

ABSTRACT

Title of Dissertation: HORUS: A WLAN-BASED INDOOR
 LOCATION DETERMINATION SYSTEM

Moustafa Amin Abdel Azim Yousief Abdel Rehim, Doctor of Philosophy, 2004

Dissertation directed by: Professor Ashok K. Agrawala
 Department of Computer Science

As ubiquitous computing becomes more popular, the need for context-aware applications increases. The context of an application refers to the information that is part of its operating environment. Typically this includes information such as location, activity of people, and the state of other devices. Algorithms and techniques that allow an application to be aware of the location of a device on a map of the environment are a prerequisite for many of these applications.

Many systems over the years have tackled the problem of determining and tracking user position. Examples include GPS, wide-area cellular-based systems, infrared-based systems, magnetic tracking systems, various computer vision systems, physical contact systems, and radio frequency (RF) based systems. Of these, the class of RF-based systems that use an underlying wireless data network, such as the IEEE 802.11 wireless network, to estimate user location has gained attention recently, especially

for indoor applications. RF-based techniques provide more ubiquitous coverage than other indoor location determination systems and do not require additional hardware for user location determination, thereby enhancing the value of the wireless data network.

However, using a wireless network for location determination has the challenge of dealing with the noisy characteristics of the wireless channel. Current location determination techniques for the 802.11 wireless networks suffer from these noisy characteristics, leading to coarse grained accuracy. A key feature of current techniques is the dependence on building a radio map for the area of interest and using this radio map to infer the user location. Using the radio map to infer the user location is a computationally intensive process and may consume the scarce energy resource of the mobile units.

The *Horus* system is concerned with developing accurate methods for determining the user location with low computation requirements. Our goal is to build a location determination system that is capable of determining the user position with high accuracy, is light enough to be implemented on energy-constrained devices such as handheld computers, and is scalable to track a large number of users and to be used with large areas.

We identify different causes of the wireless channel variations and we develop techniques to handle these variations. The results show that the *Horus* system is able to achieve accuracy significantly better than the current WLAN location determination systems. Moreover, the number of operations required to run the algorithm is better than the current systems with more than an order of magnitude.

HORUS: A WLAN-BASED INDOOR
LOCATION DETERMINATION SYSTEM

by

Moustafa Amin Abdel Azim Yousief Abdel Rehim

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2004

Advisory Committee:

Professor Ashok K. Agrawala, Chair/Advisor
Professor Raymond E. Miller
Professor A. Udaya Shankar
Associate Professor Amitabh Varshney
Associate Professor Steven Tretter

© Copyright by

Moustafa Amin Abdel Azim Yousief Abdel Rehim

2004

DEDICATION

To my parents, Nagia and Farah

ACKNOWLEDGEMENTS

I am indebted to my advisor Dr. Ashok Agrawala for his continuous guidance, support and encouragement.

Special thanks are due to Dr. Udaya Shankar for this advice and support.

I gratefully acknowledge the support of the computer science department family especially the MIND Lab group.

TABLE OF CONTENTS

List of Tables	viii
List of Figures	ix
1 Introduction	1
1.1 Approach	6
1.2 Contribution	8
1.3 Organization	9
2 Related Work	11
2.1 Location Determination Technologies	11
2.2 WLAN Location Determination Systems	15
2.2.1 Ad-hoc mode based techniques	15
2.2.2 Infrastructure mode based techniques	16
3 Wireless Channel Characteristics	20
3.1 Sampling Process	20
3.2 Noisy Characteristics of the Wireless Channel	22
3.2.1 Temporal variations	22
3.2.2 Spatial characteristics	26
3.3 Experimental Testbeds	28

3.3.1	Testbed 1	28
3.3.2	Testbed 2	28
3.4	Summary	29
4	Problem Formulation	31
4.1	Mathematical Model	31
4.2	Non-Parametric Signal Strength Distributions	32
4.2.1	Training Phase	32
4.2.2	Location Determination Phase	33
4.3	Parametric Signal Strength Distributions	35
4.3.1	Training Phase	36
4.3.2	Location Determination Phase	37
4.4	Analysis	39
4.4.1	Average Distance Error	39
4.4.2	Probability of Error	40
4.4.3	Optimality	41
4.5	Experimental Results	45
4.6	Discussion	45
4.7	Summary	47
5	Handling Samples Correlation	48
5.1	Background	48
5.2	Autoregressive Model	49
5.2.1	Relation Between the Mean of s_t and v_t	50
5.2.2	Relation Between the Variance of s_t and v_t	50
5.2.3	Relation Between s_t and s_0	51

5.3	Estimating the Value of α	51
5.4	Distribution of the Average of n Correlated Samples	52
5.4.1	Mean of the Distribution of the Average of n Samples	53
5.4.2	Variance of the Distribution of the Average of n Samples	53
5.5	Modified <i>Horus</i> Algorithm	55
5.6	Experimental Evaluation	55
5.7	Summary	57
6	Continuous-Space Estimation	60
6.1	Technique 1: Center of Mass of the Top Candidate Locations	60
6.2	Technique 2: Time-Averaging in the Physical Space	63
6.3	Experimental Evaluation	63
6.3.1	Center of Mass Technique	63
6.3.2	Time-averaging Technique	65
6.4	Summary	65
7	Radio-Map Locations Clustering	68
7.1	Introduction	68
7.2	Explicit Clustering	69
7.3	Implicit Clustering	72
7.4	Discussion	73
7.5	Experimental Results	75
7.5.1	Joint Clustering Technique	75
7.5.2	Incremental Triangulation Technique	77
7.5.3	Discussion	77
7.6	Summary	80

8	Small-Scale Compensation	81
8.1	Small-Scale Variations	81
8.2	The Perturbation Technique	83
8.2.1	Detecting Small Scale Variations	83
8.2.2	Compensating for Small Scale Variations	84
8.3	Experimental Evaluation	84
8.4	Summary	85
9	Performance Evaluation	89
9.1	Comparison with the Radar System	89
9.2	Effect of the Training Parameters	93
9.2.1	Training time	93
9.2.2	Radio-map spacing	95
9.3	Applying <i>Horus</i> Techniques to the <i>Radar</i> System	96
9.4	Summary	98
10	Conclusions and Future Work	99

LIST OF TABLES

9.1	Estimation parameters for the first testbed	90
9.2	Estimation parameters for the second testbed	90
9.3	Comparison of the <i>Horus</i> system and the <i>Radar</i> system for the first testbed	91
9.4	Comparison of the <i>Horus</i> system and the <i>Radar</i> system for the second testbed	92

LIST OF FIGURES

1.1	Relation between signal strength and distance.	4
1.2	<i>Horus</i> system components.	7
2.1	A taxonomy of WLAN location determination systems.	16
3.1	Active scanning process in the 802.11 standard.	21
3.2	An example of the normalized signal strength histogram from an access point.	23
3.3	An example of the autocorrelation between samples from an access point.	24
3.4	Signal strength received from an access point over time.	24
3.5	Relation between the average signal strength of an access point and the percentage of samples received from it during a 5-minute interval.	25
3.6	Large-scale variations: Average signal strength over distance.	26
3.7	Small-scale variations: Signal strength contours	27
3.8	Floor plan for the first testbed.	29
3.9	Floor plan for the second testbed.	30
4.1	An example of the signal strength normalized histogram from an access point.	34

4.2	An example of fitting a Gaussian distribution to the signal strength normalized histogram from an access point.	37
4.3	Expected error for the special case of two locations	44
4.4	Performance of the basic algorithm of the <i>Horus</i> system for the first testbed testbed.	46
4.5	Performance of the basic algorithm of the <i>Horus</i> system for the second testbed.	46
5.1	Ratio between the variances of the averaging process and the original process for different values of α and n	54
5.2	Average distance error with and without taking correlation into account for the first testbed.	58
5.3	Average distance error with and without taking correlation into account for the second testbed.	58
6.1	An example of the center of mass technique.	62
6.2	Average distance error using the center of mass technique for the first testbed.	66
6.3	Average distance error using the center of mass technique for the second testbed.	66
6.4	Average distance error using the time-averaging technique for the first testbed.	67
6.5	Average distance error using the time-averaging technique for the second testbed.	67
7.1	Effect of the parameter q on the performance of the clustering process for the first testbed.	76

7.2	Effect of the parameter q on the performance of the clustering process for the second testbed.	76
7.3	Effect of the parameter q on the average distance error for the two testbeds.	78
7.4	Effect of the parameter q on the average number of operations per location estimate for the two testbeds.	78
7.5	Effect of the parameter <i>Threshold</i> on the average distance error for the two testbeds.	79
7.6	Effect of the parameter <i>Threshold</i> on the average number of operations per location estimate for the two testbeds.	79
8.1	Small-scale variations: signal strength contours for different access points.	82
8.2	Effect of changing the perturbation percentage on average distance error.	86
8.3	Effect of changing the number of perturbed access points on average distance error.	86
8.4	CDF for the distance error for the first testbed.	87
8.5	CDF for the distance error for the second testbed.	87
8.6	Effect of changing the number of perturbed access points on computational requirements.	88
9.1	Performance of the <i>Horus</i> system and the <i>Radar</i> system for the first testbed.	91
9.2	Performance of the <i>Horus</i> system and the <i>Radar</i> system for the second testbed.	92

9.3	Average number of operations per location estimate for the <i>Horus</i> system and the <i>Radar</i> system for the two testbeds.	93
9.4	Effect of the training time on the performance of the <i>Horus</i> system for the first testbed.	94
9.5	Effect of the training time on the performance of the <i>Horus</i> system for the second testbed.	94
9.6	Effect of the radio map spacing on the performance of the <i>Horus</i> system for the first testbed.	95
9.7	Effect of the radio map spacing on the performance of the <i>Horus</i> system for the second testbed.	96
9.8	Effect of applying the <i>Horus</i> system's techniques to the <i>Radar</i> system for the first testbed.	97
9.9	Effect of applying the <i>Horus</i> system's techniques to the <i>Radar</i> system for the second testbed.	97

Chapter 1

Introduction

As ubiquitous computing becomes more popular, the need for context-aware applications increases. The context of an application refers to the information that is part of its operating environment. Typically this includes information such as location, activity of people, and the state of other devices [1]. Algorithms and techniques that allow an application to be aware of the location of a device on a map of the environment are a prerequisite for many of these applications. Examples of location-aware applications [2–6] include location-sensitive content delivery, where tailored information is sent to the user based on his current location, direction finding, asset tracking, and emergency notification.

The growing need for location support systems underscores the importance of addressing location-awareness problem. For example, government initiatives require that cellular phone providers should develop a way to locate any phone that makes an emergency call [7]. In outdoor settings, the Global Positioning System (GPS) [9] has been used in many commercial applications, as in the case of locating automobiles. Despite the extraordinary advances in GPS technology, many indoor spaces cannot reliably receive GPS signals. An indoor system must use different sensors, such as infrared [10, 11], computer vision [12–14], physical contact [15], ultrasonic [16–18],

or radio frequency (RF) [19–42].

The class of RF-based systems that use an underlying wireless data network [21–42], such as the IEEE 802.11 wireless network [43], to estimate the user location has gained attention recently, especially for indoor applications. Many mobile devices and many buildings, both commercial and residential, are already equipped with off-the-shelf IEEE 802.11b wireless Ethernet. Furthermore, most wireless Ethernet devices already measure signal strength of the received packets as part of their standard operation and the signal strength varies noticeably as the distance and obstacles between wireless nodes change. If an accurate localization system could be developed using only this technology, then many existing systems could be retrofitted in software and new systems could be deployed using readily available parts.

However, using a wireless network for location determination has the challenge of dealing with the noisy characteristics of the wireless channel. The IEEE 802.11b standard uses radio frequencies in the 2.4 GHz band, which is attractive as it is license-free in most places around the world. The available adapters are based on spread spectrum radio technology, where the information signal is spread over several frequencies [44], so interference on a single frequency does not block the signal. The main problem is that an accurate prediction of the signal strength in every position of the environment is a very complex and difficult task because the signal propagates in many unpredictable ways [45]. The received signal is further corrupted by unwanted random effects such as noise, interference from other sources and interference between channels.

As waves propagate through an environment, the environment scatters the waves in a variety of different ways. Reflection, absorption, and diffraction occur when the waves encounter opaque obstacles; refraction occurs when the waves encounter translucent obstacles. Scattered waves can either decrease or increase the signal

strength at the reception point. Changes in atmospheric conditions like air temperature can also affect the propagation of waves and the resulting signal strengths. Unfortunately, 2.4 GHz is a resonant frequency of water, so people absorb radio waves in the 2.4 GHz frequency band.

