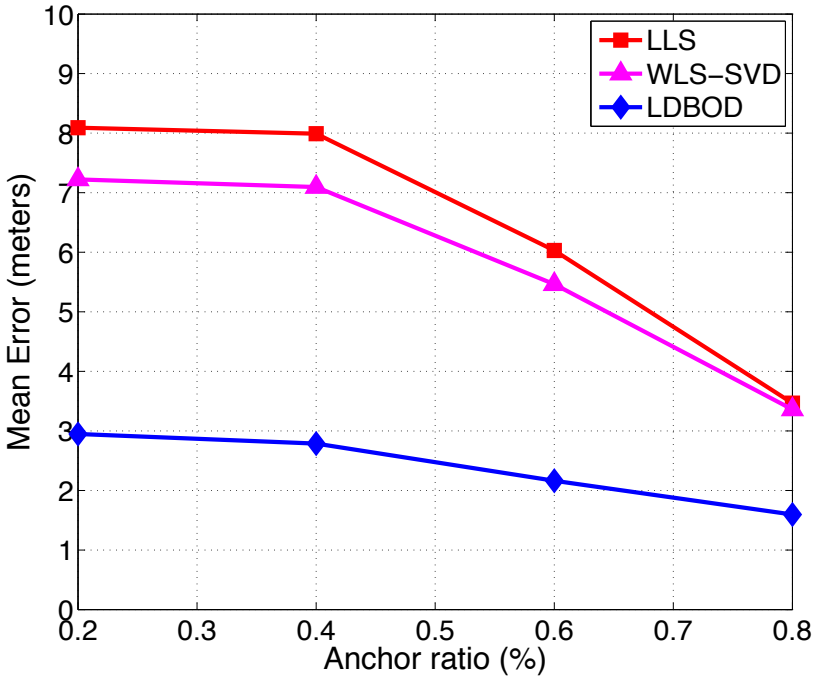


Figure 5.34: Mean error versus anchors ratio.

nodes localizability is limited by 0.6 due to the range, which is fixed at 10 meters.

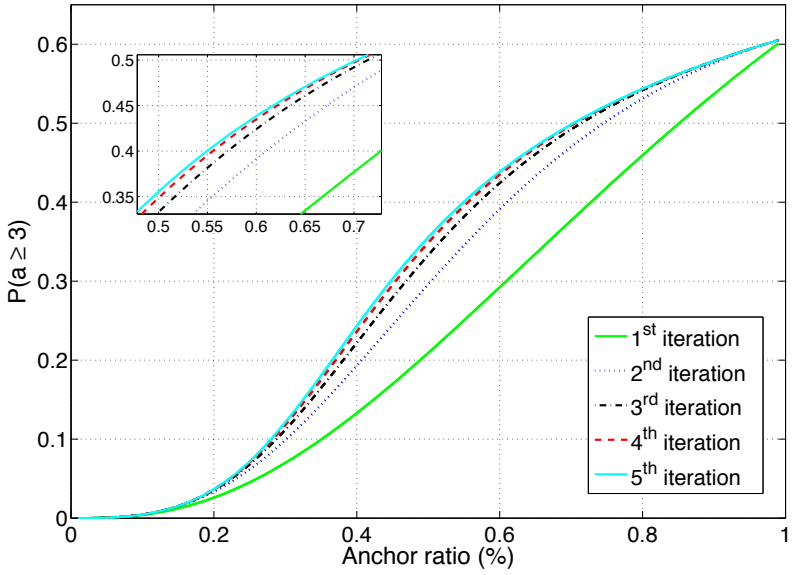


Figure 5.35: Probability of node localizability based on anchor ratio.

Next, we consider the effect of the node geometry on performance, with GGDOP used as the geometry measure. Three anchor nodes are deployed on a circle with a fourth unlocalized node in the center. The transmission range is set to 30 meters to ensure that the unlocalized node is within range of the three anchors, and the distance error variance is set to 10%. The anchors a_1, a_2, and a_3 are distributed around the unlocalized node u at angles $\measuredangle a_1 u a_2$ and $\measuredangle a_2 u a_3$, ranging from 1° to 101° (with both angles the same). The results are averaged over 10^5 trials and are shown in Fig. 5.36 for GGDOP and in Fig. 5.37 for angles between anchors. Both figures show that the proposed algorithm performs better, particularly when the geometry is poor, i.e., low GGDOP or small angles. The LLS and WLS-SVD algorithms perform similarly at all angles and GGDOP values because only four nodes are deployed. At small angles, the geometry is poor, and as the angles increase the geometry approaches the ideal case where the anchors are distributed uniformly on the circle. Thus, angles of 101° perform very well, and all three algorithms perform similarly.

Finally, we evaluated the algorithms with different anchor ratios and distance error variances. Figure 5.38 shows the resulting mean error surfaces. Clearly LDBOD outperforms the other algorithms at high distance error variances (90%) and low anchor ratios (10%), which is the most typical, but also the most challenging environment. The performance is similar at high anchor ratios (90%) and low distance error variances (10%), which is close to the ideal case, and therefore not likely to occur in practice.

