A LOCATION FINGERPRINT FRAMEWORK TOWARDS EFFICIENT WIRELESS INDOOR POSITIONING SYSTEMS

by

Nattapong Swangmuang

B.Eng., Chulalongkorn University, Thailand, 1998

M.S.,Telecommunications, University of Pittsburgh, 2002

Submitted to the Graduate Faculty of
the School of Information Sciences in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2008

UNIVERSITY OF PITTSBURGH

SCHOOL OF INFORMATION SCIENCES

This dissertation was presented

by

Nattapong Swangmuang

It was defended on

September 11th 2008

and approved by

Prashant Krishnamurthy, PhD, Associate Professor

Richard Thompson, PhD, Professor

Martin Weiss, PhD, Professor

Hassan Karimi, PhD, Associate Professor

Ching-Chung Li, PhD, Professor

Dissertation Director: Prashant Krishnamurthy, PhD, Associate Professor

ABSTRACT

A LOCATION FINGERPRINT FRAMEWORK TOWARDS EFFICIENT WIRELESS INDOOR POSITIONING SYSTEMS

Nattapong Swangmuang, PhD

University of Pittsburgh, 2008

Location of mobile computers, potentially indoors, is essential information to enable location-aware applications in wireless pervasive computing. The popularity of wireless local area networks (WLANs) inside and around buildings makes positioning systems based on readily available received signal strength (RSS) from access points (APs) desirable. The fingerprinting technique associates location-dependent characteristics such as RSS values from multiple APs to a location (namely location fingerprint) and uses these characteristics to infer the location. The collection of RSS fingerprints from different locations are stored in a database called radio map, which is later used to compare to an observed RSS sample vector for estimating the MS's location.

An important challenge for the location fingerprinting is how to efficiently collect fingerprints and construct an effective radio map for different indoor environments. In addition, analytical models to evaluate and predict "precision" performance of indoor positioning systems based on location fingerprinting are lacking. In this dissertation, we provide a location fingerprint framework that will enable a construction of efficient wireless indoor systems. We develop a new analytical model that employs a proximity graph for predicting performance of indoor positioning systems based on location fingerprinting. The model approximates probability distribution of error distance given a RSS location fingerprint database and its associated statistics. This model also allows a system designer to perform analysis of the internal structure of location fingerprints. The analytical model is employed to identify and eliminate unnecessary location fingerprints stored in the radio map, thereby saving on computation while performing location estimation. Using the location fingerprint properties such as clustering is also shown to help reduce computational effort and create a more scalable model. Finally, by study actual measurement with the analytical results, a useful guideline for collecting fingerprints is given.

Keywords: indoor position location system, location fingerprint, performance model, efficient radio map.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I would like to dedicate this dissertation to my parents, Boonmee and Penchan, my brother, Sariddech, who always love and believe in me. Without their support, I could not have finished this work and could not have been where I am now. They are invaluable to me.

I would like to acknowledge my advisor, Dr. Prashant Krishnamurthy, for his encouragement and guidance towards this dissertation. I deeply appreciate his help, technical insight, and dedication. After several years, he has tremendously enhanced on how I view research problems on modeling of indoor positioning and wireless networking in general. I also like to thank all my committee for spending time reading my work and giving all valuable comments. I gratefully acknowledge Telecommunications program's financial committee, especially Prof. Richard Thompson, for providing the financial support during the last years of my study. I would like to express my gratitude to the Royal Thai Government, specifically to Ministry of Science and Technology and Chiang Mai University, for granting me support for the first six years since I started my Master study at the University of Pittsburgh. Also, I want to thank Dr. Prashant Krishnamurthy's research funding from the National Institute of Standard and Technology (NIST) for partially providing service grant for me during summer 2003.

I further wish to thank all my friends in Pittsburgh, SIS, and the Technology department of SOE for assistance and all wonderful time at Pitt. Special thank to Dr. Kamol Kaemarungsi who provides previous measurement data and relevant background for this study. In addition, I want to thank Dr. Richard Donato of the SOE, who helped refine my research statement while I was applying for the National Science Foundation (NSF) student travel grant to the PerCom 2008 conference in Hong Kong.

I. INTRODUCTION

The evolution of networking and computing technology promises to make life simpler via digital environments that can sense, recognize, and respond to human needs. Location-aware computing gives a computing device the ability to pinpoint users in the environment and to react to real world situations. Technology required for provision of location information in both outdoor and indoor environment has been researched and developed over the past several decades. While its roots are in military applications (e.g., the global positioning system(GPS)), location information has become important in many applications such as routing, logistics, safety and emergency response, asset tracking, and consumer marketing. Since a mobile device could roam anywhere and its location can change with time, providing location information sufficiently and efficiently is not easy and can impose several challenges.

Location determination is described as a process used to obtain the location information of a mobile device with respect to a set of reference positions within a predefined space [1]. The process has been termed differently in the research literature as radiolocation, position location, geolocation, location sensing, or localization. A system deployed to determine the location of an entity is called a position location system or positioning system. A wireless indoor positioning system provides indoor location information to the requested user (or system) using a wireless network infrastructure. A set of coordinates or reference points within the predefined space is used to indicate the physical location of the entity. For example, GPS uses latitude, longitude, and altitude, constructing a geographic coordinate system on the Earth's surface, to express the position of the mobile device. An indoor positioning system, on the other hand, may combine floor number, a room number, and other reference objects to represent the mobile device's location. Although the term position (a point) and location (a point or a region) are different in scope, they are used interchangeably in this thesis.

As previously mentioned, there are many possible applications using indoor location information to accomplish tasks. Location-based service (LBS) applications include intelligent management of information in 802.11 (Wi-Fi) hot spots. For example, at the airport a traveler turns on his Wi-Fi-enabled handheld device and immediately gets information about the floor plan, direction to specific gates, baggage claim, restaurants, or the nearest restroom. Applications like Microsoft Location Finder [2] can turn a Wi-Fi user's device into a location determining device and could be used in such scenarios. In the manufacturing and logistics industry, the capability to track assets or products in a plant, movement of work in process, tools, vehicles, or even personnel is helpful. Location information can boost productivity, optimize equipment utilization, and also reduce turn around times. Location information for emergency services is also vital. Knowing the locations of all fire fighters in a building during a mission helps track personnel and harmonize the operation, thereby saving trapped fire fighters and rescuing individuals. Retail or shopping experiences will never be the same if location information is employed. Stores can launch personalized promotions based on profiles and locations of customers through a mobile unit or a personal device. A library or museum can apply the same idea to broadcast relevant information such as an exhibition detail or a new book in the bookshelf close to a user. These are just a few examples of location-related applications that users can benefit from in real life.

This chapter introduces the fundamentals of indoor positioning systems and current indoor positioning systems based on location fingerprinting. We describe an indoor positioning system based on a current network infrastructure and challenge in deploying such system. Finally, research approaches and focus of the dissertation are presented.

A. FUNDAMENTAL OF INDOOR POSITIONING SYSTEMS

As discussed previously, wireless positioning technology has been receiving increasing attention. With the ability to determine a location of an entity or an object, the technology can be applied to a variety of applications and services that help facilitate a human's daily activities. As a matter of fact, people's activities are mostly taking place inside buildings such as homes, schools, supermarkets, and offices. Most of the time social interactions between individuals tend to occur more in indoor areas than outdoor areas. With conventional GPS, although it is the most famous positioning technology, a GPS receiver cannot work well inside buildings and urban environments due to the absence of a line of sight to satellites [3]. Positioning systems using cellular networks also fail to provide good accuracy for location determination. So, this brings about the need to develop a new positioning system that can perform efficiently in the indoor environment. There are many proposed indoor positioning systems using different technologies. RF, infrared, and ultrasound are three major signaling technologies used for indoor positioning systems. Generally, different types of sensors are used to detect signals which have different characteristics depending on the location. A sensor transduces the particular physics of the environment (i.e., heat, light sound, pressure, motion) for each location and a sensor process converts them into measurable metrics such as distance, time, or angle for later location determination. These metrics are processed by a *positioning algorithm* in order to estimate the position. The complexity of the indoor area due to obstructions, different types of constructional materials, and the dynamic nature of environment creates many challenges and difficulties for accurate location finding. Thus, a fundamental understanding of issues related to indoor radio propagation is needed for the design and performance evaluation for new positioning systems.

The popularity of local wireless networks (WLANs) has increased in recent years where they have already been installed in common buildings including homes, offices, or campuses. The prevalence of WLANs gives great opportunity for providing location-aware services. This type of commodity wireless technology like IEEE 802.11 allows an existing network infrastructure to provide indoor location service with minimal modification and no additional equipment. Hence, WLANs that are also cheap and easy to deploy, become an attractive solution. In existing WLANs, a wireless interface card at a mobile station (MS) is used to measure RF signals from access points (APs) nearby and can be considered as the sensor part of the positioning system. There are many commercialized products (such as Ekahau[4], MicroSoft Location Finder [2], the Skyhook's WPS [5]), which can be used as indoor location engines. The engine can later be integrated for many location-aware applications.

However, provision of the positioning system poses numerous interesting challenges. There are not many papers that provide a good analytical and theoretical background for indoor positioning systems. A good framework for system design and performance evaluation is required to guarantee a success of the technology. Krishnamurthy [6] identifies important challenges and issues in locating the position of mobile terminals. Many of the issues are interrelated. These issues are summarized as followings.

- **Performance:** In deploying an indoor positioning system, there are many performance benchmarking metrics the need to be considered [6]. The most fundamental metric is accuracy of location information which is usually reported as an error distance between the estimated location and the actual mobile location in meters or feet. The accuracy of the system depends on the sensing technology deployed, radio propagation characteristics of environment, and signal processing technique used to estimate the location. Another metric is called "precision". The precision of location estimation reports the probability of successful (or unsuccessful) location estimates with a given accuracy. With 0-meter error distance (0-meter accuracy), the location precision will correspond to the probability of returning the correct location. Some other essential metrics include delay, capacity, coverage, scalability, and interoperability of the positioning system. The delay metric refers to the time taken for the system from sensing the location information to reporting it to the requesting entity. The capacity metric measures the number of queries for location estimation that the system can process and handle per unit time. The coverage metric reports the area boundary where the positioning system is available to compute location information. Scalability of the positioning system is concerned with how well the system performs when handling larger coverage areas and larger numbers of requests for location estimation. Interoperability is the capability of the system to operate and combine with different systems in order to support better location services. An efficient indoor position system should satisfy the desired performance metrics as mentioned above.
- **Cost and Complexity:** Provision of an indoor positioning system results in cost of required infrastructure, additional communication bandwidth, fault tolerance and reliability, and nature of technology deployed. For some systems, the cost also includes

installation of equipment, software upgrade, survey time, and development of a location database. Integrating the indoor positioning system with the commodity equipment makes the system cheaper and fast for deployment. It is also desirable to reuse existing communication infrastructure and signals for location sensing, thereby simplifying the deployment process. Finally, a mobile device involved in location estimation process should consume as minimal power as possible. The complexity of signal processing and algorithms for location estimation has to be considered as a trade-off with performance of the positioning systems.

- **Application requirements:** Different applications in mobile and distributed data environment may have different position location requirements. The major requirements are the granularity of position location information, the performance for location estimation, and the availability. The granularity consists of spatial granularity and temporal granularity. The spatial granularity determines level of detail of location information whereas the temporal granularity determines the rate at which the location information is requested. For performance requirements, applications may need different combinations of performance metrics. Some applications such as advertising may require only moderate delay response time (within couple minutes or hours) and moderate location accuracy (within tens of meters), while emergency response application may require shorter response times (within couple seconds) and higher accuracy (within few meters). In the latter application, the performance aspects become crucial. Moreover, positioning systems can be created using either self-positioning and remote-positioning approaches. The availability in location estimation refers to an ability to obtain location information for an entity under a particular situation. The availability metric is closely related to privacy concerns for indoor positioning systems.
- **Security:** Location information should be made available only to those with proper authorized access. Lack of privacy of location information could provide knowledge of activities of any individuals whose location can be unobtrusively tracked. Service providers, who know location of users, can exploit location information to provide location dependent information or services not wanted by users. This is sometimes referred to as user personalization. Personalization, combining location information and logging informa-

tion, has the potential to be extremely contentious. This can seriously damage trust in the system. So it is suitable to develop security protocols to prevent misuse of location information. Especially, a wireless mobile terminal that transmits and receives signals with the intent to capture and process signals to derive location information, makes it difficult to secure such signals. On the other hand, some applications such as emergency response and homeland security need to be able to access such signals during critical conditions as soon as possible. Securing of location information in such situations can be burdensome. Hence, a good positioning system should preserve privacy without sacrificing accessability and functionality of the system. This is extremely difficult especially when a mobile terminal has to operate across different control boundaries.

B. INDOOR POSITIONING SYSTEMS BASED ON LOCATION FINGERPRINTING

Increasing deployment of commodity computers with WLAN equipment (e.g. access points, built-in wireless access cards) has the advantage that adding indoor localization functions can be done so as to leverage existing infrastructure with minimal modification. So, the use of existing radio characteristics of an IEEE 802.11 network to localize a user emerges as a favorable option. Location fingerprinting, which utilizes radio signals, is a technique that identifies the location of a user by characterizing the radio signal environment of the user. The location fingerprinting approach exploits the relationship between physical-related measures and a specific location. In other words, a physical location of the user is mapped into a unique measure in radio signal space and it is used for location determination. The measures may include received signal strength (RSS), signal-to-noise ratio (SNR), or packet loss rate. The RSS is the measurement by the receiver of the power (usually expressed in decibels) of each received packet and it has shown strong correlation to distance [7]. In addition, the RSS is typically available in a normal wireless interface card, and its use for localization has been adopted in many indoor positioning systems [4, 5, 8].

An outdoor positioning system can use triangulation-based techniques, such as angle of arrival (AOA) and time difference of arrival (TDOA), to efficiently implement a localization system. For an indoor environment, however, wireless signals encounter the problem of dense multipath and none-line-of-sight effects which renders these techniques ineffective and complex for actual implementation. Also, a mobile station may not always see three or more access points in an indoor environment at all places and at all times, which is essential for triangulation computation for both AOA and TDOA. Compared to these techniques, an indoor positioning system using location fingerprinting technique is relatively simple to deploy.

Generally, fingerprinting based indoor positioning systems comprise of *offline* and *online* phases. Locations in the entire area of interest are usually considered as rectangular grid points. The *grid spacing*, which is defined as the distance between the closest positions, is typically reported in meters or feet. During the offline phase, by site-surveying, the RSS from multiple access points (AP) at different grid points are collected and stored in a database called a *radio map*. The vector of RSS values at a point on the grid is called the location fingerprint of that point [9]. This RSS vector is measured with enough statistics such that it creates a specific RSS pattern on a predefined point of the grid. The central tendency measure such as the mean of the RSS is used to represent fingerprint vectors of a location. Then a MS's location is determined by a positioning engine during the online phase. The positioning engine, where the radio map is stored, estimates a MS's location using an appropriate algorithm and it can be implemented either at a WLAN infrastructure or at a MS.

During the online phase, first a MS will measure one or more sample fingerprint vectors of RSSs from different APs at its current location. If the positioning engine is located at the WLAN infrastructure, the sample fingerprint is sent to a central server in the WLAN infrastructure. This server compares the measured fingerprint to fingerprints stored in the radio map for determining the MS's location on the grid. The estimated result is then reported back to the MS. If a positioning engine is located at the MS, the fingerprint comparison is done locally. Commonly, the Euclidean distance between the measured fingerprint and each fingerprint in the radio map is computed and used for location estimation. The grid coordinate associated with the fingerprint that provides the smallest Euclidean distance is selected as the estimate of the position. Other methods using Bayesian modeling [10] and Statistical Learning [11] are also suggested in order to map a sample fingerprint to the fingerprint in the radio map.

Existing literature (will be discussed in Chapter II) have demonstrated an effort to adopt the indoor positioning systems based on location fingerprinting technique with WLAN infrastructure. Many research groups have introduced various ways to enhance the system performance with several approaches. Previous work varies among model analysis, simulation, experimental measurement, and actual implementation studies. However, provisioning an indoor location fingerprint system still faces many challenges and limitations especially with issues related to the performance of the system. In general, the performance of the system depends on the technology employed, the characteristics of radio channel in the environment, and the complexity of signal processing technique used.

For benchmarking of indoor fingerprinting systems, the most fundamental metrics used are location accuracy and location precision as defined in section I.A. It is intuitive to believe that a small grid spacing would help to achieve good accuracy with acceptable precision. So, conventionally, to achieve desired performance, a large number of fingerprints are collected from a fine-grained symmetric square grid system during the offline phase. This process can be laborious. In addition to the labor cost, the size of the database of the radio map can have a direct impact on the delay, capacity, and granularity performance of the positioning system. In fact, a constellation of fingerprints is typically scattered and asymmetric due to different signal propagation for different locations. The Euclidean distance among some fingerprints on a particular area could be small compared to the variation of the RSSs in the area. The variation can influence a decision of the location system such that most of the time it will select one location as a correct location even though the MS is actually located at another location. As such, keeping all those fingerprints in the radio map can reduce location precision.

Collecting many fingerprints with small signal distances in the radio map will not be helpful and also cause extra computational effort while determining the location. Such fingerprints should be eliminated from the radio map or not be collected in the first place (if they can be identified a priori) during the offline phase. With a large grid spacing with a small number of location fingerprints, although the precision performance may be improved, it may not achieve the desirable accuracy by the user. Unfortunately, a system designer usually lacks a sound approach to deal with the situation and determine which fingerprints or how many of them should be included in the radio map database. Especially for large indoor buildings with several floors, it is necessary to have a cost/time effective approach to deploy and construct the radio map for the positioning system.

So, from the above reasoning, one challenge for deploying a location fingerprinting technique is how to efficiently collect fingerprints and construct a radio map so that it contains only *necessary* location fingerprints without sacrificing performance. Even though research on location fingerprinting has been investigated for years, no literature has studied this issue of determining the radio map. Moreover, an existing model of location fingerprinting [9] did not study the distribution of probability of selecting location, which is important to construct an efficient radio map.

C. APPROACHES AND RESEARCH FOCUS

In this dissertation, we study an indoor positioning system based on location fingerprinting technique. The question we try to answer is whether it is possible to employ the properties of fingerprint constellations, structure of the radio map and probability of selecting a location in the system to make location determination more efficient. The process behind answering this question is as follows. We derive an analytical model for analyzing indoor positioning systems. The model must enhance the existing analytical model(s) by considering the *distribution* of the probability of selecting a location and *distribution* of error distance. Next we find a method that helps determine and eliminate unnecessary location fingerprints stored in a radio map database. Then, from the performance study of measured location fingerprints, a usable design framework to create an efficient radio map with less time and possibly labor for indoor location fingerprinting systems is derived.

Base on the above discussion, a framework, which is used toward the design of an efficient location fingerprinting based indoor positioning system, is given in Figure 1.

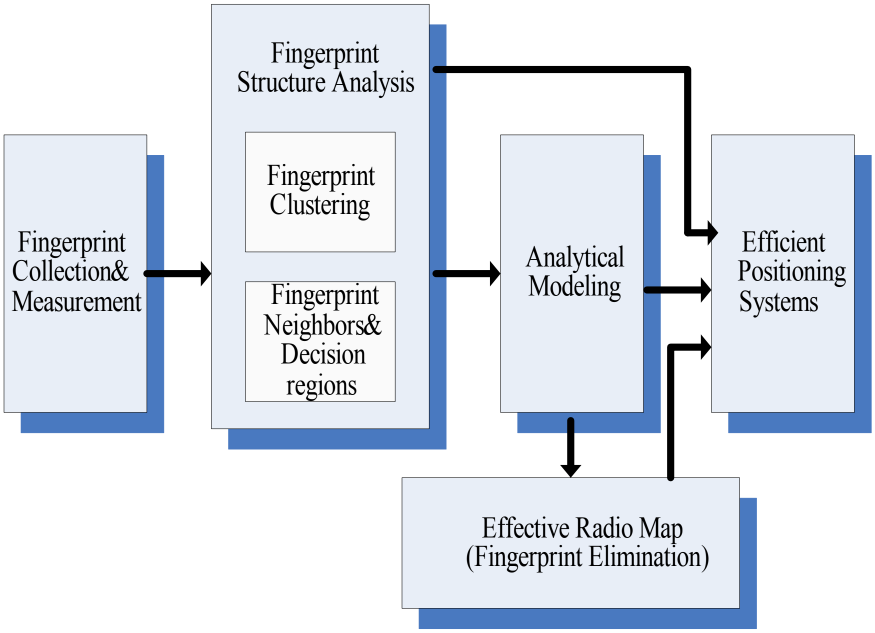
Fingerprint Collection& Measurement
Fingerprint Structure Analysis
Fingerprint Clustering
Fingerprint Neighbors& Decision regions
Analytical Modeling
Efficient Positioning Systems
Effective Radio Map (Fingerprint Elimination)

Figure 1: An Efficient Location Fingerprint Framework

In this framework, first a site-survey of RSS measurement in a defined area is performed, producing a fingerprint collection or radio map. The next component analyzes the fingerprint structure to better understand the fingerprint system which will be used to estimate a MS's position. In this stage many characteristics of the fingerprint system are extracted such as location fingerprint decision regions, neighbor set, and clusters. Then the analytical modeling component is used to help predict performance of the system before the actual deployment. An efficient radio map is then constructed by employing a fingerprint elimination procedure. With all these components, finally the positioning system can efficiently estimate the MS's position.

Since the fingerprint collection and measurement has already been done in previous research [1], our contribution in this dissertation include the rest of all components in the framework.

As stated, an improved analytical performance model as well as a fingerprint elimination technique for indoor positioning systems based on location fingerprinting are studied. The dissertation begins by studying the structure of indoor location fingerprint vectors which later allows insight into the details of the radio map and better modeling of the performance. Our work focuses on the accuracy and precision performance metrics of positioning system and suggests a performance modeling and evaluation methodology. The performance model will be used by a new fingerprint elimination technique to enable an efficient radio map construction and consequently reduce effort for fingerprint searching.

There are assumptions used to define the scope of this work. First, we only study stationary mobile stations. No mobility tracking is considered in this work. Second, we will not consider the search of an optimal positioning algorithm but assume a generic algorithm using the nearest neighbor method with Euclidean distance as the classifier. Third, we consider only indoor fingerprinting systems based on a WLAN infrastructure. A system with other sensing techniques, technologies, or a hybrid approach is beyond the scope of this dissertation. Fourth, the location system is considered only on one floor of a building. However, we believe that our system is flexible and it can be applied to a multi-floor situation. Fifth, we assume that the indoor positioning system is overlaid on top of the existing WLAN infrastructure. Hence, we will not consider optimal placement of the WLAN infrastructure.

D. DISSERTATION ORGANIZATION

The rest of the dissertation is organized as followed. In Chapter II, we present previous literature work on wireless indoor positioning systems and relevant mathematical concepts of Voronoi diagram and proximity graphs, which are used to analyze location fingerprint structure. Then the location fingerprint framework for designing an efficient indoor positioning system is described in Chapter III. An analytical model for location system performance prediction, a new fingerprint elimination technique, and its performance results are also shown and discussed in this chapter. Results from the sensitivity study of precision and accuracy metrics with varying grid systems and wireless characteristics are shown in Chapter IV. Chapter V provides a study of tradeoffs with fingerprint clustering where we first divide fingerprints into many clusters and compute separate proximity graphs towards improving the scalability in the computation of the models and fingerprint elimination procedures. In

Chapter VI, we look into the use of the new radio map for real devices. We also present a study of modeling with different RSS variations, where we employ a better RSS distribution (i.e., skewed-Normal) to test whether it improves precision modeling. Then we illustrate the use of our off-line phase fingerprint collecting guideline, derived from the analytical study. Finally, the conclusion and the future research work are described in Chapter VII.

II. LITERATURE REVIEW

In this chapter, we present a literature review of wireless indoor positioning systems based on location fingerprinting. Then we present background material on the concepts of the Voronoi Diagram and Proximity Graph. The meaning of term “proximity” is quantified by a decomposition of space into regions using the Voronoi diagram. We will describe the concept of the Voronoi diagram and its construction. Then, the concept of the proximity graph to analyze proximity structure of the location fingerprints is given.

A. INDOOR POSITIONING SYSTEMS USING WLANS AND LOCATION FINGERPRINTING

In this section, RF-based wireless LAN indoor positioning systems are reviewed. Excellent comprehensive surveys of indoor positioning systems can be found in [1, 12, 13].

The popular deployment of IEEE 802.11 wireless LANs in past few years has attracted the idea of utilizing such a network for future positioning systems based on location fingerprinting. This type of positioning system can be overlaid on top of any existing WLAN. Hence, the system can be built with a small cost of minimal additional infrastructure. Moreover, such systems utilize radio frequency (RF) signals which can penetrate most of the indoor materials resulting in a larger range and reducing the number of required access points for positioning purposes. Since the RSS can be measured by all WLAN network interface cards, no dedicated tag or badge is required for current laptops and PDAs with built-in IEEE 802.11 interfaces. The system is flexible because a system designer can select whether to have a centralized positioning server or a mobile station determine its own position. However, the fingerprinting technique requires a potentially laborious training phase (the off-line phase) to collect location fingerprints for all positions in the operating area, before the actual deployment (the on-line phase).

Initial and experimental studies of the WLAN location fingerprinting system for feasibility have been successfully shown in [8, 14]. After that, several research works have proposed improving location estimation algorithms as well as system performance [7, 15, 11, 16]. Some machine learning techniques such as neural network and support vector machines (SVMs) techniques have been introduced to improve the performance with RSS fingerprinting. Here, we will describe the fingerprinting-based indoor positioning system based on its fundamental components. The characteristics of the RF signal propagation in indoor environments can directly affect the capability of location determination by the system. For this reason, we will first consider previous studies on the impact of indoor environments. Next, general definitions and representations of location fingerprints are explained. Then, several methods used for location estimation are described. Finally, a performance summary of existing indoor positioning systems is presented.

1. Indoor Environment

Radio propagation indoors is influenced by a variety of aspects. These aspects include frequency of operation, layout of the building, constructional material, presence of objects or humans in the building, and dynamics of the environment. The RF signal propagation in indoor environment is dominated by reflection, diffraction, and scattering of radio waves. The power of the RF signal, namely the signal strength, significantly attenuates over distance. Also, the transmitted signal generally reaches the receiver by more than one path, resulting in a phenomenon known as *multipath* propagation. Signal attenuation as well as multipath propagation have great impact on the received signal strength used for indoor positioning systems. Several studies have tried to characterize properties of received signal strength for indoor environment. An initial study was done by J. Small et al [17] where the received signal strength from access points was measured at a fixed WLAN station inside an office building. They found out that the mean, mode, median from collected data were close together, and the data was log-normally distributed.

Dr. Kamol Kaemarungsi had done a comprehensive measurement campaign inside the Hillman Library and School of Information Science (IS) building at the University of Pittsburgh [1]. As opposed to the previous findings, the result from measurements revealed that the signal data (in decibels) has a left-skewed normal distribution when there is a line-of-sight between an access point and a WLAN station. It is approximately a normal distribution when there is no line-of-sight. In addition, results over multiple days also showed that there could be multiple modes in the distribution. It was indicated that the farther the WLAN station is from the access point, the smaller is the degree of standard deviation of the received signal. Also, the effects of user presence and orientation play a significant impact on the mean and the standard deviation of the received signal strength. Water, which has a resonance frequency at 2.4 GHz(one used in most 802.11 WLANs), is the most common molecule in the human body and can greatly attenuate the WLAN signal. In fact, the observation in [8, 1] showed that different user orientations can cause a variation in signal strength up to 5 dB. In some case, the human body can completely block the WLAN signal from reaching the WLAN station. So it is usually suggested that these effects of the user are needed to be taken into consideration while estimating the user location. Other factors such as dependency of signal strength over time of the day and different makes of WLAN cards apparently can result in differences in the signal strength distribution as well.

2. Location Fingerprints

A "location fingerprint" based on RF characteristics such as RSS is the basis for representing a unique position or location. It is created under the assumption that each position or location inside a building has a unique RF signature [18]. A fingerprint $\mathcal{F}$ is generally defined as a vector of RSS values from each access point and a corresponding location label $\mathcal{L}$. The RSS values are measured from a set of predefined locations in the building. Collection of location fingerprints and associated location labels are maintained in database called a *radio map* denoted as a tuple $(\mathcal{L}, \mathcal{F})$. A set of tuples representing relationship between the radio signal component and spatial component of the measurement is called a *training set* [19].

According to Battiti et al [19], the location information $\mathcal{L}$ for indoor location can be recorded in two forms depending on the type of the problem to be solved in terms of statistical learning theory. For a decision or classification problem, location information could be expressed by an indicator variable which is a single variable from a two-valued set, e.g. $\mathcal{L} = \{-1, 1\}$. The example for the decision problem is when an indoor positioning system decides whether the object is inside or outside the area. For a regression problem, location information used by the indoor positioning system is given by a tuple of real coordinates. Coordinates can vary from one dimension (e.g., position along a corridor) to five dimensions (e.g. position in three-dimensional space and two orientations expressed in spherical coordinates). To give an example, location information of a two-dimensional system with orientation could be expressed as the triplet: $\mathcal{L} = \{(x, y, d) \mid x, y \in R^2, d \in \{\text{North,East,South,West}\}\}$.

Because of the fact that the RSS is already available in most WLAN interface cards, it is thought to be the most effective RF signature and commonly used for location fingerprinting. Another parameter such as signal-to-noise ratio (SNR) has proved to be less location-dependent than the RSS due to the randomness of the noise component in nature [8]. In practice, noise can vary considerably from one location to another depending on external factors, resulting in a huge change in the SNR. This does not happen to the RSS although it can be impacted by small scale fading [20]. So in comparison with the SNR, the RSS tends to be more stable at a specific location.

To create a location fingerprint, a number of samples of RSS vectors are collected at each location. Samples are usually collected over a time period. Each RSS element of a vector can be considered as a random variable. The size of the vector is determined by the number of access points that can be heard at a location. The average RSS from each access point is calculated and recorded as an element in the location fingerprint. For N access points that can be heard at a location, a location fingerprint can be expressed as:

$$\mathcal{F} = (\rho_1, \rho_2, \ldots, \rho_N) \tag{II.1}$$

where each ρ_i is an average RSS element. The standard deviation of the RSS, as suggested in [14], could be added to the location fingerprint using another vector:

$$\mathcal{D} = (\sigma_1, \sigma_2, \ldots, \sigma_N) \tag{II.2}$$

where each σ_i is a standard deviation of RSS element. This approach of representing location fingerprint is called *deterministic* approach since location information is tied to a constant RSS value. An example of location estimation method using this approach is the nearest neighbor method. Detailed discussion of this method as well as other methods are presented in the next subsection.

Another approach [7, 15] for representing location information is to estimate the probability distribution of the RSS signature. This approach is referred to as the *probabilistic* approach since it is assumed that the location fingerprint is described by a conditional probability. The location fingerprint in this approach is described in the form: $P(\mathcal{F} \mid \mathcal{L})$, where $\mathcal{F}$ denotes the observation vector of RSS and $\mathcal{L}$ denotes the location information. The conditional probability $P(\mathcal{F} \mid \mathcal{L})$ is called the *likelihood function* because it provides the probability of the occurrence of the RSS vector given the known location information [15].

The process that creates the basis of location fingerprinting discussed above is considered as a part of *preprocessing* step [15]. This step is done before deployment of fingerprinting based positioning systems. Preprocessing is needed to clean up the raw data and render the dependency between the collected location fingerprints and the location information. This may include encoding, dimensionality reduction, feature extraction, clustering, or outlier elimination. Especially, finding an appropriate number of location fingerprints needed in the reference radio map is believed to increase system performance and it is a challenging problem. So far, there has been a lack of a systematic technique for reducing the number of fingerprints collected to produce an efficient radio map. We will discuss a technique we propose towards this, along with the analytical model used in this work in the following chapter.

3. Location Estimation Methods

A location estimation method, also known as a positioning algorithm, is a procedure that exploits the dependency between location information and location fingerprint basis in order to determine a position or location from samples of RSS signals. Many location estimation methods have been suggested in order to improve performance in terms of accuracy, precision, and granularity of the system. Two simple examples of location estimation methods are the random selection method and the strongest base station selection method. With the random selection method, the user's position is estimated by randomly picking one from a set of known positions. The strongest base station selection method guesses the user's position to be the same as the location of the base station that provides the strongest signal strength. Bahl et al [8] introduced a more efficient, yet simple method compared to the previous two methods by picking a position from a predefined search space with the closest signal strength data to a sample. With a reported 7 feet of accuracy and 38 percent precision, significant effort is still required to construct the signal strength database or search space.

The positioning algorithm can be viewed from a statistical learning perspective as a pattern classifier. In pattern classifiers, the procedure is to classify samples of patterns into different classes [15, 21]. The RSS data patterns that come from different locations or positions belong to individual classes. These data patterns form a training set and are used to create estimator models that relate location fingerprints and location information. The classifier then "learns" from the previous training set of location-dependent RSS fingerprints, and estimates the position or class from the samples of RSS vectors.

Considered as pattern classifiers, positioning algorithms or location estimation methods can be divided into two main types based on approaches used to model the relationship between location fingerprints and location information. They are parametric classifiers and non-parametric classifiers. A parametric classifier assumes some knowledge of the distributions of the location fingerprints such as mean of RSS or probability density function of the RSS. A non-parametric classifier, on the other hand, does not assume any knowledge on the distributions of the location fingerprints, but it uses a trainable parallel processing network to solve for the location from observed location fingerprints. In the case of the parametric classifier, the method is based on either nearest neighbor classifiers or Bayesian inference. In the case of the non-parametric classifier, the method is based on either neural network classifiers or a statistical learning paradigm such as support vector machines (SVMs). We discuss these methods used for indoor positioning systems below.

a. Nearest Neighbor Methods The first method employing the deterministic approach for estimating the user location is the nearest neighbor method. The nearest neighbor method requires parameters of fingerprints that includes mean vectors and standard deviation vectors of the RSS, thereby making this approach as a parametric classifier. To determine the location, a common discriminant function or context-dependent distance measure is used in order to classify a sample of the RSS fingerprint into an estimated position. The basic method for the nearest neighbor classifier is that it selects the class based on the the "closeness" of a sample RSS vector to the mean or average RSS location fingerprints stored as a training set. It is referred to also as a *case-based* method since it classifies location fingerprints from each location into each case or class [15].