Interference occurs when another radio frequency source generates a signal at the same frequency that is of comparable or higher strength than the transmitted signal, as measured by the recipient. The interfering device does not need to be a radio based transmission device [44, 46]. In the 2.4 GHz frequency band, microwave ovens, Bluetooth devices, 2.4 GHz cordless phones and welding equipment can be sources of interference. Due to reflection, refraction, diffraction, and absorption of radio waves by structures and people inside a building, the transmitted signal often reaches the receiver by more than one path, resulting in a phenomenon known as multi-path fading [47]. The signal components arriving from indirect paths and the direct path, if this exists, combine and produce a distorted version of the transmitted signal. Multi-path fading is the main cause of small-scale variations where a small change in the position of the receiver (order of wavelength) may lead to a significant change in the received signal strength [44].

These difficulties are particularly acute when operating indoors. Since there is rarely a line of sight between the transmitter and the receiver, the received signal is a sum of components that are often caused by some combination of the previously described phenomena. The received signal varies with respect to time and especially with respect to the relative position of the receiver and the transmitter.

Moreover, successive signal strength samples from the same access point are highly correlated. Therefore, a technique that uses multiple samples from the same access point to enhance the accuracy has to take this high correlation into account.

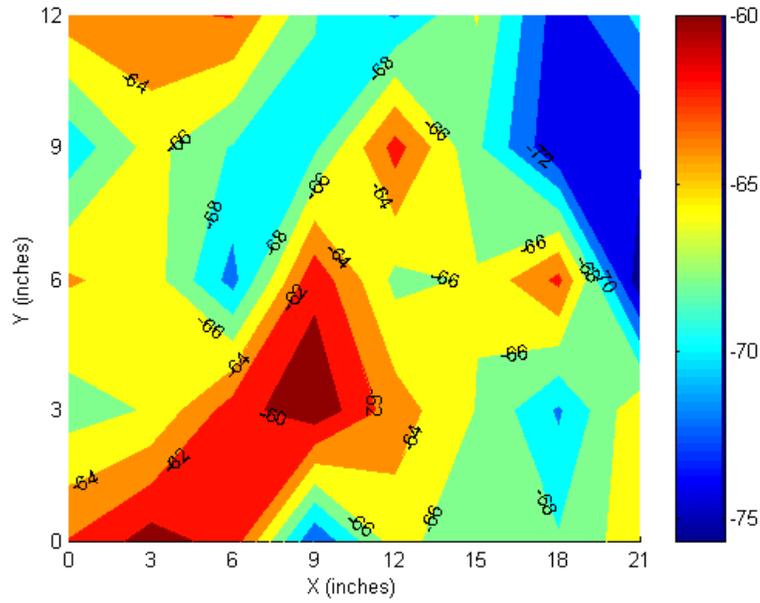


Figure 1.1: Relation between signal strength and distance.

Current location determination techniques for the 802.11 wireless networks suffer from these noisy characteristics, leading to coarse grained accuracy. Figure 1.1 shows the relation between the signal strength and distance in an area of 12×21 square inches.

As a result of the noisy wireless channel, it is difficult to capture the relation between the signal strength and distance, in an indoor environment, using a simple analytical function. Instead, indoor WLAN location determination systems capture the signature of different access points at selected locations in the area of interest. The collection of these signatures have been called in literature the *Radio Map*. Therefore, radio map-based WLAN location determination systems work in two phases: *offline* training phase, in which the radio map is constructed, and *online* location determination phase, in which the signal strength samples received from the access points are used to “search” the radio map to estimate the user location.

The location determination system can be implemented on a centralized server or on the mobile devices. A centralized implementation has the advantage of reducing the computational demand on the mobile device. However, implementing the location determination system on the mobile device has important advantages:

- *Scalability.* A centralized implementation leads to a single bottleneck and thus limits the number of mobile devices that can be tracked.
- *User privacy.* If the user wants to know his location without allowing the system to track him, the location determination code must be run at the mobile unit.

Because mobile devices are energy-constrained, it is important to reduce the computational requirement of the location determination system.

To summarize, an indoor WLAN location determination system should achieve the following goals:

- *High accuracy.* This is the main goal of any location determination system. Although the desired accuracy is dependent on the application in use, the higher the accuracy, the wider the range of applications it can support.
- *Low computational overhead.* High accuracy should not come at an increased computational cost, especially if the location determination system is to be implemented on an energy-constrained mobile device.
- *Flexible design.* Typically, a location determination system would be implemented on different hardware architectures. A flexible design allows porting the system between these architectures with minimum modifications.
- *Scalability.* The location determination system should be scalable to large areas and large number of users.

In this thesis, we describe the design and implementation of the *Horus* WLAN location determination system that achieves these goals.

1.1 Approach

The *Horus* system uses the RF signal strength as measured by an IEEE 802.11 wireless Ethernet card communicating with standard base stations. Since the signal strength information is provided during the normal card operation, the *Horus* system is a pure software solution on top of the WLAN infrastructure.

The system is based on identifying different causes of variations of the signal strength and developing techniques to handle them. Figure 1.2 shows the components of the *Horus* system. The system uses the signal strength information returned from different access points to infer the user location and provides an API for the user applications to use the system functionality.

The system works in two phases:

1. *Offline phase*: to build the radio map, cluster radio map locations, and do other preprocessing of the signal strength models.
2. *Online Phase*: to estimate the user location based on the received signal strength from each access point and the radio map prepared in the offline phase.

The radio map stores the distribution of signal strength received from each access point at each location. There are two modes for operation of the *Horus* system: one uses non-parametric distributions and the other uses parametric distributions.

The *Radio Map Builder* module is used to construct the radio map. It is described in Chapter 4.

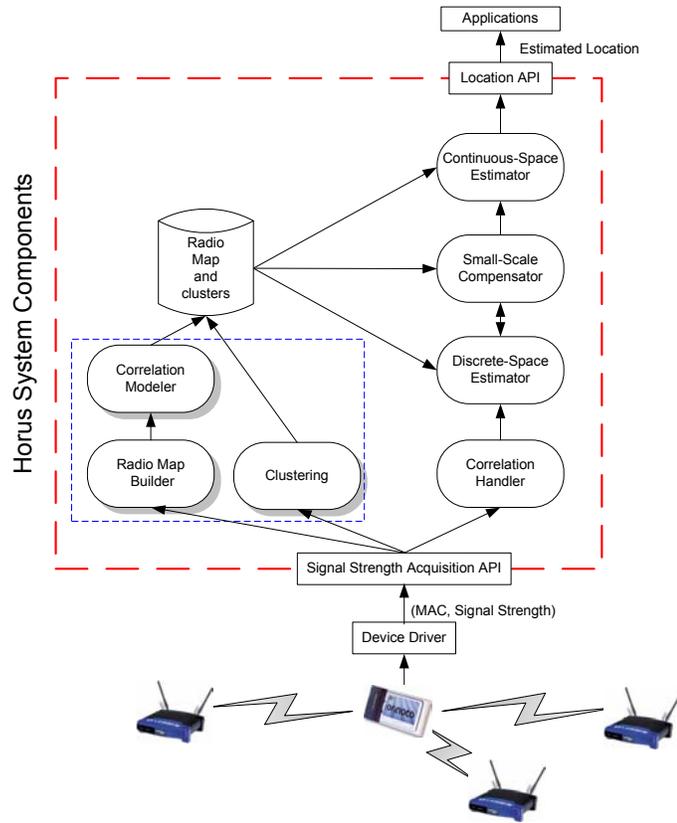


Figure 1.2: *Horus* system components: the arrows show information flow in the system. Shaded block represent modules used during the offline phase.

The *Discrete Space Estimator* module returns the radio map location that has the maximum probability given the received signal strength vector from different access points. The *Discrete Space Estimator* module is discussed in Chapter 4.

The *Correlation Modeler and Handler* modules are used to handle the correlation between successive samples from the same access point. The *Correlation Modeler and Handler* modules are discussed in Chapter 5.

The *Continuous Space Estimator* takes as an input the discrete estimated user location, one of the radio map locations, and returns a more accurate estimate of the user location in the continuous space. The *Continuous Space Estimator* module is

discussed in Chapter 6.

The *Clustering* module is used to group radio map locations based on the access points covering them. Clustering is used to reduce the computational requirements of the system and, hence, conserve power. The *Clustering* module is discussed in Chapter 7.

The *Small-Scale Compensator* module handles the small-scale variation characteristics of the wireless channel. The *Small-Scale Compensator* module is discussed in Chapter 8.

The system depends on a set of API's to achieve hardware independence; as long as the underlying wireless card driver supports the specified API, the *Horus* system code can be ported to different hardware architectures without modification.

1.2 Contribution

The contributions of this thesis are:

- We present a taxonomy of the current research in the area of WLAN location determination systems.
- We model the signal strength distributions received from access points using parametric and non-parametric distributions. By exploiting the distributions, the *Horus* system achieves accuracy better than current WLAN location determination systems. We also show analytically that these techniques are optimal among all discrete-space radio map-based WLAN location determination systems.
- We introduce an autoregressive model for handling the correlation between successive samples from the same access point. Based on this model, we present a

technique to take multiple samples from an access point into account to enhance the accuracy.

- We describe two techniques for allowing continuous space estimation.
- We introduce clustering of radio map locations as an approach to reduce the computational requirements of the location determination algorithms and achieve scalability. We describe two general approaches to clustering: explicit clustering and implicit clustering, both applicable to any of the previous radio map based approaches. We also show that clustering enhances the accuracy.
- We present a technique for handling small-scale variations.
- We show an implementation of the *Horus* system on mobile devices by writing device drivers and API's that support the location determination system.
- We compare the performance of the *Horus* system with other WLAN location determination techniques.

We also relay lessons learned and identify opportunities for future research.

1.3 Organization

Chapter 2 surveys location determination technologies in general and provides a taxonomy of WLAN location determination systems. In Chapter 3, we explain the different causes of variation in the signal strength and their effect on the accuracy of WLAN location determination systems. We formulate our problem in Chapter 4 and present the discrete-space estimation module based on non-parametric and parametric distributions. We also show analytically that our technique is optimal among all discrete-space

radio map-based WLAN location determination systems. Chapter 5 describes the correlation handling module that allows the system to use multiple samples from the same access point, to achieve better accuracy. We present the continuous-space estimation module in Chapter 6. In Chapter 7, we present the radio map clustering techniques. Chapter 8 introduces the small-scale compensation module. In Chapter 9, we evaluate the performance of the *Horus* system. Finally, Chapter 10 concludes the thesis and gives directions for extending the work presented in this thesis.

Chapter 2

Related Work

In this chapter, we survey the different solutions to the location determination problem. We start by presenting different location determination technologies and then move to the WLAN location determination systems. We present a taxonomy for the WLAN location determination systems and show where the *Horus* system belongs in this taxonomy.

2.1 Location Determination Technologies

There have been many systems over the years tackling the problem of user positioning and tracking. Examples include the Global Positioning System (GPS), wide-area cellular based systems, infrared based systems, magnetic tracking systems, various computer vision systems, physical proximity systems, ultrasonic based systems, and radio frequency (RF) based systems.

The **GPS system** [9] is very useful in outdoors environments. However, the line-of-sight to GPS satellites is not available inside buildings and hence the GPS system cannot be used indoors.

Locating users in the wide-area cellular based systems has been motivated in recent

years by the FCC 94-102 order [7], mandating wireless E911 (automatically locating 911 callers). The two most widely known location technologies used in the wide-area cellular based systems are **Time Difference of Arrival (TDOA)** and **Angle of Arrival (AOA)**. TDOA systems use the principle that the emitter location can be estimated by intersection of the hyperbolae of constant differential Time of Arrival (TOA) of the signal at two or more pairs of base stations. AOA systems use simple triangulation based on estimated AOA of a signal at two or more base stations to estimate the location of the desired transmitter [8]. While these systems are promising in outdoor environments, their effectiveness in indoor environments is limited by the multiple reflections suffered by the RF signal, which leads to inaccurate estimate of the TOA or AOA, and the lack of off-the-shelf and inexpensive hardware to provide fine-grain time synchronization.

Many **infrared based (IR)** based systems have been proposed and reported. In the Active Badge system [10], a badge worn by a person emits a unique IR signal. Fixed IR receivers pick up this signal and relay it to the location manager software. The walls of a room blocks the IR signal, thus the user can be identified accurately within a room. In [11] IR transmitters attached to known positions in the ceiling emit beacons. A head mounted optical sensor senses these beacons. This enables the system software to determine the user location. IR based techniques suffer from several drawbacks: (a) they scale poorly due to the limited range of IR, (b) they incur significant installation and maintenance costs and (c) they perform poorly in the presence of direct sunlight.