Although all four proposed algorithms outperformed recent localization algorithms in literature, they differ in their performance. The four proposed algorithms are compared side by side using different performance measures. In Fig. 5.39, LDBSCAN is the best algorithm, since it uses classification and then density-based

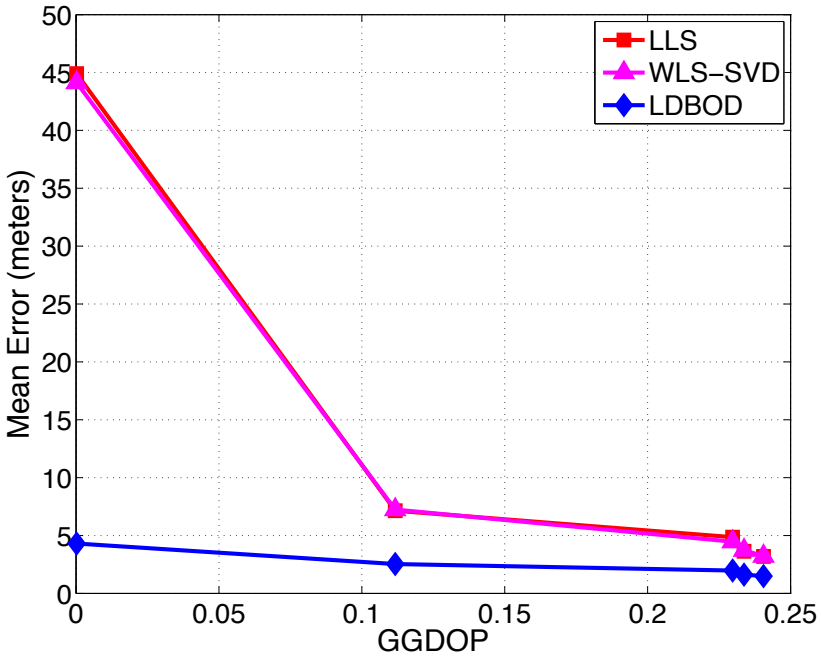


Figure 5.36: Mean error versus GGDOP.

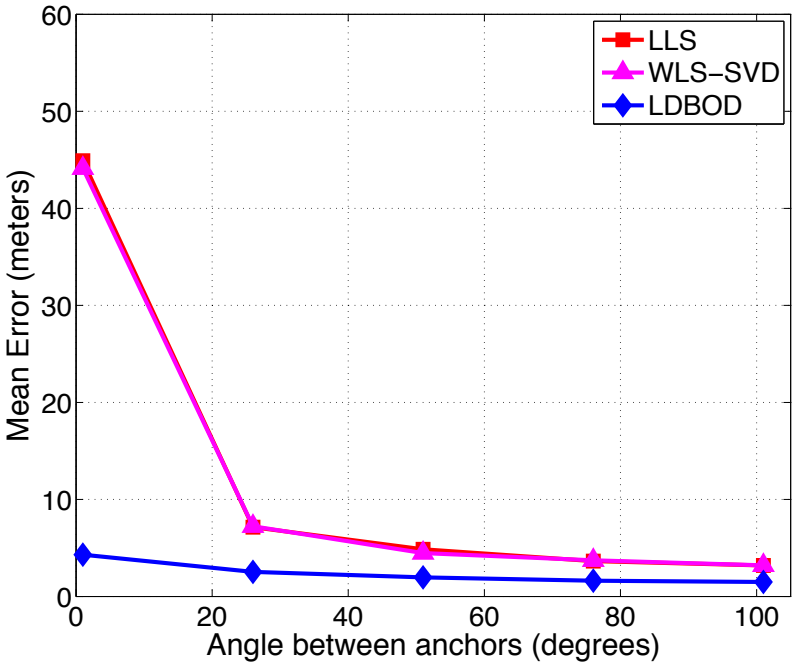
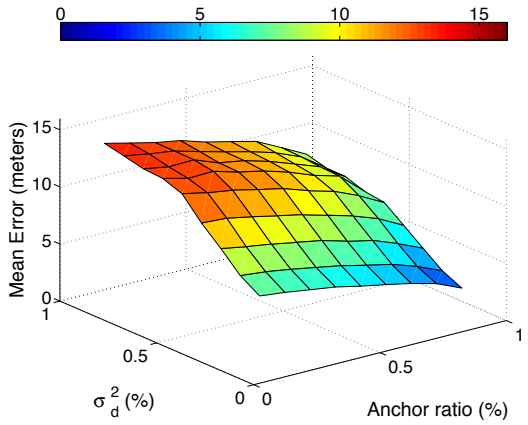
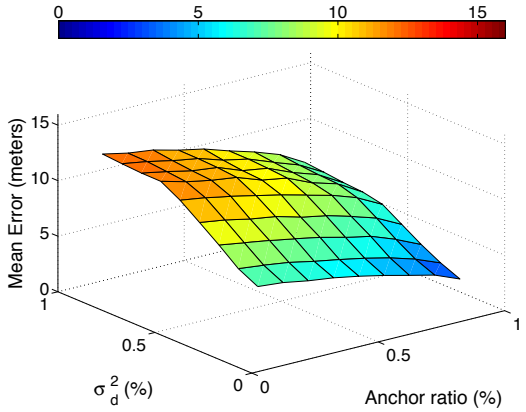


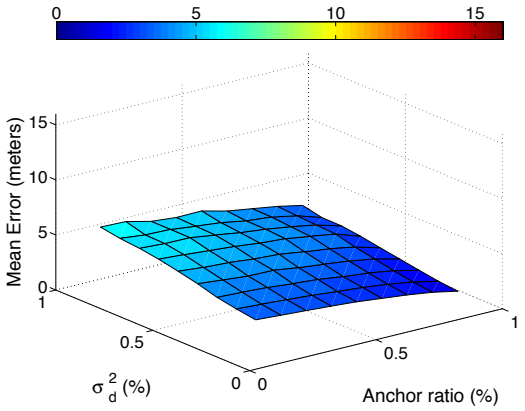
Figure 5.37: Mean error versus angle between anchors.



(a) Mean error surface for the LLS algorithm.



(b) Mean error surface for the WLS-SVD algorithm.



(c) Mean error surface for the LDBOD algorithm.