Let a set of K location fingerprints be denoted by $\{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_K\}$ and each fingerprint has a one-to-one mapping to a set of positions $\{\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_K\}$. A sample of an RSS fingerprint denoted as $\mathcal{R}$ is measured during the on-line phase. The sample can be a mean or average RSS vector of a small window of RSS samples. Assume that an indoor positioning system only considers the average RSS from N access points as a location fingerprint. The sample of RSS vector is $\mathcal{R} = (r_1, r_2, \ldots, r_N)$ and each location fingerprint i in the database can be expressed as $\mathcal{F}_i = (\rho_1^i, \rho_2^i, \ldots, \rho_N^i)$.

Given the $Dist(\cdot)$ function that computes closeness or distance measurement metric in signal space [14], the procedure with the nearest neighbor method is to pick the fingerprint j that has the shortest signal distance:

$$Dist(\mathcal{R}, \mathcal{F}_j) \leq Dist(\mathcal{R}, \mathcal{F}_k), \forall k \neq j. \tag{II.3}$$

Prasithsangaree et al [22] summarized the signal distance metric as a generalized weighted distance L_p which can be computed as follows:

$$L_p = \frac{1}{N} \left(\sum_{i=1}^{N} \frac{1}{w_i} |\rho_i - r_i|^p \right)^{1/p}, \tag{II.4}$$

where N is the dimension of signal space or the number of access points deployed by the system. Here w_i is a weighting factor ($w_i \leq 1$) and p is the norm parameter. The weighting factor is used to bias the distance parameter based on how reliable or important the RSS component is from the fingerprint measurement. The number of signal samples or standard deviation of RSS could be used as a weighting factor [22]. With $p = 1$, the distance based on the sum of absolute differences in the RSS sample vector and a location fingerprint is called the Manhattan distance L_1. With $p = 2$, the Euclidean distance L_2 is formulated and it is the most well-known distance used to classify the locations [8, 22, 14]. Another distance metric that can be used is the *Mahalanobis distance* [23]. The Mahalanobis distance has an advantage over the Euclidean distance in that it takes into account the correlations of the fingerprint components and it is scale-invariant, i.e., it is not dependent on the scale of measurements. However, as shown in [23], only a slight improvement in precision performance is achieved with some extra computation of the covariance matrix for location fingerprints.

Different modifications have been proposed to improve the performance of the nearest neighbor method. It is believed that there may be more than one closest neighbor and that a closer estimate should probably be the result from averaging over these locations. Therefore, instead of using only one closest neighbor, a method using k nearest neighbors are suggested [8, 22] and a weighted k nearest neighbor method has been also suggested [19, 22]. The estimated location from these methods is the average of those k neighbor's coordinates. With a small k, there is a small improvement over the single nearest neighbor method [8]. However, Phongsak et al [22] reported that for $k > 8$ location estimation error became worse.

As discussed earlier, the standard deviation of the RSS fingerprint can also be used to provide additional information for the nearest neighbor classifier [14]. For example, when a sample fingerprint lies outside a region with two standard deviations on each side of the mean RSS, the sample vector is categorized as a *non-classifiable pattern* where the sample can not be associated with any position in the database. The mathematical expression for the additional criterion is written as [14]:

$$\begin{aligned}
\rho_1^i - 2\sigma_1^i \leq \quad & r_1 & \leq \rho_1^i + 2\sigma_1^i, \\
\rho_2^i - 2\sigma_2^i \leq \quad & r_2 & \leq \rho_2^i + 2\sigma_2^i, \\
\vdots \quad & \vdots & \vdots \\
\rho_N^i - 2\sigma_2^i \leq \quad & r_N & \leq \rho_N^i + 2\sigma_N^i. \qquad \text{(II.5)}
\end{aligned}$$

It turns out that by using the above criterion, the error distance between the actual position and estimated position is reduced compared to the one without it [14]. There is also a research work trying to suggest improvement on the nearest neighbor search method. A multidimensional search algorithm such as R-Tree, X-Tree, and optimal k-nearest neighbor search are among those search techniques [8].

The advantage of nearest neighbor methods is that they are easy to deploy and that they require minimal training or tuning of the system. Performance of positioning algorithms using nearest neighbor method depends upon how much location fingerprints can be separated in signal space. In addition, as the number of components in fingerprints or the number of fingerprints in database increases, the computational complexity of the methods also increases and may be prohibitive for deployment in a large area.

b. Probabilistic Methods A second parametric classifier for location determination is based on the probabilistic method. The probabilistic approach models the location fingerprint using conditional probability and estimates location using Bayesian inference [7, 15, 16]. It assumes a prior knowledge about the probability distribution of the user's location and the probability distribution of the RSS at each location. The prior location distribution can be found by maintaining user location profiles [15]. The profiles can be useful for location tracking applications [7]. The prior distribution of the RSS, on the other hand, is obtained from either real measurement training data or in the form of radio propagation models with estimated parameters to represent the actual environment.

For each location coordinate $\mathcal{L}$, we can estimate the conditional probability density function or the likelihood function $P(\mathcal{F}|\mathcal{L})$ from a training set consisting of samples of location fingerprints and their labels. Two methods were suggested for estimating the likelihood function: the kernel method and the histogram method [15]. For n samples of the RSS from an access point (one dimension) at a location, the kernel method imposes a probability mass (such as a Gaussian distribution) on each sample of RSS values. As a kernel function, each Gaussian distribution has a mean value ρ which equals to one of n RSS samples and a proper standard deviation σ which is an arbitrary adjustable kernel width. Then, the resulting likelihood function of a sample RSS r given a location $\mathcal{L}$ is an equally weighted sum of all n Gaussian kernel functions:

$$P(r|\mathcal{L}) = \frac{1}{n}\sum_{i=1}^{n}\left[\frac{1}{\sqrt{2\pi}\sigma}exp\left(-\frac{(r-\rho)^2}{2\sigma^2}\right)\right]. \tag{II.6}$$

From Equation II.6, the kernel width σ will have a smoothing effect on the probability density estimation if its value is large. The kernel method can be extended for multiple dimensions (multiple access points) by making an independence assumption and multiplying all conditional probabilities together as $P(\mathcal{F}|\mathcal{L}) = P(r_1|\mathcal{L})P(r_2|\mathcal{L})\cdots P(r_N|\mathcal{L})$. It was also noted that the Euclidean nearest neighbor method is derived from a Gaussian kernel with kernel width approaching zero [15].

Another method for estimating the density function is the the histogram method. This method uses a discrete density function (histrogram) to estimate the continuous density function of the RSS value. The method requires a fixed set of bins to count the frequency of occurrence of RSS samples and produces a histogram. A bin's range is calculated from an adjustable number of bins and the known values of the minimum and the maximum RSS values. Simple equal-width bins of 3, 7 and 27 were used in [15]. The larger the number of bins is, the better the histogram can approximate the probability density function of the RSS. Ladd et al [7] presented a new way to compute $P(\mathcal{F}|\mathcal{L})$ using two different conditional probabilities from two different histograms. The first conditional probability is derived from the frequency of an access point's observations at location $\mathcal{L}$ or how often the signal from that access point is observed. Another conditional probability represents the distribution of the RSS from that access point at the same location. Then these two probabilities are multiplied together to produce the conditional probability of receiving a particular observation.

Each location is initially assumed to have *a priori probability* $P(\mathcal{L})$ which could be equally likely for every location in set L. Then, the location estimation algorithms based on the probabilistic approach apply the Bayes' rule in order to find *a posteriori distribution* of that location which is the conditional probability of the location $\mathcal{L}$ given the location fingerprint $\mathcal{F}$ as:

$$P(\mathcal{L}|\mathcal{F}) = \frac{P(\mathcal{F}|\mathcal{L})P(\mathcal{L})}{P(\mathcal{F})} = \frac{P(\mathcal{F}|\mathcal{L})P(\mathcal{L})}{\sum_{k\in L} P(\mathcal{F}|\mathcal{L}_k)P(\mathcal{L}_k)}. \tag{II.7}$$

From the equation II.7, the probabilistic approach classifies the location fingerprints according to the maximum estimated posterior probability; that is, it selects the location fingerprint according the likelihood functions. Hence, the location estimate $\hat{\mathcal{L}}$ is the maximum likelihood estimator:

$$\hat{\mathcal{L}} = \mathrm{maxarg}_{\mathcal{L}_i\in L} P(\mathcal{L}_i|\mathcal{F}) = \mathrm{maxarg}_{\mathcal{L}_i\in L} P(\mathcal{F}|\mathcal{L}_i)P(\mathcal{L}_i). \tag{II.8}$$

The probabilistic approach can be used to reduce the position search space during the online phase. One example is to use the Joint Clustering technique proposed in [16]. In their system, a cluster is a set of positions where signals from the same set of access points are received. After knowing which cluster a current user's position belongs to, the Bayes's rule (as shown in equation II.8) is applied to positions only within the cluster. However, finding the choice of dimension (a set of access points) for the joint cluster is a critical aspect for this technique.

The probabilistic method can provide better performance over nearest neighbor methods since it has additional information on the probability distribution. The disadvantage of this method is that it needs a large training set to precisely estimate the conditional probability distribution. Many probabilistic methods require explicit knowledge of the probability distributions of the location fingerprints. So it is necessary to characterize the WLAN's received signal strength and the location fingerprints. Because the probabilistic methods incorporate some information of radio propagation, they could provide insight on the underlying mechanism of indoor positioning.

c. Neural Network Methods Neural network methods consider the solution to a user's location from the received signal strengths inside a building or in a complex urban geometry as a complex problem. Hence, the method uses a parallel distributed processing network that consists of interconnected processing elements called *neurons* to collaboratively produce the estimated output location. The neuron for location determination basically consists of a set of input links that are weighted with synaptic weights, a combiner (e.g., an adder) that sums all weighted inputs, and an activation function that limits the amplitude of the output of the neuron. The activation function, also known as the transfer function, is usually a non-linear function. For "yes/no" classification problem, the sigmoid function,$f(x) = 1/(1 + e^{-x})$, is considered as a suitable non-linear function [19].

The neural network that many neurons are interconnecting in both serial and parallel manner to create layers of the network is referred to as a multilayer perceptron (MLP) neural network. Signals flow sequentially through the different layers from the input to the output layer. Hidden layers are layers in the middle between the input and output layers. The MLP neural network is trained with samples of labeled location fingerprints in order to iteratively calculate all synaptic weights inside the neurons. The training process is interactive for each input sample of a location fingerprint in which the synaptic weights are tuned so that the output is the correct location. As a result, the training process automatically creates complex boundaries for location fingerprint classes.

Many works [19, 24, 14] have proposed the use of MLP neural networks to solve the location estimation problem with fingerprints. A simple feed-forward and fully connected neural network which consists of one hidden layer was used to determine location in [14]. The inputs consist of three features, each representing the RSS from three access points. The hidden layer is composed of 20 neurons and the output layer has 19 nodes corresponding to 19 positions on their map. The synaptic weights in the hidden layer are iteratively computed using a gradient of the error propagating backwards from the output layer. This training technique is called the error back-propagation algorithm. Battiti et al [24] implemented a MLP neural network with one hidden layer that uses the sigmoidal function and the output layer that uses an identity function,$f(x) = x$. Their architecture, referred to as 3→8→2, has three RSS inputs for three access points, eight hidden units, and two outputs of 2D coordinates. They used a training technique called one-step-secant (OSS) algorithm to iteratively adjust synaptic weights with second derivatives information.

The advantage of using the MLP neural network is that it requires no prior knowledge such as the location of access point, building geometry, or signal characteristics of the environment (path loss exponent). However, as reported in [14, 24], performance in terms of accuracy and precision do not show significant improvement over the nearest neighbor method. The neural network methods also requires a training process, which can be very slow especially when a large training set is used to improve location estimation performance. Overtraining and overfitting could potentially result in worse performance as well [24]. Analytical models for error performance with this method can be too complicated and only measurement tests are performed to evaluate data not in the training set. Neural network methods abstract out all underlying mechanisms of fingerprinting and leave no insight for understanding the nature of the indoor positioning systems.

d. Support Vector Machine Methods The support vector machines (SVMs) method has been used as a non-parametric non-linear classifier for estimating indoor locations [19, 11]. The SVM method is considered as a tool from statistical learning theory that can be used to derive *the unknown functional dependency* based on observations. The dependency for the study is between the RSS fingerprint and the location information.

The basic concept of the SVMs method is based on the Structural Risk Minimization (SRM) principle that tries to minimize a bound on an expected risk functional or generalization error [19]. The risk functional is defined as the expected value of a loss function. The loss function is a measure of how much the function used to approximate the pattern mapping differs from the real pattern mapping. The overall risk function is showed to be bounded by the empirical risk function and Vapnik-Chervonenkis (VC) confidence interval [19].

The classification operation used for SVMs methods can be briefly summarized as follows. First, the vectors of location fingerprints are mapped into a higher dimensional space called *feature space* [19] by using a function called *kernel* of the SVM to perform vector transformation. There are many SVMs kernel functions that can be used such as polynomial function, radial basis function (RBF), Sigmoid kernel, and Analysis of Variance (ANOVA) kernel. Battiti et al [19] used a Radial Basis Function (RBF) as the kernel of the SVM in their system. Finally, the SVMs algorithm creates an optimal separating hyperplane or decision surface in that feature space and uses the hyperplane to perform classification. The separating hyperplane is not unique in general and is optimal when it has a largest possible distance from the closest training point or a maximal margin. Support vectors are those training vectors that are necessary to define the hyperplanes [19]. In other words, the support vector machine is the learning algorithm (machines) based on support vectors.

The SVMs method is believed as the most sophisticated technique used in the pattern recognition area. However, when applied to indoor positioning systems, the method shows only comparable performance results to the weighted k nearest neighbor method [19, 11]. Suitable kernel function of the SVMs with its parameters, which is later reflected by classifier performance, is not easy to select and there are a variety of kernel functions available. From a practical point of view, the algorithmic complexity of the SVMs may become a prohibitive factor for deploying the method in an indoor positioning system.

4. Performance Summary of Existing Systems

In this subsection, we summarize the performance of existing indoor positioning systems based on WLANs location fingerprinting. Two major performance metrics studied by all systems are location accuracy and location precision. As mentioned in Chapter I, the accuracy of the system is reported in the form of error distance measurement between the estimated location and the actual location of a mobile station. The accuracy metric is typical, shown with the confidence interval or percentage of successful location estimation (also called location precision). It is noted that each existing system has different parameter settings and environment. Hence, a comparison of performance results among these systems can be easily misleading and should be carefully done. Table 1 shows system parameters of different systems while Table 2 exhibits the best reported performance of the systems. These two tables are augmented from the summary tables in [1] to also include recently proposed systems.

Table 1: Parameter Settings of indoor positioning systems

System	**Spacing**	**Positions**	**Samples/Pos.**	**APs**	**Orient.**	**Env.**
RADAR[8]	Nonuniform	70	80($\frac{1}{4}$sec/samp.)	3	4	Hallway
Saha et al[14]	Min. 3.12m	19	1200	3	N/A	1-floor
Roos et al[15]	Uniform 2m	155	40	10	N/A	1-floor
Battiti et at[19]	N/A	257	N/A	6	N/A	1-floor
Ladd et al[7]	3m	11	1307 packets	5	2	Hallway
Prasithsangaree et al[22]	1.5m, 3m	60	40	2-7	4	1-floor
Youssef et al[16]	1.5m	110	300	4	N/A	Hallway
Xiang et al[25]	N/A	100	300 (2sec/samp.)	5	4	1-floor
King et al[26]	1m	166	20(offline), 3(online)	2-6	8	Hallway

From these tables, the performance in terms of accuracy and precision of these systems, although it varies, is quite comparable. In fact, in the Battiti et al [19] study, the best accuracy performance results from four estimation methods were reported to be similar at about 3 meters. Intuitively, higher dimensions of location fingerprint vectors (i.e., more access points deployed) should provide better uniqueness among vectors, thereby improving the performance. Also, performance of the system that has positions only along hallways is better than those with an array of positions spread all over the floor due to fewer positions to confuse with. A recent system is suggested in [26] that applies a recognition of user orientation to a probabilistic method to help boost up some accuracy. However, a larger training space to represent distributions from different positions and orientations is required.

Table 2: Performance comparison of indoor positioning systems

System	**Estimation Method**	**Accuracy and Precision**
RADAR[8]	Nearest Neighbor	within 7 feet, 38%
Saha et al[14]	Nearest Neighbor & Neural Network	no specified accuracy, 90%
Roos et al[15]	Bayesian	best within 8.28 feet, 90%
Battiti et at[19]	SVMs, Bayesian, Neural Network & Weighted k Nearest Neighbor	all within 16-17 feet, 90%
Ladd et al [7]	Bayesian	within 5 feet, 77%
Prasithsangaree et al[22]	Weighted k-Nearest Neighbor	25 feet at 75% & 40 feet at 95%
Youssef et al[16]	Bayesian with Joint Clustering	within 7 feet, more than 90%
Xiang et al[25]	Bayesian with RSS distribution model	within 6 feet, 90% (static device)
King et al[26]	Bayesian with known orientation	within 5.5 feet, more than 50%

Also, user orientation is a needed information (implying additional messaging) to send back when a centralized positioning server is used.

Among the existing systems, unfortunately performance in terms of delay, capacity, coverage, and scalability are not covered by most of these study. Computational complexity during the offline and the online phases for three estimation methods (specifically weighted k nearest neighbors, Bayesian probabilistic, and multilayer perceptron neural network) is discussed in [19]. Besides WLANs-based systems, recent work by Otsason et al [27] suggested the use of GSM channels and cells for indoor location fingerprint system. It is reported that the GSM fingerprints with large coverage, despite an accuracy slightly below that with WLANs, showed good performance for differentiation between floors of a residential building [27]. Combining the two technologies could possibly expand coverage of the positioning system as well. The hybrid approach, which consists of two or more sensing techniques or technologies can also improve the accuracy and precision performance of the system. An example is a combined AOA and TDOA system is suggested in [28].

In addition to the above discussion, one important challenge facing by most conventional WLANs-based systems is that a large measurement data set from many reference positions is needed in order to yield desired performance. Building a huge radio map database can take tremendous amount of the time and may result in computational burden. To deal with database generation, Li et al [29] suggested two possible methods: the *weighted distance inverse* (WDI) and the *universal kriging* (UK) methods. The WDI method is a simple interpolation where the RSS at a given position is estimated by weighted sum of the RSSs from surrounding positions. The weight for each surrounding position is a reciprocal of the distance to the given position. The UK method is based on an iteration method and it involves computation of a spatial correlation (known as a variogram) that is typically used in the mining industry. However, there are many iterations that are needed to be performed and unknown factors needed to be estimated in the UK method. In fact, both methods try to generate a fingerprint without the study of whether the new fingerprint could potentially be an efficient fingerprint or not.

To the best of our knowledge, we have not seen any work that does an analysis on how important a single fingerprint is among other fingerprints in the radio map. *The adoption* *of future location-based services will require an efficient design methodology and sufficient analysis. Understanding the structure of the radio map and the impact of individual fingerprint on others can help predict performance (such as the probability distribution of location selection and distribution of the error distance).* This has never been studied by the research community so far. The design of indoor system based on location fingerprinting should consider an analytical model that explains relationship between system parameters, fingerprint structure, and the system performance. There are few and limited studies of the analytical model in the literature [9, 30]. Unfortunately, none of the models has incorporated knowledge about fingerprint structure and studied its impact on the probability distribution of location selection from an analytical perspective. In this dissertation, we try to enhance the existing study of Dr. Kamol Kaemarungsi in [9] by consolidating aspects that have never been investigated as discussed above. Finally, a design framework is to be supplied to a system designer in order to produce an efficient system deployment.

In the next sections, we will consider tools for analyzing the location fingerprint structure. They are the Voronoi diagram and the proximity graph. These tools are important and later used to model the probability distribution of fingerprint selection and thus distance error.

B. VORONOI DIAGRAM

The Voronoi diagram is simple and it starts from an appealing problem. Given a finite set of distinct, isolated points in a continuous space, we can associate all locations in that space with the closest member of the point set. The solution to the problem is a partitioning of the space into a set of regions. The resulting configuration is called a Voronoi diagram.

Starting back in 1908, a famous Russian mathematician, Georgy Voronoi, was the first to consider this structure of points in space. His original concern was the distribution of set of points which are regularly place in the d-dimensional space generated by linear combination of d linearly independent vectors with integer coefficients. The set contains infinitely many points and the Voronoi diagram generated by this set gives the partition of the space into mutually congruent polyhedra [31]. It is often found that many kinds of natural structures closely resemble the Voronoi diagram. However, different names have been used to refer to the concept in different field of study. It is called medial axis transform in biology and physiology, Wigner-Seitz zones in chemistry and physics, domains of action in crystallography, and Thiessen polygons in meteorology and geography.

The Voronoi diagram has been widely used and proven successful for many decades in various fields including physics, biology, physiology, computer science, and engineering [31]. For example, an astronomer studies the structure of the Universe. An archaeologist applies the diagram in order to identify the parts of a region under the influence of different Neolithic clans. A biomedical scientist studies shape similarity of proteins. An urban planner wants to locate public schools in the city. An engineer and computer scientist try to solve the coverage problem of wireless sensor networks, to determine whether a region of interest is sufficiently covered by a given set of sensors. The diagram is also used for studying point location problem in computational geometry [32].

For ease of exposition, we first give a mathematical definition of the Voronoi diagram in 2-dimensional space or a plane. Given a set of finite number n of points (known as generators) in the Euclidean plane and assume $2 \leq n < \infty$. Then n points are labeled by $p_1, \dots, p_n$, with coordinates $(x_{11}, x_{12}), \dots, (x_{n1}, x_{n2})$ or location vectors $\mathbf{x_1}, \dots, \mathbf{x_n}$. The n points are *distinct* in the sense that $\mathbf{x_i} \neq \mathbf{x_j}$ for $i \neq j, i, j \in I_n = \{1, \dots, n\}$. Let p be an arbitrary point in the Euclidean plane with coordinates (x_1, x_2) or location vector $\mathbf{x}$. Then the Euclidean distance from a generator p_i to p is given by $d(p_i, p) = ||\mathbf{x_i} - \mathbf{x}|| = \sqrt{[(x_{i1} - x_1)^2 + (x_{i2} - x_2)^2]}$. If p_i is the nearest generator point from p or p_i is one of the nearest generator points from p, we have the relation $||\mathbf{x_i} - \mathbf{x}|| \leq ||\mathbf{x_j} - \mathbf{x}||$ for $j \neq i, j \in I_n$. We call the region $V(p_i)$ given by

$$V(p_i) = \{\mathbf{x} : ||\mathbf{x_i} - \mathbf{x}|| \leq ||\mathbf{x_j} - \mathbf{x}||, \text{ for } j \neq i, j \in I_n\} \tag{II.9}$$

the *Voronoi polygon or region* associated with p_i and the set $\mathbb{V}$ given by

$$\mathbb{V} = \{V(p_1), \dots, V(p_n)\} \tag{II.10}$$

the *Voronoi* or *Voronoi tessellation* diagram. Example of the Voronoi diagram for a set of points in the plane is given in Figure 2. Notice that a Voronoi region can be either bounded or unbounded. The boundary of a Voronoi region may consist of line segments,

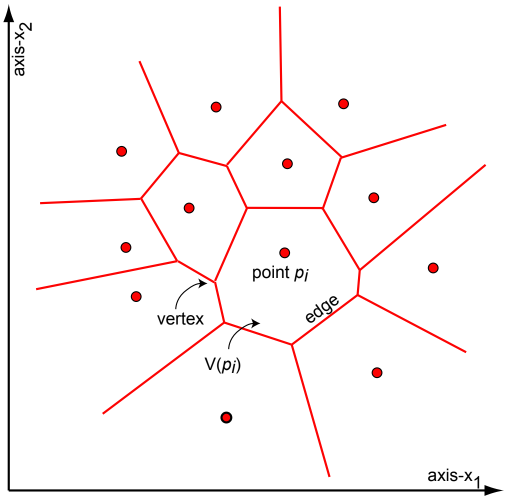

Figure 2: The planar Voronoi Diagram Example

half lines or infinite lines, which we call *Voronoi edges*. We denote a Voronoi edge by e. An end point of a Voronoi edge is called a *Voronoi vertex* and it is a point shared by three or more Voronoi regions.

From basic geometry, a straight line splits a plane into two disjoint regions. We call one of the regions with the line as a *half space* or *half plane*. A Voronoi diagram can be alternatively defined in terms of half planes. Consider the line perpendicularly bisecting the line segment $\overline{p_i p_j}$ joining two generators p_i and p_j. We call this line the *bisector* between p_i and p_j and denote it by $b(p_i, p_j)$. Since a point on the bisector $b(p_i, p_j)$ is equally distant from the generators p_i and p_j, $b(p_i, p_j)$ is written as

$$b(p_i, p_j) = \{\mathbf{x} : ||\mathbf{x_i} - \mathbf{x}|| = ||\mathbf{x_j} - \mathbf{x}||\}, j \neq i.$$

The bisector divides the plane into two half planes and gives

$$H(p_i, p_j) = \{\mathbf{x} : ||\mathbf{x_i} - \mathbf{x}|| \leq ||\mathbf{x_j} - \mathbf{x}||\}, j \neq i.$$

We call region $H(p_i, p_j)$ the dominance region of p_i over p_j since the distance (from a point p) to generator p_i is shorter than the distance to generator p_j. Now if we consider the regions from point p_i to all other points p_j, we find that the intersection of all $H(p_i, p_j)$ regions yields the Voronoi region associated with p_i. Hence, an alternative definition of the Voronoi region can be written as

$$V(p_i) = \bigcap_{j \in I_n - \{i\}} H(p_i, p_j) \tag{II.11}$$

and a set $\mathbb{V} = \{V(p_1), \dots, V(p_n)\}$ is the Voronoi diagram.

The construction of the Voronoi diagram has been studied using various methods. A straightforward method is done based on the intersection of the half planes as given in equation II.11. It turns out that time complexity for this method is $O(n^3)$ [31]. The convex hull method is another Voronoi diagram construction method used by a well-known commercial software such as MATLAB. The convex hull for set of points in the space is the minimal convex set containing all points. The method transforms generators in the d-dimensional space into points in (d+1)-dimensional space. The convex hull is then computed and transformed back to the original space to obtain the Voronoi diagram. Worst case time complexity for d-dimensional Voronoi diagram using the method is $O(n^{\lceil d/2 \rceil})$ [33]. Other methods has been proposed for the Voronoi diagram construction including incremental, edge flipping, divide and conquer, and plan sweeping methods. Construction of the Voronoi diagram is not the focus for our study. However, extensive details for different methods can be found in [31, 34].

In next section, we will look at analysis of proximity structures which is related to the Voronoi diagram. We will review different proximity graphs used to study proximity structures of generator points in space.

C. PROXIMITY STRUCTURE AND PROXIMITY GRAPHS

The universe is made up of all manner of things, each seemingly unique. Those things, however, interact and relate on one another in all sorts of ways. The existence of interaction and relation of objects or incidents produces specific structure of the data observed within a given domain space. Study the underlying structure or pattern of the data can help one acquire closeness or proximity information about the data of particular subject.

In mathematics and computer science, a graph is a mathematical tool that is used to study structure of data and model relations between data collection, represented by set of points in space. A graph is a set of points or nodes connected by links or edges. A proximity graph is a particular type of graph used to observe a structure of data and exhibit a relation between points by connecting pairs of points that are deemed *close* by some proximity measure. In other words, proximity graph represents neighbor relationships between data points. Different proximity graphs have been defined by different proximity measures or "forbidden regions" for the data set. The proximity graphs have found applications in diverse areas including computer vision, pattern classification, database design, geographic analysis, and computer networks. For instance, in wireless ad hoc or sensor networks, proximity graphs can be used to determine a power efficient topology for the nodes in a completely distributed environment [35].

In following subsection, we will describe three important proximity graphs that will be used in our study of the indoor positioning system. These graphs include Delaunay proximity graph(DG), Gabriel proximity graph(GG), and Relative Neighborhood proximity graph(RNG).

1. The Delaunay Proximity Graph (DG)

The *Delaunay proximity graph* is a graph structure that has close relationship to the Voronoi diagram. It is referred as *Delaunay tessellation* or *Delaunay triangulation* [36] and it is a dual graph of the Voronoi diagram. To obtain the DG from a Voronoi diagram, let us consider a graph in the Figure 3. Given a Voronoi diagram (the same as in Figure 2), we can construct the DG by first choosing a Voronoi edge (the heavy broken line in Figure 3). This edge is shared by two Voronoi regions. Then we join the generator points of these Voronoi regions by a line segment (heavy solid line in Figure 3). We carry out this line generation with respect to all Voronoi edges in the Voronoi diagram. This way we can divide the map of points into several triangles. Each triangle, called a Delaunay triangle, contains the boundary consisting of line segments called Delaunay edges. A triangulation of all the points results in the dual graph for the Voronoi diagram and it is the DG.

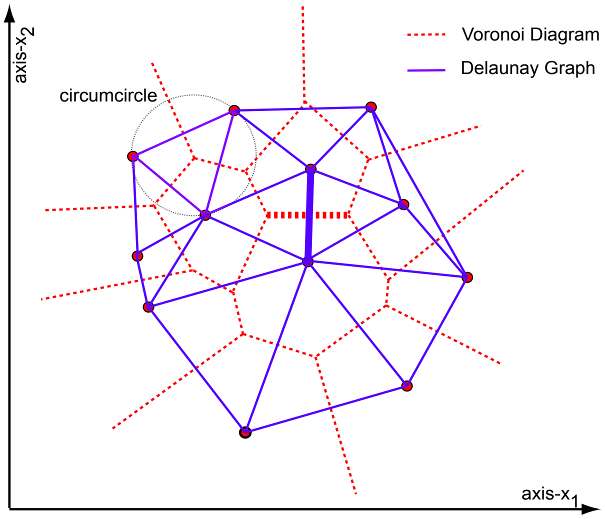

Figure 3: The Delaunay Proximity Graph

Alternatively, we can construct the DG by generating line segments with respect to every Voronoi vertex. Let $\mathbb{V}(P)$ be a Voronoi diagram generated by a set of distinct points $P = \{p_1, \ldots, p_n\}$ ($3 \leq n < \infty$) that satisfies the non-collinearity assumption.[1] Let $Q = \{q_1, \ldots, q_{n_v}\}$ be the set of Voronoi vertices in $\mathbb{V}(P)$; $\mathbf{x_{i1}}, \ldots, \mathbf{x_{ik_i}}$ be the location vectors of the generators whose Voronoi regions share vertex q_i. Then we define the set by

$$T_i = \{\mathbf{x} : \mathbf{x} = \sum_{\mathbf{j=1}}^{\mathbf{k_i}} \lambda_\mathbf{j} \mathbf{x_{ij}} \text{ ,where } \sum_{\mathbf{j=1}}^{\mathbf{k_i}} \lambda_\mathbf{j} = 1, \lambda_\mathrm{j} \geq 0, \mathrm{j} \in \mathrm{I}_{\mathrm{k_i}}\}, \tag{II.12}$$

and let

$$\mathbb{D} = \{T_1, \ldots, T_{n_v}\}. \tag{II.13}$$

If $k_i = 3$ for all $i \in I_{n_v}$, we called set $\mathbb{D}$ the *Delaunay* triangulation or graph of P. If there exists at least one $k_i \leq 4$, we partition T_i having $k_i \leq 4$ into $k_i - 2$ triangles by non-intersecting line segments joining the vertices, altogether resulting in the DG [31]. Note that the DG is constructed so that a circumcircle (centered at a Voronoi vertex) of any Delaunay

[1] the *non-collinearity* assumes that for a given set of points $P = \{p_1, \ldots, p_n\} \subset \mathbb{R}^m (n \geq 3), p_1, \ldots, p_n$ are not on the same line.

triangle is *empty* (i.e. no points is inside the circumcircle of any triangle). An example of a circumcircle is shown by a circle in Figure 3. Moreover, we can define a Delaunay neighbor of a point p_i as a generator point that has a Delaunay edge to point p_i.

With the fact that the DG is a dual graph of the Voronoi diagram, the worst case computational complexity for construction of the d-dimensional DG is the same as that from the d-dimensional Voronoi diagram, $O(n^{\lceil d/2 \rceil})$ [33].

The DG can be regarded as the connected geometric graph and it contains many subgraphs. Next two subsections describe two subgraphs of the DG used in this study, namingly the Gabriel proximity graph and the relative neighborhood proximity graph.

2. The Gabriel Proximity Graph (GG)

A Gabriel proximity graph is a graph that connects a set of points in the Euclidean space. Given a set of points P as before, two points p_i, p_j in P are connected by the edge in the GG whenever the circle having line segment $\overline{p_i p_j}$ as its diameter contains no other points from the set P. The circle is called a diametral circle and the edge is called a Gabriel edge. An example of a GG is shown in Figure 4 and note that it is a subgraph of the DG in Figure 3.

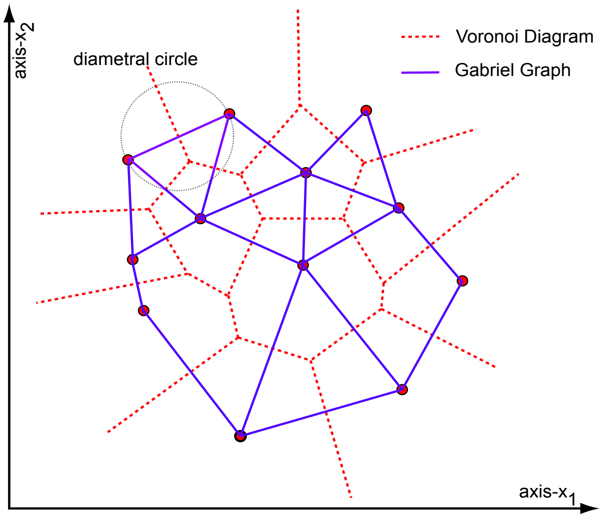

Figure 4: The Gabriel Proximity Graph

A mathematical definition of the GG can be given as following. Let a set of edges, E, such that an edge (p_i, p_j) satisfies a certain condition. Given $d(.,.)$ be the Euclidean distance in $\mathbb{R}^d$. The GG is a proximity graph with the set of edges defined as;

$$\begin{aligned}(p_i, p_j) \in E \Leftrightarrow^{iff} d^2(p_i, p_j) \leq \\ d^2(p_i, p_k) + d^2(p_j, p_k), \text{for all } p_k, k \neq i, j\end{aligned} \tag{II.14}$$

Note that the GG is constructed so that a diametral circle for each Gabriel edge is empty (i.e. no points is inside a diametral circle). An example of a diametral circle is shown by a circle in Figure 3. We can see that, given the DG, a Delaunay edge $\overline{p_i p_j}$ is a Gabriel edge if and only if this edge intersects its dual Voronoi edge. Like in the GG, we can also define a Gabriel neighbor of a point p_i as a generator point that has a Gabriel edge to point p_i.