Magnetic tracking has been used to support virtual reality and motion capture for computer animation [48, 49]. For example, Ascension [48] offers a variety of motion capture solutions such as the MotionStar DC magnetic tracker. These tracking systems generate axial DC magnetic field pulses from a transmitting antenna in a fixed

location. The system computes the position and orientation of the receiving antennas by measuring the response in three orthogonal axes to the transmitter field pulse, combined with the fixed effect of the earth's magnetic field. Such systems suffer from steep implementation costs and the need to tether the tracked object to a control unit. Furthermore, the sensors must remain within 1 to 3 meters of the transmitter, and accuracy degrades with the presence of metallic objects in the environment.

Several groups have explored using **computer vision technology** for locating objects [12–14]. Microsoft Research's Easy Living [12] provides one example of this approach, in which real-time 3D cameras provide a stereovision positioning capabilities in a home environment. Computer-vision based techniques have two drawbacks: (a) they use substantial processing power to analyze frames captured with comparatively low-complexity hardware; (b) the analysis becomes more complex when the scene complexity increases or more occlusive motion occurs.

Georgia Tech's Smart Floor proximity location system [15] is an example of **physical proximity** systems. Here, embedded pressure sensors capture footfalls, and the system uses Hidden Markov Models to recognize the users according to their profiles. The system has the disadvantages of poor scalability and high incremental cost, because the floor of each building in which Smart Floor is deployed must be physically altered to install the pressure sensor grids.

An example of ultrasonic based systems is the **Active Bat** [1, 16] system and the **Cricket** [17] system and the **Dolphin** system [18]. In the *Active Bat* system, a short pulse of ultrasound is emitted from a transmitter (a Bat) attached to the object to be located in response to an RF request from a local controller. The local controller sends, at the same time as the request packet, a synchronized reset signal to the ceiling sensors using a wired serial network. The system measures the times-of-flight of the pulse to

the mounted receivers on the ceiling. The speed of sound in air is used to calculate the distances from the Bat to each receiver. The local controller forwards these distances to a central controller that performs the location determination computation. The lack of scalability and difficulty of deployment are disadvantages to this approach.

The **Cricket** location support system [17] uses a combination of RF and ultrasound technologies to provide a location-support service to users and applications. Wall- and ceiling-mounted beacons are spread through the building, publishing information on an RF signal operating in the 418 MHz AM band. With each RF advertisement, the beacon transmits a concurrent ultrasonic pulse. Listeners attached to devices and mobiles listen for RF signals, and upon receipt of the first few bits, listen for the corresponding ultrasonic pulse. When this pulse arrives, they obtain a distance estimate for the corresponding beacon. The listeners run maximum-likelihood estimators to correlate RF and ultrasound samples and pick the best pair. The disadvantages lie in the lack of centralized management or monitoring and the computational, and hence the power consumption burden, at the receiver due to timing and processing the RF data and ultrasound pulses.

The **Dolphin** system proposed in [18] uses spread spectrum signaling to allow simultaneous access and to provide better performance in the presence of noise, compared to the above described narrow-band techniques.

RF based techniques have recently become the subject of ongoing research. For example, the **3D-iD RF** tag built by PinPoint Corporation [20]. Antennas planted around a facility emit RF signals. Tags, acting like RF mirrors, transmit a response signal along with an identification code. Various antennas receive the response signal and send the results to a central controller that triangulates the user location. The cost of the entire system is quite high.

All these techniques have two common disadvantages: requirement of specialized hardware leading to more deployment and maintenance cost; and poor scalability.

In the last few years, researchers have started looking at location determination systems that does not require any additional hardware. In an 802.11 WLAN [43], the wireless card uses the signal strength returned from the access points to roam between different access points. This signal strength information is available to the application level. Therefore, a location determination system based on signal strength information in an 802.11 network can be implemented without requiring any specialized hardware. This has the advantage of increasing the utility of the data network. A similar idea can be applied to **FM** radio signals to determine the location of an FM receiver [50].

In the rest of this chapter we focus on the WLAN location determination systems. We start by a taxonomy of the current WLAN location determination techniques.

2.2 WLAN Location Determination Systems

Figure 2.1 shows a taxonomy of WLAN location determination systems. The techniques can be categorized into two main categories: ad-hoc mode based and infrastructure mode based.

2.2.1 Ad-hoc mode based techniques

Determining location in ad-hoc networks has been an active area of research, e.g. [51–61]. In the 802.11 ad-hoc mode, there are no access points and location determination systems working in this mode cannot assume any preprocessing of the environments. Therefore, these system, e.g. [62, 63], use empirical radio propagation models to triangulate the user location.

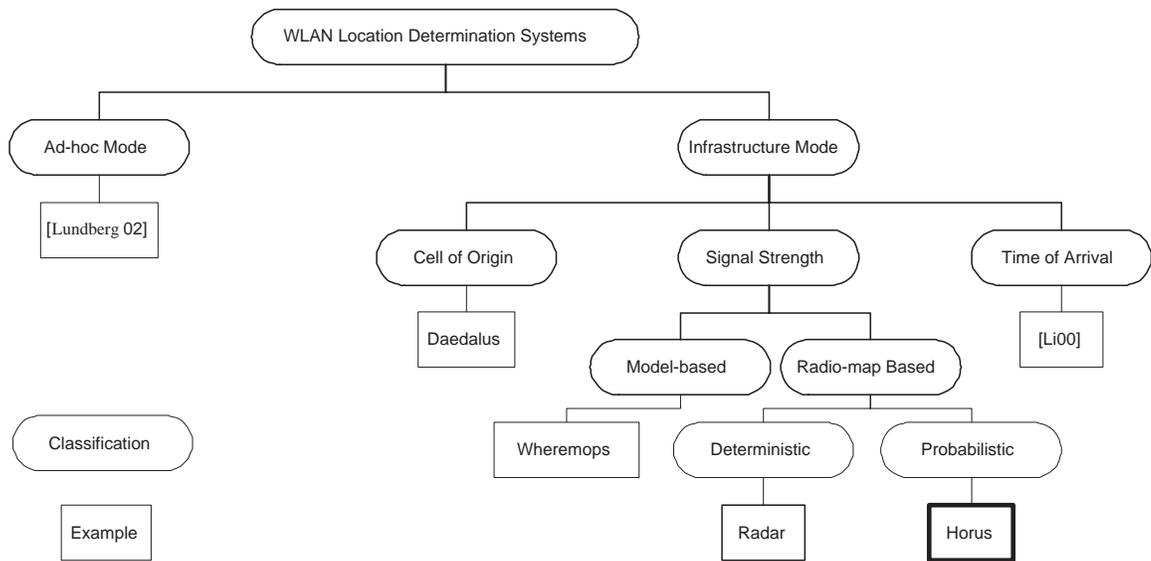


Figure 2.1: A taxonomy of WLAN location determination systems.

2.2.2 Infrastructure mode based techniques

Systems based on the infrastructure mode take the access points as reference points and triangulate the user location relative to these access points. These techniques can be categorized into cell of origin based techniques, time of arrival based techniques, and signal strength based techniques.

Cell of origin

Cell of origin based techniques, e.g. the **Daedalus** project [64], estimate the user location as the location of the access point that the wireless card is associated with. Therefore, the accuracy of the cell of origin based techniques is limited to the range of the access point.

Time of Arrival Based

The system in [65] uses delay measurement techniques to estimate the user location in an 802.11 network. They propose a *TDMA* measuring method to avoid the synchronization required in traditional *TOA* based methods. However, they require a high-precision timer in the order of ns-accuracy, which is not available in the current 802.11 implementations.

Signal Strength Based

Signal strength based techniques are purely software technique. They can be categorized into model based techniques and radio-map based techniques.

Model based techniques, e.g. [21–25], use empirical models to capture the relation between the signal strength and distance. These models can capture the internal structure of the building, e.g. the number of walls between the access point and the receiver [22]. However, they suffer from the noisy characteristic of the indoor wireless channel which makes the relation between the signal strength and distance a complex function that is difficult to capture using a model.

To solve this problem, radio-map based techniques try to capture the signature of the access points at selected locations in the area of interest. They work in two phases:

1. offline phase, in which the signal strength received from the access points at each location is tabulated in some form, and
2. online phase, in which the received signal strength vector, one entry for each access point, is compared to the stored radio map and the nearest match is returned as the estimated user location.

Radio-map based techniques can be further divided into deterministic techniques and

probabilistic techniques.

Deterministic techniques use deterministic algorithm to infer the user location [22, 23, 25–27]. During an offline phase of the **RADAR system** [22, 23], the system stores in the radio-map the average signal strength of each access point that can be heard. During the online phase, the RF signal strength of the transmitter is measured by a set of fixed receivers and is sent to a central controller. The central controller uses a K-nearest approach to determine the location from the radio map that best fits the collected signal strength information.

The **TMI system** proposed in [27] is based on triangulation, mapping and interpolation (TMI). In the TMI technique, the physical position of all the access points in the area needs to be known and a function is required to map signal strength onto distances. They generate a set of training points at each trained position. The interpolation of the training data allows the algorithm to use less training data. During user location determination phase, they use the approximate function they got from the training data to generate contour and they calculate the intersection between different contours yielding the signal space position of the user. The nearest set of mappings from the signal space to the physical space is found by applying a weighted average, based on proximity, to the signal space position.

The systems proposed in [25, 26] use a neural network approach to estimate the user location.

Probabilistic techniques use probability theory to infer the user location [31–37, 40]. For example, the **Nibble location system** [31, 32] from UCLA uses a Bayesian network to infer a user location. Their Bayesian network model include nodes for location, noise, and access points (sensors). The signal to noise ratio (SNR) observed from an access point at a given location is taken as an indication of that location. The

system also quantizes the SNR into four levels: high, medium, low, and none.

The **Horus** system belongs to the probabilistic radio-map based category.

None of the above systems takes into account smoothing the signal strength histogram through continuous distribution approximation, continuous space estimation, nor reducing correlation. All of these factors make the *Horus* system achieve accuracy better than all of these techniques. Moreover, the location clustering performed by the *Horus* system makes it energy efficiency, which is not provided by any of the above techniques. In addition, none of these systems handles small-scale variations, which is a major factor affecting the accuracy of WLAN based location determination systems.

Chapter 3

Wireless Channel Characteristics

In this chapter, we identify the different causes of variations in the wireless channel and how they affect the WLAN location determination systems. We are mainly concerned with the variations that affect the received signal strength. We start by describing our sampling process. Then, we categorize the variations in the wireless channel as temporal variations and spatial variations. We end the chapter by describing the experimental testbeds that we will use in the rest of the thesis.

3.1 Sampling Process

A key function required by all WLAN location determination systems is signal strength sampling. We used a Lucent Orinoco silver network interface card (NIC) supporting up to 11 Mbit/s data rate [66].

We modified [67] the *Lucent Wavelan* driver for Linux so that it returns the signal strength of beacon frames received from all access points in the NIC range using active scanning [43]; our driver was the first to support this feature under Linux. Figure 3.1 shows how the active scanning works. The NIC scans all the available 802.11 channels. On each channel, corresponding to a specific frequency, the NIC sends a probe

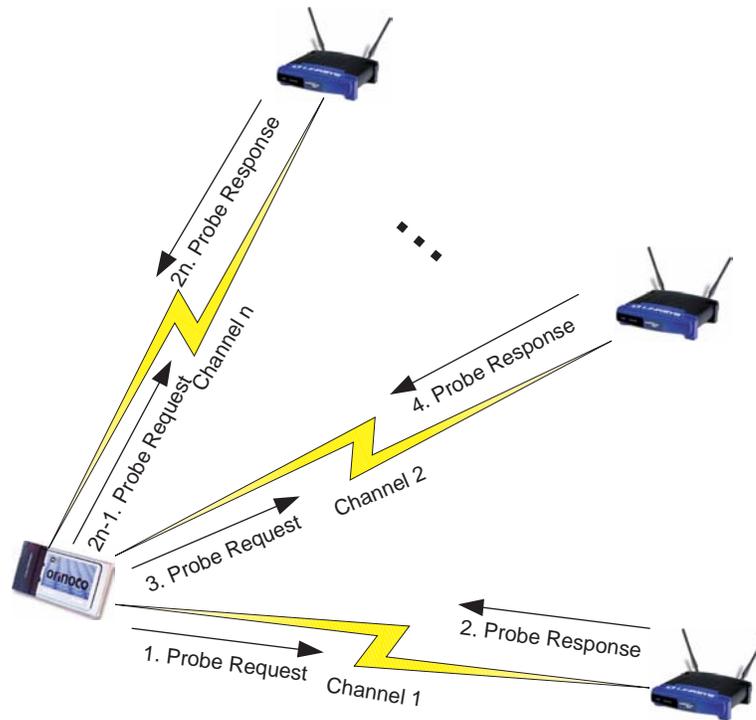


Figure 3.1: Active scanning process in the 802.11 standard.

request frame and waits for probe response frames from access points running on the same channel. The NIC reports back the signal strength that was observed during the reception of the probe response frames. There is a lower bound and an upper bound on the amount of time the NIC spends on a certain channel. Therefore, for the purpose of the location determination systems, the scanning process output is a list of the *MAC* addresses of the access points associated with the signal strength observed in this scan.