Figure 5.38: Mean error surfaces.

clustering. In Fig. 5.40, LDBOD is the best one although it uses only density-based clustering. In Fig. 5.41, at high anchor ratios, both LKmeans and LDBSCAN outperformed other techniques. In Fig. 5.42, LDBSCAN and LDBOD have similar performance and they outperformed others. Finally in Fig. 5.43, it is clear that LDBOD outperformed other algorithms. As as result of this comparison, we can say that the performance of each method is affected by the error type. For example, at high GGDOP, LDBOD is the best. However, at high distance noise variance, LDBSCAN is the best.

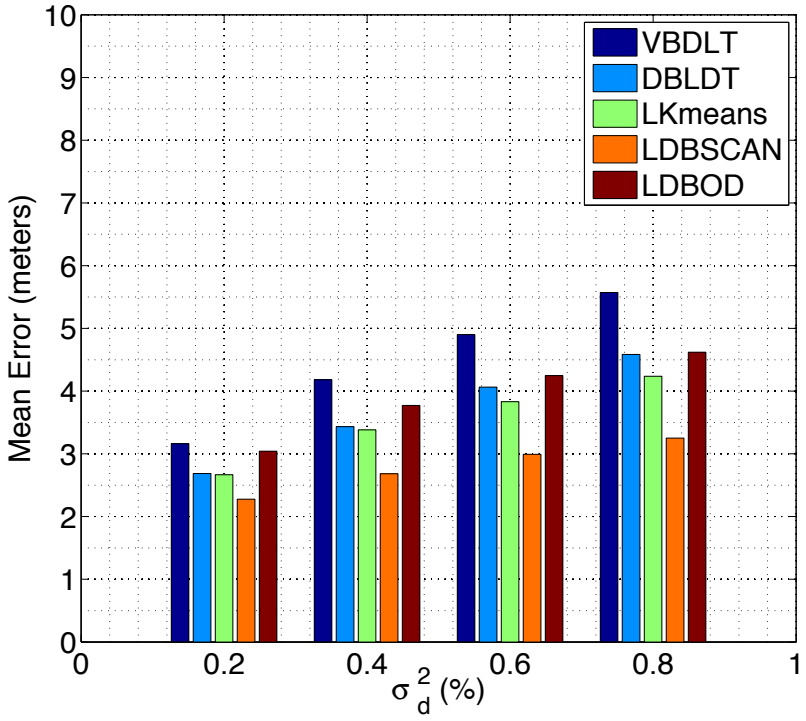


Figure 5.39: Mean error versus distance variance.

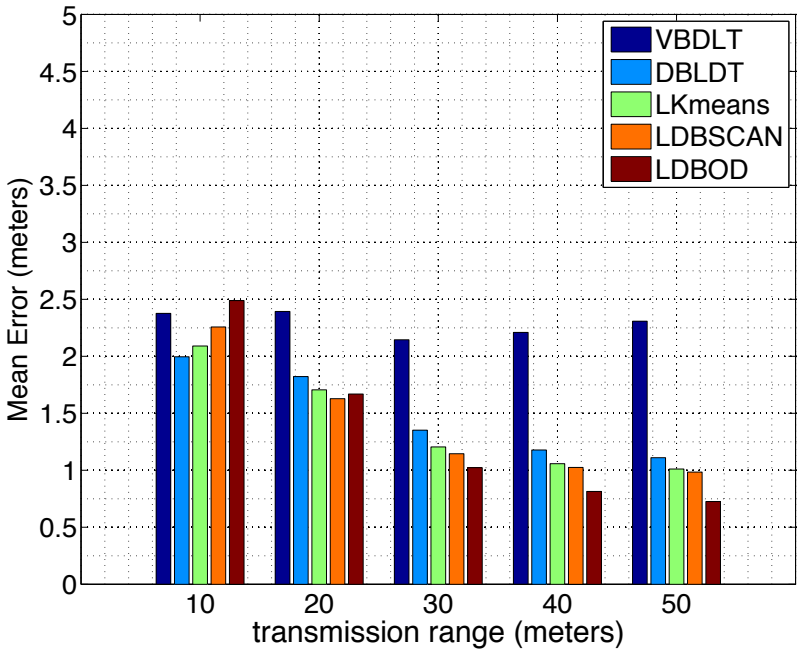


Figure 5.40: Mean error versus transmission range.

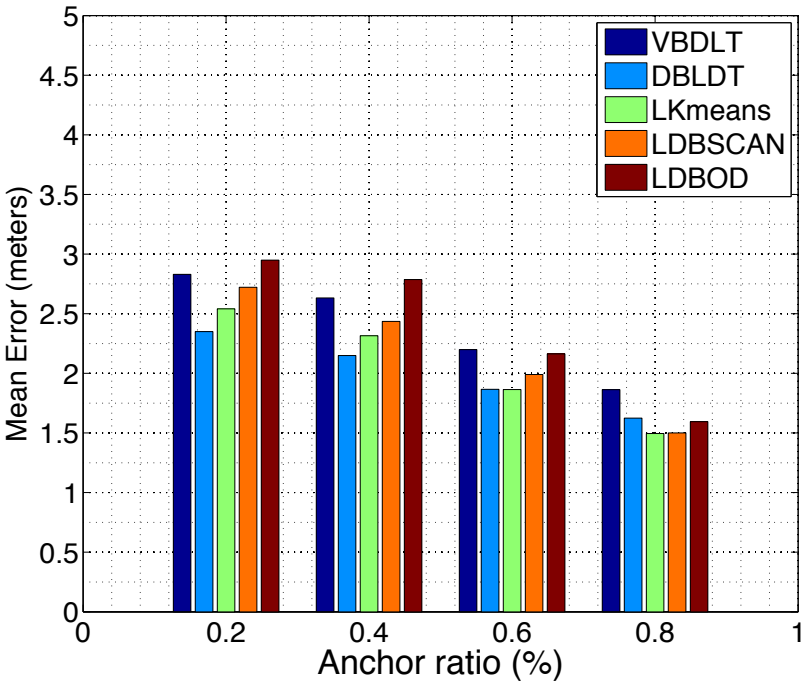


Figure 5.41: Mean error versus anchor ratio.