As a subgraph, the GG can be found in linear time if the DG or Voronoi diagram is given as explained above. However, this method to construct the GG is not quite efficient and not desirable with higher dimensions. Bhattacharya et el [37] proposed two methods for construct the GG. The first method using a brute force approach to search every possible pair of points (a total of $\frac{n(n-1)}{2}$), and test all p_k as in equation II.14. Hence, the complexity of this method is $O(dn^3)$. This method is still prohibitive if n is large. The second method uses a heuristic approach to reduce number of pairs to be tested and it reduces complexity to $O(dn^2)$ (see [37] for details).

3. The Relative Neighborhood Proximity Graph (RNG)

A Relative Neighborhood Proximity graph is a graph of a finite set of points P and it is also a subgraph of the Gabriel graph (i.e. RNG $\subset$ GG $\subset$ DG). Given definitions as in previous subsection, the set of edges, E, in the RNG is mathematically defined as

$$\begin{aligned}(p_i, p_j) \in E \Leftrightarrow^{iff} d(p_i, p_j) \leq \\ max[d(p_i, p_k), d(p_j, p_k)], \text{for all } p_k, k \neq i.\end{aligned} \tag{II.15}$$

Equivalent definition is based on the concept of *lune*, defined as the disjoint intersection between two circles (or hyperspheres) centered at p_i and p_j and whose radii are equal to the distance between them. Examples of a RNG and a lune are shown in Figure 5.

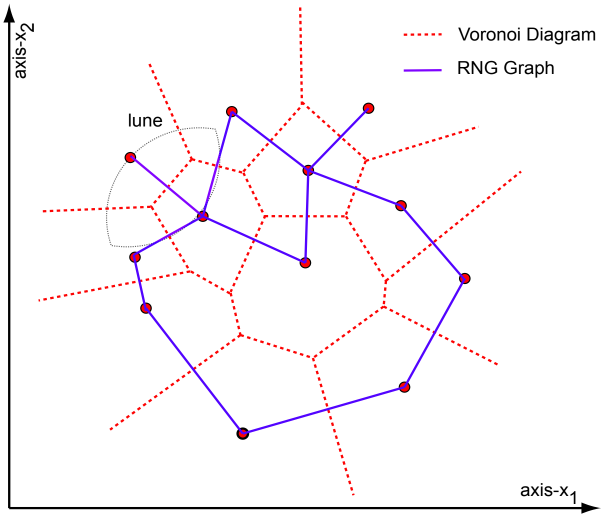

Figure 5: The Relative Neighborhood Proximity Graph

In addition, we define a RNG neighbor of a point p_i as a generator point that has a RNG edge to point p_i. The construction of the RNG can be performed in the same way as the GG construction. Both brute force and heuristic approaches similar to GG can be used [37].

To summarize, data points are meaningless without understanding of the underly structure and relations among them. The concept of Voronoi diagram and Proximity graphs enables one to extract proximity information such as neighbors in a set of data space based on different measures or criteria. Table 3 summarizes different concepts introduced in this chapter that is used to analyze a proximity information and solve a proximity problem.

Table 3: Summary of Relevant Concepts for Proximity Problem

Diagram/Graph	**Brief Description**	**Complexity**
Voronoi	partitioning of plane into convex regions, each contains one generator and every point in given region is closer to its generator	$O(n^{\lceil d/2 \rceil})$
DG	a triangulation of points in which every *circumcircle* of a triangle is empty circle	$O(n^{\lceil d/2 \rceil})$
GG	a graph where its edge from a pair of points produces a *diametral* circle that is empty	$O(dn^3)$, heuristic $O(dn^2)$
RNG	a graph where its edge from a pair of points produces a *lune* that is empty	$O(dn^3)$, heuristic $O(dn^2)$

III. ANALYTICAL MODELING OF LOCATION FINGERPRINTS FOR INDOOR POSITIONING

This chapter contains the study used to analyze fingerprinting based indoor location systems. The first section discusses a mathematical model for indoor location fingerprinting. The model characterizes the distance in signal space between the measured fingerprint and fingerprints in the radio map as random variables. This model is used to determine the probability of selecting fingerprint when a set of a) two and b) multiple locations are considered. Then, we present a new analytical model to estimate the probability distribution of fingerprint selection. We study proximity information, in terms of the defined neighbor set, which is extracted from the structure of location fingerprints using both the Voronoi diagram and proximity graphs discussed in Chapter II. Comparison results from the new model, the simulations, and using measurement data are also shown.

A. INDOOR LOCATION FINGERPRINT MODEL

In this section we describe the preliminary model described previously in [9] which is a precursor to our model. Consider an indoor positioning system overlaid on a WLAN in a single floor inside a building. We assume that there are N access points (APs) in the area and they are all visible throughout the area under consideration. A square grid is defined over the two-dimensional floor plan and any estimate of a MS's location is limited to the points on this grid. Assuming that the grid spacing results in L points along both the x and y axes, we have $L \times L = L^2$ positions in the area. Any location can be represented by a label (x, y) where x and y represent the 2D coordinates on the floor. We assume zero height (i.e., $z = 0$) for all coordinates.

During the offline phase a total of $K = L^2$ of the RSS vectors are collected from site-survey at predetermined grid points. All K entries are recorded in a radio map database and each entry includes a mapping of the grid coordinate (x, y) to the vector of corresponding RSS values from all APs in the area. Each element in each vector in the database is assumed to be the *mean* of the RSS from each of the N access points in the area. This is typically done by collecting a large number of samples of the RSS for different orientations of the MS, and calculating an average value. This approach reduces variations due to orientation and time in the system. During the online phase, to determine the MS's location, a sample of the RSS from all APs at the current position is obtained. This sample vector is compared with all K existing entries in the database. The fingerprint entry that has the closest match to the users sample of RSS is used by the system as the estimate of the user's current location.

To derive mathematical models for predicting performance, two vectors are used in estimating the location of the MS, *a sample vector* and *a fingerprint vector*. The sample vector consists of samples of the RSS measured at the MS from N access points in the the area. The sample vector is denoted as: $R = [r_1, r_2, r_3, ..., r_N]$. Each component in the vector is assumed to be a random variable such that:

- The random variables r_i (in dBm) for all i are mutually independent.
- The random variables r_i (in dBm) are normally (or Gaussian) distributed.
- The (sample) standard deviation of all the random variables r_i is assumed to be identical and denoted by σ (in dB).
- The mean of the random variable r_i or $E\{r_i\}$ is denoted as ρ_i (in dBm).

The fingerprint vector in the radio map consists of the means of all the RSS random variables at a particular location from the N access points and it is denoted as: $\tilde{R} = [\rho_1, \rho_2, \rho_3, ..., \rho_N]$.

The assumption that the RSS is a normally distributed random variable is acceptable. Our previous study in [38] observed that the RSS's distribution often exhibits left-skewness and varies according to its average value or its location. However, when the AP is far from the measurement location and the RSS contains no direct line-of-sight, the distribution can be closely approximated by a Gaussian distribution. Moreover, this assumption allows tractability of the mathematical model. Also, there is no observable relationship between the RSS variations transmitted by different APs. Hence, the assumption of independence is reasonable.

As discussed earlier, the "signal distance" between the sample RSS vector and the fingerprint is used to determine which of the points on the grid corresponds to the position of the MS. The (x, y) coordinates corresponding to the fingerprint that has the smallest distance from the sample RSS vector is returned as the estimated location. This approach is sometimes referred as *the Nearest Neighbor Point in Signal Space* (NNSS) [8]. The signal distance, being different from physical distance, is calculated by the Euclidean distance between $\tilde{R}$ and R and it is given as:

$$Z = [\sum_{i=1}^{N}(\rho_i - r_i)^2]^{1/2}. \tag{III.1}$$

A detailed analysis of the characteristics of the Euclidean distance metric Z for indoor location fingerprinting can be found in [9]. For example, Z can have either a central or non-central chi distribution.

Next we discuss the mathematical model modified from [9] for predicting the probability of selecting the correct location fingerprint when the grid system contains two locations and multiple locations.

1. Probability of Selecting the Correct Location Fingerprint from a Set of Two

Consider a grid system with two grid points, indexed as i and k, and assume a MS is at the i^{th} grid point. We define the *pairwise error probability (PEP)* as the probability that a sample vector R_i is closer to the fingerprint vector $\tilde{R}_k$ than the target fingerprint vector $\tilde{R}_i$. In fact, it is the probability that we have an incorrect estimate of the location (picking the k^{th} grid point instead of the i^{th} grid point). Given sd_{ik}, the Euclidean signal distance between $\tilde{R}_i$ and $\tilde{R}_k$, we can compute the pairwise error probability $PEP(\tilde{R}_i, \tilde{R}_k)$ between the target (correct) fingerprint vector $\tilde{R}_i$ and another (incorrect) fingerprint vector $\tilde{R}_k$ as follows:

$$
\begin{aligned}
PEP(\tilde{R}_i, \tilde{R}_k) &= P\{R_i \text{ is closer to } \tilde{R}_k \text{ than } \tilde{R}_i\} \\
&= P\{||\tilde{R}_k - R_i|| < ||\tilde{R}_i - R_i||\} \\
&= \int\limits_{x=\frac{sd_{ik}}{2}}^{x=\infty} \frac{1}{\sqrt{2\pi\sigma^2}} e^{(\frac{-x^2}{2\sigma^2})}\,\mathrm{d}x \\
&= Q(\frac{sd_{ik}}{2\sigma}). \qquad \text{(III.2)}
\end{aligned}
$$

$||.||$ denotes the magnitude (i.e., Euclidean distance) of an RSS vector. $Q(x)$ represents the right-tail probability for a standard Gaussian random variable where the random variable exceeds x. Note that $PEP(\tilde{R}_i, \tilde{R}_k) = PEP(\tilde{R}_k, \tilde{R}_i)$ (i.e., the MS is at the k^{th} grid point, but its location is estimated to be the i^{th} grid point).

The chance of the event that the distance between the sample RSS vector R_i and the correct location fingerprint $\tilde{R}_i$ is smaller than the distance between the sample RSS vector R_i and the incorrect neighboring location fingerprint $\tilde{R}_k$ is recognized as the probability of returning the correct location. When considering only two location fingerprints, the pairwise probability of returning the correct location or *pairwise correct probability (PCP)* between the correct fingerprint vector $\tilde{R}_i$ and an incorrect fingerprint vector $\tilde{R}_k$ can be computed as:

$$PCP(\tilde{R}_i, \tilde{R}_k) = 1 - PEP(\tilde{R}_i, \tilde{R}_k) = 1 - Q(\frac{sd_{ik}}{2\sigma}). \qquad \text{(III.3)}$$

2. Probability of Selecting the Correct Location Fingerprint from a Set of Many

In a positioning system, the radio map database contains several location entries and fingerprints. To find the probability of returning a correct location, the joint probability density function (PDF) of the location fingerprints needs to be known. Let $C_k = ||\tilde{R}_i - R_i|| - ||\tilde{R}_k - R_i||$ be the comparison variable. The variable C_k compares the distance between the sample RSS vector and the correct fingerprint $\tilde{R}_i$ and the corresponding distance to the incorrect fingerprint $\tilde{R}_k$. The index k runs from 1 to K excluding the correct location index (the index i in this case). So, the probability of correct decision is described by:

$$\begin{aligned} Prob\{\text{Correct Estimation}\} &= \mathcal{P}_c \\ = P\left\{C_1 \leq 0, \cdots, C_{i-1} \leq 0, C_{i+1} \leq 0, \cdots, C_K \leq 0\right\} & \end{aligned} \tag{III.4}$$

Unfortunately, deriving such a probability analytically proves to be difficult and may not be practical when there is a large number of location fingerprints in the database. The model in [9] applies a simple approximation that assumes independence among the many different comparison variables. So

$$\begin{aligned} Prob\{\text{Correct Estimation}\} = \mathcal{P}_c &\approx \prod_{\substack{k=1 \\ k\neq i}}^{K} Pr\{C_k \leq 0\} \\ Prob\{\text{Error Estimation}\} = \mathcal{P}_e &\approx 1 - \mathcal{P}_c. \end{aligned} \tag{III.5}$$

This simple model yields a reasonable estimation for the probability of selecting the correct location. However, the above analytical model is not sufficient to find the *probability distribution* of the *error distance*. To obtain the distribution of the error distance, we need an estimate of the probability of selecting an arbitrary location (and then associating it with the corresponding error in physical distance). Further, we want to find the chance of picking one location against other locations in order to determine the "level of importance" of a corresponding fingerprint in terms of how it impacts the probability of returning the correct location. In fact, the internal structure or distance relationships among location fingerprints has a direct impact on the performance of the positioning system. Although seemingly unique, each fingerprint will have different influence level in terms of the chance of selecting the fingerprint and thus the distance error in the estimated location. Therefore, understanding the fingerprint structure that dictates both a decision region and the probability of fingerprint selection is critical to the design of a good wireless positioning system.

In the next section, we will discuss an extended analytical model that considers fingerprint structure in order to better model the probability distribution of fingerprint selection and thus the distance error.

B. ANALYTICAL MODEL FOR PROBABILITY DISTRIBUTION OF PICKING FINGERPRINTS

To create a model for the probability distribution, we apply a Voronoi diagram and various proximity graphs which are explained below.

1. Voronoi Diagram and Proximity Structure of Location Fingerprint

The concepts of the Voronoi diagram and the proximity graphs have been adopted in many research area.These concepts are closely related. *However, they have never been studied and applied to a context of the indoor positioning before.* We believe that these concepts altogether can constitute a fundamental theory which is essential to analysis the structure of location fingerprints for indoor positioning systems.

The Voronoi diagram is a tool introduced to find "decision regions" for each fingerprint in the system (explained below). A proximity graph is a tool that helps analyze the fingerprint structure and yield proximity information or a "neighbor set" of a given fingerprint. A neighbor is a fingerprint that is believed to be more important to the precision of selecting the target fingerprint, i.e., one that is "relatively close" to the target fingerprint in signal space. Applying these tools, we create a mathematical model for approximating the probability distribution of fingerprint selection.

To restate, the Voronoi diagram of a set of K sites (denoted as set S) is a partition of space into regions, one per site, where the region for a site s is the set of points that are closer to s than to any other sites of S.

The Voronoi diagram of a set of fingerprints $\mathcal{FP} = \{\tilde{R}_1, \tilde{R}_2, \cdots, \tilde{R}_K\}$ is defined as a division of the space according to the nearest-neighbor rules, where each fingerprint from $\mathcal{FP}$ is associated with a region of the Euclidean space closest to a given fingerprint from $\mathcal{FP}$. Such a region is called a *Voronoi region* (or a *decision region*) for a fingerprint. From the definition, the Voronoi region of a fingerprint $\tilde{R}_i$ in N-dimensional Euclidean space (N access points) can be expressed as:

$$V(\tilde{R}_i) = \{R : ||\tilde{R}_i - R|| \leq ||\tilde{R}_j - R||, \text{ for } \forall j \neq i\},$$

Combining of the Voronoi regions of all the fingerprints yields the Voronoi diagram for the fingerprints. Alternatively, the Voronoi diagram can be defined by a bisector $B(\tilde{R}_i, \tilde{R}_j) = \{R : ||\tilde{R}_i - R|| = ||\tilde{R}_j - R||\}$ between two fingerprints $\tilde{R}_i$ and $\tilde{R}_j$. As defined in chapter II, the bisector is a line perpendicular to the line segment $\overline{\tilde{R}_i\tilde{R}_j}$ that bisects this segment in Euclidean 2D space. It is a plane (hyperplane) perpendicular to the segment $\overline{\tilde{R}_i\tilde{R}_j}$ that bisects this segment in 3D (higher-D).

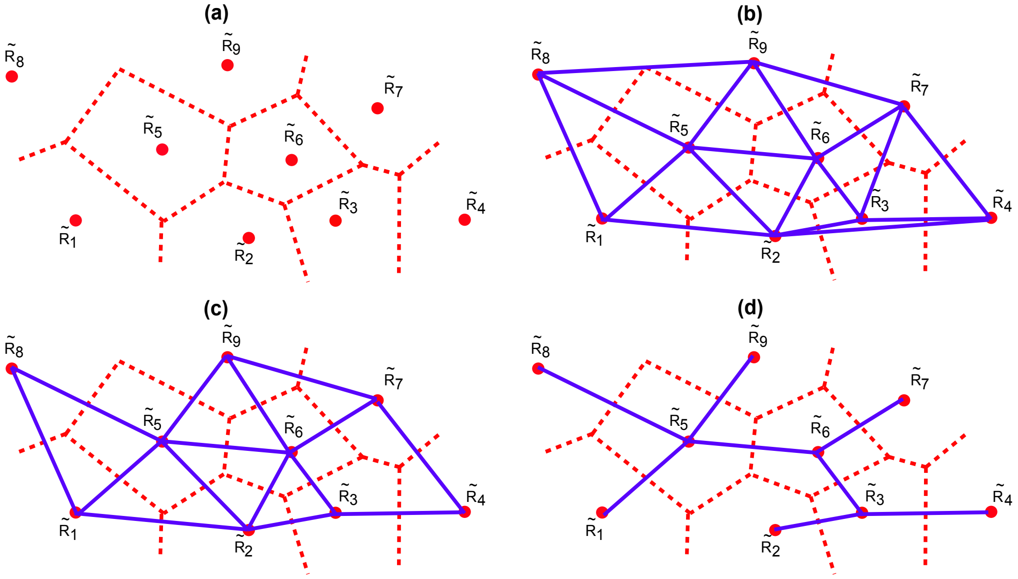

Figure 6: Example Structures for 9 Fingerprints: (a) Voronoi diagram (b) DG (c) GG (d) RNG

The Voronoi diagram is created using bisectors between any two fingerprints to derive decision regions for all fingerprints in the radio map. A bisector is also referred as a *Voronoi edge*. Figure 6(a) shows an example of a Voronoi Diagram (red dash lines) for 9 fingerprints in 2D space. If a sample RSS vector falls in the Voronoi region of a fingerprint $\tilde{R}_m$, it is closest to that fingerprint in terms of Euclidean distance. Thus, the NNSS approach will pick

$\tilde{R}_m$ (or decide that location on the grid corresponding to $\tilde{R}_m$ is the correct location). The Voronoi regions can be used to determine the probability of picking a particular fingerprint given the statistics of the random RSS vector. The method of doing this is to determine the probability that the RSS vector falls in the Voronoi region. This is mathematically tractable for rectangular Voronoi regions, but not so for irregular polygonal Voronoi regions. Instead, we use the related concept of proximity graphs to approximate this probability.

It is a general fact that nearby objects tend to exert a greater influence and have greater relevance than more distant objects. This is also true for the location fingerprints. A fingerprint with small Euclidean distance tends to have more influence to a given fingerprint than one at farer Euclidean distance. Hence, a concept of proximity structure would be helpful to express proximity information of the location fingerprints in the radio map database.

We use the idea of proximity graphs to extract structure information (especially proximity information such as a neighbor set) for fingerprints. Two fingerprints are "close together" and they are neighbors if there are no other fingerprints in a certain "forbidden region" defined differently by different proximity graphs. We consider three proximity graphs; the Deluanay graph (DG), the Gabriel graph (GG), and the relative neighborhood graph (RNG) as discussed in Chapter II. Examples of these graphs are shown by solid lines in Figures 6(b), (c), and (d) respectively. From the particular proximity graph, we define a neighbor fingerprint as a fingerprint point that has edge connected to a given fingerprint.For example, in DG a neighbor fingerprint is one that has a Delaunay edge connected to a given fingerprint (from Figure 6(b), neighbors of fingerprint $\tilde{R}_6$ are fingerprint $\tilde{R}_2$,$\tilde{R}_3$,$\tilde{R}_5$,$\tilde{R}_7$, and $\tilde{R}_9$). The same idea is applied for defining neighbors with GG and RNG.

Different proximity graphs can yield different sets of neighbors. A good graph (in our case) must give us the right set of neighbors such that they represent the top candidates for location fingerprint selection (given the location of a MS). Note that the DG yields the largest set of neighbors while an RNG yields the smallest set of the three graphs. The way we employ proximity graphs is described in the next subsection.

2. Approximate Probability Distribution using Proximity Graphs

As mentioned in subsection III.A.2, to find the exact probability of selecting a fingerprint and thus a location on the grid, the joint probability density function (PDF) of fingerprints is needed. A mathematical expression for such a function is very difficult to derive. Moreover, we simply wish to find the probability of selecting one fingerprint against others in order to evaluate the influence level of a fingerprint on the probability of correctly selecting a fingerprint. Given a MS at the i^{th} grid point, the probability of selecting fingeprint $\tilde{R}_k$ is approximated using a new model as follows:

$$\begin{aligned} Prob\{\text{Selecting Fingerprint } \tilde{R}_k\} \approx PEP(\tilde{R}_i, \tilde{R}_k) \times \\ \prod_{j \in \text{neighbor of } i} PCP(\tilde{R}_k, \tilde{R}_j). \end{aligned} \tag{III.6}$$

The idea behind this equation is as follows. Instead of using all of the comparison variables C_k as in (III.5), we use only the neighbors as the most significant candidates – and still use an independence assumption. Given that the MS is at grid point i, it is the probability of selecting $\tilde{R}_k$ and not $\tilde{R}_i$, AND the probability of selecting $\tilde{R}_k$ and not any of the other neighbors of $\tilde{R}_i$. That is, the above approximation weighs the $PEP(\tilde{R}_i, \tilde{R}_k)$ with all PCPs between fingerprint $\tilde{R}_k$ and only "neighbors" (as defined by the proximity graphs) of the correct fingerprint $\tilde{R}_i$. The influence from remote fingerprints is ignored by using this approach. The set of neighbors to be employed in the approximation depends on the choice of the proximity graph. Moreover, this approach allows us to compute the probability of not only picking the correct location, but also the probability of picking any of the neighbors in the set. However, fingerprints outside the neighbor set are assumed to be never picked (although there is always a negligible probability that this may happen). To find the probability of selecting the correct fingerprint, the first line in (III.5) can be used. However, instead of using all K fingerprints we use only the neighbor set of the correct fingerprint in the computation, for better estimation (a claim that is validated by our results in section III.C).

C. PERFORMANCE EVALUATION

In this section, we evaluate the analytical model discussed in section III.B. We study the results of the probability of fingerprint selection (for correct and incorrect fingerprints). We then look at the results of the distribution of the probability of fingerprint selection. The distribution of the error distance of the location estimate is also studied. We do this for a simple system model described below as well as for a real radio map derived from measurement.

1. System Model

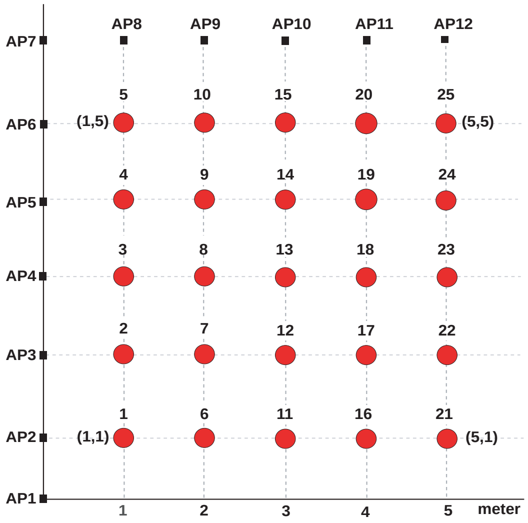

Figure 7: 25 Grid points for an indoor positioning system

The system model considered for evaluation is as follows. We use a 25 grid point system as shown in Figure 7, with a grid spacing of 1 meter ($\approx$ 3 feet). We place access points along the outer most positions (small dark rectangles in Figure 7). Initially we consider only two access points. The two access points are AP1 = (0,0) and AP8 = (1,6). The position of the mobile station could be at any one of the 25 locations in Figure 7. Suppose the physical distance of the k^{th} grid point from the j^{th} AP is $d_{j,k}$ meters. The mean or expected value of r_j for the grid point can be calculated from the mean path loss given by:

Table 4: EXAMPLE RADIO MAP

Access Points	**AP1** (dBm)	**AP4** (dBm)	**AP8** (dBm)	**AP12** (dBm)
Coordinate	**(0,0)**	**(0,3)**	**(1,6)**	**(5,6)**
Loc7(2,2)	-57.1918	-53.1094	-63.7390	-67.0888
Loc8(2,3)	-61.4089	-51.1712	-59.1300	-64.2355
Loc9(2,4)	-65.1506	-53.1094	-53.1094	-61.4089
Loc12(3,2)	-61.4089	-59.1300	-65.1506	-65.1506
Loc13(3,3)	-64.2355	-58.2149	-61.4089	-61.4089
Loc14(3,4)	-67.0888	-59.1300	-57.1918	-57.1918
Loc17(4,2)	-65.1506	-63.7390	-67.0888	-63.7390
Loc18(4,3)	-67.0888	-63.2124	-64.2355	-59.1300
Loc19(4,4)	-69.2330	-63.7390	-61.4089	-53.1094

$$Pl(d_{j,k}) \quad = \quad Pl(d_0) + 10.\alpha.log_{10}(d_{j,k}) \qquad \text{(III.7)}$$

Here $Pl(d_0)$ is the free-space loss at the reference distance of $d_0 = 1$ m (i.e., 54.13 dBm for line-of-sight propagation (LOS) and 37.3 dBm for non-line-of-sight propagation (NLOS) as reported in some measurements [39]). The variable α denotes the path loss exponent, which for indoor locations could be between 1-6 [40]. The mean received signal strength $E\{r_j\}$ can be computed using:

$$E\{r_j\} \quad = \quad \rho_j = Pt - Pl(d_{j,k}) \qquad \text{(III.8)}$$

where P_t is the transmit power of the access point which we will fix at 15 dBm for IEEE 802.11b based WLANs. The standard deviation of the RSS for this indoor positioning system is assumed to be $\sigma = 4$ dB as reported in [20]. Other values of σ for indoor location systems are reported in [17]. A more accurate path loss prediction model, such as those including wall and floor attenuation factors suggested in [8], could also be used. The path loss equation only provides us with the mean received signal strength value. It is possible to use values from actual measurements as well without changing our analytical model.

Table 4 shows the database of location fingerprints when access point AP1, AP4, AP8, and AP12 in Figure 7 are deployed. The table contains the location fingerprints of locations 7-9, 12-14,and 17-19, which are located around the center of the system. Note that if only one access point is present, the fingerprints, as listed in the second column, may not be unique. This happens when two points on the location grid are at the same distance from the access point. Additional access points make the fingerprint unique. An example of a Voronoi diagram of the 25 location fingerprints when AP1 and AP8 are deployed, along with its DG, is shown in Figure 8. The Voronoi diagram with the Gabriel graph and the Relative neighbor graph are shown in Figure 9 and 10 respectively. Note that, although a symmetric square physical location grid is used, the resulting fingerprints are not necessary symmetric in signal space. Fingerprints ($\tilde{R}_{21} - \tilde{R}_{25}$) that are far from an AP tend to stay closer while fingerprints closer to the APs ($\tilde{R}_1 - \tilde{R}_5$) tend to be apart in signal space. The decision regions for fingerprints are also different in shape and size. Bounded (unbounded) regions are found at inner (outer) grid points.

Next, we compare the results from simulation and the analytical model. We simulated 10,000 RSS samples from a given MS location and applied the nearest neighbor computation to estimate its location. Then we computed relevant performance metrics (i.e., error probabilities, error distances) from simulations.

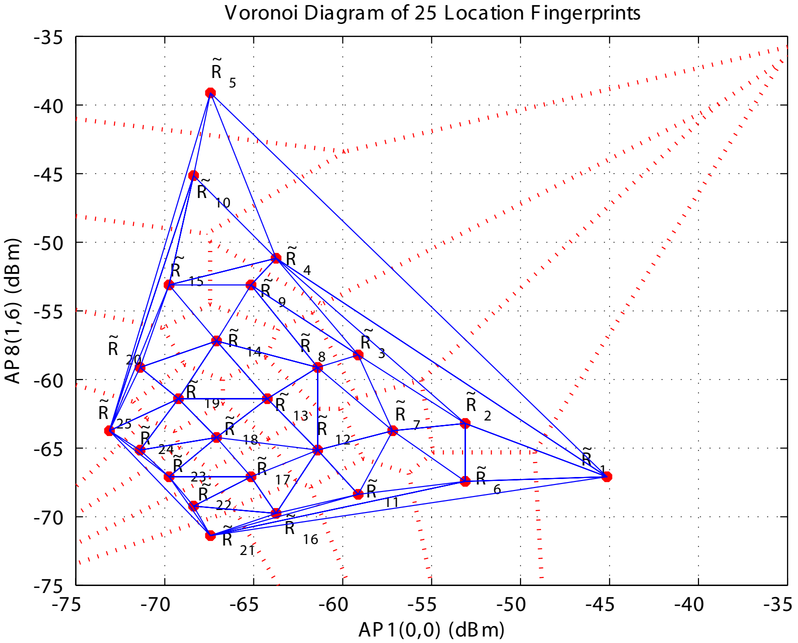

Figure 8: Voronoi Diagram and DG of 25 Location Fingerprints

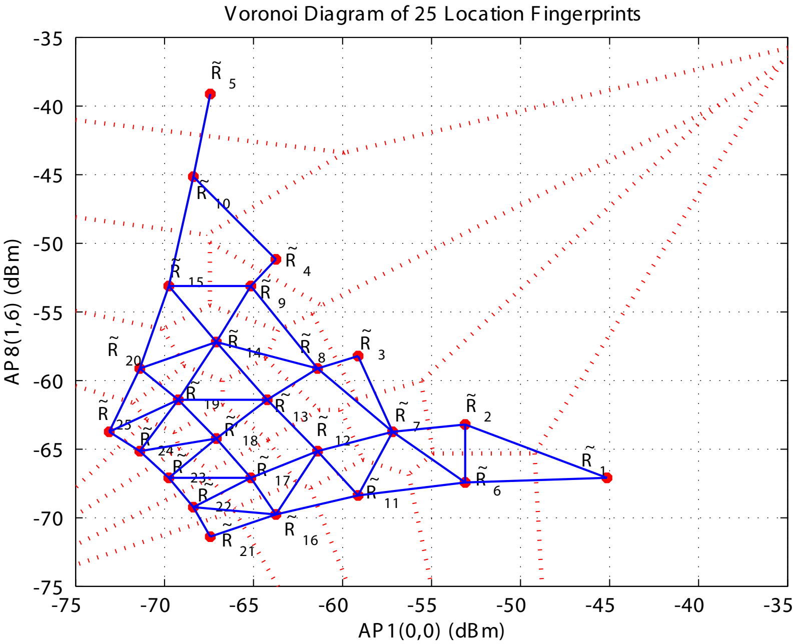

Figure 9: Voronoi Diagram and GG of 25 Location Fingerprints

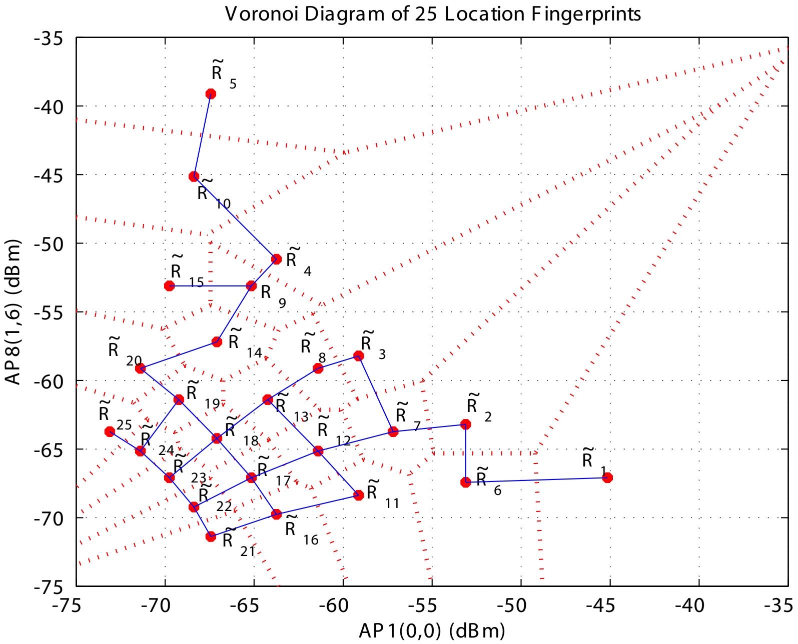

Figure 10: Voronoi Diagram and RNG of 25 Location Fingerprints

2. Results of Error Probability of Fingerprint Selection

We first study the precision at zero-meter accuracy in terms of the probability of error of fingerprint selection ($\mathcal{P}_e$ – the probability of not picking the correct fingerprint). In particular, the impact of the standard deviation σ of RSS on $\mathcal{P}_e$ is considered. We choose the grid point 13, in Figure 7, as the MS's actual location (it is at the center of our grid system). We consider the standard deviation σ between 1-7 dB which corresponds to values seen in extensive experiment results [1]. The results from both simulation and the analytical model are given in Figure 11.

In the case of the analytical results, we consider different approximations depending on the number of fingerprints involved in the probability estimation. First, we select all the other 24 fingerprints (*Ana:all-nb*) to compute the probability of error as given in (III.5). Second, we use only the neighbor set (*Ana:dg-nb, gg-nb, rng-nb*) derived from the different proximity graphs to estimate the probability value. Clearly, when the standard deviation

σ increases, the error probability also increases. The error probability estimation using Delaunay neighbors shows the closest results to the simulation. The same result is obtained by using Gabriel neighbors because both graphs yield the same neighbor set for the grid point 13. In *Ana:all-nb*, remote fingerprints (those that are far away from the correct fingerprint in signal space) are included in the comparison variables $[Pr\{C_k \leq 0\}]$ in (III.5). Since such probabilities are multiplied with the independence assumption, this decreases the probability of correct decision and increases the error probability thereby giving higher results compared to simulations. Note that the random variables C_k are not really independent. On the other hand, using the relative neighborhood graph [*Ana:rng-nb*] gives a lower probability of error compared to simulation since it underestimates the number of significant neighbor fingerprints used in the approximation.