We also developed a wireless API [67] that interfaces with any device driver that supports the wireless extensions [68]. The device driver and the wireless API have been available for public download and have been used by others in wireless research.

3.2 Noisy Characteristics of the Wireless Channel

We divide the causes for the variation of the wireless channel into two categories: temporal variations and spatial variations.

3.2.1 Temporal variations

These variations occur when the receiver position is fixed. This section describes the behavior of the wireless channel over time.

Samples from one access points

We conducted an experiment where we measured the signal strength from a single access point over a five minutes period. Samples were taken one second apart for a total of 300 samples. Figure 3.2 shows the normalized histogram of the received signal strength. When the user is standing at a fixed position, the signal strength measured varies over time. The histogram range can be as large as 10 dBm or more. This time variation of the channel can be due to changes in the physical environment such as people movement.

These variations suggest that depending on a single statistical value, e.g. the average signal strength, to capture the signature of the access point at a fixed position leads to the loss of a lot of information and affects the accuracy of the location determination system significantly. We show how the *Horus* system deals with these temporal variations in Chapter 4.

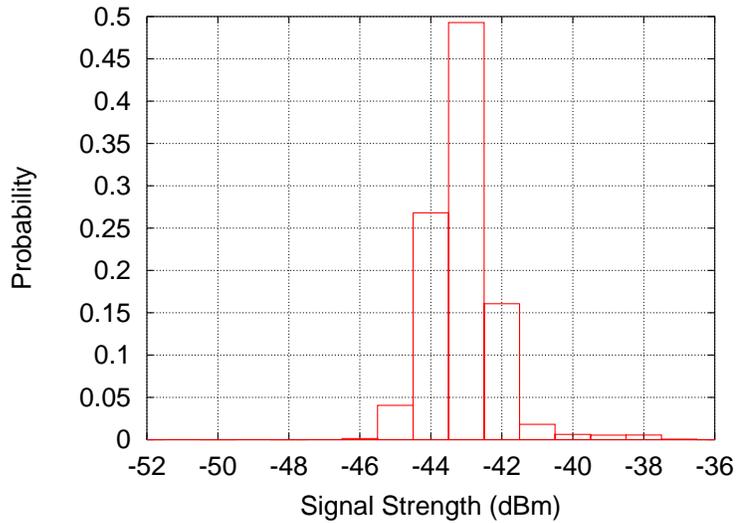


Figure 3.2: An example of the normalized signal strength histogram from an access point.

Samples Correlation

Figure 3.3 shows the autocorrelation function of the samples collected from one access point (one sample per second) at a fixed position. The figure shows that the autocorrelation of consecutive samples ($lag = 1$) is as high as 0.9. This high autocorrelation is expected as over a short period of time the signal strength received from an access point at a particular point is relatively stable (modulo the changes in the environment discussed in the previous section). Figure 3.4 shows the samples received from an access point for about 2.5 hours. The figure shows the high correlation between successive samples.

This high autocorrelation value has to be considered when using multiple samples from one access point to enhance accuracy. Assuming independence of samples from the same access point leads to the undesirable result of degraded system performance as the number of samples is increased. This is discussed in more details in Chapter 5.

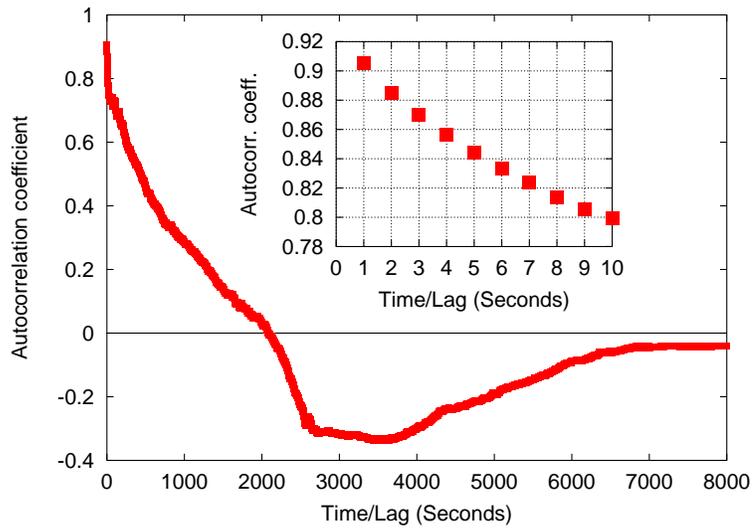


Figure 3.3: An example of the autocorrelation between samples from an access point. The sub-figure shows the autocorrelation for the first 10 lags.

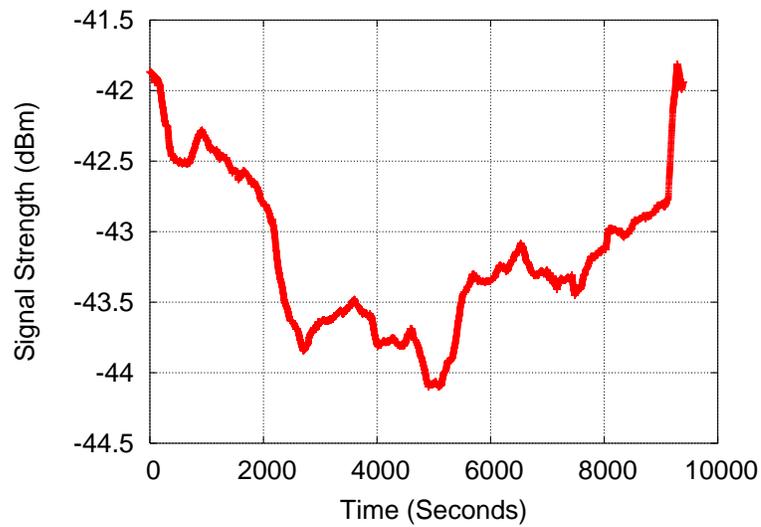


Figure 3.4: Signal strength received from an access point over time (smoothed with moving average of 600 seconds).

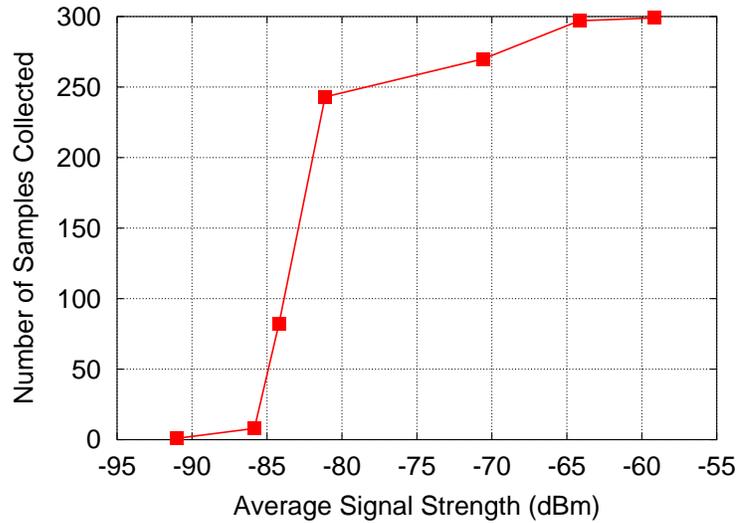


Figure 3.5: Relation between the average signal strength of an access point and the percentage of samples received from it during a 5-minute interval.

Samples from different access points

We performed an experiment to test the behavior of access points with different average signal strength at the same location. During this experiment, we sampled the signal strength from each access point at the rate of one sample per second. Figure 3.5 shows the relation between the average signal strength received from an access point and the percentage of samples we receive from it during a period of 5 minutes. The figure shows that the number of samples collected from an access point is a monotonically increasing function of the average signal strength of this access point. Assuming a constant noise level, the higher the signal strength, the higher the signal to noise ratio and the more probable it becomes that the 802.11b card will identify the existence of a packet. The sharp drop at about -81 dBm can be explained by noting that the receiver sensitivity for the card we used was -82 dBm.

We use this fact in the clustering techniques described in Chapter 7.

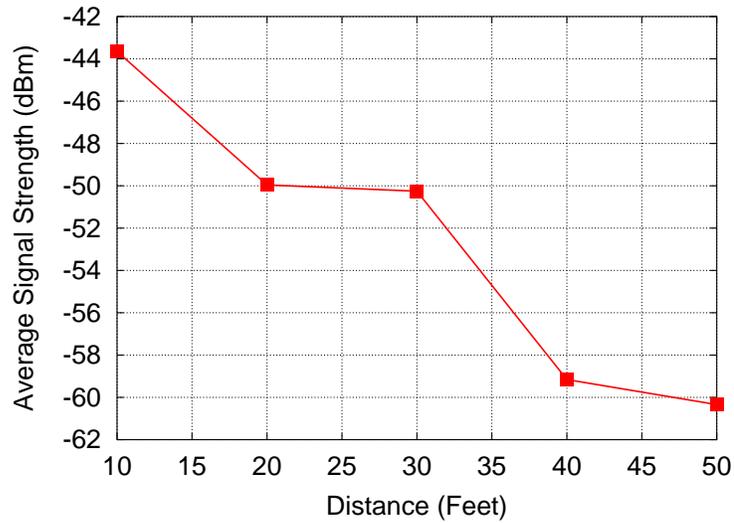


Figure 3.6: Large-scale variations: Average signal strength over distance.

3.2.2 Spatial characteristics

These variations occur when the receiver position is changed. We further divide these variations into large-scale variations and small-scale variations.

Large-Scale Variations

Figure 3.6 shows the average signal strength received from an access point as the distance from it increases. The signal strength varies over a long distance due to attenuation of the RF signal.

Large-scale variations are desirable in RF-based systems as they lead to changing the signature stored in the radio map for different locations and, hence, better differentiation between these locations.

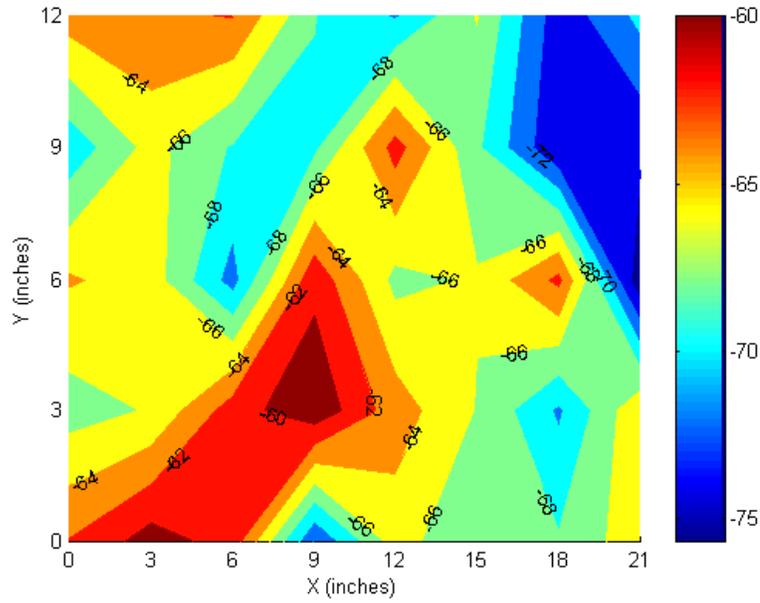


Figure 3.7: Small-scale variations: Signal strength contours

Small-Scale Variations

These variations happen when the user moves over a small distance (order of wavelength). This leads to changes in the average received signal strength. For the 802.11b networks working at the 2.4 GHz range, the wavelength is 12.5 cm and we measure a variation in the average signal strength up to 10 dBm in a distance as small as 3 inches (Figure 3.7).