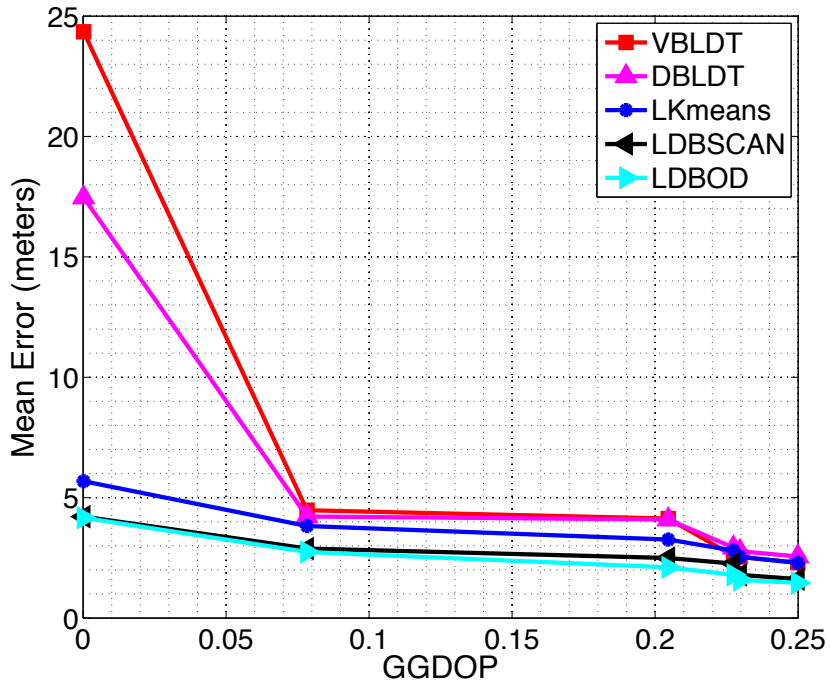


Figure 5.42: Mean error versus GGDOP.

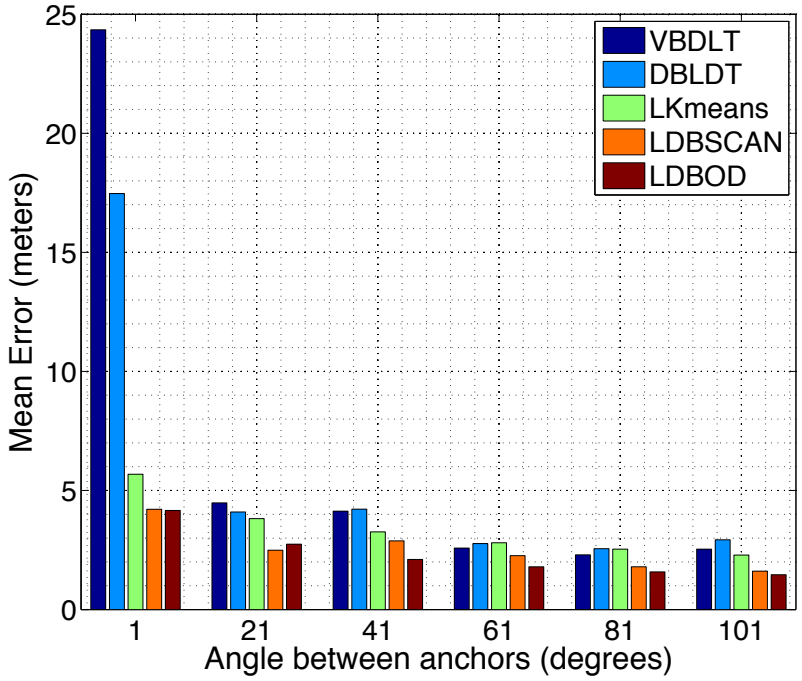


Figure 5.43: Mean error versus angle between anchors.

Chapter 6

Conclusions and Future Work

This dissertation considered two important problems, OFDM symbol timing offset (STO) estimation and localization in wireless networks. An STO estimation algorithm was proposed for ZP-OFDM systems and another for CP-OFDM. A range-free localization algorithm was proposed, and four different range-based localization algorithms were also proposed.

For OFDM synchronization, the first proposed algorithm uses a metric which is calculated recursively. Two estimation methods were considered: one using the average of the metric results, and the other using the median. In the median case sending 10^3 symbols is considered to calculate the median. The second approach used a preamble designed to have a maximum timing metric for the correct location and very small values otherwise. Both methods assumed that the CFO is estimated and compensated for already. In all simulations, zero mean Gaussian noise is assumed to combine thermal noise and co-channel interference. The ZP-OFDM method can be applied to systems using the WiMedia standard which is based on zero padding instead of a cyclic prefix. On the other hand, the CP-OFDM method can be applied to systems using the 802.11 standard to enhance the rang-

ing capabilities. Part of my future work will be to analyze both methods in the presence of CFO.

For range-free localization, a new distributed range-free localization algorithm was proposed where every unlocalized node forms two sets of anchors. The first set contains one-hop anchors from the unlocalized node. The second set contains two-hop and three-hop anchors away from the target. Each unlocalized node uses the intersections between the ranging radii of these anchors to estimate its position. The proposed algorithm outperforms recent algorithms in the literature such as DRLS [38], CPE [30], centroid [31]. The proposed algorithm is also far less complex than DRLS [38] since it does not use a grid scan which takes a lot of time to determine how many anchors are within range of a grid cell. Range-free approaches do not need any distance estimation which reduces the complexity but with less accuracy, i.e., a rough location estimate. Applications that need a rough location estimate are determining if a piece of equipment is in laboratory A or if an employee is in a meeting room.