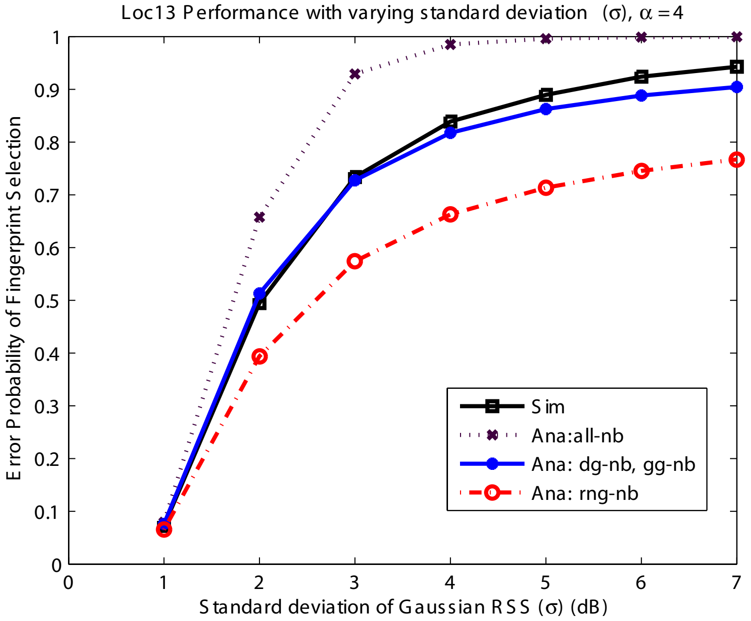

Figure 11: Impact of the standard deviation on error probability

Table 5 summarizes the comparison of the error probability for estimating location 13 when the number of access points in the system increases. We study different cases when we deploy 2 APs (AP1-AP8), 3 APs(AP1-AP8-AP12), 4 APs(AP1-AP4-AP8-AP12), and 5 APs(AP1-AP4-AP8-AP10-AP12) in the system. Here, we assume the same standard deviation of RSS ($\sigma = 4$ dB) from all APs. As expected, increasing the number of APs will reduce the error probability. We can see that *Ana:all-nb* provides a poor upper bound approximation compared to simulation results. The *Ana:dg-nb* and *Ana:rng-nb* provide better upper and lower bounds as the number of APs increases. The *Ana:gg-nb* is the closest to the simulation results. The neighbor set derived from GG reflects fingerprints that have a better chance to be picked outside of the correct fingerprint. We also did an extensive study by locating the MS at different locations with different combinations of the APs. It turned out that similar results were observed. From this, we conclude that using the neighbor set derived from a Gabriel Graph makes the most sense in estimating the probability of error.

Table 5: COMPARISON OF ERROR PROBABILITY, $\sigma = 4$

APs	**Sim**	**Anal: all-nb**	**Anal: dg-nb**	**Anal: gg-nb**	**Anal: rng-nb**
2	0.8334	0.9849	0.8176	0.8176	0.6631
3	0.7603	0.9409	0.8592	0.7891	0.6830
4	0.6337	0.8285	0.7770	0.6792	0.5759
5	0.5278	0.7111	0.6677	0.5815	0.4890

3. Results of Probability Distribution of Fingerprint Selection

Probability Distribution by Location

Next we evaluate our analytical model to see how well it approximates the probability distribution of picking specific locations given the MS is at one grid point, as discussed in subsection III.B.2. Figure 12 and 13 show examples for the comparison of the distributions between the analytical model using GG and simulations, when the MS is located at location 4 and 9 respectively. Here we assume that $\sigma = 4$ as before. Note that the results from the analytical model are found to be close to the simulation results in both cases. We also found that using DG (Figures 14-15) and RNG (Figures 16-17) give relatively close results to the simulation with only small differences in shape and height of the histograms, but the GG is the best. We considered several different MS's locations – in each case the analytical model gives distributions that are close to those from simulation.

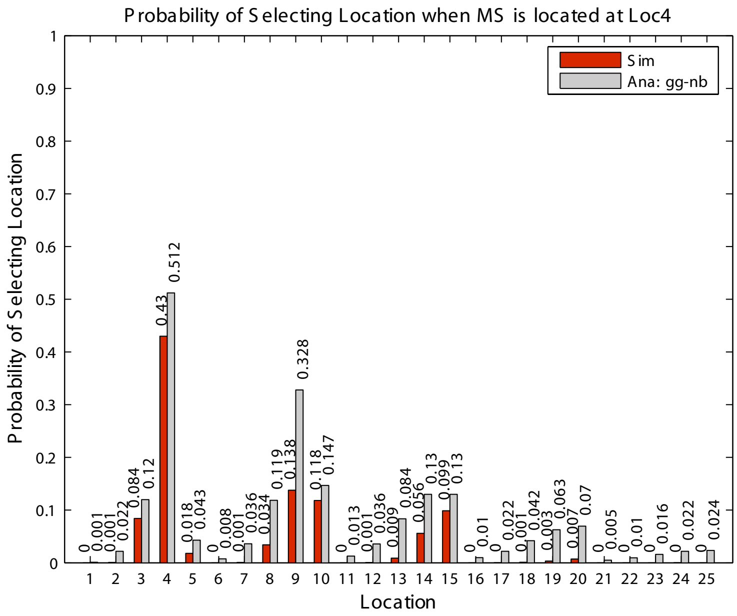

Figure 12: Prob. Distribution of Fingerprint Selection (MS is at Loc4)-GG

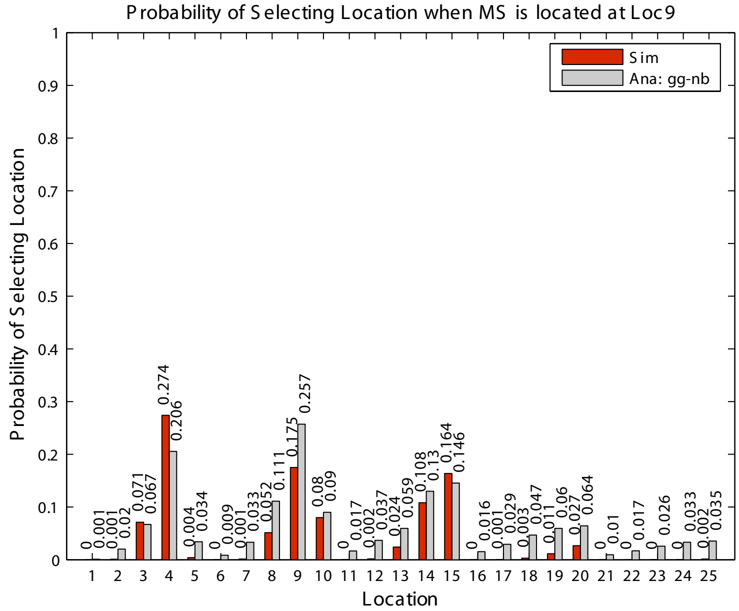

Figure 13: Prob. Distribution of Fingerprint Selection (MS is at Loc9)-GG

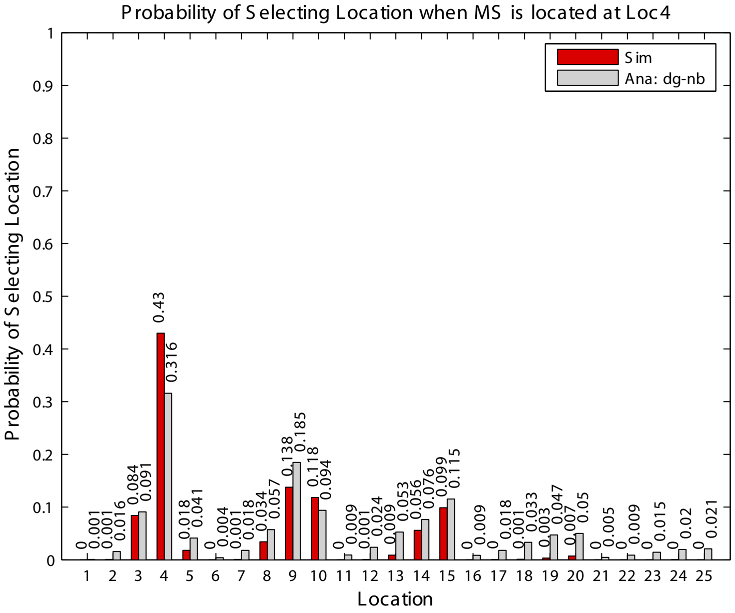

Figure 14: Prob. Distribution of Fingerprint Selection (MS is at Loc4)-DG

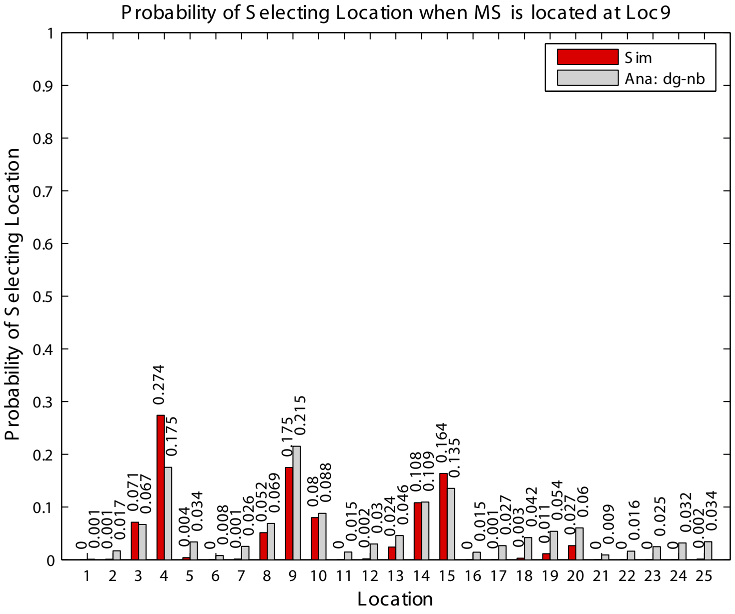

Figure 15: Prob. Distribution of Fingerprint Selection (MS is at Loc9)-DG

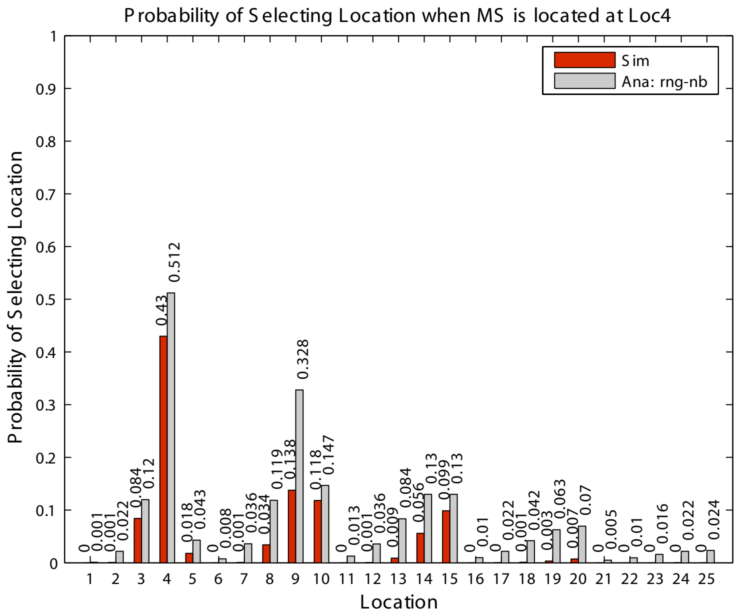

Figure 16: Prob. Distribution of Fingerprint Selection (MS is at Loc4)-RNG

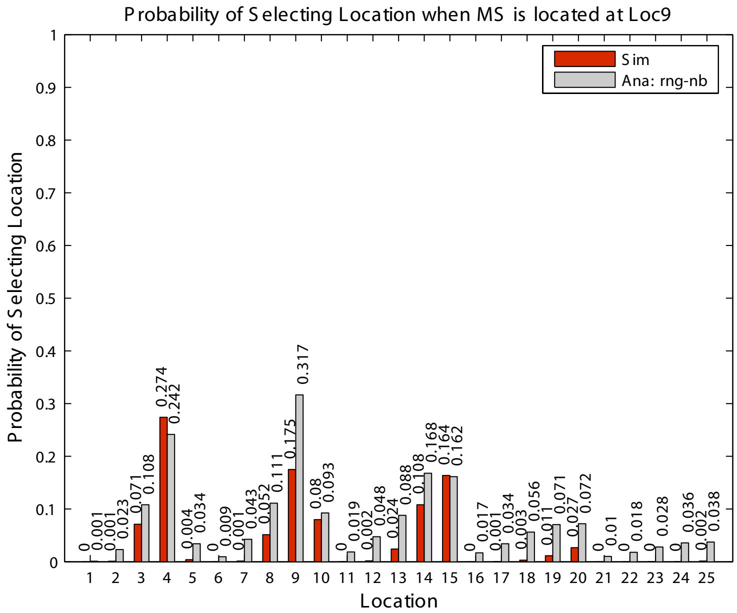

Figure 17: Prob. Distribution of Fingerprint Selection (MS is at Loc9)-RNG

As it provides reasonable estimates for the probability distribution, the analytical model can now be used to find some fingerprints that, if retained in a radio map, can degrade the overall performance of the location estimation. Those fingerprints also cause inefficient nearest neighbor computation during the online phase with a complexity of $O(n * D)$, where n is the cardinality or number of the location fingerprints and D is dimensionality (number of RSSs from the APs) in the fingerprint system.

To give an example, let consider all the analytical results in Figure 12. We can see that, when the MS is at a grid location 4, the probability of selecting the grid location 4 (0.512) is a lot higher than the next highest probability of selecting a different grid location (location 9 with probability 0.328). However, when the MS is located at the grid location 9 in Figure 13, the two highest probability values are almost equal (0.206 for location 4 and 0.257 for location 9). This means that retaining the fingerprint at the grid location 9 in the radio map results in the system having a relatively high chance of falsely returning the grid location 4 as a correct location of the MS. In fact, the simulation results in the figure report that the system picks the grid location 4 as the correct location more often than the actual grid location 9 (0.274 over 0.175). Therefore, this suggests that we should not include location 9's fingerprint ($\tilde{R}_9$), and it should be eliminated from the database. If possible, such locations must be avoided in the laborious offline phase as well.

Note that using the Gabriel and Relative Neighborhood graphs also provides the same results. However, with the complexity given in Table 3, Using the Gabriel and Relative Neighborhood graphs is faster to compute than using the Delaunay graph (especially when both n and D are large). Therefore, in general we can describe the procedure for fingerprint elimination as follows:

Fingerprint Elimination Procedure:

```
1) Compute all prob. distributions of all locations
2) For each L1's distribution:
        - If $prob\{picking\ L1\}$ is not the highest, eliminate L1
        - If there exists L2 where $|prob\{picking\ L1\} - prob\{picking\ L2\}| < threshold$,
          - Check L2's distribution. Eliminate L1 if $prob\{picking\ L2\}$ is the highest &
          it differs from $prob\{picking\ L1\} > threshold$
```

We can repeat this procedure to the probability distributions when a MS is located at different grid locations and determine whether there are incorrect grid locations that have a comparable probability of being selected to the probability of selecting a correct grid location. Threshold value of 0.2 is chosen in this work based on trial and error.

Fingerprint elimination procedures are effective and make sense when the positioning system is deployed in a large area such as entire floor of a building where hundreds of grid locations are present. Fewer numbers of fingerprints in the database mean smaller numbers of comparisons needed during the online phase. From an extensive (analytical) evaluation, in our system of 25 fingerprints, we can identify four more fingerprints ($\tilde{R}_{17}$,$\tilde{R}_{19}$,$\tilde{R}_{22}$,$\tilde{R}_{24}$) that should be eliminated. By doing so, we can save about $5/25 = 20\%$ of computation required for nearest neighbor search during the online phase.

Probability Distribution by Meters

The probability distribution of fingerprint selection can be represented in terms of physical distance error as well (which is necessary for determining precision). Figures 18-20 give examples of the distribution plots of error distance in meters, using different proximity graphs, when the MS is at grid location 9. As expected, the probability decreases when error distance increases. Again, the distributions from simulation and analysis, though not perfectly precise, are close. Note that the probabilities from analysis do not add to one, since they are determined using an approximation that assumes independence between comparisons of random variables.

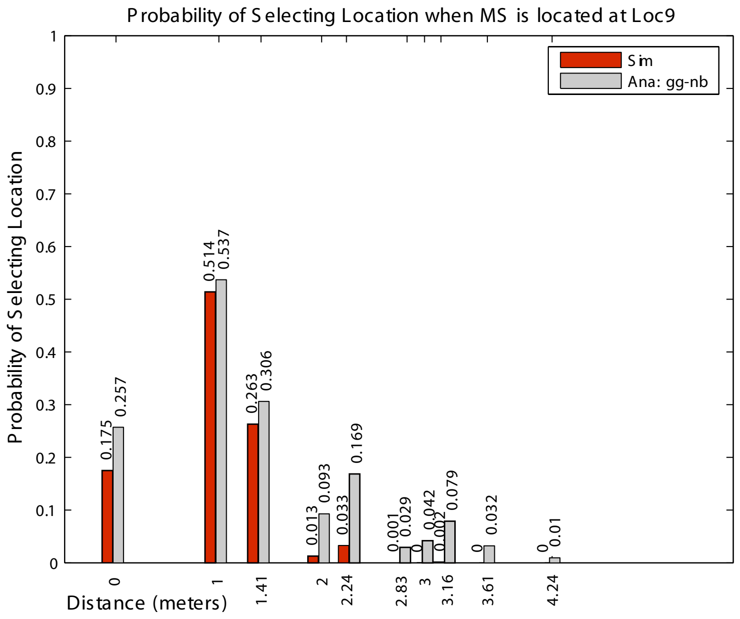

Figure 18: Prob. Distribution (by meters) of Fingerprint Selection-GG

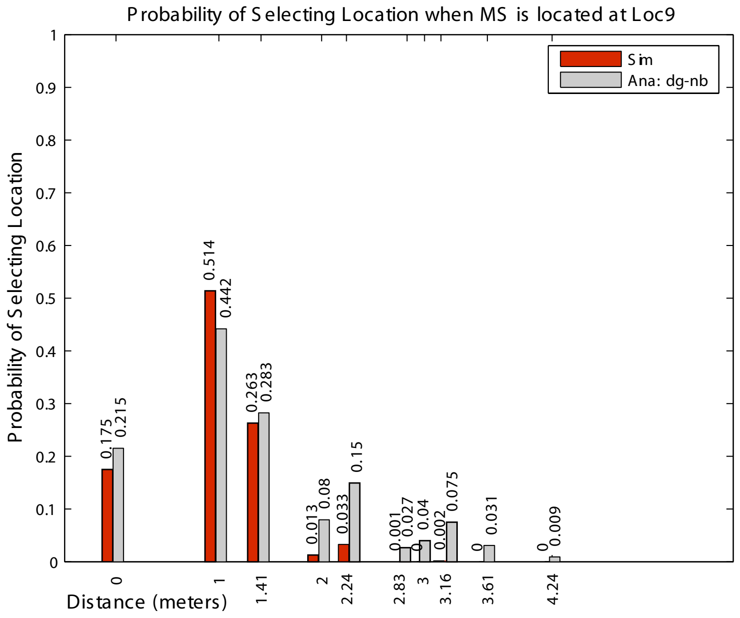

Figure 19: Prob. Distribution (by meters) of Fingerprint Selection-DG

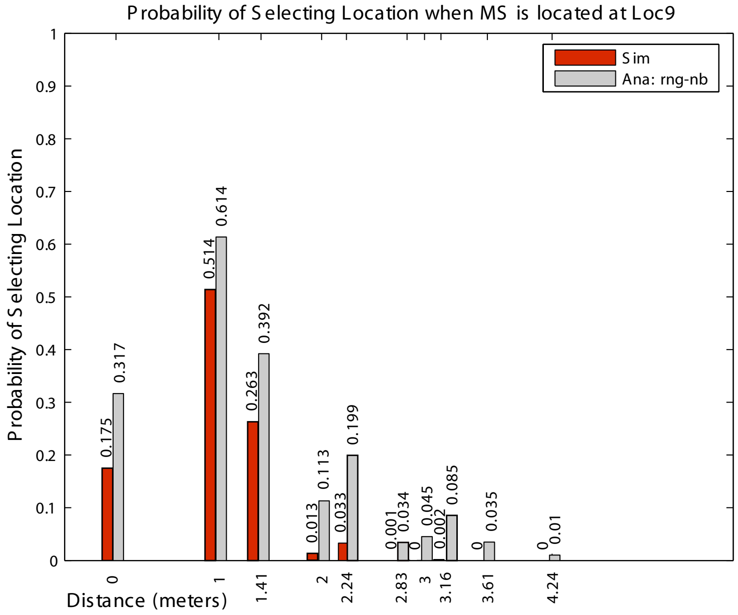

Figure 20: Prob. Distribution (by meters) of Fingerprint Selection-RNG

Cumulative Probability Distribution

A question that arises is how eliminating certain fingerprints impacts performance. In Figure 21, the cumulative probability distribution of the error distance in meters, with and without elimination of fingerprints from the radio map, are shown. The results are averaged based on simulations from all 25 locations. Again, we keep $\sigma = 4$. We can see that, after eliminating some fingerprints ($\tilde{R}_9$, $\tilde{R}_{17}$, $\tilde{R}_{19}$, $\tilde{R}_{22}$, $\tilde{R}_{24}$), the cumulative distribution is only slightly different compared to the case where all fingerprints are kept in the system. The difference in the distributions occurs only in the first few meters of error and then diminishes as the error distance increases. Note that, there is about a 70 % chance that the system still maintains the distance error within 1 m after fingerprint elimination and it is close to the case when all fingerprints are used. Therefore, by applying fingerprint elimination, we do not lose much in terms of precision (probability), and yet maintain acceptable accuracy (error distance).

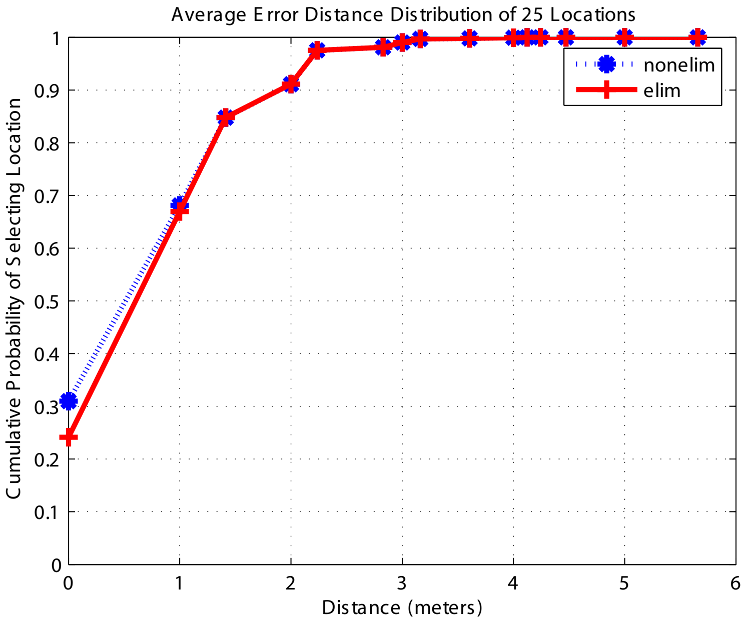

Figure 21: Average Cumulative Distribution of Error Distance

4. Results with Fingerprint Measurement

We apply the analytical model and fingerprint elimination technique as discussed in subsection III.C.3 to a radio map from past measurements [1]. Unlike the simple model used for evaluation, the measured fingerprints have different σ's for different APs and locations. Therefore, instead of using (III.2), we approximate the new PEP by (the derivation is given in Appendix A):

$$PEP(\tilde{R}_i, \tilde{R}_k) \quad = \quad Q(\frac{sd_{ik}^2}{2[\sum_{j=1}^{N} \beta_{ij}^2 \sigma_{ij}^2]^{1/2}}). \qquad \text{(III.9)}$$

The above equation is a modification – in the form of a PEP – of the derivation based on a sum of multiple Gaussian variables in [9] with different σ_{ij}. $\beta_{ij} = \rho_{ij} - \rho_{kj}$ (difference of $\tilde{R}_i$ and $\tilde{R}_k$ at the j^{th} AP of fingerprint) and σ_{ij} is the standard deviation of the RSS from the j^{th} AP at the i^{th} location. The new PEP is then used in (III.6) to approximate the probability distribution.

The measurement was conducted in an office environment on the 4th floor of the Information Sciences (IS) building as shown in Figure 22. The measurement setup consists of a grid area of 25 locations where 20 locations are inside room 410 and 5 locations are along the corridor. The grid spacing is approximately 1 meter. The figure also shows locations of nearby APs that can be detected at grid locations. The same location labels as in the simple model are used.

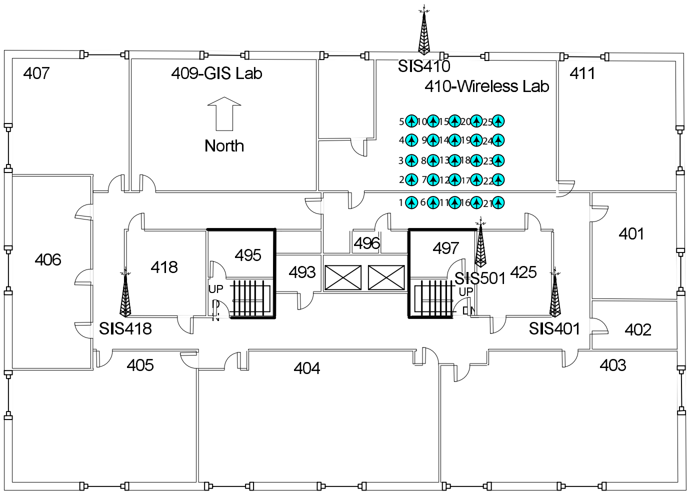

Figure 22: Measurement Setup at the fourth floor IS building

Fingerprints are based on the 2 APs named SIS410 and SIS501. The fingerprints are shown in Figure 23 with their Voronoi regions and GG. By applying the analytical model with the Gabriel Graph, the probability distributions of picking a locations given the MS's location are approximated. Pairwise comparisons are used for fingerprint elimination. It turned out that we can identify 9 fingerprints that can be eliminated. Figure 24 shows the average CDF of the error distance in an experiment. Similarly, we can see minimal difference of average CDFs when we eliminate fingerprints. This result indicates that we can sacrifice little performance but save $9/25 = 36\%$ by reducing the search space in the online phase. Note that this figure shows the *average CDF* over all locations. The error CDFs for individual locations are different (depending on the number of neighbors and distances to neighbors). However, individual plots of the difference between the CDFs from before and after fingerprint elimination provides no significant difference (see results in Appendix B).

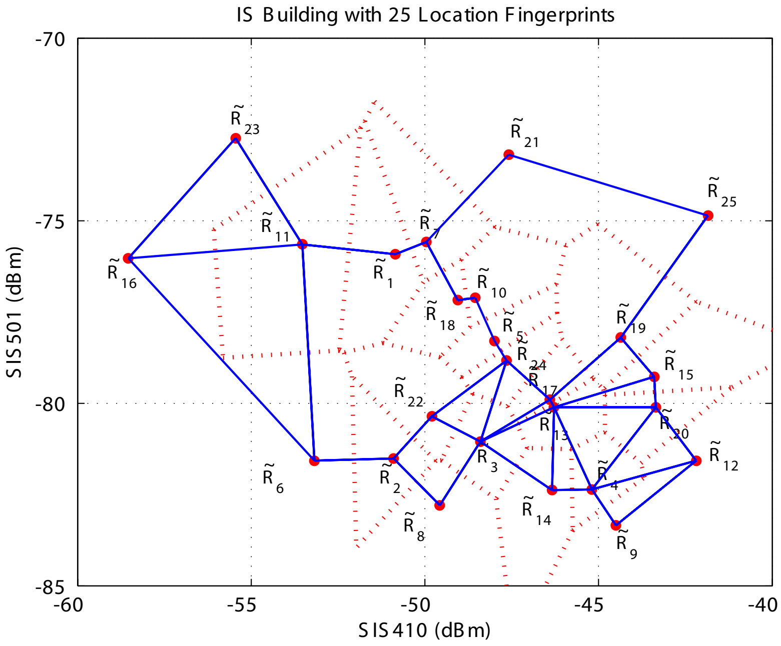

Figure 23: Voronoi Diagram and GG of Measured Radio Map

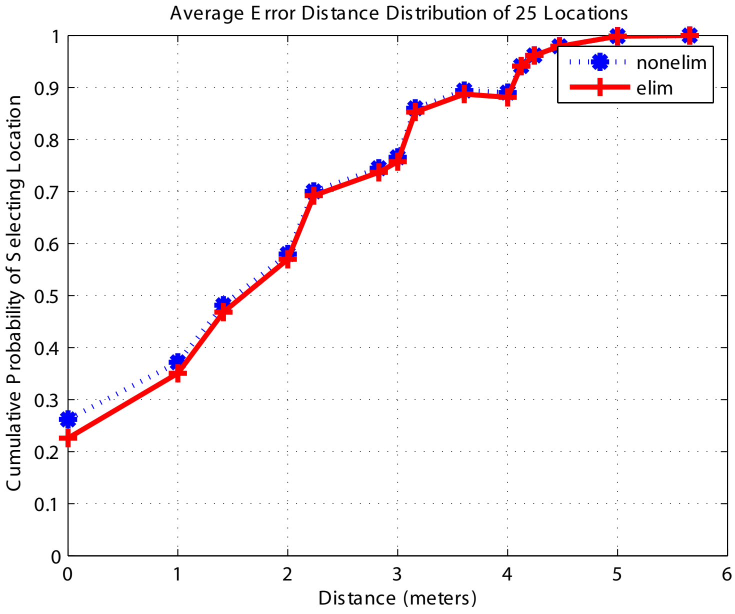

Figure 24: Average Cumulative Distribution of Error Distance from Measurement

D. ANALYTICAL MODELING STUDY SUMMARY WITH SIMPLE GRID SYSTEMS

In this section, we compile the accomplished work and what we have learned from the analytical study as follows.

Proximity Structure of Location Fingerprint

We first studied location fingerprints by assuming a simple and uniform grid system. An exponential path loss propagation model between an access point and a mobile station is generally used to predict a signal power received at each location. The first finding is that the location fingerprints from the analytical model are not uniformly distributed over the signal space as shown in Figure 6. Signal points for each fingerprint are affected by physical distances to access points that can been seen by each location. Farther grid locations tend to produce fingerprints that are close together, while near grid locations tend to produce fingerprints that are more separated (this may not be true in a real setting where more obstacles between an AP and a farther location may actually separate the fingerprints). The next finding is that fingerprints have different decision regions (or Voronoi region) with significant differences in shape and size. In fact, a border location in the grid has an unbounded decision region (theoretically) and an inner location carries a bounded decision region. The number of Voronoi edges for a fingerprint decision region is determined from not only the number of surrounding fingerprints but also from their signal distance to the respective fingerprint.

We have analyzed the constellation or structure of location fingerprints using different proximity graphs in order to extract inherent proximity information of the fingerprint system. We implemented algorithms using MATLAB for computing all three proximity graphs in this work. As shown in subsection III.B.1, we found that different graphs can produce different numbers of edges between fingerprints or numbers of fingerprints in the neighbor set. The Delaunay graph is the one with the most numbers of edges and with the highest complexity of computation. The Gabriel and Relative Neighborhood graphs respectively produce lesser numbers of edges and neighbors with asymptotically lower complexity. In our study, the neighbor set derived from the Gabriel graph gives better picks of the fingerprints having a better chance to be picked outside the correct fingerprint. Results are shown in Figures 8-10. The analysis of fingerprint structure using proximity graphs permits us to understand the nature of the location fingerprints and their interrelations in order to derive performance prediction for a given fingerprint system.

Performance Modeling with Proximity Information

Mathematical modeling for system performance evaluation has been studied in subsection III.A.2. The model with proximity information from proximity graph enables us to approximate both precision and accuracy performance metrics for a location system as results show in subsections III.C.2 and III.C.3. When considering the precision metric represented as the error probability of fingerprint selection with varying degrees of signal deviation, a simple approximation using the product of probabilities from all fingerprints gives the worst performance. We found that it is better approximated when we considered only neighbors derived from the proximity graphs into calculation. In addition, among the three proximity graphs, the error probability model computed with the Gabriel graph yields the tightest upper bound to the simulation result at a typical signal deviation of 4 dB (something reasonable in RSS measurements [20]). The Relative Neighborhood graph, on the other hand, can provide a lower bound for the error probability. As we increased a number of access points, the analytical model with the Gabriel graph becomes even better and closer to simulation. This is a result from a better approximation of the neighbor set as discussed earlier.

As we consider probability distribution of fingerprint selection, relevant neighbors extracted from the proximity graph can help us derive an approximate histogram for probability of fingerprint selection. We have modeled the distribution both for different locations and for different error distances. Example results are given in Figures 12-20. The distribution from the Delaunay graph is found to be close to the simulation results. However, with a large number of fingerprints and access points involved, to compute the distribution using the Delaunay graph can be burdensome. The other two proximity graphs, Gabriel and Relative Neighborhood, can provide a decent approximation with lesser computational effort. By knowing the probability distribution of fingerprint selection, we can identify the influence level of each incorrect fingerprint on the correct one. Finally, we can determine which fingerprint, if kept in the radio map, could have a small chance to be picked and while it could worsen overall precision performance of the system.

Fingerprint Elimination Technique

We utilized the model of the probability distribution and proposed a fingerprint elimination technique so we can keep only fingerprints with good precision performance. A procedure for fingerprint elimination is described in subsection III.C.3. We found that we can identify and eliminate some "inefficient" fingerprints. From the preliminary study, we see that it is possible to potentially save 20-36 percent of computation in terms of nearest neighbor searching during the online phase.

We have studied the cumulative probability distribution of the error distance before and after elimination of fingerprints from the radio map. With a simple location system, we found that there is about a 70 percent chance that the system maintains distance error within 1 meter after fingerprint elimination. This is good and more importantly close to the case when we use all fingerprints. In other words, we do not lose on precision and still obtain tolerable accuracy. Therefore, we believe that our approach is beneficial to construct an efficient radio map for location fingerprint systems.

In the next chapter, we will discuss the sensitivity analysis study of the analytical modeling to precision and accuracy metrics with varying positioning parameters that describe both grid system and wireless channel.