Dealing with small-scale variations is challenging and none of the current WLAN location determination systems handles it. To limit the radio map size and the time required to build the radio map, selected radio map locations are typically placed few feet apart. This means that the radio map does not capture small-scale variations leading to decreased accuracy in the current WLAN location systems. In Chapter 8, we present how the *Horus* system handles small-scale variations.

3.3 Experimental Testbeds

In this section, we present the experimental testbeds we used to evaluate the performance of the *Horus* system and compare its performance to other systems.

3.3.1 Testbed 1

We performed our first experiment in the south wing of the fourth floor of the A. V. William's building in the University of Maryland at College Park. The layout of the floor is shown in Figure 3.8. The wing has a dimension of 224 feet by 85.1 feet. The technique was tested in the University of Maryland wireless network using *Cisco* access points. The entire wing is covered by 21 access points.

The radio map has 110 locations along the corridors and 62 locations inside the rooms. On the average, each location is covered by 6 access points. The test set was collected by different persons on different days and time of days. The *Horus* system was running on *Windows XP* professional operating system.

3.3.2 Testbed 2

We performed the second experiment in the Fujitsu Lab of America-East (FLA) located in the third floor of the 8400 Baltimore Ave. building in College Park (Figure 3.9). The area of the experiment site is approximately 39 feet by 118 feet covering corridors, cubicles, and rooms. The test area was covered by 6 access points. Five of the 6 access points were *LinkSys* and the remaining one was a *Cisco* access point.

We have a total of 110 locations in the radio map. On the average, each location is covered by 4 access points. The test set was collected by different persons on different days and time of days. The *Horus* system was running under *Linux* (kernel 2.5.7)

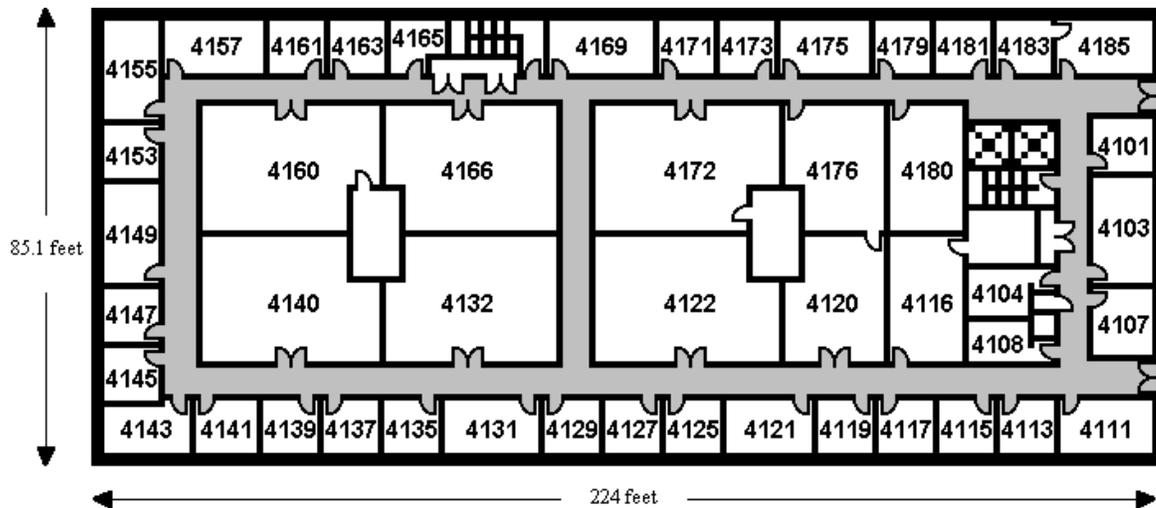


Figure 3.8: Plan of the south wing of the 4th floor of the Computer Science Department building where the experiment was conducted. Readings were collected in the corridors (shown in gray).

operating system.

3.4 Summary

In this chapter, we showed different causes for the noisy characteristics of the wireless channel. We divided them into two main categories: temporal variation and spatial variations. Temporal variations are related to the characteristic of the signal strength received from the access points at a fixed location over time. Spatial variations are related to how the signal strength from an access point changes over distance. The rest of the thesis shows how the *Horus* system handles these variations. We also showed the experimental testbeds that we use to evaluate the performance of the *Horus* system in the rest of the thesis.

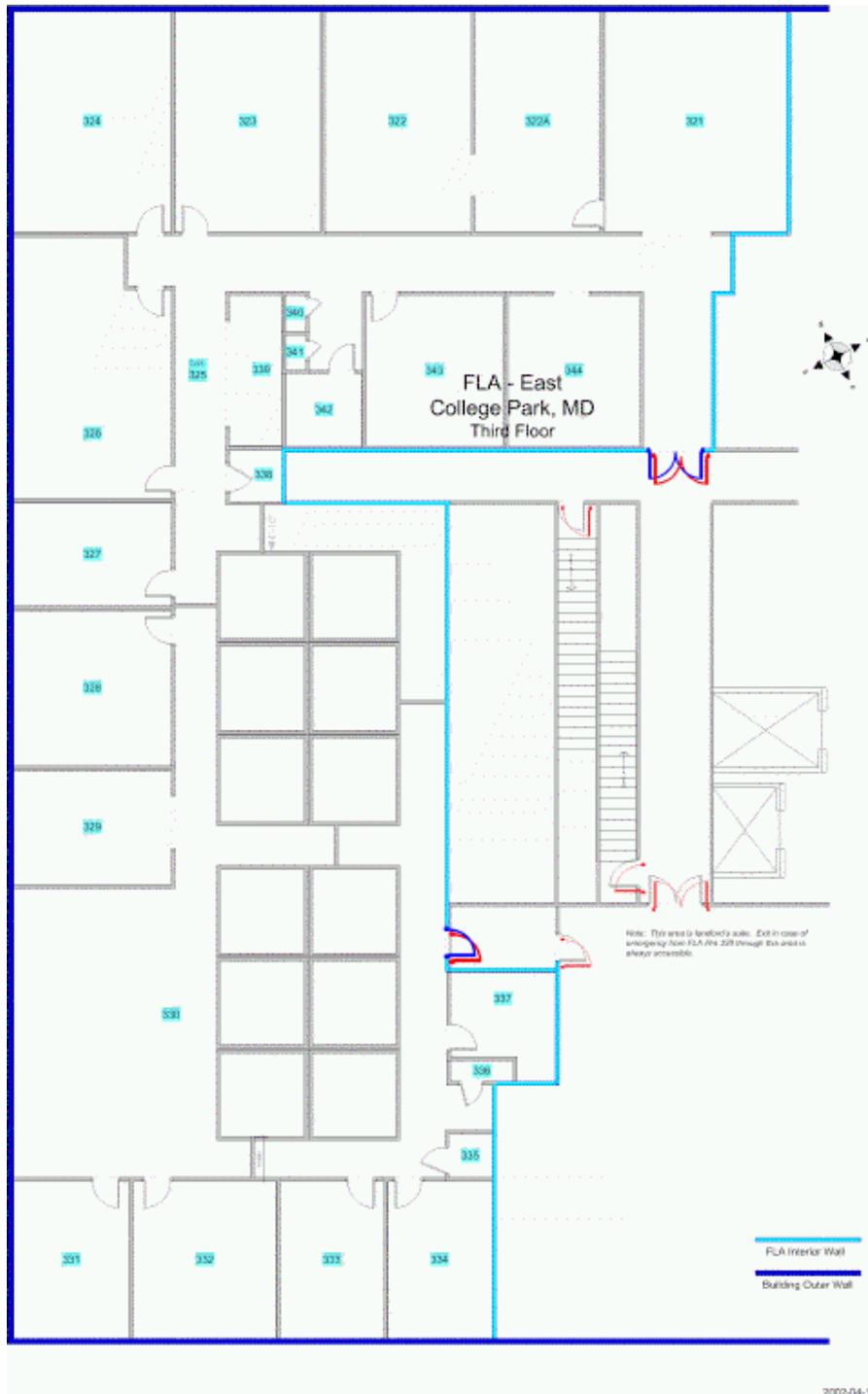


Figure 3.9: Floor plan of the Fujitsu Lab of America-East where the second experiment was conducted.

Chapter 4

Problem Formulation

In this chapter, we present our approach for solving the location determination problem. We start by laying out the mathematical framework for the approach in Section 4.1. We present two different techniques for solving the problem in Sections 4.2 and 4.3. Section 4.4 analyze the performance of the *Horus* system analytically. Finally, in Section 4.5, we present the empirical results on different testbeds.

4.1 Mathematical Model

Let \mathbb{X} be a 2 (or 3) dimensional physical space. At each location $x \in \mathbb{X}$, we can get the signal strength from k access points. We denote the k -dimensional signal strength space as \mathbb{S} . Each element in this space is a k -dimensional vector whose entries represent the signal strength readings from different access points. Sampling from the signal strength space \mathbb{S} at location x at time t , denoted by $s(x, t)$, can be viewed as a vector stochastic process. We assume that this process is stationary, therefore we can drop the time index. For the remaining of the thesis, we refer to this process as $s(x)$ and may, some time, drop the location index x and refer to the signal strength vector as s when it is clear to do so. We also assume that the samples from different access

points are independent.

The problem becomes, given a signal strength vector $s = (s_1, \dots, s_k)$, we want to find the location $x \in \mathbb{X}$ that maximizes the probability $P(x/s)$.

In the rest of this chapter, we assume a discrete \mathbb{X} space. We defer the continuous space case to Chapter 6.

We present two approaches for the location determination problem:

1. When the signal strength distributions at any location are represented by non-parametric distributions.
2. When the signal strength distributions at any location are represented by parametric distributions.

4.2 Non-Parametric Signal Strength Distributions

In this section, we present the location determination algorithm of the *Horus* system when the signal strength distributions are modeled by non-parametric distributions.

The algorithm runs in two phases:

1. *training phase*: to estimate the signal strength distributions and build the radio map.
2. *location determination phase*: to estimate the user location based on a received signal strength vector, using the constructed radio map.

4.2.1 Training Phase

At each location $x \in \mathbb{X}$, we store a model for the joint probability distribution of the access points at this location. Therefore, our radio map is stored as a collection of

models for joint probability distributions.

The problem of estimating the joint probability distribution can be done in different ways with different accuracy levels. The problem can be stated as: given k access points AP_1, \dots, AP_k , we want to estimate $P(AP_1 = s_1, AP_2 = s_2, \dots, AP_k = s_k)$ where s_i is a signal strength value from AP_i at a given location x . One good way to estimate this joint distribution is to use the Maximum Likelihood Estimation (MLE) method which estimate the joint probabilities as:

$$P(AP_1 = s_1, AP_2 = s_2, \dots, AP_k = s_k) = \frac{\text{Count}(s_1, s_2, \dots, s_k)}{\text{Size of Training Data}} \quad (4.1)$$

that is, the number of times that the signal strength vector (s_1, s_2, \dots, s_k) appeared in the entire training set divided by the size of the training set at location x .

The problem of this approach is that it requires a large training set to obtain good estimate of the joint distribution and the required size increases exponentially with k . Since access points running on different channels are independent, as per assumption, the *Horus* system estimates the joint probability distribution as the product of the marginal probability distributions of the individual access points as:

$$P(AP_1 = s_1, AP_2 = s_2, \dots, AP_k = s_k) = P(AP_1 = s_1) \dots P(AP_k = s_k) \quad (4.2)$$

For a given location, $P(AP_i = s_i)$ can be estimated using the normalized histogram of the access point AP_i at this location. Figure 4.1 gives a typical example of the signal strength normalized histogram from an access point.

4.2.2 Location Determination Phase

Given a signal strength vector $s = (s_1, \dots, s_k)$, we want to find the location $x \in \mathbb{X}$ that maximizes the probability $P(x/s)$, i.e. we want

$$\text{argmax}_x [P(x/s)] \quad (4.3)$$

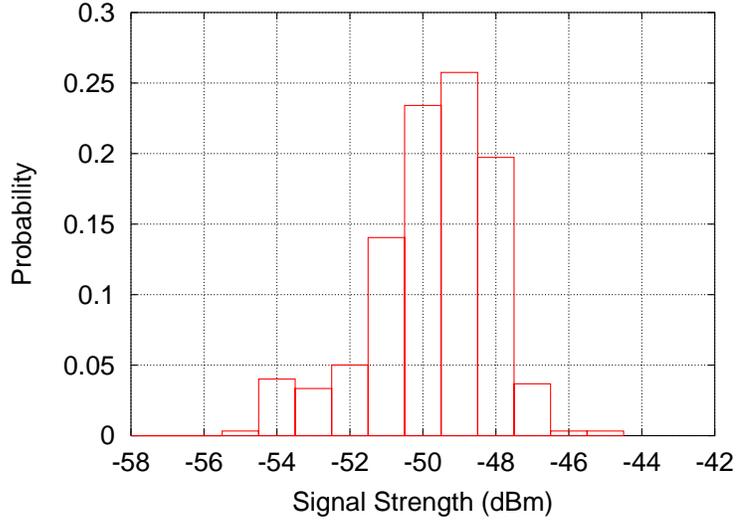


Figure 4.1: An example of the signal strength normalized histogram from an access point.