In range-based localization, four approaches were proposed. The first uses decision tree classification to divide the intersection points into inner and outer points. The mean of the inner points is then calculated to find the location estimate. In LKmeans, a K-means clustering algorithm is proposed with decision tree classification to refine the selection of candidate points among the inner points.

LDBSCAN is the third proposed approach to range-based localization. It is based on the DBSCAN algorithm which is a density based clustering technique. Density-based clustering locates regions of high density that are separated from one another by regions of low density. DBSCAN is a simple and effective density-based clustering algorithm [2]. Among the four range-based algorithms proposed, it is the best in terms of accuracy in general, since it uses classification and density-

based clustering to classify the inner points. The idea is to group dense inner points together and calculate the mean to obtain the location estimate of the unlocalized node. There is a recent enhancement to the basic DBSCAN algorithm called P-DBSCAN [57]. In P-DBSCAN a density threshold and adaptive density are introduced. This could be a great enhancement to the proposed LDBSCAN algorithm.

The last range-based algorithm is LDBOD. It is the best algorithm in terms of complexity, since no intersection point classification is needed as a first step. This could be enhanced by using more enhanced versions of the DBOD clustering algorithm. All the proposed range-based algorithms, DBLDT, VBLDT, LKmeans, LDBSCAN and LDBOD outperform the LLS [51] and recent WLS-SVD [52] algorithms.

Data mining has many classification and clustering methods which can be used separately or jointly to solve the localization problem. Also, classification and clustering can be applied to unlocalized and anchor nodes in addition to intersection points to enhance the localization accuracy. This is a very interesting topic.

For future work, three different topics are proposed. The first is increasing the signal resolution for ranging purposes by combining the received signals from different subbands. The second is carrier frequency offset (CFO) estimation in OFDM systems. Finally, enhancing the localization accuracy using hybrid distance estimation approaches will be considered.

6.1 Increased Signal Resolution

Signal resolution is an important area that leads to more accurate distance estimates. Increasing the resolution of the ranging system taking advantage of the multiband OFDM signals or multiple received OFDM symbols. In order to have

an accurate range, we need to estimate the TOA of the first MPC accurately, as mentioned previously. Actually, when the first MPC experiences deep fading in one band or at a certain time, there is less chance for it to suffer from the same deep fading in different bands [58] or other times. Thus, even if we miss the first MPC in one band, we may get it in another.

To increase the range resolution we can combine the signal from different bands. Actually, in multiband systems: we can send a signal with a bandwidth equal to that of a single subband in a given time interval. This would appear to decrease the achievable range and time resolution by a multiband system. The idea is to process the corresponding received signals from different subbands to achieve a time resolution proportional to the inverse of the entire bandwidth, as opposed to that of a single subband.

In [59], a scheme was proposed that coherently processes received signals from different subbands in multiband systems to achieve a resolution proportional to the inverse of the total system bandwidth. This can be done by aligning the outputs of the matched filters of the received signals from all subbands and applying a fast fourier transform (FFT) in each time bin along the subband index [59]. In other words, the overall scheme achieves a resolution inversely proportional to the number of subbands multiplied by the bandwidth of a subband, i.e., to the total bandwidth of the multiband system. However, they did not study this scheme in the presence of multipath fading.

In [60], the same authors proposed a different way to achieve higher resolution in multiband systems. They combined several low bandwidth signals at the receiver to generate a delayed version of a virtual large bandwidth signal. The time delay is then estimated from this large bandwidth signal. The combining is done using a full-tree wavelet reconstruction filter bank. The output of the re-

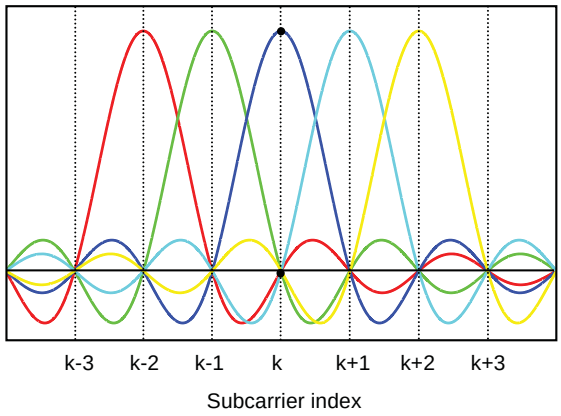
construction filter bank, in this case, is the delayed version of a template signal. This template signal can be obtained by putting the original transmitted preamble signal through the same reconstruction filter bank. The combined signal at the receiver is correlated with this template to estimate the delay. Finally in [61], the same authors compared the previous two schemes and showed that the scheme proposed in [60] provides better resolution than the one proposed in [59].

Combining different received signals in MB-OFDM can provide more advantages to the system, particularly in multipath fading where the first component may not be the strongest one.

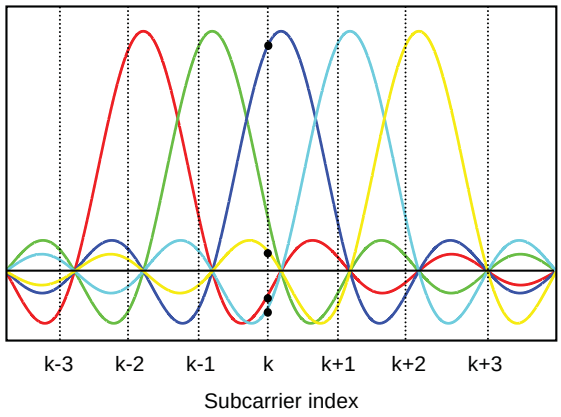
6.2 Carrier Frequency Offset (CFO) Estimation

OFDM systems are very sensitive to synchronization errors caused by Doppler shift and/or oscillator instabilities. Orthogonality among subcarriers can be damaged at the receiver due to carrier frequency offset (CFO). Actually, the degradation in OFDM in the presence of CFO is caused by two main phenomena: reduction of the amplitude of the desired subcarrier and ICI caused by neighbouring carriers [62]. The amplitude loss occurs because the FFT values of the desired subcarriers are not at the peak of the signal . Adjacent carriers cause interference because the values obtained are not at the zero-crossings of the sinc functions [62], as shown in Fig. 6.1. The CFO destroys the orthogonality among subcarriers.