IV. SENSITIVITY ANALYSIS OF THE SYSTEM MODEL

In this chapter, we provide a sensitivity analysis of the performance modeling in Chapter III. The performance of the positioning system depends on many factors such as grid system properties (i.e., grid spacing and number of access points) and wireless signal properties (i.e, variation of RSS and indoor path loss exponent). Results from the sensitivity study can also help determine recommended values for positioning parameters.

A. SENSITIVITY OF MODELING WITH GRID SYSTEM PROPERTIES

We perform a sensitivity analysis study of the analytical modeling with varying grid system properties. The factors considered are number of APs (i.e. dimensions of a fingerprint vector) and grid spacing in the positioning system.

First, a 25 grid system like one in Figure 7 is studied. We then vary grid spacing (using a step size of 0.25 m) and number of APs in the system. The correct probability of fingerprint selection (precision at zero error distance) is observed assuming a MS is at a grid location 13. The probability is computed based on the first line in (III.5) by using only the neighbor set of the correct fingerprint defined by the Gabriel graph in the computation. The result is shown in Figure 25. Here we use $\sigma = 4$dB and $\alpha = 4$. We can see that increasing both number of APs and grid spacing can improve the correct probability of fingerprint selection. However, when the number of APs is already high (6 APs for instance), an additional AP does not significantly improve the probability. Also, we found that small grid spacing worsens the probability of fingerprint selection greatly.

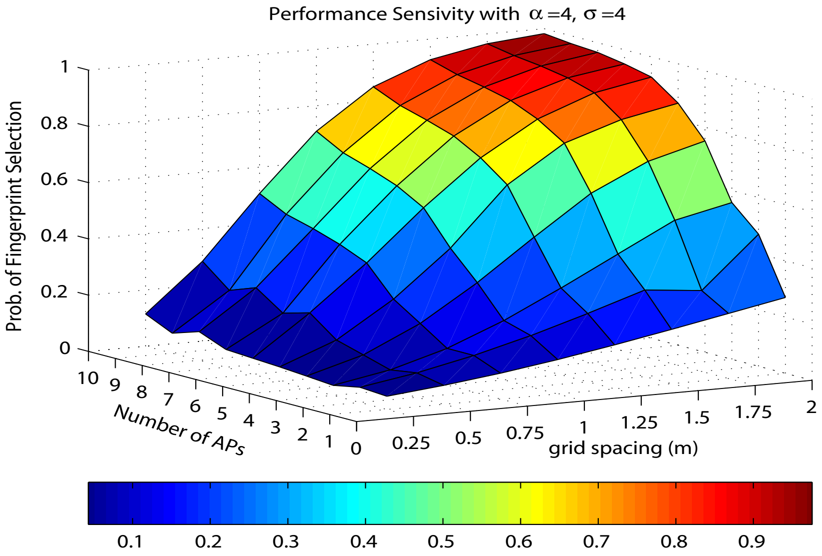

Figure 25: Performance Sensitivity Between No. of APs and Grid Spacing

We perform further analysis where the correct probability of fingerprint selection is plotted against grid spacing, variation of the RSS and path loss exponent, as shown in Figure 26. In the left-hand plot of the figure, we can see that increasing grid spacing does not improve the correct probability if variation of the RSS is high (6 dB and higher). In the right-hand plot of the figure, however, we found that increasing grid spacing will improve the correct probability, especially with higher path loss exponent ($\alpha = 4$ or higher). This supports the intuition that we have – if the fingerprints are very different, there is a smaller chance of making errors between them. Fingerprints become different as the path loss exponent increases and become similar when the RSS variation increases or the grid spacing reduces.

We study the effect of varying grid system properties on the average cumulative probability distribution of the error distance (precision at higher error distances). We model the probability distribution of fingerprint selection using (III.6) and apply the fingerprint elimination procedure described in subsection III.C.3. Comparison results are shown in Figure 27. In the left-hand plot of the figure, we found that increasing the number of APs from 2 to

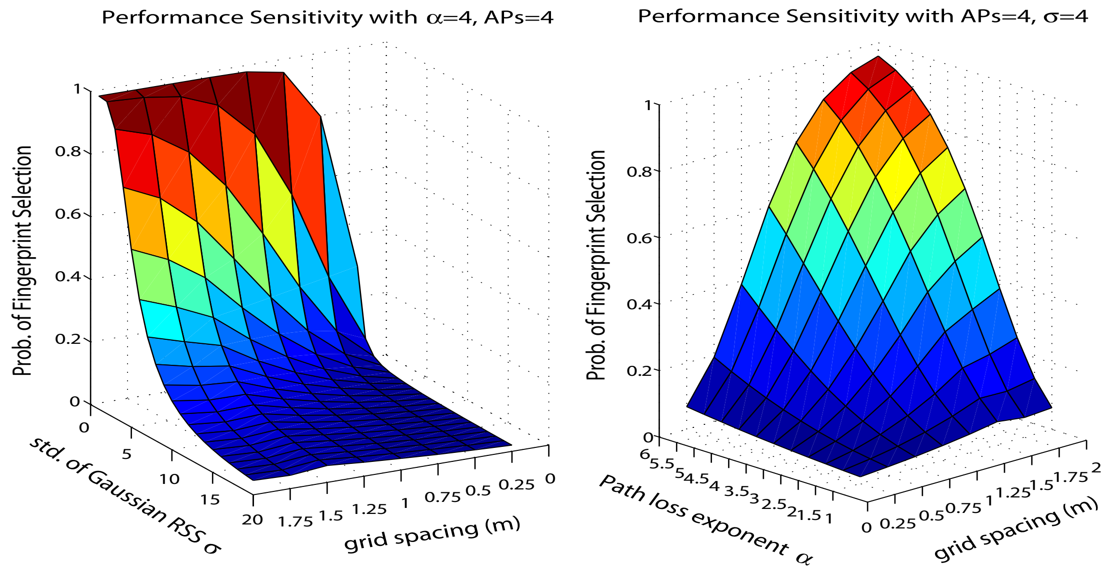

Figure 26: Performance Sensitivity: Grid Spacing, STD.of RSS, and Path Loss Exponent

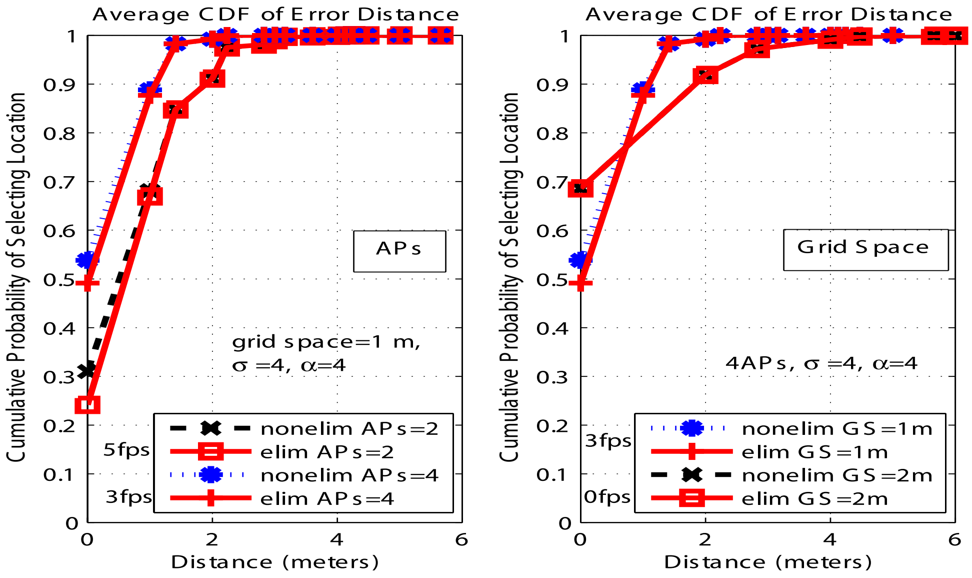

Figure 27: Cumulative Performance Sensitivity: No. of APS and Grid Spacing

4 causes a higher cumulative probability at the same error distance. This is due to the fact that additional APs add dimensions to the fingerprint vector, thereby increasing the ability to distinguish among different locations (fingerprints) within a given error distance. A total of 5 and 3 inefficient fingerprints are eliminated from systems with APs = 2 and APs= 4 respectively. It means that the more the number of APs, the better prototype a fingerprint becomes and this is reflected by a smaller number of inefficient fingerprints.

In the right-hand plot of the figure, we can see a cross over between the cumulative probability graphs when using a grid spacing of 1 and 2 meters. Initially, although a 2-meter grid spacing has a higher probability at zero error distance (i.e., 0.7 compared to 0.5), the probabilities with a 1-meter grid spacing accumulate faster and become higher at an error distance of 0.8 meters and higher. There are 3 inefficient fingerprints with a 1-meter grid spacing while there are none in a 2-meter grid spacing. The result shows that a sparser grid spacing causes higher probability of selecting a location or contains many already “unique” fingerprints, but it can increase the error distance (or less accuracy) in location estimation.

B. SENSITIVITY OF MODELING WITH WIRELESS SIGNAL PROPERTIES

Next, we check the sensitivity of the modeling with varying wireless signal (or RSS) properties including variation of RSS σ and path loss exponents α. The result is shown in Figure 28. Here we use grid spacing = 1 meter and number of APs = 4. We can see that increasing α can improve the probability of fingerprint selection when the variation of RSS is small (i.e. σ= 4dB or lower). With large RSS variation, higher α can improve the probability (which is already low) just by a little. Note that these two RSS properties are those that cannot be controlled, but ones that must be considered during the design of fingerprinting based positioning systems.

We perform further analysis where the correct probability of fingerprint selection is plotted against the number of APs, variation of the RSS, and path loss exponent, as shown in Figure 29. In the left-hand plot of the figure, we can also see that increasing the number of

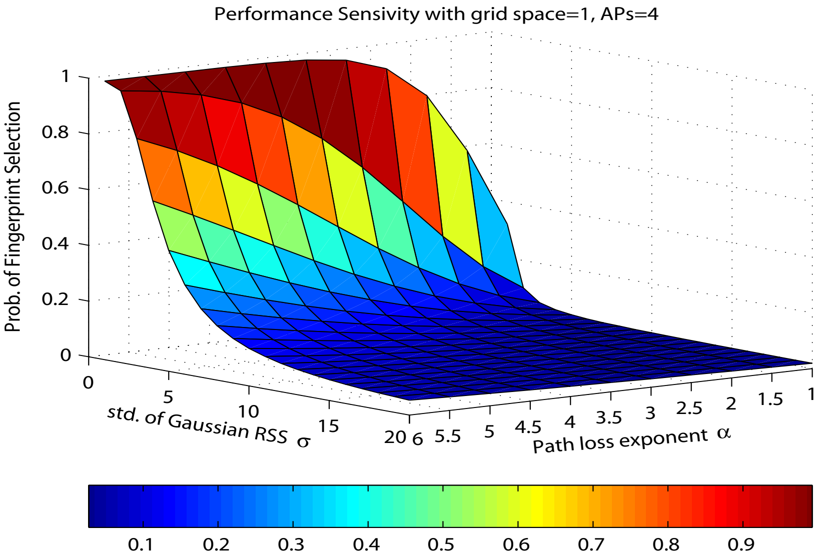

Figure 28: Performance Sensitivity Between STD.of RSS and Path Loss Exponent

APs does not improve the correct probability if the variation of the RSS is high (6 dB and higher). In the right-hand plot of the figure, however, we found that increasing the number of APs will improve the correct probability, especially with a high path loss exponent. ($\alpha =$ 4 or higher).

When we vary both RSS properties (σ and α), we observe the average cumulative probability distribution of the error distance as shown in Figure 30. In the left-hand plot of the figure, we found that increasing σ from 4 dB to 8 dB causes a lower cumulative probability at the same error distance. This is because a high variation of the RSS reduces the ability to distinguish among different locations (fingerprints) within a given error distance. A high variation of the RSS can cause higher number of inefficient fingerprints that should be eliminated (9 fingerprints for $\sigma = 8$dB and 3 fingerprints for $\sigma = 4$ dB). While in the right-hand plot of the figure, we can see that increasing α from 4 to 6 causes a higher cumulative probability at the same error distance. This is due the the fact that a high path loss exponent increases the Euclidean distance among fingerprints at different locations and increases the

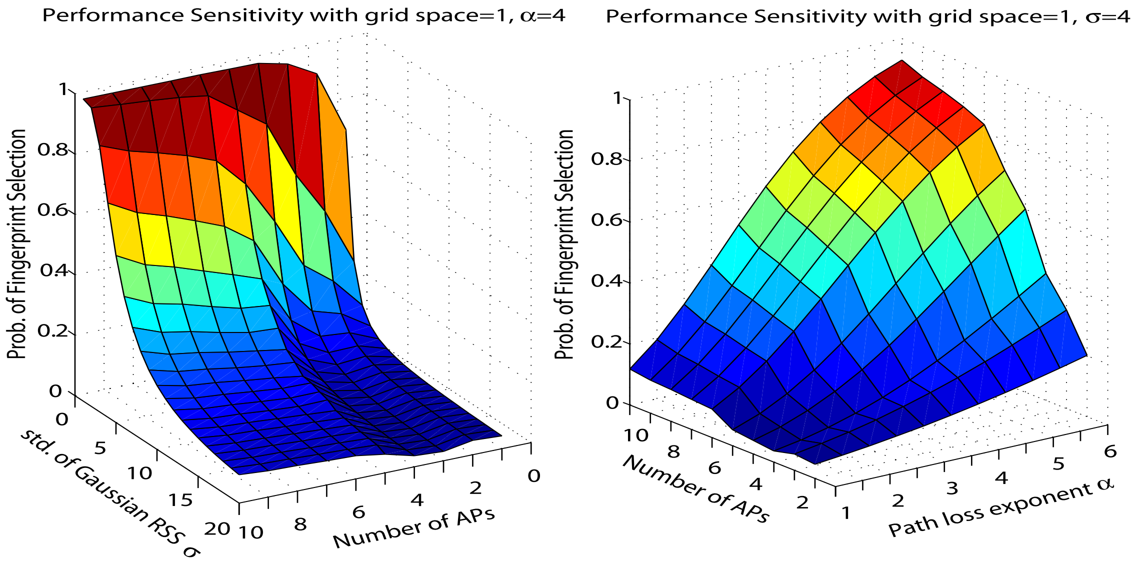

Figure 29: Performance Sensitivity: No. of APs, STD.of RSS, and Path Loss Exponent

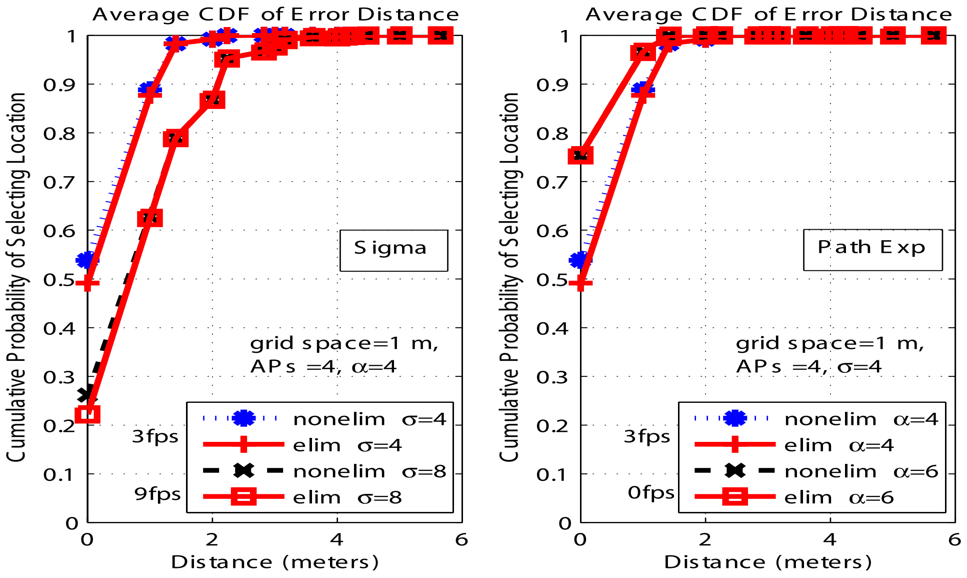

Figure 30: Cumulative Performance Sensitivity: STD. of RSS and Path Loss Exponent

ability to distinguish among different locations (fingerprints) within a given error distance. High path loss exponent causes many unique fingerprints and fewer numbers of inefficient fingerprints (3 inefficient fingerprints are found when $\alpha = 4$ while none are found when $\alpha = 6$).

C. RECOMMENDED VALUES FOR POSITIONING PARAMETERS

Based on the sensitivity study of the simplified system, we can suggest methods for basic improvements in positioning performance. We summarize the effect of system parameters on the precision performance and give an example of values of parameters to improve the precision metric in indoor positioning systems. To select a suitable value of a parameter, one needs to consider both the zero error distance precision (Figures 25-26 and Figures 28-29) and a higher error distance precision (Figures 27 and 30). The performance guideline is summarized in Table 6 given a requirement of more than 85% precision at 1 meter of error distance (or 1 meter accuracy).

Table 6: Recommended values for location system parameters

Parameters	**Value Increased**	**Desired Range**
σ-STD. of Gaussian RSS	precision decreases and accuracy decrease	$\sigma < 4dB$
N-No. of APs	precision increases and accuracy increase	$N \geq 4$
α-Path loss exponent	precision increases and accuracy increase	$\alpha > 4$
g-grid spacing	precision increases and accuracy decrease	$g > 1$ meter

In general, most existing WLAN infrastructures are deployed to satisfy only certain criteria such as coverage and bandwidth for users. To enhance existing infrastructure for an overlay positioning system and better performance, additional changes are desired to improve the precision and accuracy. For example, from Figure 27, one additional AP (resulting in total of 5 APs) could be used to improve the precision at zero error distance (the highest accuracy) since the correct probability at a zero error distance is only 0.5 with 4 APs. The guideline provided in Table 6 may not be applicable to every indoor positioning system. More details about the environment and accurate path loss models with wall and floor attenuation could result in a more realistic recommendation for the deployment process.

V. SCALABILITY OF ANALYTICAL MODELING VIA CLUSTERING

When location fingerprints from a large number of grid points are collected during actual deployment, computational requirement for the analytical model can be high, especially for computing the proximity graph. We have seen from our previous study that fingerprint scattering is typically asymmetric and some may be even clustered together (see Chapter III). The probability of selecting a fingerprint not within the same cluster could be low. Those remote fingerprints can be ignored during proximity analysis as they are hardly picked as a neighbor while constructing the proximity graph. From this observation, therefore, in this chapter we look at fingerprint structure analysis by applying a *fingerprint clustering*. The fingerprint clustering is applied in order to reduce effort for computation of one large proximity graph, which requires high polynomial time complexity (will be seen later in Table 3). Dividing fingerprints into smaller clusters is useful to improve scalability of the analytical model computation. The analytical model can then be applied based on derived proximity graphs of smaller sizes.

Dividing fingerprints into small clusters can also reduce number of comparison pairs (between a sample fingerprint and location fingerprints in the radio map) performed during the online phase by considering only those fingerprints in a particular cluster. However, computing a proximity graph using separate clusters may cause a loss in precision performance. Therefore, a tradeoff between precision and computation of the fingerprint clustering is studied. Ideally, we want fingerprint clusters to be as evenly distributed as possible (i.e., the number of fingerprints or the online comparison pairs are equal for each cluster) yet reflect actual signal distance relationship among them. Also, a clustering method should be simple and an iterative method may be acceptable. For initial study, we only study the clustering of fingerprints into two clusters. However, the clustering can later be used for case with many clusters.

In the next section we will describe two fingerprint clustering methods used in our study to improve the scalability of the analytical model. They are Median clustering and K-Mean clustering methods.

A. FINGERPRINT CLUSTERING METHODS

1. Median Clustering Method

The Median clustering method, as the name implies, divides all fingerprints in a radio map into two clusters using the median of the fingerprint's RSS values. The method produces one cluster with a half of all fingerprints having its RSS values below the median and another cluster with the remaining half. The median RSS is derived from one dimension (i.e., one AP) of location fingerprint vectors. Such a dimension (or an AP) is selected to have the largest RSS range (the difference between the maximum and a minimum of the RSS in the radio map from that AP). Intuitively, the "largest range" AP provides a large *signal distance* on average. Thus it provides a better ability to distinguish among fingerprints. If needed, the median clustering can be re-applied within clusters to further divide fingerprints (i.e., using the AP with the second largest RSS range within the former two clusters to produce four clusters). This method is relatively simple.

2. K-Mean Clustering Method

The K-Mean clustering method is a well-known pattern recognition clustering method [21]. Here, natural centers (known as *centroids*) of fingerprint clusters are determined. Each fingerprint is assigned to the cluster having the nearest centroid. K indicates the total number of clusters. The objective is to minimize total intra-cluster variance, which is the sum of squares of distances between fingerprints and the corresponding cluster centroids. The Euclidean distance is commonly used to measure proximity.

A simple algorithm for K-Mean clustering can be described by these three steps: (for our study, we use $K = 2$)

```
1) Initialize K cluster centers ($\mu_i$ for i = 1,..., K) by randomly
   picking K fingerprints
2) (2a) For all fingerprints, assign each fingerprint $\tilde{R}$ to the
   nearest centers and corresponding clusters
   (2b) Recompute new mean centers;
        $\mu_i = (1/K_c) \times \sum_{\tilde{R}_c \in i^{th}\ cluster} \tilde{R}_c,$
for all i = 1,..., K and $K_c$ = number of fingerprints in a cluster
3) Repeat step 2 until clustering converges.
```

Here $\tilde{R}_c$ represents a current fingerprint member of a cluster. Note that a derived center is a mean vector of fingerprints within the same cluster. Also, centers from all clusters are needed to determine cluster membership for a sample RSS vector during the online phase. The K-Mean clustering method is iterative and it can be used even if many clusters (for $K>2$) are desired. However, this is a tradeoff between additional clustering computation and improvement in predicting precision performance. We evaluate results from clustering methods in the following section.

B. PERFORMANCE EVALUATION OF CLUSTERING METHODS

1. Clustering Experiment Setup

To evaluate clustering methods, we consider two grid systems from the measurement. The first system (*Scenario 1*) uses a square grid system of 25 locations as previously shown in Figure 22. The second system (*Scenario 2*) is a larger grid system where fingerprints were collected from 71 grid points at the Hillman Library building as shown in Figure 31. In the second system, all RSS measurements were done inside the area on the 1st floor, where there is a large open space that shares the ceiling with the 2nd floor. Measurement grid points, as labeled by number 1 to 71, are shown by small arrows in the figure. 6 APs, as being placed on different floors, are shown and overlaid in this figure. 2 APs are placed each on the ground floor, the 2nd floor, and the 4th floor. Number (or alphabet) after “hl” indicates floor in

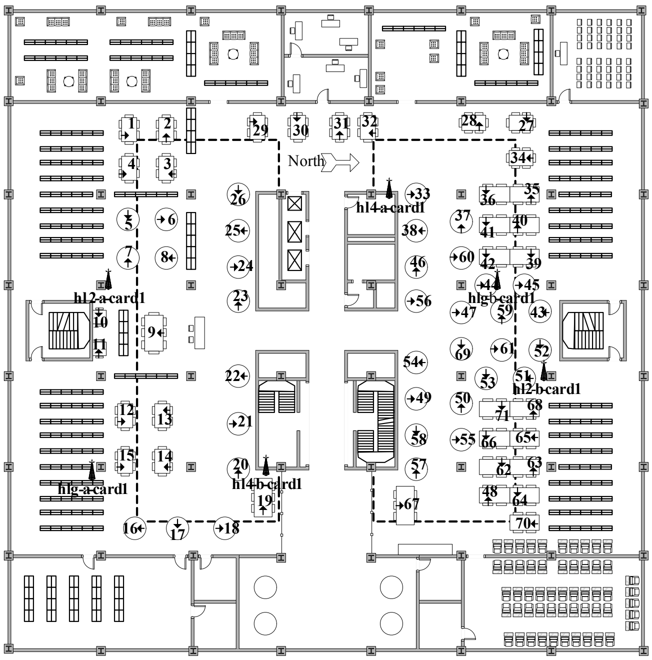

Figure 31: Measurement Setup: Scenario 2 at the Hillman Library building

the building where an AP is located (i.e. 2 = 2nd floor, g = ground floor). Although signals from all 6 APs can be detected on the 1st floor, their coverage is not complete throughout the floor. Grid spacing between locations is non-uniform (> 2 m). This is because grid points were picked according to locations of reading tables inside the library. More details about this measurement can be found in [1].

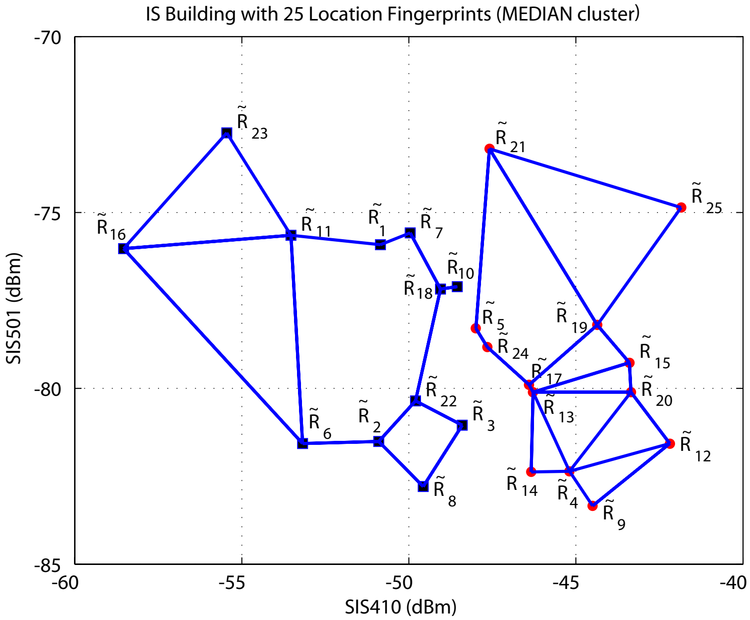

Figure 32: Median Clustering and GGs: Scenario 1

Next we apply the Median clustering and the K-Mean clustering methods to both scenarios. Figure 32 shows the Median clustering and the two GGs (one graph for each cluster) in the Scenario 1. There are two clusters; one cluster with 12 fingerprints and the other with 13 fingerprints. Markers with dots (or squares) indicate fingerprints within the same cluster. Each fingerprint is based on 2 APs (SIS410 and SIS501) as shown in Figure 22, represented as a RSS vector denoted by [SIS410, SIS501]. SIS410 is selected for the median for clustering. RSS values from SIS410 in the fingerprints provide the highest range. The RSS median value is -47.99 dBm. Note that each GG will be used separately in computing the probability distributions. Later the fingerprints needed to be eliminated within each cluster are determined. The K-Mean clustering and the GGs for the Scenario 1 are shown

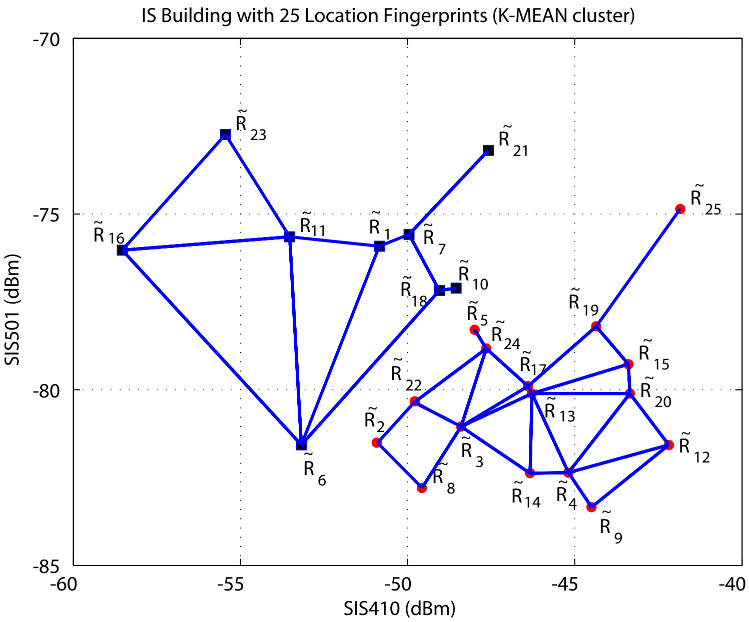

Figure 33: K-Mean Clustering and GGs: Scenario 1

in Figure 33. The K-Mean clustering method produces 9 fingerprints in one cluster and 16 fingerprints in the other cluster. Centers for two clusters are [-51.86, -76.12] and [-46.14, -80.31] respectively.

In the Scenario 2, fingerprints are based on 2 APs (hl2_b_card1 and hl4_b_card1) as shown in Figure 31. Hence, a fingerprint is represented as a RSS vector from both APs denoted by [hl2_b_card1, hl4_b_card1]. However, there are only 42 out of 71 locations that we can have coverage from both APs. The Median clustering in the Scenario 2 is shown in Figure 34. Fingerprints are divided into clusters of 21 fingerprints each indicated by dots and squares. AP hl2_b_card1 is selected to derive a median clustering with the RSS median value of -67.31 dBm. Two GGs are constructed but not shown in the figure for a better view of clustering. The K-Mean clustering method for the Scenario 2 is shown in Figure 35. There is a cluster with 18 fingerprints and the another cluster with 24 fingerprints. Centers for the two clusters in this scenario are [-85.69, -89.42] and [-60.74 -81.46] (dBm) respectively. Again, two GGs are constructed (not shown) and used separately for further calculations.

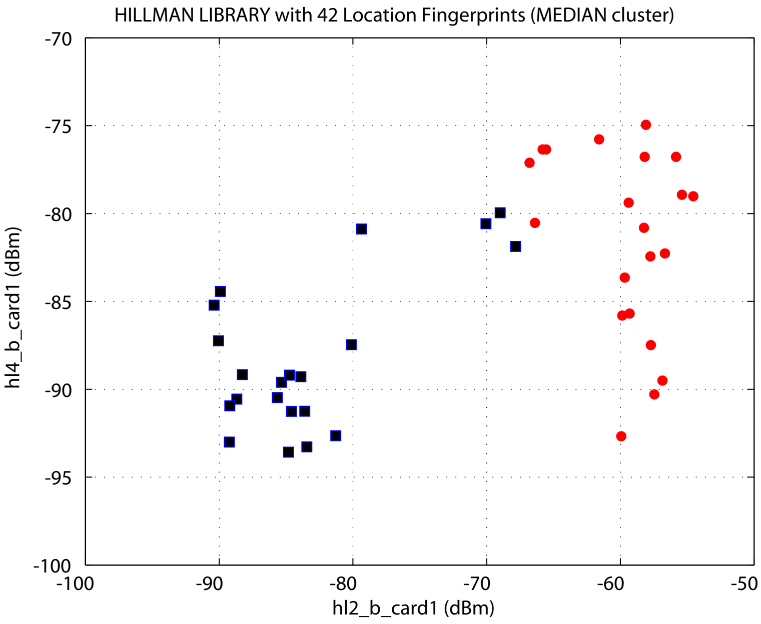

Figure 34: Median Clustering: Scenario 2

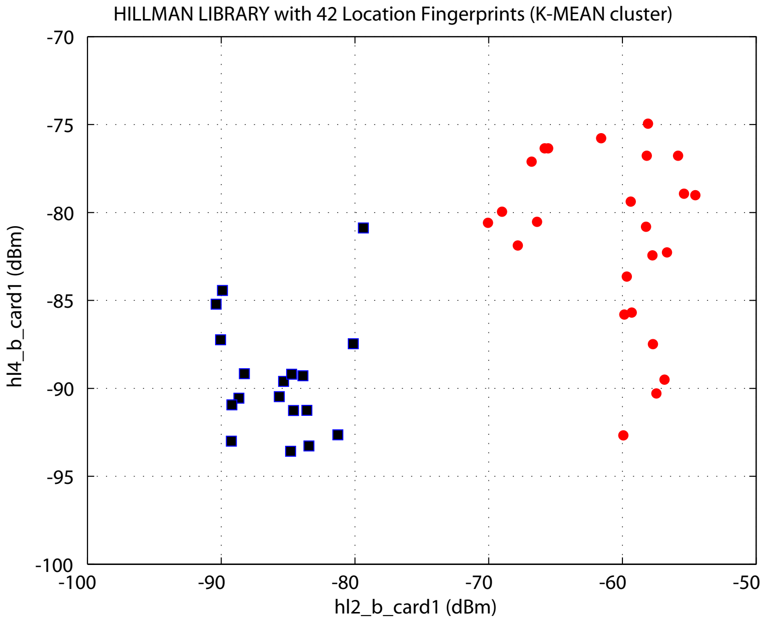

Figure 35: K-Mean Clustering: Scenario 2

2. Results with Fingerprint Clustering

In each scenario, for each cluster, we use the separate GGs to find the approximate probability distributions of picking a location, and use the fingerprint elimination procedure as discussed in the Chapter III. We also simulate results for the average CDFs of error distance after fingerprint elimination. We compare the two results when clustering is used and without clustering. Without clustering, a single GG with all fingerprints is constructed. The average CDFs of the error distance in Scenario 1 and 2 are shown in Figure 36 and Figure 37 respectively.