Using Baye's theorem, this can be rewritten as:

$$\operatorname{argmax}_x [P(x/s)] = \operatorname{argmax}_x \left[\frac{P(s/x) \cdot P(x)}{P(s)} \right] \quad (4.4)$$

since $P(s)$ is constant for all x , we can rewrite equation 4.4 as:

$$\operatorname{argmax}_x [P(x/s)] = \operatorname{argmax}_x [P(s/x) \cdot P(x)] \quad (4.5)$$

$P(x)$ can be determined from the user profile based on the fact that if the user is at a given location, it is more probable that he will be at an adjacent location in the future. If the user profile information is not known, or not used, then we can assume that all the locations are equally likely and the term $P(x)$ can be factored out from the maximization process. Equation 4.5 becomes:

$$\operatorname{argmax}_x [P(x/s)] = \operatorname{argmax}_x [P(s/x)] \quad (4.6)$$

The remaining term is calculated using the radio map:

$$P(s/x) = \prod_{i=1}^k P(s_i/x) \quad (4.7)$$

Algorithm 1 gives the detail of the location determination algorithm for the parametric distribution case.

Algorithm 1 $x = \text{GetLocation}(s, \mathbb{X}, RM)$

Input:

s : Measured signal strength vector ($s = (s_1, \dots, s_k)$).

\mathbb{X} : Radio map locations.

RM : Radio map, where $RM[s_a][a][x]$ represent the probability of receiving signal strength s_a from access point a at location $x \in \mathbb{X}$.

Output:

The location $x \in \mathbb{X}$ that maximizes $P(x/s)$.

```

1: Max  $\leftarrow 0$ 
2: for  $l \in \mathbb{X}$  do
3:    $P \leftarrow \prod_{i=1}^k RM[s_i][i][l]$ 
4:   if  $P > \text{Max}$  then
5:      $x \leftarrow l$ 
6:      $\text{Max} \leftarrow P$ 
7:   end if
8: end for

```

4.3 Parametric Signal Strength Distributions

In this section, we present the location determination algorithm of the *Horus* system when the signal strength distributions are modeled by parametric distributions. Similar

to the case of non-parametric distributions, the algorithm runs in two phases: training phase and location determination phase.

4.3.1 Training Phase

Since the signal strength values received from different access points are independent, our problem reduces to finding a parametric distribution that fits the signal strength received from a single access point. The Gaussian distribution fits the signal strength distribution from an access point (measured in dBm). The PDF of the Gaussian distribution is given by:

$$pdf(q) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(q-\mu)^2/(2\sigma^2)} \quad (4.8)$$

where μ and σ are the mean and standard deviation of the distribution respectively.

Given n signal strength samples from an access point AP_i , the *Horus* system uses the MLE estimation method to estimate the distribution parameters. The estimate for the parameter μ is:

$$\hat{\mu} = \frac{1}{n} \sum_{j=1}^n s_i(j) \quad (4.9)$$

where $s_i(j)$ is the j^{th} signal strength sample from AP_i .

The estimate for the parameter σ is:¹

$$\hat{\sigma} = \sqrt{\frac{1}{n} \sum_{j=1}^n [s_i(j) - \hat{\mu}]^2} \quad (4.10)$$

Figure 4.2 shows an example of a Gaussian distribution fitted to the normalized histogram in Figure 4.1.

¹This estimator assumes that the samples are independent. We address the independence issue in Chapter 5.

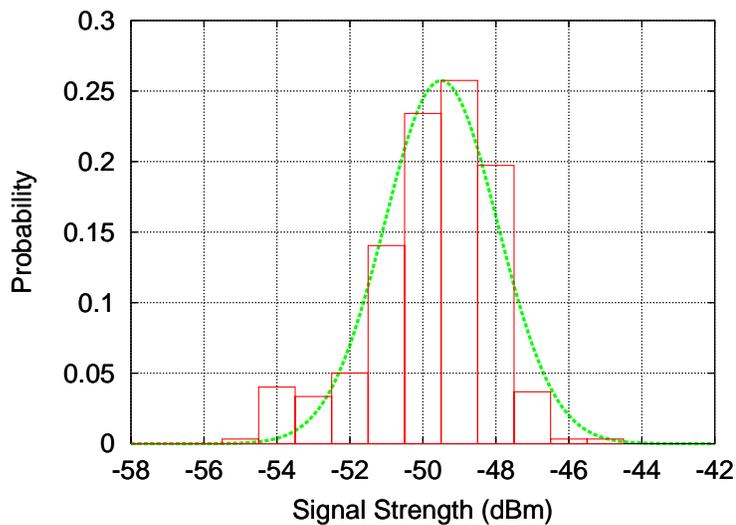


Figure 4.2: An example of fitting a Gaussian distribution to the signal strength normalized histogram from an access point.

4.3.2 Location Determination Phase

Given a signal strength vector $s = (s_1, \dots, s_k)$, we want to find the location $x \in \mathbb{X}$ that maximizes the probability $P(x/s)$. Similar to the non-parametric distribution case, this probability is given by Equation 4.6. To obtain the probability $P(s_i/x)$ using the Gaussian PDF, we use:

$$P(s_i/x) = \int_{s_i-0.5}^{s_i+0.5} pdf(q) dq \quad (4.11)$$

Since the RSSI returned from the wireless NIC is quantized to integer values.

Algorithm 2 gives the detail of the location determination algorithm in the parametric distribution case.

Algorithm 2 $x = \text{GetLocation}(s, \mathbb{X}, RM)$

Input: s : Measured signal strength vector ($s = (s_1, \dots, s_k)$). \mathbb{X} : Radio map locations. RM : Radio map, where $RM[a][x](q)$ represents the *pdf* of signal strength from access point a at location $x \in \mathbb{X}$.**Output:**The location $x \in \mathbb{X}$ that maximizes $P(x/s)$.

```
1: Max  $\leftarrow 0$ 
2: for  $l \in \mathbb{X}$  do
3:    $P \leftarrow \prod_{i=1}^k \int_{s_i-0.5}^{s_i+0.5} RM[i][x](q) dq$ 
4:   if  $P > \text{Max}$  then
5:      $x \leftarrow l$ 
6:      $Max \leftarrow P$ 
7:   end if
8: end for
```

4.4 Analysis

In this section, we give an analytical method to analyze the performance of WLAN location determination techniques. We provide two expressions: one for calculating the average distance error of a given technique and the other for calculating the probability of error (i.e. the probability that the location technique will give an incorrect estimate). Using the probability of error, we show that the *Horus* system strategy is optimal among all strategies that returns an estimated location from the radio map locations.

4.4.1 Average Distance Error

We want to find the average distance error (denoted by $E(\text{DErr})$). Using conditional probability, this can be written as:

$$E(\text{DErr}) = \sum_{x \in \mathbb{X}} E(\text{DErr}/x \text{ is the correct user location}).P(x \text{ is the correct user location}) \quad (4.12)$$

where $P(x \text{ is the correct user location})$ depends on the user profile.

We now proceed to calculate $E(\text{DErr}/x \text{ is the correct user location})$. Using conditional probability again:

$$\begin{aligned} & E(\text{DErr}/x \text{ is the correct user location}) \\ &= \sum_{s \in \mathbb{S}} E(\text{DErr}/s, x \text{ is the correct user location}).P(s/x \text{ is the correct user location}) \\ &= \sum_{s \in \mathbb{S}} \text{Euclidean}(f_{\mathcal{A}}^*(s), x).P(s/x \text{ is the correct user location}) \end{aligned} \quad (4.13)$$

where $f_{\mathcal{A}}^*(s)$ represents the estimated location returned by the WLAN location determination technique \mathcal{A} when supplied with the input s and $\text{Euclidean}(f_{\mathcal{A}}^*(s), x)$ represents the Euclidean distance between the estimated location and the correct location.

Equation 4.13 says that to get the expected distance error given we are at location x , we need to get the weighted sum, over all the possible signal strength values $s \in \mathbb{S}$, of the Euclidean distance between the estimated user location ($f_{\mathcal{A}}^*(s)$) and the actual location x .

Substituting equation 4.13 in equation 4.12 we get:

$$E(\text{DErr}) = \sum_{s \in \mathbb{S}} \sum_{x \in \mathbb{X}} \text{Euclidean}(f_{\mathcal{A}}^*(s), x) \cdot P(s/x \text{ is the correct user location}) \cdot P(x \text{ is the correct user location}) \quad (4.14)$$

Note that the effect of the location determination technique is summarized in the function $f_{\mathcal{A}}^*$. We seek to find the function that minimizes the probability of error. We defer the optimality analysis till we present the *probability of error* analysis in the next section.

4.4.2 Probability of Error

In this section, we want to find an expression for the probability of error which is the probability that the location determination technique will return an incorrect estimate. This can be obtained from equation 4.14 by noting that every non-zero distance error (represented by the function $\text{Euclidean}(f_{\mathcal{A}}^*(s), x)$) is considered an error. More formally, we define the function:

$$g(x) = \begin{cases} 0 & : x = 0 \\ 1 & : x > 0 \end{cases}$$

The probability of error can be calculated from equation 4.14 as:

$$P(\text{Error}) = \sum_{s \in \mathbb{S}} \sum_{x \in \mathbb{X}} g(\text{Euclidean}(f_{\mathcal{A}}^*(s), x))$$

$$.P(s/x \text{ is the correct user location}).P(x \text{ is the correct user location}) \quad (4.15)$$

In the next section, we present a property of the term $g(\text{Euclidean}(f_{\mathcal{A}}^*(s), x))$ and use this property to get the optimal strategy for selecting the location.

4.4.3 Optimality

We base our optimality analysis on the probability of error.

Lemma 1 *For a given signal strength vector s , $g(\text{Euclidean}(f_{\mathcal{A}}^*(s), x))$ will be zero for only one location $x \in \mathbb{X}$ and one for the remaining $N - 1$ locations.*

Proof For a given signal strength vector s , the location determination technique will return a single location. If this location matches the correct location x , the distance error will be zero and hence the function g . If not, the distance error will be greater than zero and the function g will equals one. The estimated location $f_{\mathcal{A}}^*(s)$ can only match one of the possible N locations. \square

The lemma states that only one location will give a value of zero for the function $g(\text{Euclidean}(f_{\mathcal{A}}^*(s), x))$ in the inner sum. This means that the optimal strategy should select this location in order to minimize the probability of error. This leads us to the following theorem.

Theorem 1 (Optimal Strategy) *Selecting the location x that maximizes the probability $P(s/x).P(x)$ is both a necessary and sufficient condition to minimize the probability of error.*

Proof [Sufficient part] Selecting the location that maximizes the probability $P(s/x).P(x)$ will lead to making the function g in the inner sum equals zero for this

probability. Since this technique removes the maximum probability for all $s \in \mathbb{S}$, this minimizes the overall probability of error.

[Necessary part] By contradiction: Assume not, then there exist an optimal strategy \mathcal{A}_1 that for at least one signal strength vector s , selects a location x' that does not maximize the product $P(s/x').P(x')$. Let the probability of error using this strategy be $E1$. Consider another strategy \mathcal{A}_2 that take the same decisions as \mathcal{A}_1 except for the signal strength vector s , where it returns the location x that maximizes the product $P(s/x).P(x)$. Let the probability of error using this strategy be $E2$. Clearly, $E2$ is less than $E1$ which contradicts our assumption that \mathcal{A}_1 is optimal. \square

Theorem 1 suggests that the optimal location determination technique should store in the radio map the signal strength distributions to be able to calculate $P(s/x)$. Moreover, the optimal technique needs to know the user profile in order to calculate $P(x)$.

Corollary 1 *Deterministic techniques are not optimal.*

Proof Since deterministic techniques do not store any information about the signal strength distribution at each location, it follows from Theorem 1 that they are not optimal. \square

Note that we did not make any assumption about the independence of access points, user profile, or signal strength distribution in order to get the optimal strategy.