If we have some mobile nodes inside a wireless network using OFDM signals, we have to also consider the Doppler shift effect. When a transmitter and a receiver are moving relative to one another, the frequency of the received signal will not be the same as at the source. The detected frequency increases for objects moving towards the receiver. Conversely, frequency decreases when the source moves



(a) CFO=0.



(b) CFO=0.2 Δf.

Figure 6.1: Effect of the carrier frequency offset on the subcarrier orthogonality.

away from the receiver. The frequency change due to the Doppler effect depends on the relative motion between the transmitter and receiver. The Doppler shift in frequency is given by:

$$\Delta f \approx \pm f_0 \frac{v}{c} \tag{6.1}$$

where Δf is the change in frequency of the source seen at the receiver, f_0 is the frequency of the source, v is the speed difference between the source and transmitter, and c is the speed of light.

The acquisition range is an important parameter used to compare different CFO estimators. The acquisition range is calculated in terms of subcarrier frequency spacing. For example, Moose [63] proposed a maximum likelihood estimator (MLE) using two OFDM symbols for frequency offset estimation. He assumed that the symbol timing is known. The drawback of this approach is that the range of detectable frequency offsets is limited to one subcarrier spacing. Schmidl and Cox [21] increased the acquisition range up to ± 2 subcarrier spacing. In [64], the acquisition range is $\pm N/2p$, where p is a user-selected even integer such that $p \in \{2, \ldots, N-2\}$. The proposed CFO estimator is based on the correlation between the received samples spaced by p samples.

6.3 Enhanced Localization using Hybrid Approaches

Although range-free localization is less expensive to implement in networks with limited resources like sensor networks, a wireless network with more powerful nodes like a WLAN should use available resources to enhance location accuracy. A WLAN consisting of routers and computers or advanced mobile devices like smart phones, can implement different ranging approaches to get the maximum possible location accuracy, especially in indoor environments. In such cases, we can combine ranging techniques like TOA, TDOA, RSS, and AOA to maximize the accuracy. Although hybrid methods are also implemented in range-free localization such as in [65], indoor high localization accuracy can be achieved only by range-based techniques. In [66], TDOA and AOA are used to enhance the localization accuracy in a UWB systems. On the other hand, both range-free and range-based methods were combined to achieve higher accuracy in [67]. There are many questions regarding the design of hybrid approaches, including: What loca-

tion metrics should be used? What localization methods are the best? How can we combine different approaches effectively?

Bibliography

[1] P. F. S. Tim C.W. Schenk and E. R. Fledderus, "Multiple carriers in wireless communications – curse or blessing?," *Tijdschrift van het NERG*, vol. 70, pp. 112–123, Dec. 2005.

[2] P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Pearson Addison Wesley Boston, 2006.

[3] M. Pun, M. Morelli, and C. Kuo, *Multi-carrier Techniques for Broadband Wireless Communications: A Signal Processing Perspective*. Imperial College Press, 2007.

[4] B. Muquet, W. Zhengdao, G. B. Giannakis, M. de Courville, and P. Duhamel, "Cyclic prefixing or zero padding for wireless multicarrier transmissions?," *IEEE Transactions on Communications*, vol. 50, pp. 2136–2148, Dec. 2002.

[5] Z. Wang and G. Giannakis, "Wireless multicarrier communications," *IEEE Signal Processing Magazine*, vol. 17, pp. 29–48, May 2000.

[6] T. Araújo and R. Dinis, "Efficient detection of zero-padded OFDM signals with large blocks," in *IASTED Signal and Image Processing Conference*, pp. 358–361, Aug. 2006.

[7] IEEE 802.11 standard, "http://www.ieee802.org/11/."

[8] Multiband OFDM, "http://www.multibandofdm.org."

[9] Wireless USB, "http://www.usb.org."

[10] WiMAX, "http://www.ieee802.org/16/."

[11] L. Hazy and M. El-Tanany, "Synchronization of OFDM systems over frequency selective fading channels," in *IEEE Vehicular Technology Conference*, pp. 2094–2098, 1997.

[12] G. Li and G. Stuber, *Orthogonal Frequency Division Multiplexing for Wireless Communications*. Springer, 2006.

[13] J. Van de Beek, M. Sandell, and P. Borjesson, "ML estimation of time and frequency offset in OFDM systems," *IEEE Transactions on Signal Processing*, vol. 45, pp. 1800–1805, July 1997.

[14] D. Lee and K. Cheun, "A new symbol timing recovery algorithm for OFDM systems," *IEEE Transactions on Consumer Electronics*, vol. 43, pp. 767–775, Aug. 1997.

[15] M. Speth, F. Classen, and H. Meyr, "Frame synchronization of OFDM systems in frequency selective fading channels," in *IEEE Vehicular Technology Conference*, vol. 47, pp. 1807–1811, May 1997.

[16] *OFDM Based WLAN Systems*. Aalborg University, Feb 2005.

[17] Y. You, E. Shim, and H. Song, "A frame detector for zero-padded OFDM systems," *IEICE Transactions on Communications*, vol. E89-B, pp. 963–965, Mar. 2006.

[18] S. Yoon and J. Chong, "Packet detection and symbol timing synchronization algorithm for Multi-Band OFDM UWB," *IEICE Transactions on Communications*, vol. E89-B, pp. 1433–1435, Apr. 2006.