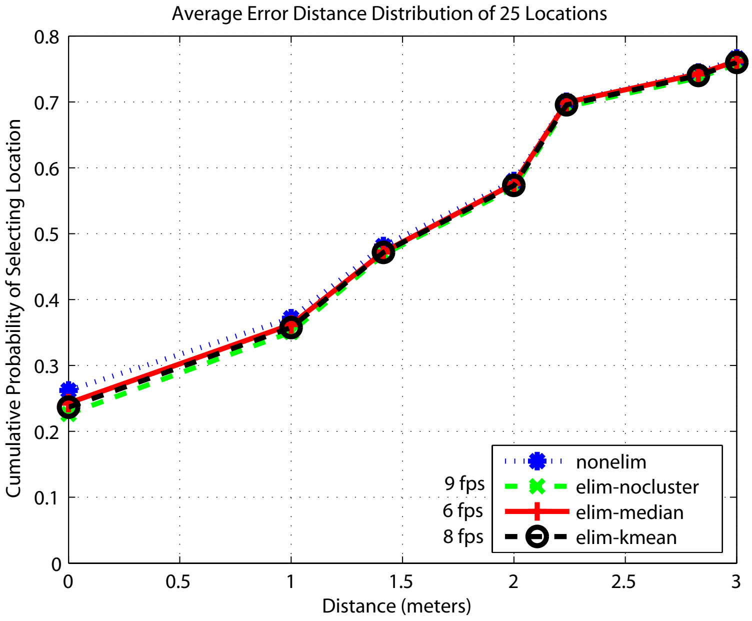

Figure 36: The Average CDF of Error Distance with Clustering in Scenario 1

From Figure 36, we can see that simulation results of the cumulative probability from the Median and K-Mean clustering methods (indicated by the "elim-median" and "elim-kmean" lines respectively) show only small difference (less than 0.05) from the result when the elimination procedure was used but no clustering method was applied (indicated by the "elim-nocluster" line). The result when all 25 fingerprints are used for positioning (indicated by the "nonelim" line) is shown as a baseline performance. Number of eliminated fingerprints are lower when the clustering methods are used; 9 from no clustering, 6 from the Median

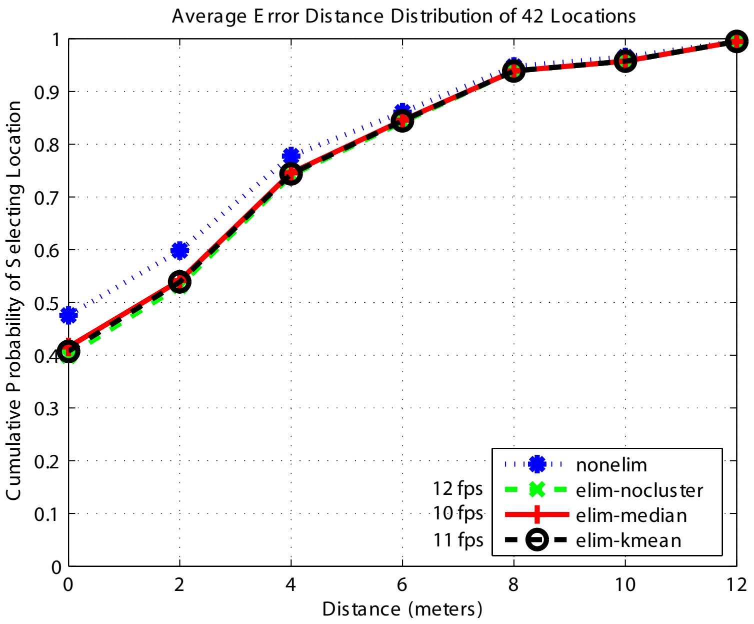

Figure 37: The Average CDF of Error Distance with Clustering in Scenario 2

clustering, and 8 from the K-Mean clustering. This can be explained as follows. Because of the separate GGs used for determining the approximate probability distributions, pairwise comparison for eliminating fingerprints is only applied to fingerprints within the same cluster. As such, a fingerprint at the border of one cluster completely ignores the chance of picking a border fingerprint from the other cluster, although they may have a small Euclidean distance between them. Therefore, the elimination procedure results in fewer numbers of eliminated fingerprints as compared to the case where a single GG is used without clustering. In some ways what we would accomplish by eliminating fingerprints is accomplished with minimal penalty by clustering, since fingerprints from other clusters are eliminated automatically from comparison.

Such elimination, however, is performed without considering pairwise error probabilities. However, it is quite possible that a measured RSS vector was associated with a wrong cluster during the online phase. The minimum error distance in such case is between a pair of fingerprints, one from each cluster, with the smallest Euclidean distance. The maximum error could be larger, but is highly unlikely to occur because the probability that a measured RSS vector is associated with a fingerprint in the wrong cluster that has a large Euclidean distance to the correct fingerprint in the correct cluster is negligible. As summarized in Table 7, this minimum physical error is smaller with the median clustering compared to K-Mean clustering. Further, simulations show that average CDFs of error distance for MSs located at different points cannot be distinguished from one another whether or not clustering was used, or based on the clustering method, at error distances greater than 2m . So we can conclude that, by applying clustering methods, we can still maintain almost the same precision performance as without clustering.

This is also true from Scenario 2 as indicated by the results in Figure 37. In Scenario 2, the number of eliminated fingerprints are 12, 10, and 11 with no clustering, Median clustering, and K-Mean clustering respectively. The difference of the average CDFs of the error distance between the two clustering methods and without clustering is minimal (less than 0.05, for error distances less than 4m and negligible at larger distances). In other words, no significant difference in the probability of selecting a location is observed if clustering is used.

Table 7: Error Distance with Clustering (meter)

Sceanario	**Median**	**K-Mean**
1	1	1
2	8	12

Results of Clustering with $K > 2$

We further apply clustering methods for the case when more than two clusters are used. Here we only look at fingerprint system from the Scenario 1. Figure 38 shows the Median clustering with $K = 4$. In the figure, SIS410 and SIS501 is selected for the first and the second highest RSS range APs respectively. The RSS median value from SIS501 is -79.27 dBm. Hence, using the medians from both APs produces four fingerprint clusters. Figure 39 shows the K-Mean clustering with $K = 3$. Centers for three clusters are [-50.37, -81.46], [-44.96, -79.94], and [-51.69, -75.43] respectively.

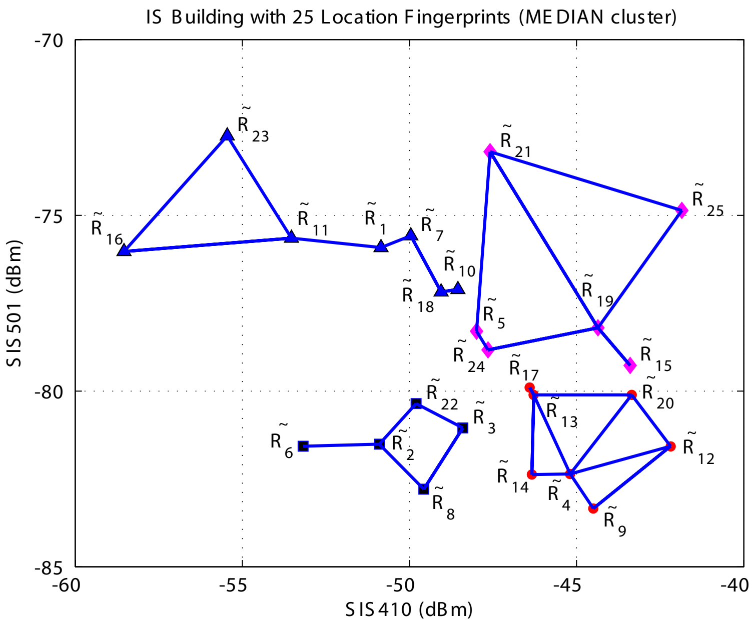

Figure 38: Median Clustering with More Clusters ($K = 4$)

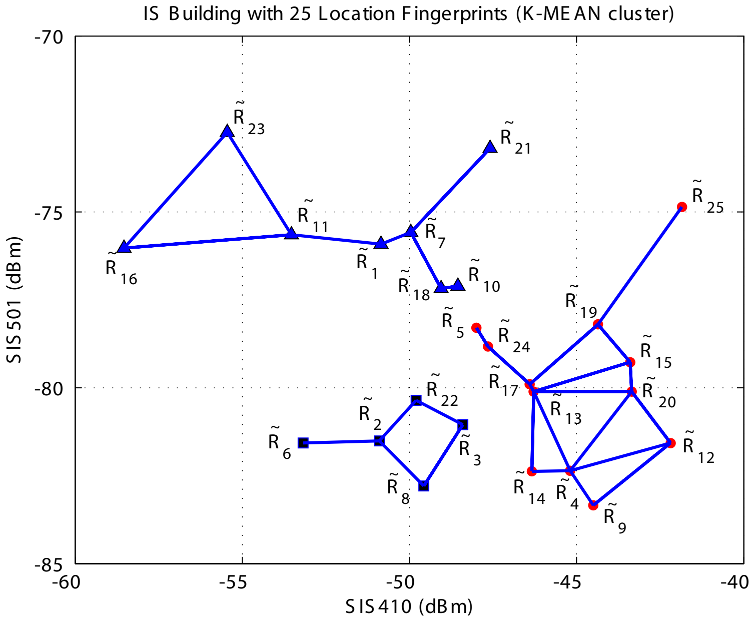

Figure 39: K-Mean Clustering More Clusters($K = 3$)

In a manner similar to the previous results, we simulate and compare results for the average CDFs of error distance after fingerprint elimination. The average CDFs from different number of clusters are shown in Figure 40. For Median clustering with $K = 4$ and K-Mean clustering with $K = 3$, it is turned out that the number of eliminated fingerprints are lower than the case when two clusters are used. The number of eliminated fingerprints are 5 and 7 with Median clustering and K-Mean clustering respectively. However, we observe that the difference of the average CDFs of the error distance is minimal and disappears at larger distances. In other words, no significant change in the probability of selection a location is observed from the study.

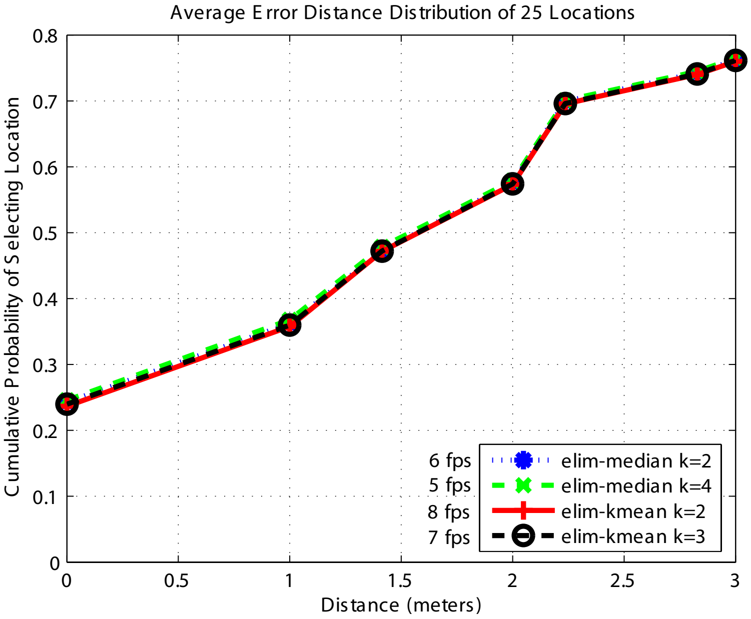

Figure 40: The Average CDF of Error Distance with 3(K-Mean) and 4(Median) Clusters

We further evaluate the possibility that clustering can cause performance to get worse. We vary number of clusters (K) in the K-Mean clustering and look at the best and the worst results of the average CDFs of error distance. It is expected that a K value that represents actual clustering of a given fingerprint system will produce the best average CDF of error distance. The CDF results are shown in Figure 41. We found that the CDF result increases when K increases, starting from 1 cluster. The result reaches its best when $K = 6$. Figure

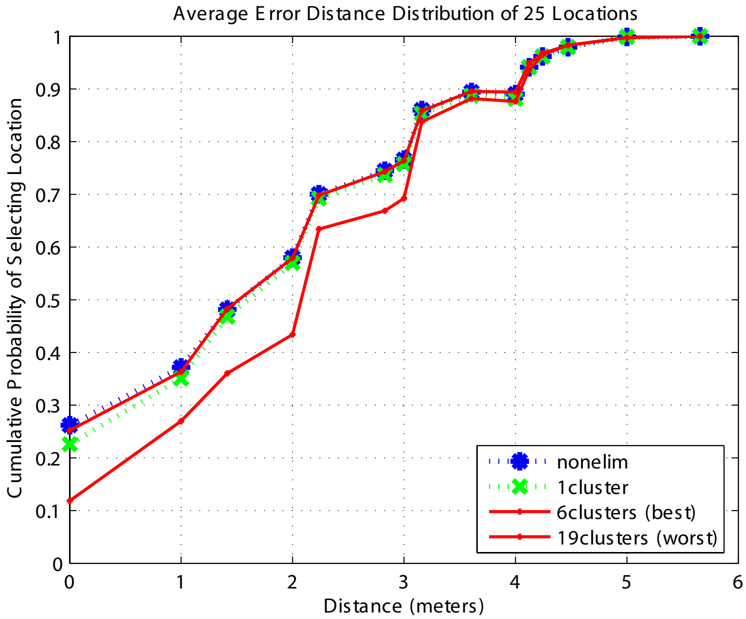

Figure 41: The Average CDF of Error Distance with 6 and 19 Clusters

42 shows the 6 clusters of fingerprints and corresponding 6 GGs. If we increase K further, the CDF results begin to drop and reach the worst result when $K = 19$. As shown in Figure 41, the average probability of selection of a location can drop from the best result to almost 20% below at smaller error distance (i.e,. 0 to 2 meters). The difference between the best and the worst probability becomes smaller as the error distance increases. Hence, from this observation of the impact of number of clusters on the performance, we can conclude that changing the number of clusters can cause the average CDFs of error distance to be better or worse. Selecting the number of clusters that represents how given fingerprints are actually clustered can yield the best result.

3. Comparison of Computational Effort With and Without Clustering

In this subsection, we analyze the complexity for the total computation with fingerprint clustering and without clustering in both the offline and online phases. This will provide us

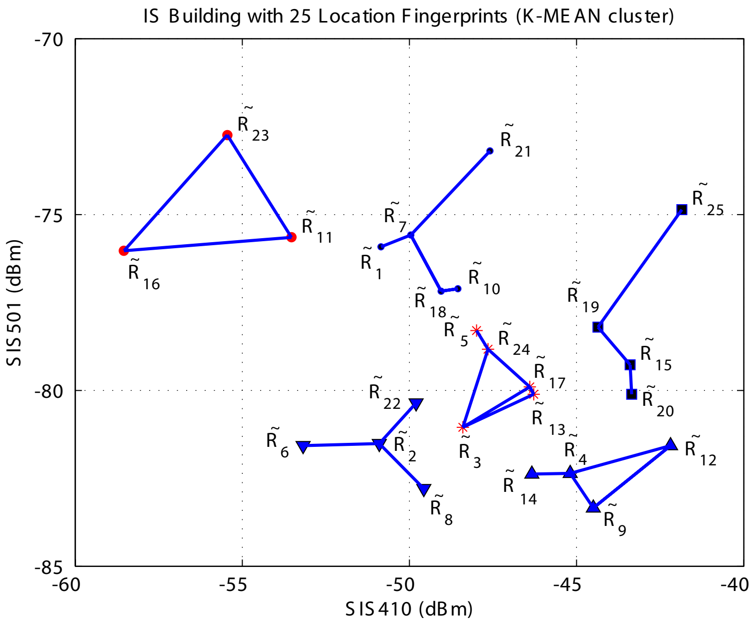

Figure 42: K-Mean Clustering with more Clusters($K = 6$)

a basis for concluding whether or not clustering helps in saving on computation (we already know that there is minimal impact on the performance based on the error CDF results shown earlier). With K clusters, we assume M_{ci} as the number of fingerprints in cluster C_i. Ideally, $M_{ci} = M/K$ and we use this below although the following can be done with knowledge of individual M_{ci}.

During the offline phase, there are 3 sequential tasks: 1) clustering 2) GG construction and 3) pairwise comparison for fingerprint elimination:

1) *Clustering Setup:* Here, either the Median or the K-Mean clustering is performed. With Median Clustering, for each of N APs, we first sort M fingerprints using a sorting algorithm, find the RSS range (max - min), and find the median. Sorting takes $O(M \log M)$ while finding the range and median ($O(1)$) can be ignored. Totally, this step needs $N \times O(M \log M) = O(NM \log M)$ operations. Next, we sort the computed ranges from N APs and find the largest which takes $O(N \log N)$ operations. Finally, we assign a cluster to each of M fingerprints by comparing each to the median. For K clusters (or K medians), this

needs $O(KM)$ operations. Hence, the total number of operations with Median clustering is $O(NM\log M + N\log N + KM)$. If $N, K \ll M$ as is usually the case, the above result reduces to $O(NM\log M + KM)$. If K-Mean clustering is used, we first randomly pick K cluster centers from all fingerprints, which takes $O(K)$ operations. Next we assign a cluster to each of M fingerprints by computing Euclidean distances to K centers and pick the smallest. Computing K Euclidean distances to a fingerprint takes $K \times O(N) = O(KN)$ (because of the N dimensions in the Euclidean distance) and picking the smallest distance takes $O(K)$ [1]. To compute K new centers, we need to consider M fingerprints each with N dimensions for a total of $O(NM)$ steps. Then each fingerprint has to be assigned to the clusters again. Given t iterations before convergence, totally we need $O(KM(N+1)t + NMt)$ steps. Hence, the total number of operations with K-Mean clustering is $O(K + KM(N+1)t + NMt)$. If $N, K \ll M$, the above result reduces to $O((KN + K + N)Mt)$.

2) *GG Construction:* Without clustering, the computation of a single GG takes $O(NM^3)$. With clustering and K GGs, we need $O(NM_{c1}^3 + \cdots + NM_{cK}^3)$ operations. Ideally, with M/K fingerprints/cluster, we need $O(NM^3/K^2)$ steps.

3) *Pairwise Fingerprint Elimination:* With no clustering, we need $O(M^2)$ steps for the probability distribution comparisons. With clustering, we need $O(M^2/K)$ operations.

During the online phase, to determine a MS's location, a system with fingerprint clustering performs two sequential tasks: 4) determining cluster membership and 5) finding nearest neighbor in signal space (*NNSS*).

4) *Determining Cluster Membership:* To determine cluster membership, with Median clustering, the measured RSS sample in the dimension of the pre-selected "highest RSS range" AP, is compared with the pre-computed median (i.e., -47.99 dBm in Scenario 1 or -67.31 dBm in Scenario 2). With K clusters, we need $O(K)$ operations for this task. If K-Mean clustering is used, the Euclidean distance (N dimensions) between the measured RSS vector and the K cluster centers are computed. The cluster whose center has the smallest Euclidean distance to the sample vector is picked. Hence, determining the membership takes $O(KN)$ operations.

[1] We assume distance computation in one dimension and comparisons are equivalent in complexity.

Table 8: COMPUTATION OF FINGERPRINT CLUSTERING

Phase	**Task**	**Clustering**		**No Clustering**
		Median	**K-Mean**	
Offline	1)	$O(NM \log M + KM)$	$O((KN + K + N)Mt)$	-
	2)	$O(NM^3/K^2)$	$O(NM^3/K^2)$	$O(NM^3)$
	3)	$O(M^2/K)$	$O(M^2/K)$	$O(M^2)$
Online	4)	$O(K)$	$O(KN)$	-
	5)	$O(NM/K)$	$O(NM/K)$	$O(NM)$

5) NNSS: Once the cluster membership is determined, the Euclidean distance from a sample vector to all fingerprints in this cluster is computed. The the nearest fingerprint points to the estimate of the MS's location. Hence, on average, we need $O(NM/K)$ for the NSSS operation. Without clustering, we need $O(NM)$ operations. In fact, since some fingerprints were already eliminated from the offline phase, the number of comparisons needed is actually lower than that shown above. Table 8 summarizes the computational complexity of the five tasks.

To illustrate this numerically, we compare the number of operations that a system takes to complete both the online and offline tasks in Scenarios 1 and 2. The number of operations are based on the "Big O" notation in Table 8. Based on Figure 32 and 33, the following parameters are used to compute the number of operations needed in Scenario 1: $M = 25$, $K = 2$, $N = 2$, $M_{c1} = 12$(Median) or 9(K-Mean), $M_{c2} = 13$(Median) or 16(K-Mean), and $t = 4$(from simulations). Likewise, the following parameters are used to compute the number of operations needed in Scenario 2: $M = 42$, $K = 2$, $N = 2$, $M_{c1} = 21$(Median) or 18(K-Mean), $M_{c2} = 21$(Median) or 24(K-Mean), and $t = 6$.

In the offline phase, number of operations used in a location system with the fingerprint clustering is a combination of all three tasks as described earlier. However, only the last two tasks are necessary for a system without clustering. As such, a system with the Median clustering would take $[2(25)(log 25)+2(log 2)+2(25)]+[2(12^3)+2(13^3)]+[(12^2)+(13^2)] = 8,284$ operations. A system with the K-Mean clustering would take $[2 + 2(25)(3)(4) + 2(25)(4)] +$

$[2(9^3)+2(16^3)]+[(9^2)+(16^2)] = 10,789$ operations. A system without clustering would take $[2(25^3)] + [25^2] = 31,875$ operations. Therefore, we can save about one-third of the number of operations with the fingerprint clustering.

In the online phase, number of operations used in a location system with the fingerprint clustering is a combination of two tasks as described earlier. However, only the NNSS task is necessary for a system without clustering. So, a system with the Median clustering would take $[2] + [2(12 + 13)/2] = 27$ operations. A system with the K-Mean clustering would take $[2(2)] + [2(9 + 16)/2] = 29$ operations. A system without clustering would take $[2(25] = 50$ operations. So we could save about a half of the number of operations with the fingerprint clustering.

Similar to the calculation in Scenario 1, for Scenario 2 in the offline phase a system with the Median and the K-Mean clustering would take 38,147 and 42,230 operations respectively. A system without clustering would take 149,940 operations. In the online phase, a system with the Median and the K-Mean clustering would take 44 and 46 operations respectively. A system without clustering would take 84 operations. Summarized results for the operations of fingerprint clustering is shown in Table 9.

Table 9: Operations Comparison

Method	Scenario 1		Scenario 2	
	Offline	**Online**	**Offline**	**Online**
Median	8,284	27	38,147	44
K-Mean	10,789	29	42,230	46
No Clustering	31,875	50	149,940	84

To empirically evaluate an impact of number of clusters to the number of operations in a fingerprint system, we plot its relationships as shown in Figure 43. We consider the total number of operations from both the offline and online phases. Here, we give an example when K-Mean clustering is applied in system of 2 APs and 25 fingerprints ($N = 2$,$M = 25$), similar to those from Scenario 1. We assume a constant $t = 4$. From the figure, we can see that we could achieve the lowest number of operations when we have 6 clusters ($K = 6$). Considering this result and the performance result with 6 clusters in Figure 41, we can say that using

6 clusters is a suitable number of clusters for a fingerprint system in Scenario 1 as it yields the best performance with the smallest computational effort. Note that the suitable number of clusters, K, can be different in system with different measurement data. An interesting research work is to evaluate a suitable K for system from different indoor environments. This is beyond the focus of this dissertation. However, it is worth to mention that a good value of K should provide acceptable performance and less computational effort. In addition, one is also interested in using a non-parametric clustering method (i.e. one without determining K) and observe its performance. An example of such method is a mean-shift clustering [41].

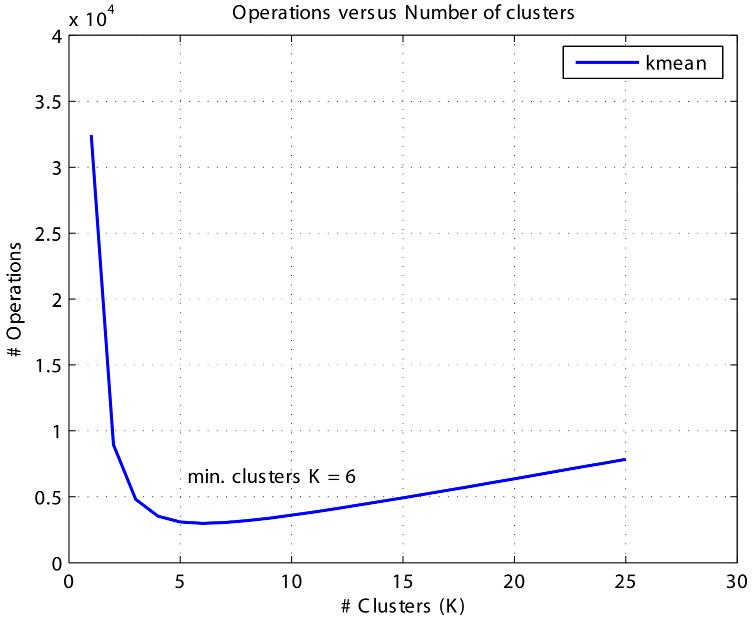

Figure 43: Operations versus Number of Clusters

In summary, fingerprint clustering is shown to help reduce computational effort for a location fingerprint system making the performance prediction and actual position determination scalable. Future work will be focussed on generalizing this approach further and quantifying its impact.

In the next chapter, we look at the use of a radio map with a real device. A study of modeling using a close-to-actual RSS distribution is also presented as we want to know if the distribution can improve the precision performance model.

VI. MODELING AND RADIO MAP IN REAL SITUATION

In this chapter we look at radio map construction based on a real situation. First we motivate the study of a modified analytical model that takes into account signal strength variations that are different for neighboring fingerprints and non-Normal distributions of the RSS at different locations. Then a usage scenario of the radio map for a real mobile device is given. This is based on the assumption that geospatial information such as the floor plan of a building, or even the user's travel profile, could potentially provide useful hints for storing the radio map in the device. A simple guideline and application of the guideline to radio map construction during the offline phase is also presented. Finally, some remarks on miscellaneous issues are provided.

A. ANALYTICAL MODEL WITH BETTER RSS DISTRIBUTION

In this section, we propose an analytical model based on more realistic assumptions. In section III.A, we assumed the mean of the received signal strength (RSS) envelope at a mobile station (MS) is a normally distributed random variable[1]. The normal assumption not only allows tractability of the mathematical model, but it is also widely used to describe large-scale fading (known as a *shadowing fading*) effects in wireless channels [20]. We note that, although the received signal is affected by both large-scale and small-scale fading components, the RSS measurement at a wireless adapter averages out the effects of the small-scale fading component [1]. So only the large-scale component is the one of interest here.

[1] To be precise, the probability density function (PDF) of the RSS envelope is assumed as log-normally distributed. We simply refer it as "Normal" or Gaussian, since the RSS measurement is typically reported on a log scale (i.e., in dBm).

From extensive measurements, however, different distributions and variations of RSS are observed at different grid locations [1, 7, 17, 38]. The RSS distribution can be skewed from the presumed Normal distribution. So using the NNSS, with the normal assumption and computing of distance between fingerprints represented by mean values, can make the model of probability of fingerprint selection less accurate. However, there is no previous work to study if the use of a more realistic RSS distribution model (such as a distribution with shape parameters) can actually enhance the model of precision performance. Therefore we try to partially address this non-Normal distribution modeling issue.

Based on the RSS histograms in different locations for Scenarios 1 and 2 (see scenario descriptions in subsection V.B.1), certain shapes of distribution were found at particular average values of the RSS. Different shapes of the RSS distribution are caused by upper and lower limit of measurable RSS at each location. The maximum transmitted power provides the upper limit and the receiver's sensitivity gives the lower limit for the received signal. The analysis of measurements by Dr. Kamol Kaemarungsi showed that many histograms possess a long tail to the left (can be described as left-skewed). There are 64 out of 75 (85%) and 191 out of 299 (64%) left-skewed histograms in Scenario 1 and Scenario 2, respectively [1]. These histograms are ones often found with strong average RSS or when there is a line-of-sight (LOS) between an AP and a MS. It was previously explained that the left-skewed distribution is the effect of the range limitation imposed by the maximum RSS at each location [38].

Due to the complexity of the radio propagation and simplicity of measurements at the WLAN interface card, the actual distribution of the RSS is difficult to obtain. However, it would be a great benefit if, besides the normal distribution, we can find an approximate distribution of the underlying RSS process [38]. Therefore, we try to fit the measured RSS distribution using a more flexible probability distribution than the Normal distribution. We choose the *Beta distribution* in this study for the above reason. The Beta distribution is chosen as it also satisfies the following properties: (1) the capability of modeling various distribution shapes from left skewed to symmetrical to right skewed (not always symmetric, as in the normal distribution), and (2) the capability of modeling distributions with a finite range (as opposed to the Normal distribution, which extends infinitely to both sides of the distribution). These properties make the Beta distribution an excellent candidate for representing many characteristics in real life. With the Beta distribution, a variety of distribution shapes can be modeled with two shape parameters, λ_1 and λ_2 as depicted in Figure 44.

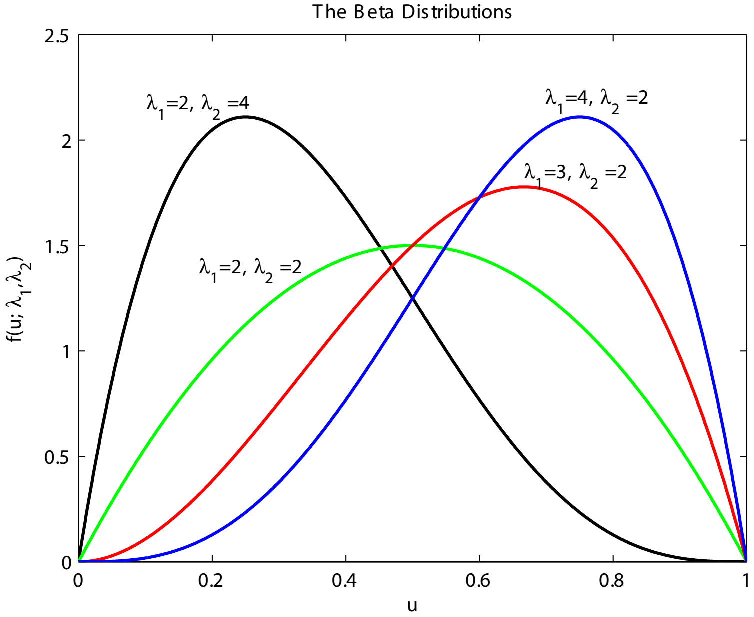

Figure 44: Probability Density Function of the Beta Distribution

A Beta distribution is defined over the interval (0,1) [42]. However, the generalized Beta distribution can be defined over the interval (a,b), where a represents the minimum value and b represents the maximum value respectively. The unit beta probability density function is given as follows:

$$f(u;\lambda_1,\lambda_1) \quad = \quad \frac{\Gamma(\lambda_1+\lambda_2)}{\Gamma(\lambda_1)\Gamma(\lambda_2)}u^{\lambda_1-1}(1-u)^{\lambda_2-1}\ ;\ 0<u<1\ ;\ \lambda_1,\lambda_2>0 \qquad \text{(VI.1)}$$

,where $\Gamma(.)$ represents the *gamma function* [42]. The unit cumulative probability distribution function is defined as

$$F(U;\lambda_1,\lambda_2) \quad = \quad \frac{\Gamma(\lambda_1+\lambda_2)}{\Gamma(\lambda_1)\Gamma(\lambda_2)}\int_0^U u^{\lambda_1-1}(1-u)^{\lambda_2-1}du. \qquad \text{(VI.2)}$$

$F(U;\lambda_1,\lambda_2)$ is the probability that u falls within the interval (0,U). The probability can be calculated given the shape parameters, λ_1 and λ_2, and the distribution range U. The mean value (μ) and variance (σ^2) of a unit Beta distribution can be calculated from the following equations:

$$\mu = \frac{\lambda_1}{\lambda_1+\lambda_2}\ ;\ \sigma^2 = \frac{\lambda_1\lambda_2}{(\lambda_1+\lambda_2)^2(\lambda_1+\lambda_2+1)} \qquad \text{(VI.3)}$$

A generalized Beta random variable x over the interval (a,b) is derived by re-scaling and re-allocating the unit Beta random variable u over interval (0,1) of the same shape by the following transformation:

$$x = a + (b-a)u \qquad \text{(VI.4)}$$

We fit both the Normal and Beta distributions to an RSS distribution from the measurement. We use Statical Analysis System ($SAS^{®}$, [43]) software for fitting distributions and estimating its parameters (e.g., mean, variance, and shape parameters). Figure 45 shows an example of an RSS histogram collected, at a fixed location, from the AP SIS410 in Scenario 1. Data was collected at approximately 0.25 seconds per sampling interval for a five-minute period. So there are approximately 1,200 RSS samples. The curve in the figure shows a fitted Normal distribution. Figure 46 shows a fitted Beta distribution on the same histogram. From both figures, we can see that the Beta distribution, with estimated shape parameters $\lambda_1 = 7.09$ $\lambda_2 = 1.9$, is a better fit to the histogram than the Normal distribution.

We also use a probability plot to compare ordered values of the RSS with the percentiles of a specific theoretical distribution model. If they match, the points on the plot form a linear pattern. The probability plots from the Normal distribution and the Beta distribution are depicted in Figure 47 and 48 respectively. A diagonal reference line corresponding to the distribution with estimated parameters is shown in each of the probability plots. We can see that more numbers of RSS samples are linearly matched to the reference line in Figure 48 than those in Figure 47. Therefore, we can conclude from these results that the Beta distribution serves as a better model for the measured RSS distribution.