If all user locations are equi-probable, then $P(x) = \frac{1}{N}$ and Theorem 1 can be rewritten as:

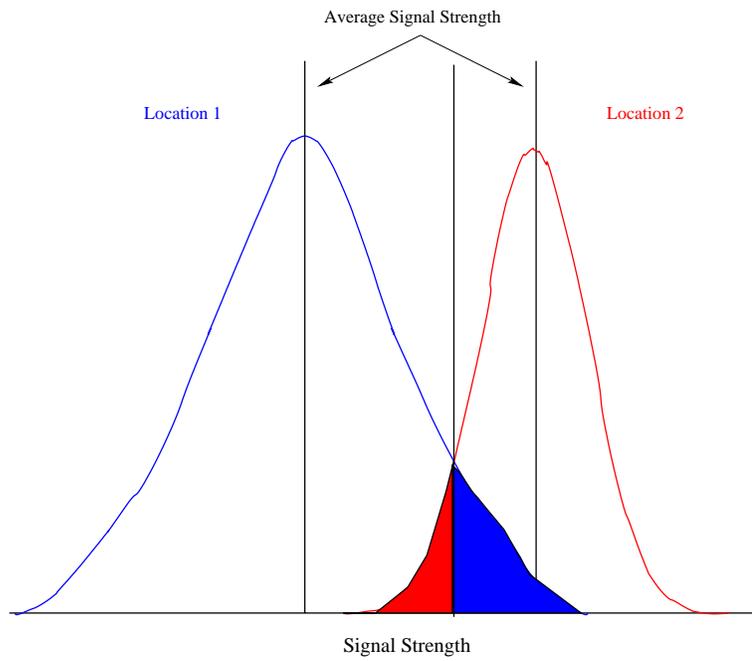
Theorem 2 *If the user is equally probable to be at any location of the radio map locations \mathbb{X} , then selecting the location x that maximizes the probability $P(s/x)$ is both a necessary and sufficient condition to minimize the probability of error.*

Proof The proof is a special case of the proof of Theorem 1. \square

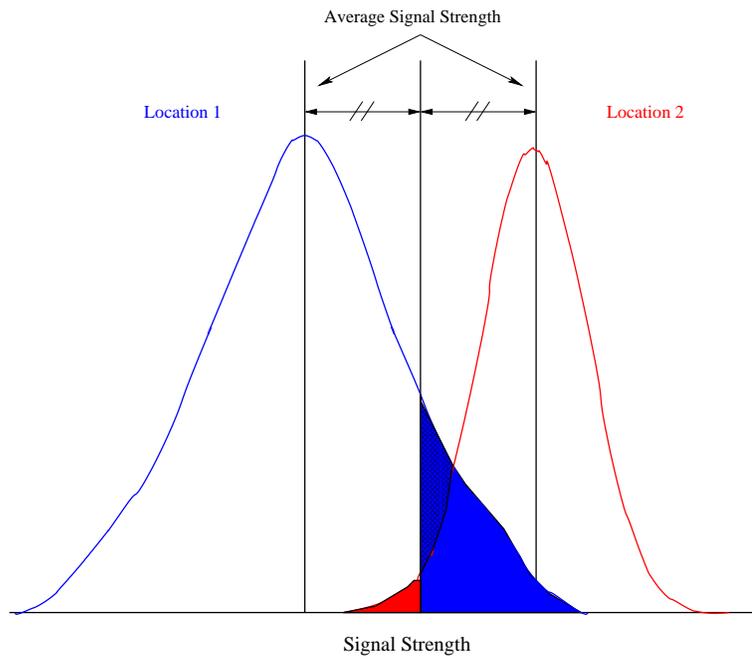
The theorem indicates that, for this special case, it is sufficient for the optimal technique to store the histogram of signal strength at each location. This is exactly the technique used in the *Horus* system. We show the effect of the user profile on performance in Chapter 9.

Figure 4.3 shows a simplified example illustrating the intuition behind the analytical expressions and the theorems. In the example, we assume that there are only two locations in the radio map and that at each location only one access point can be heard whose signal strength, for simplicity of illustration, follows a continuous distribution. The user can be at any one of the two locations with equal probability. For the *Horus* system (Figure 4.3.a), consider the line that passes by the point of intersection of the two curves. Since for a given signal strength the technique selects the location that has the maximum probability, the error if the user is at location 1 is the area of curve 1 to the right of this line. If the user is at location 2, the error is the area of curve 2 to the left of this line. The expected error probability is half the sum of these two areas as the two locations are equi-probable. This is the same as half the area under the minimum of the two curves (shaded in figure).

For the *Radar* system (Figure 4.3.b), consider the line that bisects the signal strength space between the two distribution averages. Since for a given signal strength the technique selects the location whose average signal strength is closer to the signal strength value, the error if the user is at location 1 is the area under curve 1 to the right of this line. If the user is at location 2, the error is the area under curve 2 to the left of this line. The expected error probability is half the sum of these two areas as the two locations are equi-probable (half the shaded area in the figure).



(a) Horus System



(b) Radar System

Figure 4.3: Expected error for the special case of two locations

From Figure 4.3, we can see that the *Horus* system outperforms the *Radar* system since the expected error for the former is less than the later (by the hashed area in Figure 4.3.b). The two systems would have the same expected error if the line bisecting the signal strength space of the two averages passes by the intersection point of the two curves. This is not true in general. This has been proved formally in the above theorems.

4.5 Experimental Results

Figures 4.4 and 4.5 show the performance of the basic algorithms of the *Horus* system for the two testbeds. The figures show that the performance of the *Horus* algorithms is consistent under the two testbeds. The system can achieve an accuracy of 5-6 feet 90% of the time. The performance of the parametric and non-parametric methods is comparable with a slight advantage for the parametric method. This will be explained in the next section.

4.6 Discussion

Since the performance of the parametric and non-parametric algorithms are comparable, the question is when to choose one over the other. The main advantage of the non-parametric technique is the efficiency of calculating the location estimate which involves an array lookup of the frequency of a given signal strength at a location and all operations are performed using integer arithmetic.

The parametric technique requires the evaluation of an integral which can be obtained by numerical methods. Although this involves complex calculation (exponentiation and square root) in floating point arithmetic, lookup tables for the standard normal

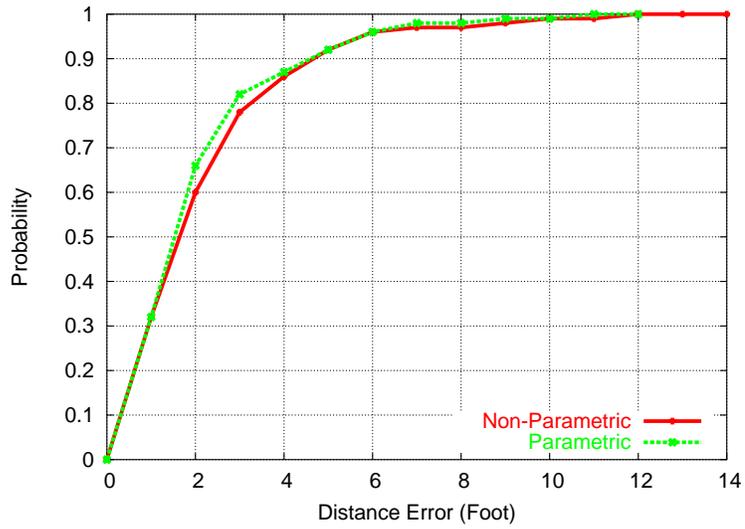


Figure 4.4: Performance of the basic algorithm of the *Horus* system for the first testbed.

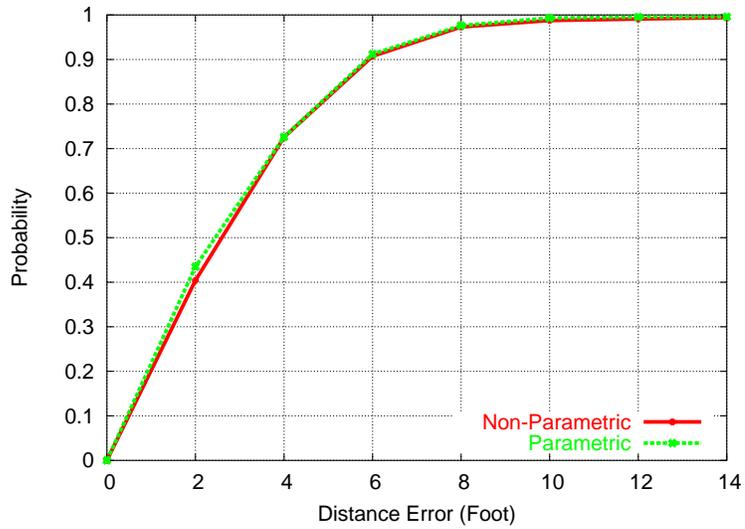


Figure 4.5: Performance of the basic algorithm of the *Horus* system for the second testbed.

distribution can be used to eliminate these calculations. On the other hand, the parametric technique reduces the radio map size as the system needs to store only the mean and variance of the Gaussian distribution, compared to storing the entire histogram in the case of the non-parametric technique. Moreover, using a parametric distribution to estimate the signal strength distribution smooths the distribution shape to account for missing signal strength values in the training phase (due to the finite training time). This avoids obtaining a zero probability for any signal strength value that was not obtained in the training phase and hence enhances the accuracy. This explains the slight performance advantage of the parametric technique compared to the non-parametric technique. In addition, using parametric distributions makes the problem of handling samples correlations more tractable as will be explained in Chapter 5.

4.7 Summary

In this chapter, we presented our mathematical model and showed how to model the signal strength distributions received from access points using parametric and non-parametric distributions. The main advantage of the non-parametric technique is the efficiency of calculating the location estimate, while the parametric technique reduces the radio map size and smooths the distribution shape which leads to a slight performance advantage of the parametric technique over the non-parametric technique. We also showed analytically that these techniques are optimal among all discrete-space radio map-based WLAN location determination systems.

Unless otherwise specified, we use the parametric distribution case as our reference point in performance evaluation. We compare all the techniques that we introduce in the thesis to this case.

Chapter 5

Handling Samples Correlation

In order to account for the signal strength variations, it is important to average multiple signal strength samples from the same access point. As we showed in Chapter 3 (Figure 3.3) the autocorrelation of successive samples collected from one access point is as high as 0.9. This high autocorrelation value has to be considered when using multiple samples from one access point to enhance accuracy. Assuming independence of samples from the same access point leads to the undesirable result of degraded system performance as the number of averaged samples is increased.

In this chapter, we present how the *Horus* system handles this high autocorrelation value. We start by presenting an autoregressive model to capture the autocorrelation of the samples. Following that, we present a technique that uses this model to calculate the distribution of the average of n correlated samples. Finally, we modify the *Horus* location determination system to incorporate the new technique.

5.1 Background

The technique described in this chapter treats the samples received from an access point as a time series and uses time series based-techniques to analyze the correlation

between the samples. A *time series* [69] is a set of observations generated sequentially in time. If the set is discrete, the time series is said to be discrete.

We refer to successive equi-spaced samples from a discrete time series s as s_1, s_2, \dots . A statistical phenomenon that evolves in time according to probabilistic laws is called a *stochastic process* [69]. The time series to be analyzed may be thought of as a particular realization of the system under study. A *stationary* stochastic process is based on the assumption that the process is in a particular state of statistical equilibrium. More formally, a discrete process is strictly stationary if the joint probability of any set of observations is unaffected by shifting all the times of the observation forward or backward by any amount.

Autoregressive models are stochastic models used to analyze stochastic time series. In these models, the current values of the process is expressed as a finite, linear aggregate of previous values of the process and a white noise process v_t . Therefore, if we denote the values of the process as $s_t, s_{t-1}, s_{t-2}, \dots$, then

$$s_t - \bar{s} = (\phi_1 \cdot s_{t-1} - \bar{s}) + (\phi_2 \cdot s_{t-2} - \bar{s}) + \dots + (\phi_p \cdot s_{t-p} - \bar{s}) + v_t \quad (5.1)$$

is called an autoregressive process of order p , where \bar{s} is the average of the process.

In this chapter, we treat the signal strength samples from an access point as a discrete stationary time series. We model this time series using a first order autoregressive model.

5.2 Autoregressive Model

Let s_t be the *stationary* time series representing the samples from an access point where t is the discrete time index. s_t can be represented as a *first order* autoregressive

model as:

$$s_t = \alpha.s_{t-1} + (1 - \alpha).v_t \quad ; 0 \leq \alpha \leq 1 \quad (5.2)$$

where v_t is a noise process, independent from s_t , and α is a parameter that determines the degree of autocorrelation of the original samples. Moreover, different samples from v_t are i.i.d.'s¹.