[19] J. UM, S. IM, S. KIM, and H. Choi, "A robust timing synchronization algorithm for OFDM systems over multipath Rayleigh fading channels," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E89-A, pp. 1705–1709, June 2006.

[20] C. Li, D. Liu, and G. Yue, "A robust blind symbol-timing synchronization for UWB-OFDM systems," in *International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1–4, Sep. 2006.

[21] T. M. Schmidl and D. C. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Transactions on Communications*, vol. 45, pp. 1613–1621, Dec. 1997.

[22] H. Minn, M. Zeng, and V. K. Bhargava, "On timing offset estimation for OFDM systems," *IEEE Communications Letters*, vol. 4, pp. 242–244, July 2000.

[23] B. Park, H. Cheon, C. Kang, and D. Hong, "A novel timing estimation method for OFDM systems," *IEEE Communications Letters*, vol. 7, pp. 239–241, May 2003.

[24] S. D. Choi, J. M. Choi, and J. H. Lee, "An initial timing offset estimation method for OFDM systems in Rayleigh fading channel," in *IEEE Vehicular Technology Conference*, pp. 1–5, Sep. 2006.

[25] K. Fazel and S. Kaiser, *Multi-Carrier and Spread Spectrum Systems*. John Wiley & Sons Ltd, 2003.

[26] M.-G. D. Benedetto and G. Giancola, *Understanding Ultra Wide Band Radio Fundamentals*. Prentice Hall Professional Technical Reference, 2004.

[27] A. Kupper, *Location-based Services: Fundamentals and Operation*. John Wiley and Sons Ltd, 2005.

[28] M. Porretta, P. Nepa, G. Manara, and F. Giannetti, "Location, location, location," *IEEE Vehicular Technology Magazine*, vol. 3, pp. 20–29, June 2008.

[29] A. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *IEEE Signal Processing Magazine*, vol. 22, pp. 24–40, July 2005.

[30] L. Doherty, K. S. J. pister, and L. El Ghaoui, "Convex position estimation in wireless sensor networks," in *IEEE Conference on Computer Communications*, pp. 1655–1663, Apr. 2001.

[31] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications Journal*, vol. 7, pp. 28–34, Oct. 2000.

[32] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *International Conference on Mobile Computing and Networking*, pp. 81–95, Sep. 2003.

[33] D. Niculescu and B. Nath, "DV based positioning in ad-hoc networks," *Springer Journal on Telecommunication Systems*, vol. 22, pp. 267–280, 2003.

[34] I. Guvenc and Z. Sahinoglu, "Threshold-based TOA estimation for impulse radio UWB systems," in *IEEE International Conference on Ultra-Wideband*, pp. 420–425, Sep. 2005.

[35] X. Wei, L. Wang, and J. Wan, "A new localization technique based on network TDOA information," in *International Conference on ITS Telecommunications*, pp. 127–130, June 2006.

[36] A. Hatami, K. Pahlavan, M. Heidari, and F. Akgul, "On RSS and TOA based indoor geolocation - A comparative performance evaluation," in *IEEE Wireless Communications and Networking Conference*, pp. 2267–2272, Apr. 2006.

[37] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AOA," in *IEEE Conference on Computer Communications*, pp. 1734–1743, Mar. 2003.

[38] J.-P. Sheu, P.-C. Chen, and C.-S. Hsu, "A distributed localization scheme for wireless sensor networks with improved grid-scan and vector-based refinement," *IEEE Transactions on Mobile Computing*, vol. 7, pp. 1110–1123, Sep. 2008.

[39] C. Saad, A. Benslimane, and J.-C. Konig, "A distributed method to localization for mobile sensor networks," in *IEEE Wireless Communications and Networking Conference*, pp. 3058–3063, Mar. 2007.

[40] J. Caffery Jr, "A new approach to the geometry of TOA location," in *IEEE Vehicular Technology Conference*, vol. 4, pp. 1943–1949, Sep. 2000.

[41] E. Weisstein, *Concise Encyclopedia of Mathematics*. Chapman & Hall/CRC, 2003.

[42] J. Vince, *Geometry for Computer Graphics: Formulae, Examples and Proofs*. Springer-Verlag New York Inc, 2005.

[43] M. Protter and P. Protter, *Calculus with Analytic Geometry*. Jones & Bartlett Pub., 1988.

[44] B. Hofmann-Wellenhof, L. H., and J. Collins, *Global Positioning System: Theory and Practice, 5th Ed.* Springer (India) Pvt. Ltd., 2007.

[45] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, "The anatomy of a context-aware application," *Springer Journal on Wireless Networks*, vol. 8, no. 2, pp. 187–197, 2002.

[46] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket location-support system," in *IEEE International Conference on Mobile Computing and Networking*, pp. 32–43, Aug. 2000.

[47] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in Ad-Hoc networks of sensors," in *IEEE International Conference on Mobile Computing and Networking*, pp. 166–179, July 2001.

[48] P. Bahl and V. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *IEEE Joint Conference of the IEEE Computer and Communications Societies*, pp. 775–784, Mar. 2000.

[49] D. B. Jourdan, D. Dardari, and M. Z. Win, "Position error bound for UWB localization in dense cluttered environments," in *IEEE International Conference on Communications*, pp. 3705–3710, June 2006.

[50] S. Venkatesh and R. M. Buehrer, "Multiple-access design for ad hoc UWB position-location networks," in *IEEE Wireless Communications and Networking Conference*, pp. 1866–1873, Apr. 2006.

[51] I. Guvenc, C. Chong, and F. Watanabe, "Analysis of a linear least-squares localization technique in LOS and NLOS environments," in *IEEE Vehicular Technology Conference*, pp. 1886–1890, May 2007.