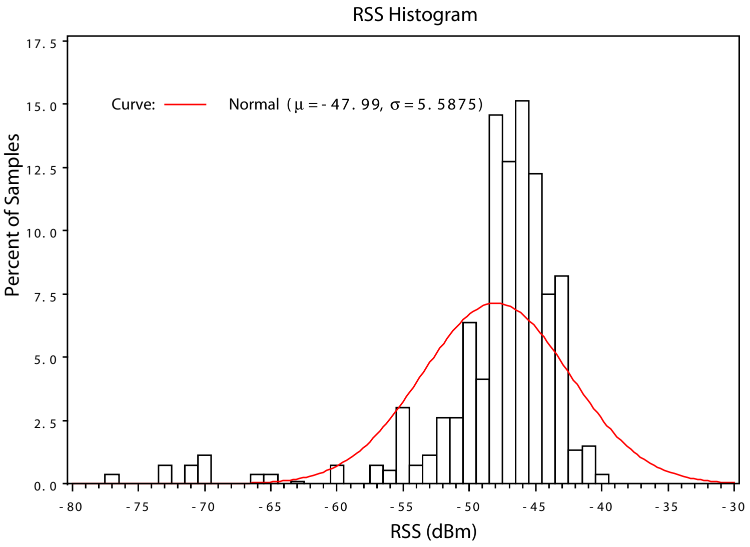

Figure 45: RSS Histogram and the Normal Distribution

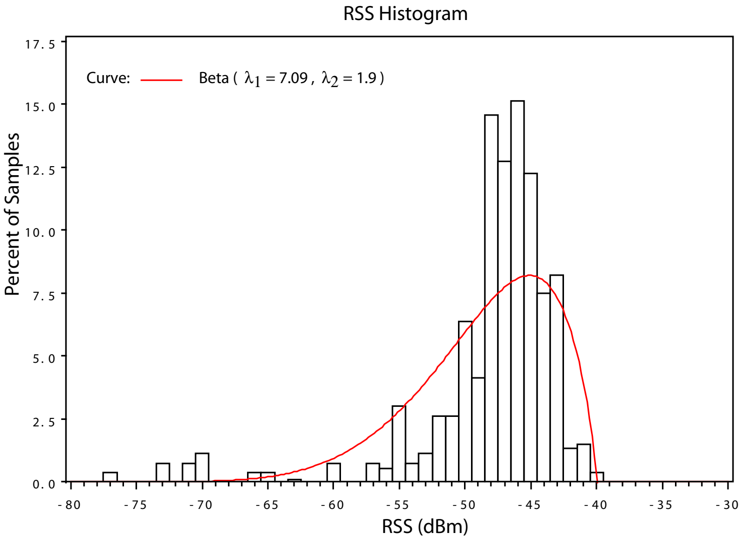

Figure 46: RSS Histogram and the Beta Distribution

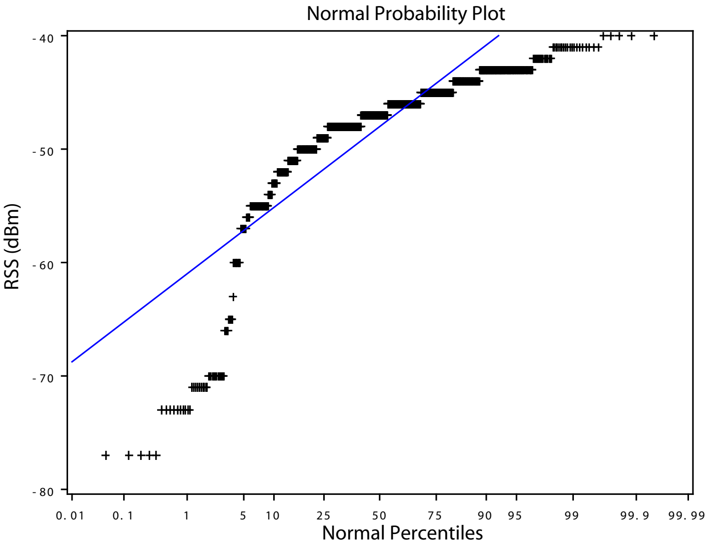

Figure 47: The Normal Probability Plot from the RSS Measurement

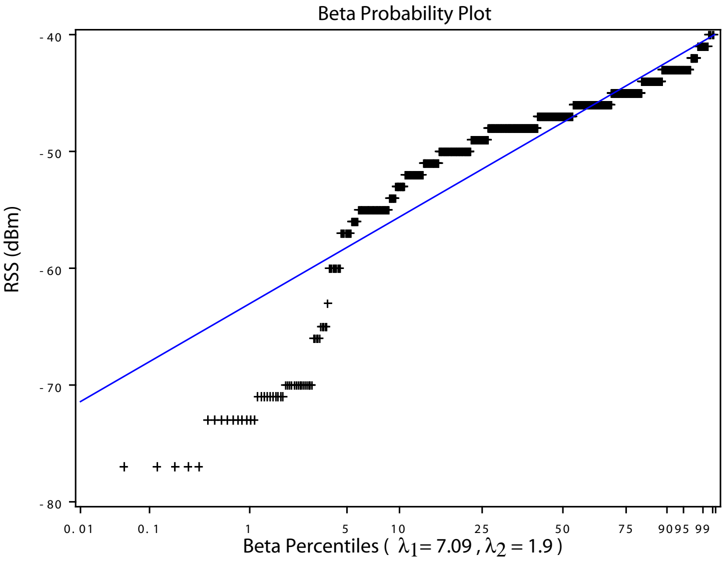

Figure 48: The Beta Probability Plot from the RSS Measurement

Although the Beta distribution provides a good fit to the RSS distribution from the real measurement, to incorporate it into analysis needs a certain modification of the model (e.g., for identifying fingerprint neighbors). In this dissertation we do not find a method to incorporate non-Normal distributions in the performance modeling (it is left as part of future research). However, we study the issue of incorporating at least the RSSs with non-identical Normal distributions in our analytical model so as to get a better understanding.

In Chapter III, constructing the analytical model employs proximity structure and the Gabriel proximity graph (GG) to extract information about neighbor and non-neighbor sets. An edge in the GG and thus a neighbor set, is obtained by considering a diametral circle with a line segment between any two fingerprints, $\tilde{R}_i$ and $\tilde{R}_j$, as the diameter (see subsection II.C.2). If we explore this in detail, we will see that each diametral circle in the GG has its center at the mid point of the line segment $\overline{\tilde{R}_i\tilde{R}_j}$. This is exactly a point on the decision boundary (i.e., a Voronoi edge) between the two fingerprints as seen in Figure 4. This boundary is symmetric as it is based on the assumption of uniform (identical) standard deviations of RSS in all fingerprints. Hence, the derived diametral circle, known as the forbidden region, between the two fingerprints has its center exactly at the mid point. Now from the above discussion, we can see that using a diametral circle to define a GG edge, and thus a GG neighbor, is not really correct as it does not consider the effect of different RSS standard deviations associated with different fingerprints. So, here a new circle for constructing GG that considers the effect of non-identical RSS distributions is studied. We call the new circle as the *equivalent diametral circle*.

To find the equivalent diametral circle and its center, we consider the intersection between distributions of two Gaussian variables (in 1-dimension) with different means and variances as shown in Figure 49. Assume one variable has a mean m_1 (representing the first fingerprint) and variance σ_1^2, while the other has a mean m_2 (representing the second fingerprint) and variance σ_2^2. We also assume $m_1 < m_2$ and $\sigma_1 \neq \sigma_2$. Instead of the mid point between m_1 and m_2 (i.e., if $\sigma_1 = \sigma_2$), an intersection point x will be used as a center of an equivalent diametral circle. It is at this point that an RSS sample has equal probability of picking either fingerprint, m_1 or m_2. The intersection point x can be computed as: (the derivation is given in Appendix C):

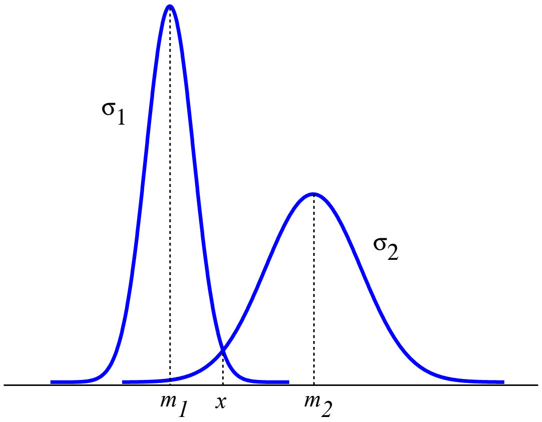

Figure 49: Intersection between 2 Gaussian Distributions Example

$$x = \frac{(\sigma_1^2 m_2 - \sigma_2^2 m_1) \pm \sigma_1\sigma_2\sqrt{(m_2-m_1)^2 + 2(\sigma_2^2-\sigma_1^2)ln\frac{\sigma_2}{\sigma_1}}}{(\sigma_1^2-\sigma_2^2)} \qquad \text{(VI.5)}$$

Between the two roots of x, the one with a value between m_1 and m_2 is used as the new center. Then the radius of the equivalent diametral circle is computed by finding the maximum between $|m_2 - x|$ and $|m_1 - x|$[2]. Once we find the equivalent diametral circles from all pairs of fingerprints and produce GG edges from the ones being empty (see subsection II.C.2), a new *skewed* GG, considering RSS distributions with different variances, is created. In this study, we will check if the new skewed GG, by taking into account of different RSS variation, could possibly be useful in improving the prediction of precision performance.

We compute the skewed GG from the previous measurement setup as shown in Figure 22. Here the RSS standard deviations measured from all 25 fingerprints are used. The skewed GG for the 25 grid system is shown in Figure 50. The skewed GG and fingerprints are based on SIS410 and SIS501 like in the previous study on subsection III.C.4.

[2] the maximum distance is selected as the new radius instead of the minimum distance, since the latter creates a circle that is almost always empty. As a result, the number of GG edges can be over estimated (i.e., two far fingerprints may be falsely interpreted as neighbors).

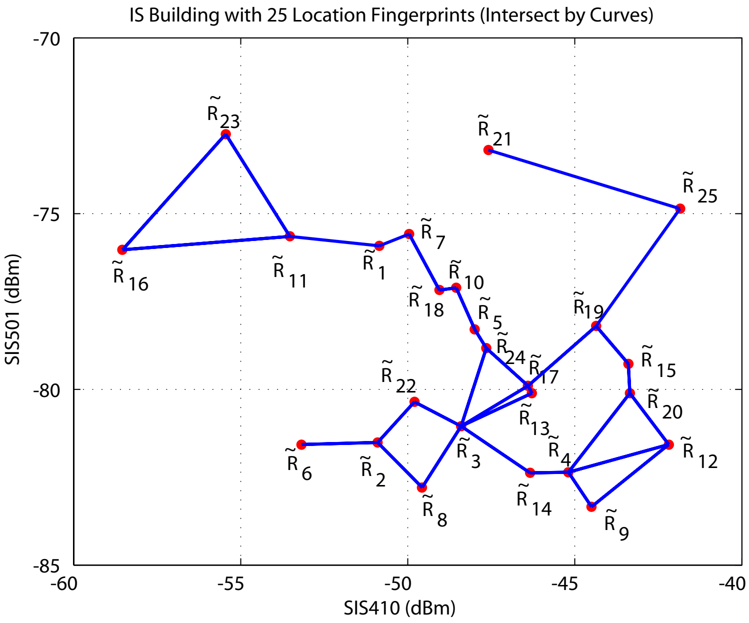

Figure 50: Skewed GG of Measured Radio Map

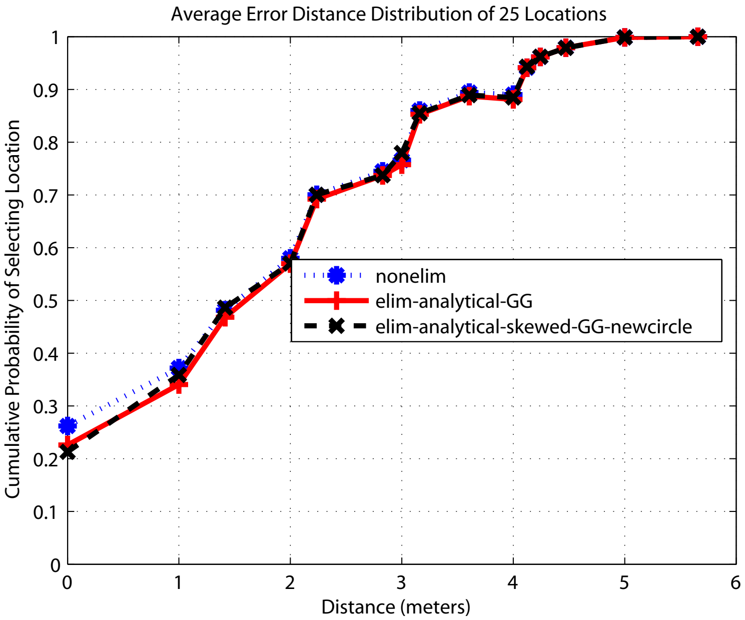

Figure 51: The Average CDF of Error Distance with a Skewed GG

Again, we apply the analytical model with the skewed GG, approximate the probability distributions of picking a location given the MS's location, and perform pairwise comparisons for fingerprint elimination. It turned out that we can identify 9 fingerprints that we can eliminate. Then we run simulations to evaluate how eliminating fingerprints, determined by different GGs, impacts performance. Figure 51 compares the average CDF of the error distance in an experiment. In the figure, the result when all 25 fingerprints are used for positioning (indicated by "nonelim") is shown as the baseline. A plot, as indicated by "elim-analytical-GG" is the result when a GG is applied. Then a plot, as indicated by "elim-analytical-skewed-GG-newcircle", is the result when a skewed GG is applied. We can see that the average cumulative distributions from the last two cases show no significant difference. The skewed GG result shows little change in the precision (i.e., its probability is slightly greater than that of the GG for the first few meters of error distance then they are the same as the error distance increases). This observation also holds when the system in Figure 31 (Scenario 2) is studied (see Appendix F).

So from this result, we conclude that, although taking into account the effect of different deviations and re-defining GG neighbors, the precision performance predicted may not change by much. However, this conclusion is based only on our best available measurement and current modeling. Future research on how to incorporate different distributions into the model or even answers to whether knowing "close-to-real" distributions is actually helpful is something to be studied (i.e., to find a new kind of a proximity graph or do more extensive experiments).

In the next section, we will discuss a usage scenario for storing the radio map in a real mobile device.

B. RADIO MAP WITH REAL DEVICE

We look at the use of the radio map for a real device. We are interested in the case where a positioning engine is located in a mobile device. In other words, the MS performs location estimation locally by comparing the measured fingerprint with a stored radio map. The case is considered important as it will allow an estimate of how many fingerprints in the radio map we should have in a mobile device. This has impact because of the reduced memory and computational capabilities of mobile devices.

Imagine many workers (known as users), each carrying a positioning-enabled mobile device with a radio map, moving around inside an office building. Each user will move differently within different areas. While moving, a mobile device estimates a user's current location. In a real scenario the movement of most users is only covered by the same certain area (i.e., places or rooms where he or she usually works everyday). In such scenario, if a mobile device contains fingerprint collection from the whole building, fingerprints from other areas (one outside user's typical working area), may not be useful and they become less important for a particular user. Hence, only some of the fingerprints should be enough for location estimation of the user. This also helps in reducing the fingerprint search space for location estimation.

From the above reasoning, we believe that a design of the radio map for each mobile device can be impacted by the *mobility profile* of a user. The mobility profile describes the user "movement history" inside a building. It can be obtained by a series of user's movement in each area, which can be represented by *a sequence*. For example, in certain time-frame, a user moves from table 1 to table 2, and then to room 5 and hallway 3, and so on. The sequence can be written as "table 1 → table 2 → room 5 → hallway 3 ...". This idea of representing visited locations as a sequence was previously adopted for the learning of inhabitant's movement profile [44]. The sequence can be influenced by a user's habit/routine, time of a day, or environment settings. This sequence then determines a set of fingerprints, which is collected at locations in the sequence, and thus the user's radio map. Deriving a radio map from a mobility profile allows users to store only fingerprints of locations actually visited by a user. This promotes an efficient use of the radio map, but it can cause performance loss due to limited mobility space (i.e., positioning system can be ineffective when a user moves into a different area).

In a fingerprinting based positioning system, location estimation during the online phase is done by two sequential tasks: 1) measuring a RSS sample and 2) computing the NNSS (see section III.A for the NNSS description). The sum of the delays from both tasks results in the total delay in estimating a user's current location. The total delay of the location estimation, however, is dominated by the measurement delay of an RSS sample in the first task. For example, the maximum RSS sampling rate of a Lucent Orinoco Gold wireless card is 4 samples/s – thus the smallest sampling period is 250 ms [1]. The delay from the second task depends on the total number of fingerprints used in the radio map for finding the nearest neighbor fingerprint. We will look at a numerical example of delays given parameters observed from a real device. We also study the "break even" number of fingerprints of the radio map where delay incurred from the two tasks are equal.

Assume there are N APs and M fingerprints in a system. Suppose a mobile device is an IPAQ pocket PC with the Motorola Dragonball RISC-based 200 MHz CPU. The clock period for this mobile device is 5 ns. The NNSS computational delay in the second task can be estimated as follows:

$$\text{The NSSS delay (CPU exec. time)} = \text{CPU clock cycles} \times \text{clock time} \tag{VI.6}$$

where CPU clock cycles = total number of instructions × average clock cycle per instruction (CPI). The average CPI of the Motorola Dragonball CPU is assumed to be 4. The total number of instructions (TNI) required by the NNSS is computed using different floating-point operations as shown in Table 10 [45].

Table 10: Floating-Point Operations

Operation	**Number of Instructions**
FCMP	70
FADD/FSUB	290
FMUL	180

To compute the NNSS, the first step we need is to find the square of the Euclidean distance between the RSS sample and each of M fingerprints. The square of the Euclidean distance is computed as $\sum_{i=1}^{N}(\rho_i - r_i)^2$. So for each of the N APs, we need one operation for floating-point subtraction (FSUB) and one operation of floating-point multiplication

(FMUL) for computing a square value. The sum of N such terms require $N-1$ operations of floating-point additions (FADD). With M fingerprints, the total operations of the first step is $M \times [N(\text{FSUB}) + N(\text{FMUL}) + (N-1)(\text{FADD})]$. The second step is to find the nearest neighbor fingerprint by selecting the fingerprint with the smallest distance among all computed M distances from the first step. This requires $M-1$ operations of floating-point comparisons (FCMP) to get the fingerprint with the smallest distance. So the total number of operations (TNO) to compute the NNSS is the sum of operations in both steps:

$$TNO = M \times [N(\text{FSUB}) + N(\text{FMUL}) + (N-1)(\text{FADD})] + (M-1)(\text{FCMP}) \qquad \text{(VI.7)}$$

Using the number of instructions required for each operation in the Table 10 and substituting in (VI.7), TNI can then be calculated from TNO. Note that the TNI implies the required number of CPU clock cycles. Finally, knowing values of all relevant variables allows the computation of the NSSS delay in (VI.6).

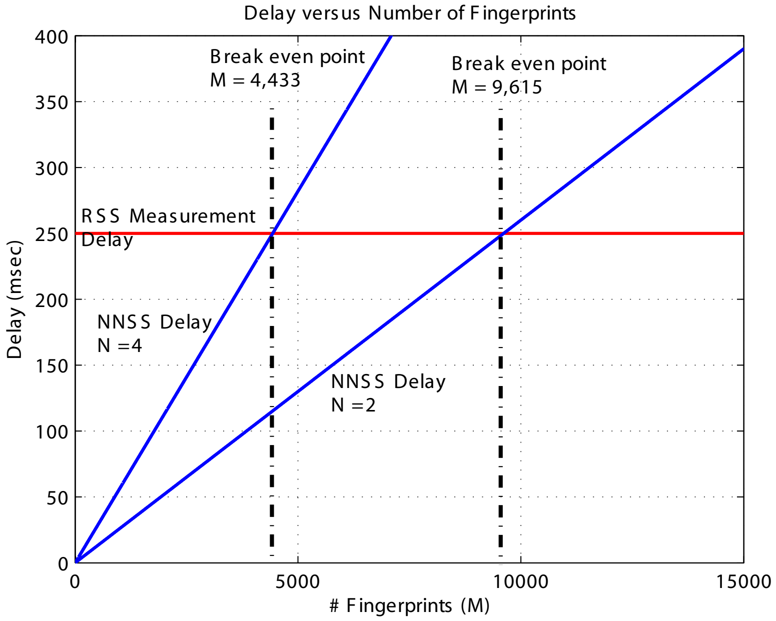

Figure 52: Delays as a Function of Number of Fingerprints

We plot the delay from the RSS sample measurement and the NNSS delay, as a function of number of fingerprints (M) in a radio map. Here we assume $N = 2$ or 4. The results are shown in Figure 52.

From the figure, we can conclude that we need at least 9,615 (4,433) fingerprints in a radio map for the system with 2 APs (4 APs) so the NNSS delay becomes the dominant delay over the RSS measurement delay. In addition, most radio maps of the proposed systems in Table 1 have the sizes only on the order of hundreds of fingerprints (or positions). So the above numerical example implies that the NNSS delay may not be a significant factor of the total delay during the online phase of such positioning systems. However, the benefit of using less number of fingerprints is already obtained as it reduces effort for collecting large numbers of RSS fingerprints during the offline phase as described next.

In the next section, we will discuss how to create a radio map and apply our analytical study for the fingerprint collection during the offline phase.

C. OFF-LINE PHASE FINGERPRINT COLLECTING GUIDELINE

Identifying and eliminating unnecessary fingerprints from a radio map database can reduce computational time spent in the online phase for fingerprinting based positioning systems. This is a “by-product” of the analytical model. However, collecting fingerprints from all predefined grid locations in space during the offline phase still proves to be very tedious and time-consuming. System designers lack hints that could help them cleverly pick locations from a site-survey during the offline phase. By studying the actual measurement data combined with the simple analytical results, we present here a few simple guidelines that we think can be handy during pre-deployment of the system. However, further work to refine and quantify these guidelines can be interesting for future study.

From the measured characteristics of the RSS, the σ of the RSS can vary from one location to another and from one AP to another. In [1], it was found that the standard deviation of the RSS is large (6-7 dB) when the MS is located near an AP and sees a strong RSS (-60 dBm to -40 dBm). These locations usually see a direct line of sight (LOS) of the received signal between the AP and the MS. On the contrary, the standard deviation is small (1-2 dB) when the MS is located far from the AP with weak RSS (-95 dBm to -85 dBm). Such locations often are in non line-of-sight (NLOS) situations in terms of the received signal between the AP and the MS. With large deviation of the signal, the error probability of fingerprint selection is high. We believe that an “inefficient” fingerprint (one that is hardly picked as a correct location) is likely to be gathered at a location close to the AP. In other words, observing a cluster of locations with a large RSS vector’s magnitude will likely contain a lot of “inefficient” fingerprints. Hence, a small grid spacing should not be used at such locations during the offline site surveying. Using a larger granularity of a grid spacing in the locations with a strong RSS vector could save on labor while potentially keeping acceptable performance. In addition, as reported in [20], large standard deviations are found inside large and open space buildings, while small standard deviations are found inside small and closed spaces. So, system designers should expect to see many inefficient fingerprints when deploying systems inside open space areas. As a consequence, they could possibly select, for example, a sparser grid spacing in open environments as compared to cluttered environments.

To illustrate the above“offline” guideline, we construct a new grid system based on the measurement setup in Figure 22. In the new system, a 2-meter grid spacing is used for locations inside room 410 (i.e., representing open space area) and 1-meter grid spacing for locations on both sides of the room’s wall (i.e., representing closed space area). The 2-meter grid locations include grid locations 3, 5, 13, 15, 23, and 25, while the 1-meter grid locations include grid locations 1, 2, 6, 7, 11, 12, 16, 17, 21, and 22 respectively.

It is now reasonable to compare Gabriel Graphs derived from two cases: one from the new grid system with sparser grid points in open areas and the other from the previous uniform grid system (i.e., 1 meter spacing throughout) – after the elimination procedure is applied. The remaining 16 fingerprints are obtained from both cases. The two GGs are shown in Figure 53. We find graphical similarity of the fingerprints on the left half of both graphs (i.e., fingerprints and their associated edges with SIS410’s RSS $<$ -50 dBm). This confirms that that the probability of location estimation for locations associated with these fingerprints will be equal. A majority of these fingerprints are from grid locations outside the

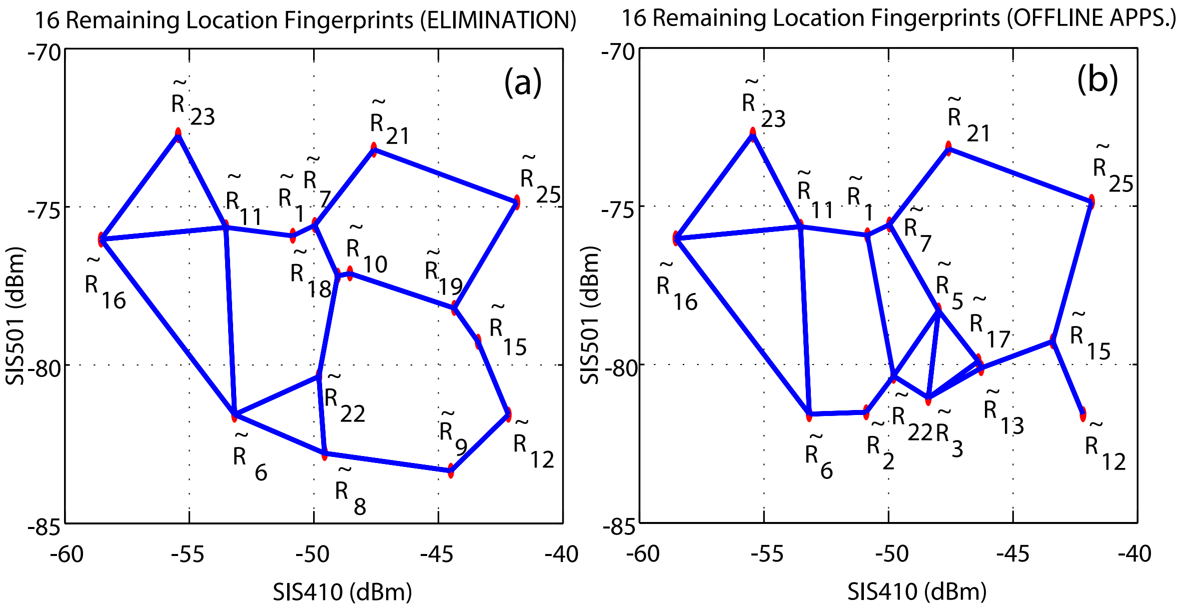

Figure 53: Comparison of GGs from (a) 1 meter grid with eliminated fingerprints and (b) 1-and-2 meter grid used during the offline phase

room. On the other hand, fingerprints with SIS410's RSS > -50 dBm are those mostly found inside the room as shown by fingerprints on the right half of both graphs. Some dissimilarity between the Gabriel Graphs occurs since it is difficult to get exactly the same "efficient" fingerprints just from the new grid system. In fact, fingerprints from grid locations such as 3, 5, and 13, although available with a 2-meter spacing, are ones being eliminated by the analytical model. However, we can also see some of the same fingerprints and their edges presented in both graphs (i.e., fingerprints $\tilde{R}_{15}$, $\tilde{R}_{23}$, and $\tilde{R}_{25}$). Therefore the difference of the probability of location estimation can be expected to be extremely small for these locations. The average CDF of the error distance, from simulations, using the offline guideline also shows a slight difference in the cumulative probability (less than 0.05 from the analytical elimination case), as seen from the black dashed line in Figure 54.

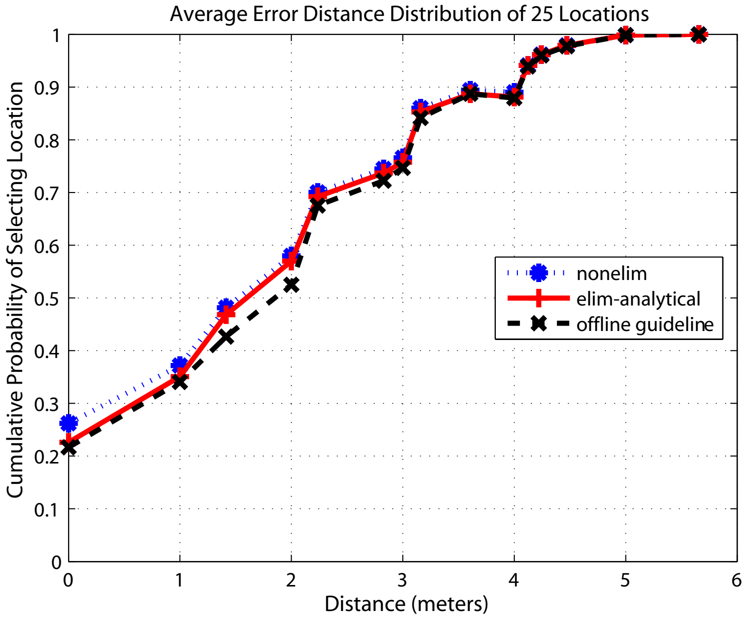

Figure 54: Comparison of Average CDFs of Error Distance from Measurement with the Offline guideline

As we can see, the offline guidelines have the potential to be useful to approximate Gabriel Graphs similar to the procedure with the analytical model. However, at this stage it is still necessary to require trial-and-error of measurements before determining grid points with only "good" fingerprints. However, as previously mentioned, further work to refine and *quantify* these guidelines is necessary and needed to research.

D. MISCELLANEOUS ISSUES

Besides issues discussed in previous sections, we believe that the proposed framework and the analytical model in this dissertation can be applied to extend its capability. For instance, we believe that the analytical model for predicting the probability of fingerprint selection can be flexible and applicable when we consider a grid system with fingerprints taken in different floors. Performance measurements of fingerprint system from multiple floors had been studied in the research literature [27]. By considering fingerprints from several floors as one set, a system designer can directly apply the analytical model and the fingerprint elimination procedure to a fingerprint set in order to identify good fingerprints and create an efficient radio map in a multi-floor building. To save on computational effort, the fingerprint clustering method, as discussed in Chapter V, can also be used in a multi-floor fingerprint system (that is likely to exhibit a cluster pattern). For instance, one might expect to see different fingerprint clusters that correspond to different floors of a building.

In addition, although we did not see much of the benefit with a self positioning system (i.e., system with a positioning algorithm implemented in a mobile device based on our simple computations), computational saving via clustering analysis may have an important role in a remote positioning system (i.e., a system with the positioning algorithm implemented at a server in an infrastructure). In such systems, saving on the number of operations required per user, for estimating location, could increase the total system capacity. In other words, the number of location estimation queries that a server can handle per unit time will increase. Moreover, if smaller operations results in a shorter delay in estimating a MS' current location during the online phase, it can provide a better possibility for deploying a fingerprinting-based system for mobility tracking purposes. For instance, if clustering results in a delay that is less than 0.5s, (as derived by the sum of RSS measurement and NNSS delay at the break-event point in the Figure 52), it is then possible to track a mobile handheld device if time taken for moving between two locations is higher than such a value.

In summary, we believe a system designer can apply the framework and the analytical model to obtain a balance among the desired accuracy, precision, fingerprint dimension, grid space, and cluster construction. Also, this work is not limited to only 802.11 based fingerprinting systems although it has been verified with such a system. We believe that the analytical model can be extended to fingerprints from a hybrid system (such as combined GSM and WiFi fingerprint system studied in [27]). Further future study could provide better understanding of application as well as limitations in different systems and environments.

VII. CONCLUSION

In this dissertation we have developed an analytical model for predicting accuracy and precision of indoor positioning systems based on location fingerprinting. The model utilizes analysis of the location fingerprint structure to help predict performance of the system and estimate a MS's location. We also proposed a framework that can be used for designing an efficient positioning system and its radio map.

We summarize research contribution and future research work as acquired from this thesis in the following sections.

A. CONTRIBUTIONS

Here is a list of the main contributions of the thesis that primarily facilitate the construction of fingerprint based indoor positioning systems.

I. This research provided a methodology to analyze the structure of location fingerprints using an analytical tool – the proximity graph. The model derived from the structure analysis, as we proposed in this research, enables computation of an approximate probability distribution of location selection. This has not been available before. The model also allows approximation of the performance without doing a complete measurement evaluation of the system.

II. Based on the analytical model, we provided a fingerprint elimination procedure as a way to identify fingerprints with less likely chances to be picked. The procedure gives a system designer the ability to create a positioning system that requires fewer computations with no significant impact on overall precision performance.

III. We enhanced the performance model to become more scalable. We introduce fingerprint clustering to reduce computational effort related to the modeling. We performed sensitivity analysis of performance modeling with different grid system properties to evaluate modeling systematically. We also extended the performance model to consider a more realistic situation such as non-identical received signal variations and provided impact study of using radio maps in a mobile device.

IV. We provided an example of a design guideline which will be useful for reducing the effort of fingerprint collection during the offline phase.

B. FUTURE RESEARCH WORK

In this section we identify the potential directions for future research.

I. Research on how to generalize performance modeling to incorporate different RSS distributions (e.g., the beta distribution) can be an interesting future direction. It is challenging to find a better analytical model, using a new kind of proximity graph, in order to provide closer approximations of the probability distribution of error distance.

II. This thesis provides a ground work for future study of refining and quantifying the offline guidelines for fingerprint collection. The idea of finding a proximity graph similar to one from the analytical model could be further explored in order to better pick grid points, potentially with “good” fingerprints. For instance, a combination of few site-survey measurements and an indoor radio propagation model may be used to help predict the fingerprint structure and corresponding proximity graph. Also, quantifying these guidelines may be possible by studying real measurements from many different building structures or settings. Finally, an implementation of a software tool or methodology that can recommend the number of APs and grid spacing that is suitable for given environment maybe useful for a system designer before deploying the system.

III. From an information theory point of view, it is also interesting to see if it is possible to quantify how much information is lost due to fingerprint elimination. Knowing this, one can estimate the overall performance of a modified system after some fingerprints are eliminated in exchange of reduced effort for fingerprint collection. An analogy to this is lossy data compression, where the final data is approximated from a larger size of the data, without losing significant information.

IV. To effectively utilize the effort for the offline fingerprint collection, a study of how to merge or share radio maps from different users or from different areas is another interesting future study. Sharing radio maps can be done in either a planned manner or opportunistically. Moreover, merging of radio maps in an overlapped area can be used to create and refine the analytical model. This issue is a challenging research problem.