[52] C.-H. Park and K.-S. Hong, "Source localization based on SVD without a priori knowledge," in *International Conference on Advanced Communication Technology*, pp. 3–7, Apr. 2010.

[53] D. Torrieri, "Statistical theory of passive location systems," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 183–198, Mar. 1984.

[54] H. So and F. Chan, "A generalized subspace approach for mobile positioning with time-of-arrival measurements," *IEEE Transactions on Signal Processing*, vol. 55, pp. 5103–5107, Oct. 2007.

[55] F. Chan, H. So, J. Zheng, and K. Lui, "Best linear unbiased estimator approach for time-of-arrival based localisation," *IET journal on Signal Processing*, vol. 2, pp. 156–162, June 2008.

[56] M. Ester, H. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. Knowledge Discovery and Data Mining*, pp. 226–231, Aug. 1996.

[57] S. Kisilevich, F. Mansmann, and D. Keim, "P-DBSCAN: A density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos," in *International Conference and Exhibition on Computing for Geospatial Research and Application*, pp. 38:1–38:4, June 2010.

[58] R. B. Christian, Z. Shengli, T. Zhi, and W. Peter, "Precise timing for multiband OFDM in a UWB system," in *IEEE International Conference on Ultra-Wideband*, pp. 269–274, Sep. 2006.

[59] E. Saberinia and A. H. Tewfik, "Enhanced localization in wireless personal area networks," in *IEEE Global Communications Conference*, vol. 4, pp. 2429–2434, Nov. 2004.

[60] E. Saberinia and A. H. Tewfik, "Ranging in multi-band communication systems," in *IEEE Vehicular Technology Conference*, pp. 2248–2251, May 2004.

[61] E. Saberinia and A. H. Tewfik, "Enhanced time resolution in band-limited communication systems," in *European Signal Processing Conference*, pp. 641–644, Sep. 2004.

[62] J. Heiskala and J. Terry, *OFDM Wireless LANs: A Theoretical and Practical Guide*. SAMS Publishing, 2002.

[63] P. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction," *IEEE Transactions on Communications*, vol. 42, pp. 2908–2914, Oct. 1994.

[64] A. Laourine, A. Stephenne, and S. Affes, "A new OFDM synchronization symbol for carrier frequency offset estimation," *IEEE Signal Processing Letters*, vol. 14, pp. 321–324, May 2007.

[65] H. Sun, Z. Fang, T. Wang, Y. Ma, and N. Ren, "CDHL: A hybrid range-free localization algorithm in wireless sensor networks," in *International Conference on Frontier of Computer Science and Technology*, pp. 180–183, Aug. 2010.

[66] C. Yang, Y. Huang, and X. Zhu, "Hybrid TDOA/AOA method for indoor positioning systems," in *The Institution of Engineering and Technology Seminar on Location Technologies*, pp. 1–5, Dec. 2007.

[67] B. Amutha and M. Ponnavaikko, "Locate: A hybrid localization scheme for wireless sensor networks," in *IEEE Symposium on Industrial Electronics Applications*, pp. 295–300, Oct. 2009.

VITA

Mr. Khalid Almuzaini was born in Riyadh, Saudi Arabia. In 1998, he was awarded a Bachelor of Science degree with second highest honours from King Saud University at Riyadh, Saudi Arabia. He joined the Saudi basic industries corporation (SABIC) at Yanbu, Saudi Arabia for about 11 months as an instrumentation engineer within the project engineering department. He then moved back to Riyadh to work at Saudi Telecom Company (STC) as a communications engineer for 4 months in the department of network operations. In 2000, he joined King Abdulaziz City for Science and Technology (KACST).

He began his master studies at the University of Southern California (USC) in September 2001. By December 2003, he had earned Master of Science in Electrical Engineering after taking 14 courses at USC in different topics.

In 2005, Khalid began work towards a Doctor of Philosophy degree in Electrical Engineering under the supervision of Dr. Aaron Gulliver. While at the University of Victoria, he participated in research as well as working as teaching assistant. Mr. Almuzaini is a member of IEEE, and his research interests include: wireless network security, sensor networks security, synchronization in OFDM systems, UWB systems, MIMO systems, localization in sensor networks, and data mining. He has authored several conference papers and journal articles.

Publications

- **Conference Publications**:

  1. Khalid Almuzaini and T. A. Gulliver, **A New Time Synchronization Technique for OFDM Systems**, *Proc. IEEE Vehic. Tech. Conf.*, pp. 1–5, Sept. 2008.

2. Khalid Almuzaini and T. A. Gulliver, **A New Distributed Range-free Localization Algorithm for Wireless Networks**, *Proc. IEEE Vehic. Tech. Conf.*, pp. 1–5, Sept. 2009.

3. Khalid Almuzaini and T. A. Gulliver, **Range-based Localization in Wireless Networks using Decision Trees**, *Proc. IEEE Globecom Workshop on Heterogeneous, Multi-hop Wireless and Mobile Networks*, pp. 131–135, Dec. 2010.

4. Khalid Almuzaini and T. A. Gulliver, **Range-based Localization in Wireless Networks using the DBSCAN Clustering Algorithm**, to be presented at *IEEE. Vehic. Tech. Conf.*, May 2011.

5. Khalid Almuzaini and T. A. Gulliver, **Localization in Wireless Networks Using Decision Trees and K-means Clustering**, submitted to *IEEE Vehic. Tech. Conf.*, Sept. 2011.

- **Journal Publications**:

1. Khalid Almuzaini and T. A. Gulliver, **Range-based Localization in Wireless Networks using Density-based Outlier Detection**, *Wireless Sensor Network*, vol. 2, no. 11, pp. 807–814, Nov. 2010.

2. Khalid Almuzaini and T. A. Gulliver, **Improved Timing Offset estimation in OFDM Systems**, submitted to *IEEE Communications Letters.*