APPENDIX A

DERIVATION OF THE *PEP* FOR FINGERPRINTS WITH DIFFERENT RSS VARIATIONS

Let $\mathcal{A}$ represent the square of the Euclidean distance between a sample RSS vector $R_i = [r_1, r_2, r_3, ..., r_N]$ and the true average RSS vector of the target (correct) location fingerprint $\tilde{R}_i = [\rho_{i1}, \rho_{i2}, ..., \rho_{iN}]$. Let $\mathcal{B}$ be the square of the Euclidean distance between the sample RSS vector R_i and the location fingerprint $\tilde{R}_k = [\rho_{k1}, \rho_{k2}, ..., \rho_{kN}]$ of another point k on the grid. Then $\{\mathcal{A} < \mathcal{B}\} = \{\mathcal{A} \leq \mathcal{B}\}$ is the event that the distance between the sample RSS vector and the correct location fingerprint is smaller than the distance between the sample RSS vector and the incorrect location fingerprint. The probability of this event,$\{\mathcal{A} \leq \mathcal{B}\}$, can be evaluated as follows:

$$\begin{aligned} \mathcal{A} &\leq \mathcal{B} \\ \Rightarrow \sum_{j=1}^{N}(r_j - \rho_{ij})^2 &\leq \sum_{j=1}^{N}(r_j - \rho_{kj})^2 \\ \Rightarrow \sum_{j=1}^{N}(r_j - \rho_{ij})^2 - \sum_{j=1}^{N}(r_j - \rho_{kj})^2 &\leq 0 \\ \Rightarrow 2\sum_{j=1}^{N} r_j\beta_{ij} + \sum_{j=1}^{N} \Gamma_{ij} &\leq 0, \end{aligned}$$

where $\Gamma_{ij} = (\rho_{ij}^2 - \rho_{kj}^2)$ and $\beta_{ij} = (\rho_{kj} - \rho_{ij})$. Note that, when all r_j are Gaussian, a comparison variable $C = 2\sum_{j=1}^{N} r_j\beta_{ij} + \sum_{j=1}^{N} \Gamma_{ij}$ is also a Gaussian variable with mean

$\mu_c = 2\sum_{j=1}^{N}\rho_{ij}\beta_{ij} + \sum_{j=1}^{N}\Gamma_{ij}$ and variance $\sigma_c^2 = \sum_{j=1}^{N}(2\beta_{ij}\sigma_{ij})^2$. The σ_{ij} is the standard deviation of the RSS from the j^{th} AP at grid point i. The probability $P\{C \leq 0\}$ is computed as follows:

$$\begin{aligned}
Pr\{C \leq 0\} &= \int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_c} e^{-\frac{(c-\mu_c)^2}{2\sigma_c^2}} dc \\
&= \frac{1}{2}\frac{2}{\sqrt{\pi}} \int_{-\infty}^{-\frac{\mu_c}{\sqrt{2}\sigma_c}} e^{-t^2} dt \\
&= \frac{1}{2} + \frac{1}{2}\text{erf}\left(\frac{-\mu_c}{\sqrt{2}\sigma_c}\right) = 1 - Q\left(\frac{-\mu_c}{\sigma_c}\right)
\end{aligned}$$

The pairwise error probability $PEP(\tilde{R}_i, \tilde{R}_k) = P\{C > 0\} = 1 - P\{C \leq 0\} = Q\left(\frac{-\mu_c}{\sigma_c}\right)$. Let sd_{ik} be the Euclidean distance between $\tilde{R}_i$ and $\tilde{R}_k$. Now since

$$\begin{aligned}
\mu_c &= 2\sum_{j=1}^{N}\rho_{ij}(\rho_{kj} - \rho_{ij}) + \sum_{j=1}^{N}(\rho_{ij}^2 - \rho_{kj}^2) \\
&= 2\sum_{j=1}^{N}\rho_{ij}\rho_{kj} - \sum_{j=1}^{N}\rho_{ij}^2 - \sum_{j=1}^{N}\rho_{kj}^2 = -\sum_{j=1}^{N}(\rho_{ij} - \rho_{kj})^2 = -sd_{ik}^2,
\end{aligned}$$

Therefore $PEP(\tilde{R}_i, \tilde{R}_k) = Q\left(\frac{-\mu_c}{\sigma_c}\right) = Q\left(\frac{sd_{ik}^2}{\sigma_c}\right) = Q\left(\frac{sd_{ik}^2}{2[\sum_{j=1}^{N}\beta_{ij}^2\sigma_{ij}^2]^{1/2}}\right)$.

APPENDIX B

THE DIFFERENCE OF CDFS FROM BEFORE AND AFTER FINGERPRINT ELIMINATION

Figure 55 shows the difference between the error cumulative probabilities (CDF) for each location before and after fingerprint-elimination (only for those locations that survive elimination in the Fourth Floor of the IS Building). It turns out that the improvement in the error probability can be seen at locations whose fingerprints are neighbors of eliminated fingerprints (i.e., fingerprints 8, 6, 22 after eliminating fingerprint 2). However, it is not quite clear from the fingerprint system how to determine the extent to which the probability difference will be at each location. However, we believe that the difference amount is a combined result from signal distance between eliminated and non-eliminated fingerprints and different RSS variations associated with fingerprints.

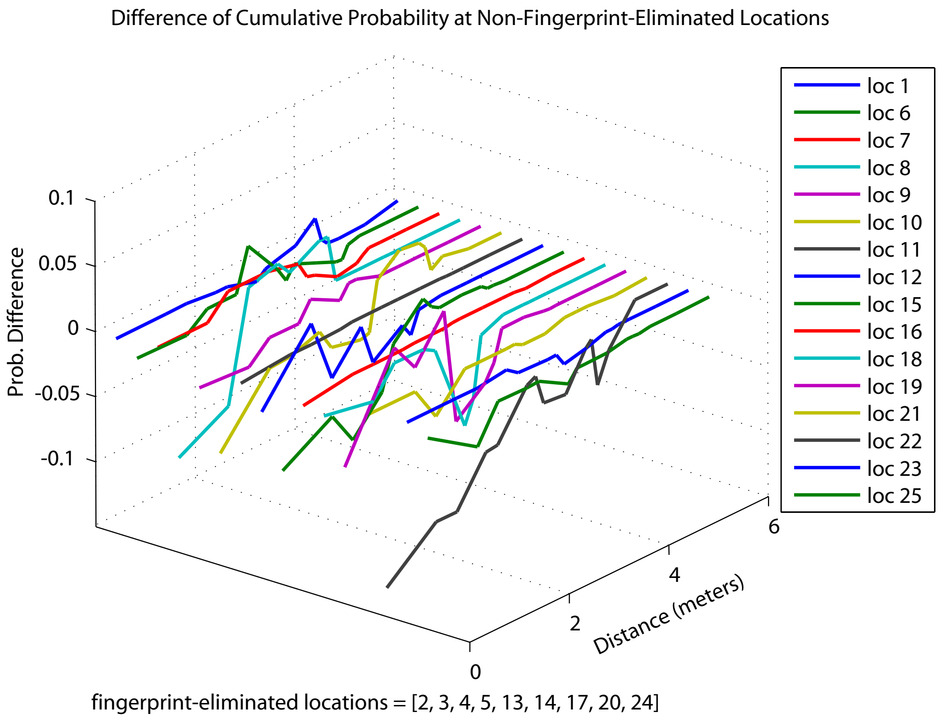

Figure 55: Difference between CDFs of Individual Location from Simulation: [Before - After]

APPENDIX C

DERIVATION OF THE POINT OF INTERSECTION BETWEEN DISTRIBUTIONS OF TWO GAUSSIAN VARIABLES

Assume one Gaussian variable has a mean m_1 and variance σ_1^2, and another Gaussian variable has a mean m_2 and variance σ_2^2. Let x be the intersection point(s) between the distributions of the two Gaussian variables. To solve for x, we need to solve the following equation:

$$\begin{aligned} \frac{1}{\sqrt{2\pi}\sigma_1} e^{-\frac{(x-m_1)^2}{2\sigma_1^2}} &= \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(x-m_2)^2}{2\sigma_2^2}} \\ \text{take natural log} \Rightarrow -ln\sigma_1 - \frac{(x-m_1)^2}{2\sigma_1^2} &= -ln\sigma_2 - \frac{(x-m_2)^2}{2\sigma_2^2} \\ \sigma_1^2(x-m_2)^2 - \sigma_2^2(x-m_1)^2 &= 2\sigma_1^2\sigma_2^2 ln\frac{\sigma_1}{\sigma_2}. \end{aligned}$$

We can rewrite above equation as $(\sigma_1^2 - \sigma_2^2)x^2 + 2(\sigma_2^2 m_1 - \sigma_1^2 m_2)x + (\sigma_1^2 m_2^2 - \sigma_2^2 m_1^2 + 2\sigma_1^2\sigma_2^2 ln\frac{\sigma_2}{\sigma_1}) = 0$. Let $A = \sigma_1^2 - \sigma_2^2$, $B = 2(\sigma_2^2 m_1 - \sigma_1^2 m_2)$, and $C = \sigma_1^2 m_2^2 - \sigma_2^2 m_1^2 + 2\sigma_1^2\sigma_2^2 ln\frac{\sigma_2}{\sigma_1}$. Then the equation can be expressed in a standard quadratic form as $Ax^2 + Bx + c = 0$. Therefore, x can be solved by $x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$. We know that

$$\begin{aligned} 2A &= 2(\sigma_1^2 - \sigma_2^2) \\ B^2 &= 4\sigma_2^4 m_1^2 - 4\sigma_1^2\sigma_2^2 m_1 m_2 + 4\sigma_1^4 m_2^2 \\ 4AC &= 4\sigma_1^4 m_2^2 - 4\sigma_1^2\sigma_2^2 m_1^2 + 8\sigma_1^4\sigma_2^2 ln\frac{\sigma_2}{\sigma_1} - 4\sigma_1^2\sigma_2^2 m_2^2 + 4\sigma_2^4 m_1^2 - 8\sigma_1^2\sigma_2^4 ln\frac{\sigma_2}{\sigma_1}. \end{aligned}$$

Therefore we have

$$\begin{aligned}
\Rightarrow B^2 - 4AC &= 4\sigma_1^2\sigma_2^2(m_1^2 + m_2^2) - 4\sigma_1^2\sigma_2^2 m_1 m_2 + 8\sigma_1^2\sigma_2^2(\sigma_2^2 - \sigma_1^2) ln\frac{\sigma_2}{\sigma_1} \\
&= 4\sigma_1^2\sigma_2^2[(m_1^2 + m_2^2) - 2m_1 m_2 + 2(\sigma_2^2 - \sigma_1^2) ln\frac{\sigma_2}{\sigma_1}] \\
&= 4\sigma_1^2\sigma_2^2[(m_2 - m_1)^2 + 2(\sigma_2^2 - \sigma_1^2) ln\frac{\sigma_2}{\sigma_1}].
\end{aligned}$$

And so x can be derived as

$$\begin{aligned}
x &= \frac{-2(\sigma_2^2 m_1 - \sigma_1^2 m_2) \pm 2\sigma_1\sigma_2\sqrt{(m_2 - m_1)^2 + 2(\sigma_2^2 - \sigma_1^2) ln\frac{\sigma_2}{\sigma_1}}}{2(\sigma_1^2 - \sigma_2^2)} \\
\Rightarrow x &= \frac{(\sigma_1^2 m_2 - \sigma_2^2 m_1) \pm \sigma_1\sigma_2\sqrt{(m_2 - m_1)^2 + 2(\sigma_2^2 - \sigma_1^2) ln\frac{\sigma_2}{\sigma_1}}}{(\sigma_1^2 - \sigma_2^2)}.
\end{aligned}$$

APPENDIX D

STANDARD DEVIATIONS FROM THE RSS MEASUREMENTS

Table 11: Standard Deviation (dB) at SIS building

Location	AP410	AP501
Loc1	2.740	1.329
Loc2	5.653	2.266
Loc3	5.007	1.244
Loc4	3.766	1.823
Loc5	5.585	1.449
Loc6	3.775	1.710
Loc7	3.276	1.010
Loc8	6.137	1.394
Loc9	3.875	0.937
Loc10	4.576	1.640
Loc11	1.473	1.129
Loc12	5.782	1.436
Loc13	5.040	1.165
Loc14	6.284	2.370
Loc15	4.680	1.234
Loc16	3.831	1.330
Loc17	3.875	0.971
Loc18	3.611	1.383
Loc19	5.003	1.052
Loc20	5.807	1.405
Loc21	2.606	1.467
Loc22	4.139	1.781
Loc23	4.012	1.216
Loc24	4.840	1.486
Loc25	3.528	1.113

Table 12: Standard Deviation (dB) from the Hillman Library

Location	hl2-b	hl4-a	hl4-b	hlg-a	hlg-b	hl2-a
L001	1.11	1.11	1.20	1.48	0.80	-
L002	1.67	1.01	1.26	2.42	0.93	-
L003	2.05	0.84	1.79	1.81	1.27	-
L004	2.06	0.91	1.70	1.97	0.86	-
L005	1.56	1.21	1.78	1.28	0.75	-
L006	1.23	1.23	1.31	1.84	0.90	-
L007	1.13	1.20	1.19	1.66	1.16	-
L008	1.39	1.14	1.33	1.43	0.60	-
L009	1.62	1.12	0.73	1.60	1.31	-
L010	1.43	1.12	1.35	1.59	1.54	-
L011	1.27	1.33	0.90	1.61	1.08	-
L012	1.54	1.84	-	1.27	1.07	-
L013	1.09	1.58	0.01	2.03	0.98	-
L014	2.11	1.48	-	2.43	0.80	1.69
L015	1.17	0.99	-	1.31	0.51	-
L016	1.43	1.59	-	0.99	0.81	1.40
L017	1.90	2.03	-	2.08	0.75	1.59
L018	1.16	1.52	-	1.44	0.95	2.11
L019	1.22	1.80	-	2.07	1.05	1.14
L020	1.10	1.42	-	1.36	1.00	1.16
L021	0.86	1.04	-	1.37	0.69	1.16
L022	1.20	1.88	-	1.68	0.83	1.27
L023	1.17	1.43	0.94	1.27	0.86	1.85
L024	1.20	1.15	1.18	1.53	0.92	1.34
L025	1.19	1.67	1.42	1.50	1.03	1.49
L026	1.57	1.36	1.55	1.24	1.51	1.09
L027	2.00	-	1.16	-	1.19	1.32
L028	1.17	-	1.61	0.65	1.82	1.47
L029	1.32	0.85	1.16	1.28	0.97	1.26
L030	2.36	-	1.25	1.25	1.81	1.39
L031	2.04	-	1.80	1.17	0.96	1.54
L032	1.77	-	1.47	0.84	2.15	1.63
L033	1.59	-	1.71	-	1.37	1.46
L034	1.83	-	1.28	-	-	-
L035	1.86	-	1.98	-	-	-
L036	1.13	-	1.39	-	-	-

Table 13: Standard Deviation (dB) from the Hillman Library (con't)

Location	hl2-b	hl4-a	hl4-b	hlg-a	hlg-b	hl2-a
L037	1.29	-	1.89	-	-	-
L038	1.39	1.38	1.23	-	-	-
L039	2.01	0.47	1.32	-	-	-
L040	1.88	-	1.95	-	-	-
L041	1.13	-	1.93	-	1.96	1.38
L042	1.12	0.59	1.43	-	1.37	1.22
L043	1.07	1.10	1.17	1.03	3.31	1.49
L044	1.58	0.73	1.53	1.07	1.07	1.17
L045	1.32	-	2.27	-	1.96	1.48
L046	1.71	-	0.99	-	1.61	1.14
L047	1.19	0.97	1.43	1.14	1.39	1.33
L048	1.75	1.59	0.91	1.53	2.16	1.68
L049	2.09	1.28	2.14	1.17	2.13	1.17
L050	1.74	1.57	1.63	1.07	1.63	1.35
L051	1.97	1.35	1.32	1.21	2.10	1.29
L052	1.62	1.29	1.31	1.05	2.66	1.86
L053	-	1.23	1.39	0.75	1.29	1.33
L054	-	1.56	1.38	1.06	2.48	1.34
L055	-	1.64	1.60	1.11	2.41	1.19
L056	-	0.96	1.23	-	1.49	1.20
L057	-	1.77	1.23	1.08	2.12	1.90
L058	-	1.53	1.24	1.37	1.19	1.11
L059	-	-	1.58	1.04	1.64	2.91
L060	-	-	1.60	-	1.42	1.66
L061	-	1.28	1.60	0.82	1.95	1.41
L062	-	1.92	0.96	1.27	1.68	1.33
L063	-	2.32	1.33	1.37	2.72	1.56
L064	-	2.50	0.93	1.59	2.27	1.58
L065	-	1.89	1.83	1.11	1.53	-
L066	-	1.61	1.31	1.40	2.15	-
L067	-	1.18	1.23	-	1.54	-
L068	-	1.48	1.50	1.07	1.35	-
L069	-	1.34	1.92	-	2.17	-
L070	-	1.41	1.06	-	2.67	-
L071	-	1.19	1.21	-	2.48	-

APPENDIX E

THE GABRIEL PROXIMITY GRAPHS FOR FINGERPRINTS IN SCENARIO 2

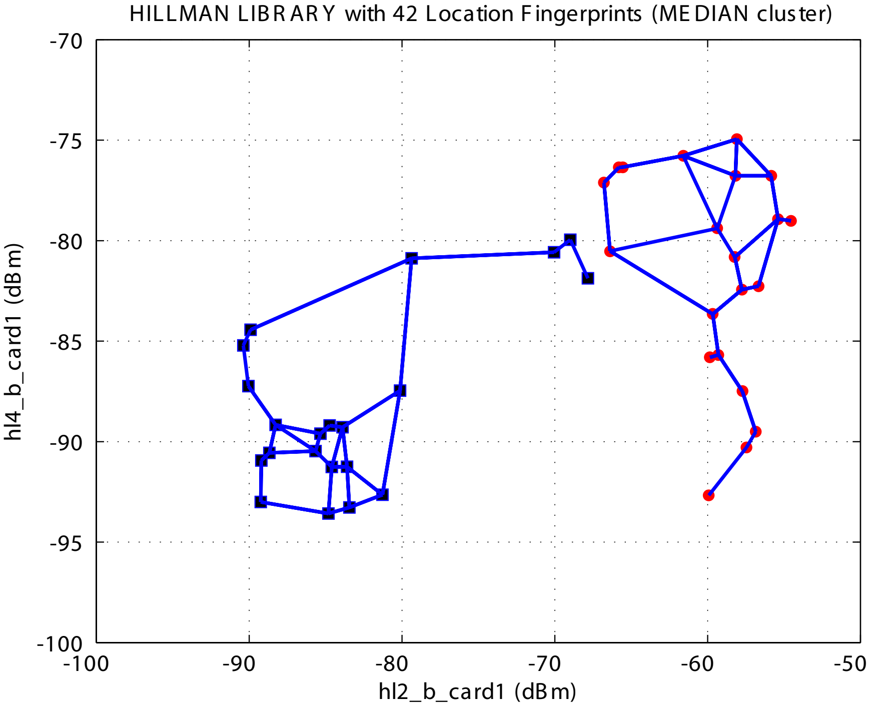

Figure 56: Two GGs with Median Clustering: Scenario 2

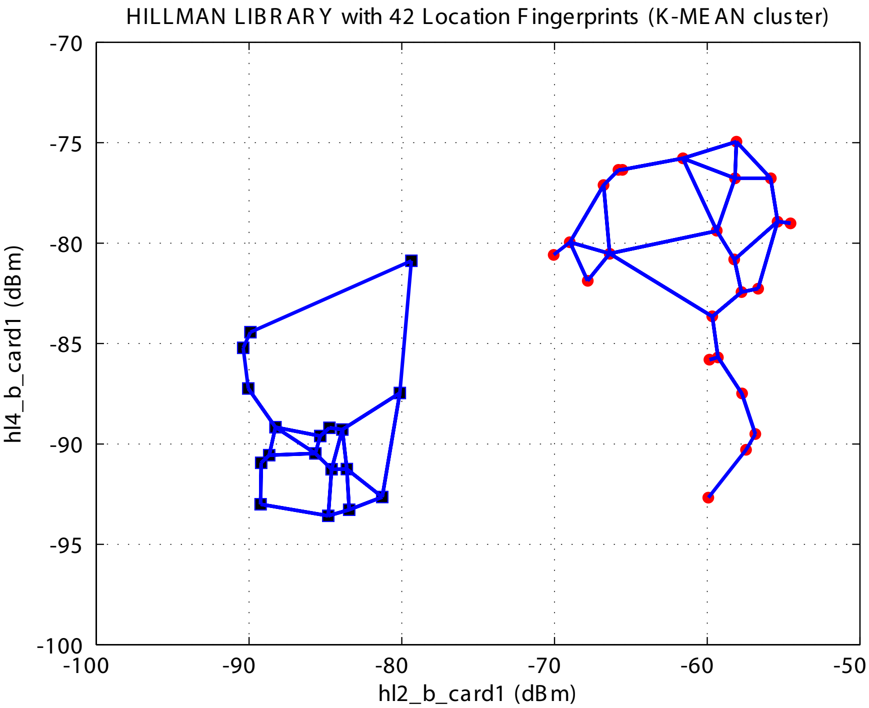

Figure 57: Two GGs with K-Mean Clustering: Scenario 2

APPENDIX F

THE SKEWED GABRIEL GRAPH STUDY FOR FINGERPRINTS IN SCENARIO 2

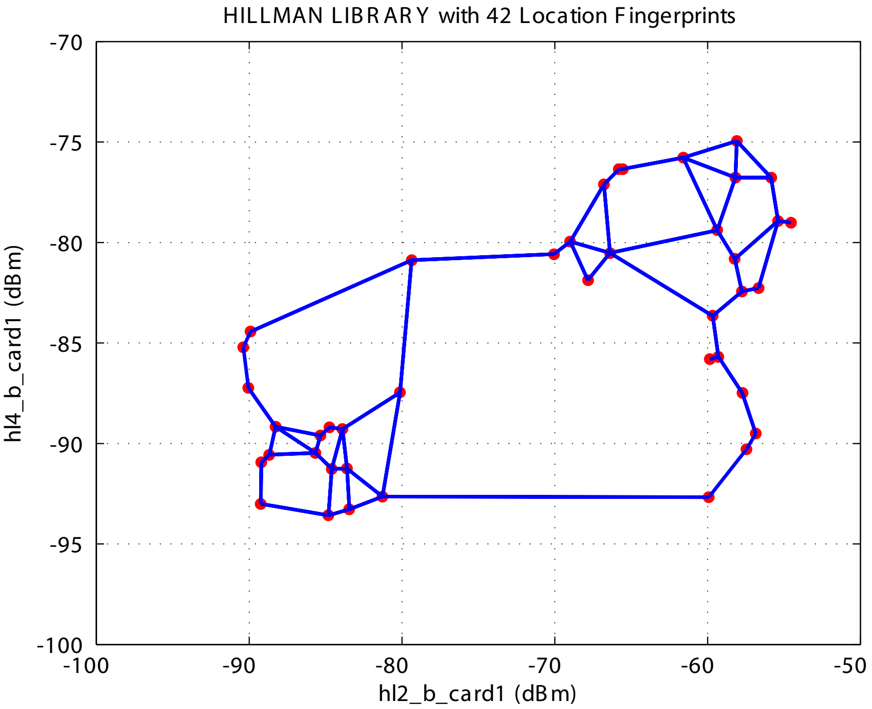

Figure 58: GG of Measured Radio Map: Scenario 2

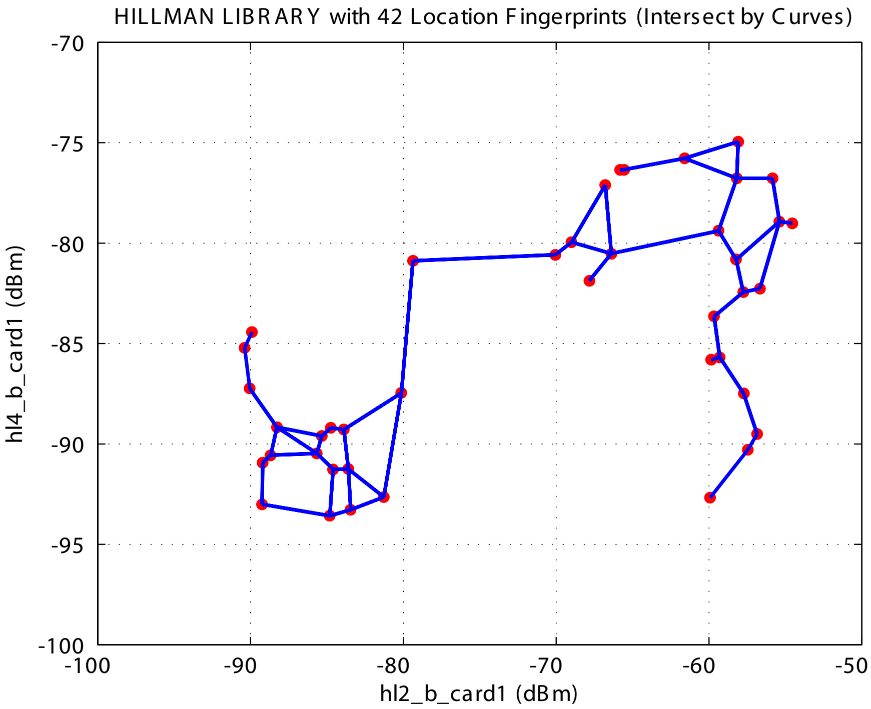

Figure 59: Skewed GG of Measured Radio Map: Scenario 2

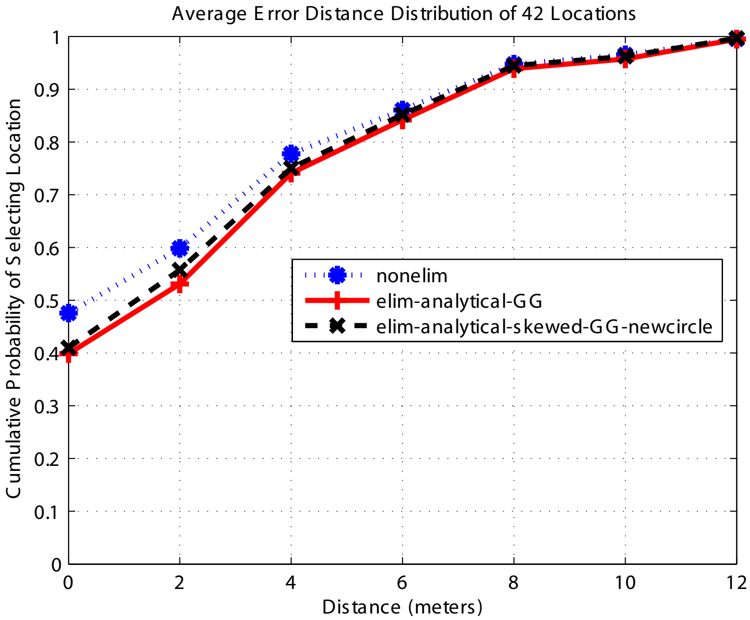

Figure 60: The Average CDF of Error Distance with a Skewed GG: Scenario 2

BIBLIOGRAPHY

[1] K. Kaemarungsi, “Design of Indoor Positioning Systems based on Location Fingerprinting Technique,” Ph.D. dissertation, Univ. of Pittsburgh, Pittsburgh, Feb. 2005.

[2] *Microsoft Location Finder*, Microsoft Download Center, Microsoft, Inc., 2006. [Online]. Available: http://www.microsoft.com/downloads

[3] G. M. Djuknic and R. E. Richton, “Geolocation and Assisted GPS,” *IEEE Computer*, vol. 34, no. 2, pp. 123–125, Feb. 2001.

[4] *Ekahau Positioning Engine 2.1*, User Guide, Ekahau, Inc., 2003. [Online]. Available: http://www.ekahau.com

[5] *The Wi-Fi Positioning System (WPS)*, Online Homepage, Skyhook Wireless, Inc., 2003. [Online]. Available: http://www.skyhookwireless.com

[6] P. Krishnamurthy, “Position Location in Mobile Environments,” in *Proc. NSF Workshop on Context-Aware Mobile Database Management (CAMM)*, Providence, RI, Jan. 2002.

[7] A. M. Ladd, K. E. Bekris, G. Marceau, A. Rudys, L. E. Kavraki, and D. S. Wallach, “Robotics-Based Location Sensing using Wireless Ethernet,” in *Proc. ACM International Conference on Mobile Computing and Networking (MOBICOM’02)*, Sept. 2002.

[8] P. Bahl and V. N. Padmanabhan, “RADAR: An In-building RF-based User Location and Tracking System,” in *Proc. IEEE Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM’00)*, Tel Aviv, Israel, Mar. 2000.

[9] K. Kaemarungsi and P. Krishnamurthy, “Modeling of Indoor Positioning Systems Based on Location Fingerprinting,” in *Proc. IEEE INFOCOM*, Mar. 2004.

[10] P. Castro, P. Chiu, T. Kremenek, and R. R. Muntz, “A Probabilistic Room Location Service for Wireless Networked Environments,” in *Proc. Ubiquitous Computing*, Oct. 2001.

[11] M. Brunato and R. Battiti, “Statistical Learning Theory for Location Fingerprinting in Wireless LANs,” *Computer Networks*, vol. 47, no. 6, pp. 825–845, 2005.

[12] J. A. Tauber, “Indoor Location Systems for Pervasive Computing,” Area Exam Report, Massachusetts Institute of Technology, Aug. 2002.

[13] K. W. Kolodziej and J. Hjelm, *Local Positioning Systems: LBS Applications and Services*, 1st ed. Boca Raton,FL: CRC Press, Taylor & Francis Group, 2006.

[14] S. Saha, K. Chaudhuri, D. Sanghi, and P. Bhagwat, “Location Determination of a Mobile Device Using IEEE 802.11b Access Point Signals,” in *Proc. IEEE Wireless Communications and Networking Conference (WCNC’03)*, New Orleans, LA, Mar. 2003.

[15] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas, and J. Sievanen, “A Probabilistic Approach to WLAN User Location Estimation,” *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, July 2002.

[16] M. A. Youssef, A. Agrawala, and A. U. Shankar, “WLAN Location Determination via Clustering and Probability Distributions,” in *Proc. IEEE International Conference on Pervasive Computing and Communications (PerCom’03)*, Dallas-Fort Worth, TX, Mar. 2003.

[17] J. Small, A. Smailagic, and D. P. Siewiorek, “Determining User Location for Context Aware Computing Through the Use of a Wireless LAN Infrastructure,” Online, Dec. 2000. [Online]. Available: http://www-2.cs.cmu.edu/~aura/docdir/small00.pdf

[18] K. Pahlavan, X. Li, and J. P. Makela, “Indoor Geolocation Science and Technology,” *IEEE Commun. Mag.*, vol. 40, no. 2, pp. 112–118, Feb. 2002.

[19] R. Battiti, M. Brunato, and A. Villani, “Statistical Learning Theory for Location Fingerprinting in Wireless LANs,” Technical Report, Oct. 2002. [Online]. Available: http://rtm.science.unitn.it/~battiti/archive/86.pdf

[20] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 1st ed. Upper Saddle River, NJ: Prentice Hall PTR, 1996.

[21] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification.* Wiley-Interscience Publication, November 2000.

[22] P. Prasithsangaree, P. Krishnamurthy, and P. K. Chrysanthis, “On Indoor Position Location with Wireless LANs,” in *Proc. IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC’02)*, Lisbon, Portugal, Sept. 2002.

[23] K. Kaemarungsi, “Efficient Design of Indoor Positioning Systems Based on Location Fingerprinting,” in *IEEE International Workshop Mobility Management and Wirelss Access (MobiWac)*, vol. 1, June 2005.

[24] R. Battiti, “Location-aware Computing: a Neural Network Model for Determining Location in Wireless LANs,” Universita degli Studi di Trento, Tech. Rep. DIT-0083, Feb. 2002. [Online]. Available: http://rtm.science.unitn.it/~battiti/archive/83.pdf

[25] Z. Xiang, S. Song, J. Chen, H. Wang, J. Huang, and X. Gao, "A Wireless LAN-based Indoor Positioning Technology," *IBM Journal of Research and Development*, vol. 48, no. 5/6, pp. 617–626, Sept./Nov. 2004.

[26] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "COMPASS: A Probabilistic Indoor Positioning System Based on 802.11 and Digital Compasses," in *Proc. ACM International Workshop on Wireless Network Testbeds,Experimental Evaluation and Characterization*, Sept. 2006.

[27] V. Otsason, A. Varshavsky, A. LaMarca, and E. de Lara, "Accurate GSM Indoor Localization," in *7th Proc. International Conference, UbiComp*, ser. LNCS., Springer (2005), vol. 3660, Sept. 2005.

[28] T. S. Rappaport, J. H. Reed, and B. D. Woerner, "Position Location Using Wireless Communications on Highways of the Future," *IEEE Commun. Mag.*, vol. 34, no. 10, pp. 33–41, Oct. 1996.

[29] B. Li, Y. Wang, H. K. Lee, A. Dempster, and C. Rizos, "Method for Yielding a Database of Location Fingerprints in WLAN," in *Proc. IEE Communications*, vol. 152, no. 5, Oct. 2005.

[30] M. Wallbaum, "A Priori Error Estimates for Wireless Local Area Network Positioning Systems," *Pervasive and Mobile Computing*, vol. 3, no. 5, pp. 560–580, 2007.

[31] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed. NYC: Wiley, 2000.

[32] S. Arya and D. M. Mount, "Computational Geometry: Proximity and Location," in *Handbook of Data Structures and Applications*, D. P. Mehta and S. Sahni, Eds. Chapman & Hall/CRC, 2005, ch. 63, pp. 1–22.

[33] V. Klee, "On the Complexity of D-dimensional Voronoi Diagrams," in *Archiv der Mathematik*, vol. 34, no. 1. Birkhuser Basel, Dec. 1980, pp. 75–80.

[34] F. Aurenhammer and R. Klein, "Voronoi Diagrams," F. Aurenhammer and R. Klein, Voronoi Diagrams, in: J.R. Sack and G. Urrutia (eds.), Handbook on Computational Geometry, Elsevier, 1996.

[35] R. Rajaraman, "Topology Control and Routing in Ad hoc Networks: A Survey," *SIGACT NEWS*, vol. 33, no. 2, pp. 60–73, 2002.

[36] F. Aurenhammer, "Voronoi Diagrams-A Survey of a Fundamental Geometric Data Structure," *ACM Computing Surveys*, vol. 23, no. 3, pp. 345–405, Sept. 1991.

[37] B. Bhattacharya, R. Poulsen, and G. Toussaint, "Application of Proximity Graphs to Editing Nearest Neighbor Decision Rules," in *The Int'l Sym. on Information Theory*, Santa Monica, 1981.

[38] K. Kaemarungsi and P. Krishnarmurthy, "Properties of Indoor Received Signal Strength for WLAN Location Fingerprinting," in *Proc. IEEE First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MOBIQUITOUS'04)*, Boston, MA, Aug. 2004.

[39] G. J. M. Janssen and R. Prasad, "Propagation Measurements in an Indoor Radio Environment at 2.4 GHz, 4.75 GHz and 11.5 GHz," in *Proc. IEEE Vehicular Technology Conference (VTC'92)*, Denver, CO, May 1992.

[40] S. Y. Seidel and T. S. Rappaport, "914 MHz Path Loss Prediction Models for Indoor Wireless Communications in Multifloored Buildings," *IEEE Trans. Antennas Propagat.*, vol. 40, no. 2, Feb. 1992.

[41] Y. Cheng, "Mean Shift, Mode Seeking, and Clustering," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, no. 8, pp. 790–799, Aug. 1995.

[42] K. Bury, *Statistical Distributions in Engineering.* Cambridge University Press, 1998.

[43] *SAS software*, SAS Institute Inc., Cary, NC USA. [Online]. Available: http://www.sas.com

[44] A. Roy, S. Bhaumik, A. Bhattacharya, K. Basu, D. Cook, and S. Das, "Location Aware Resource Management in Smart Homes," in *Proc. IEEE PERCOM*, 2003.

[45] Y. Yamada, D. Connors, and W. Hwu, "A Software-Oriented Floating-Point Format for Enhancing Automotive Control Systems," in *Workshop on Compiler and Architecture Support for Embedded Computing Systems (CASES98)*, Dec. 1998.