The model in Equation 5.2 states that the current signal strength value (s_t) is a linear aggregate of the previous signal strength value (s_{t-1}) and an independent noise value (v_t). The parameter α gives flexibility to the model as it can be used to determine the degree of autocorrelation of the original process. For example, if α is zero, the samples of the process s_t are i.i.d.'s, whereas if α is one the original samples are identical (autocorrelation=1). In the following sections we describe some properties of the autoregressive model that we will use in the rest of the chapter.

5.2.1 Relation Between the Mean of s_t and v_t

We can see from Equation 5.2 that $E(s_t) = E(v_t)$. The two processes have the same mean.

5.2.2 Relation Between the Variance of s_t and v_t

The relation between the variance of the original and noise processes can be obtained as follows:

$$\begin{aligned} Var(s_t) &= Var(\alpha.s_{t-1} + (1 - \alpha).v_t) \\ &= \alpha^2.Var(s_{t-1}) + (1 - \alpha)^2.Var(v_t) \end{aligned} \quad (5.3)$$

(s_t and v_t are independent)

¹This model is equivalent to the one given in Equation 5.1.

Since the samples of s_t are identically distributed (stationary process), $Var(s_t) = Var(s_{t-1}) = Var(s)$. Therefore equation 5.3 can be rewritten as:

$$(1 - \alpha^2)Var(s) = (1 - \alpha)^2 \cdot Var(v_t) \quad (5.4)$$

therefore

$$\begin{aligned} \frac{Var(v_t)}{Var(s)} &= \frac{1 - \alpha^2}{(1 - \alpha)^2} \\ &= \frac{1 + \alpha}{1 - \alpha} \end{aligned} \quad (5.5)$$

5.2.3 Relation Between s_t and s_0

We start from equation 5.2.

$$\begin{aligned} s_t &= \alpha \cdot s_{t-1} + (1 - \alpha)v_t \\ &= \alpha^2 \cdot s_{t-2} + \alpha \cdot (1 - \alpha)v_{t-1} + (1 - \alpha)v_t \\ &= \alpha^3 \cdot s_{t-3} + \alpha^2 \cdot (1 - \alpha)v_{t-2} + \alpha \cdot (1 - \alpha)v_{t-1} + (1 - \alpha)v_t \\ &\vdots \\ &= \alpha^t \cdot s_0 + (1 - \alpha) \cdot \sum_{i=1}^t \alpha^{t-i} \cdot v_i \end{aligned} \quad (5.6)$$

5.3 Estimating the Value of α

In this section, we show that α value can be approximated using the autocorrelation coefficient with lag one (r_1). r_1 is estimated from a sample of size N as [69]:

$$r_1 = \frac{\sum_{t=1}^{N-1} [s_t - \bar{s}] \cdot [s_{t+1} - \bar{s}]}{\sum_{t=1}^N [s_t - \bar{s}]^2} \quad (5.7)$$

where \bar{s} is the expected value of process s .

For large values of α (close to one), Equation 5.2 can be approximated as:

$$s_t \approx \alpha \cdot s_{t-1} \quad (5.8)$$

Substituting Equation 5.8 in Equation 5.7 yields:

$$\begin{aligned} r_1 &\approx \frac{\sum_{t=1}^{N-1} [s_t - \bar{s}] \cdot [\alpha \cdot s_t - \bar{s}]}{\sum_{t=1}^N [s_t - \bar{s}]^2} \\ &\approx \frac{\sum_{t=1}^{N-1} [s_t - \bar{s}] \cdot [\alpha \cdot (s_t - \bar{s}) - (1 - \alpha) \cdot \bar{s}]}{\sum_{t=1}^N [s_t - \bar{s}]^2} \quad (5.9) \\ &\approx \frac{\alpha \cdot \sum_{t=1}^{N-1} [s_t - \bar{s}]^2}{\sum_{t=1}^N [s_t - \bar{s}]^2} \quad (\alpha \text{ close to } 1) \end{aligned}$$

For large N , Equation 5.9 can be rewritten as:

$$r_1 \approx \alpha \quad (5.10)$$

Therefore for a large value of α and N , as is the case here, α can be estimated using the autocorrelation coefficient with lag one.

5.4 Distribution of the Average of n Correlated Samples

In this section, we obtain the mean and variance of the samples of a new process whose samples are the average of n samples from the original process.

5.4.1 Mean of the Distribution of the Average of n Samples

We use $A(n)$ to denote the random variable whose value is the average of n samples (from $t = 0$ to $t = n - 1$) of the original process s_t , $n > 1$. Since

$$A(n) = \frac{1}{n} \cdot \sum_{j=0}^{n-1} s_j \quad (5.11)$$

therefore, $E(A(n)) = E(s_t)$. The mean of the distribution of the average of n samples is the same as the mean of the distribution of each sample.

5.4.2 Variance of the Distribution of the Average of n Samples

From equation 5.6, $A(n)$ can be written as:

$$\begin{aligned} A(n) &= \frac{1}{n} \cdot \sum_{j=0}^{n-1} \left\{ \alpha^j \cdot s_0 + (1 - \alpha) \cdot \sum_{i=1}^j \alpha^{j-i} \cdot v_i \right\} \\ &= \frac{1}{n} \cdot \left\{ \frac{1 - \alpha^n}{1 - \alpha} \cdot s_0 + (1 - \alpha) \cdot \sum_{j=1}^{n-1} \sum_{i=1}^j \alpha^{j-i} \cdot v_i \right\} \end{aligned} \quad (5.12)$$

therefore,

$$Var(A(n)) = \frac{1}{n^2} \cdot \left\{ \left(\frac{1 - \alpha^n}{1 - \alpha} \right)^2 \cdot Var(s_0) + (1 - \alpha)^2 \cdot \sum_{j=1}^{n-1} \sum_{i=1}^j \alpha^{2 \cdot (j-i)} \cdot Var(v_i) \right\} \quad (5.13)$$

from equation 5.5

$$\begin{aligned} Var(A(n)) &= \frac{Var(s_0)}{n^2} \cdot \left\{ \left(\frac{1 - \alpha^n}{1 - \alpha} \right)^2 + (1 - \alpha^2) \cdot \sum_{j=1}^{n-1} \sum_{i=1}^j \alpha^{2 \cdot (j-i)} \right\} \\ &= \frac{Var(s_0)}{n^2} \cdot \left\{ \left(\frac{1 - \alpha^n}{1 - \alpha} \right)^2 + (1 - \alpha^2) \cdot \sum_{j=1}^{n-1} \frac{1 - \alpha^{2 \cdot j}}{1 - \alpha^2} \right\} \\ &= \frac{Var(s_0)}{n^2} \cdot \left\{ \left(\frac{1 - \alpha^n}{1 - \alpha} \right)^2 + n - 1 - \alpha^2 \cdot \frac{1 - \alpha^{2 \cdot (n-1)}}{1 - \alpha^2} \right\} \end{aligned} \quad (5.14)$$

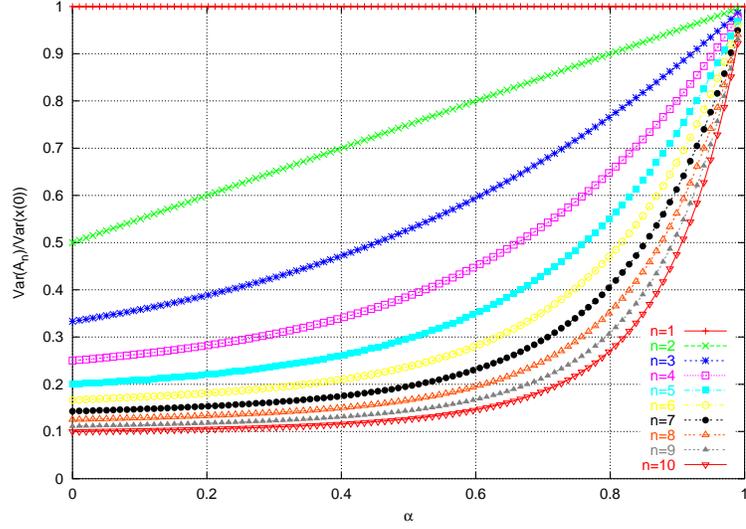


Figure 5.1: Ratio between the variances of the averaging process and the original process for different values of α and n

Since s_t is a stationary process, $Var(s_0) = Var(s)$ and the final relation between $Var(A(n))$ and $Var(s)$ is:

$$Var(A(n)) = \frac{Var(s)}{n^2} \cdot \left\{ \left(\frac{1 - \alpha^n}{1 - \alpha} \right)^2 + n - 1 - \alpha^2 \cdot \frac{1 - \alpha^{2 \cdot (n-1)}}{1 - \alpha^2} \right\} \quad (5.15)$$

Note that when $\alpha = 0$ (i.e. the samples of s_t are independent), Equation 5.16 reduces to:

$$Var(A(n)) = \frac{Var(s)}{n} \quad (5.16)$$

Figure 5.1 shows the ratio between the variance of the averaging process and original process for different values of α and n . The variance of the averaging process, $Var(A(n))$, is always less than or equal to the variance of the original process, $Var(s)$, being equal in case α equals one. Intuitively, the lower the variance of the signal strength distribution at each location, the better the ability to discriminate between different locations and the better the accuracy.

5.5 Modified *Horus* Algorithm

In this section, we use the results of the previous section to obtain the distribution of the average of n correlated samples. We use this value to determine the most probable user location. We assume that the *Horus* system is running in the parametric mode where the signal strength distribution follows a *Gaussian* distribution. Since the individual distribution of each sample follows a *Gaussian* distribution, the probability distribution of the average of n samples follows a *Gaussian* distribution whose mean and variance can be obtained using the results in Section 5.4.

The technique works as follows:

- *Offline phase*: the system calculates the parameters of the distribution of the average of n samples for each access point in the radio map.
- *Online phase*: Given n samples from an access point, the algorithm obtains their average and calculate the probability of each radio map location given this value of the average using the distribution of the average of n samples calculated during the offline phase.

Algorithm 3 shows the details of the modified *Horus* algorithm. Note that the value of α is implicitly used in the online phase as the distribution of the average of n samples depends on the value of α as discussed in Section 5.4.

5.6 Experimental Evaluation

In this section we present the result of implementing the correlation handling technique in the context of the *Horus* system.

Algorithm 3 $x = \text{Corr_GetLocation}(n, S, \mathbb{X}, P_RM)$

Input:

n : Number of samples from each access point.

S : Measured signal strength vectors from k access points ($S = (\vec{s}_1, \dots, \vec{s}_k)$).

Each $\vec{s}_i, 1 \leq i \leq k$ is a vector containing n samples from access point i .

\mathbb{X} : Radio map locations.

RM : Radio map, where $RM[a][x](q)$ represents the *pdf* of the average of n signal strength samples from access point a at location $x \in \mathbb{X}$.

Output:

The location $x \in \mathbb{X}$ that maximizes $P(x/S)$.

```
1: for  $i = 1..k$  do
2:    $Avg(i) \leftarrow average(\vec{s}_i)$ 
3: end for
4:  $Max \leftarrow 0$ 
5: for  $l \in \mathbb{X}$  do
6:    $P \leftarrow \prod_{i=1}^k \int_{Avg(i)-0.5}^{Avg(i)+0.5} RM[i][x](q) dq$ 
7:   if  $P > Max$  then
8:      $x \leftarrow l$ 
9:      $Max \leftarrow P$ 
10:  end if
11: end for
```

Figures 5.2 and 5.3 show the performance of the *Horus* system when taking the correlation into account and without taking the correlation into account for the two testbeds. The figures show that under the independence assumption, as the number of averaged samples increases, the performance degrades. The minimum value at $n = 2$ for the first testbed can be explained by noting that there are two opposing factors affecting the system accuracy:

1. as the number of averaged samples n increases, the accuracy of the system should increase.
2. as n increases, the estimation of the distribution of the average of the n samples becomes worse due to the wrong independence assumption.

At low values of n ($n = 1, 2$) the first factor is the dominating factor and hence the accuracy increases. Starting from $n = 3$, the effect of the bad estimation of the distribution becomes the dominating factor and accuracy degrades.

Using the modified technique, the system average accuracy is enhanced by about 0.5 foot for both testbeds. This is equivalent to more than 19% enhancement for the first testbed and 11% for the second testbed for $n = 5$.

5.7 Summary

In order to increase accuracy, the system needs to use multiple samples from the same access point to account for the variation of signal strength. In this chapter, we showed that the correlation of the samples from the same access point can be as high as 0.9. Assuming independence of samples from the same access point leads to degraded performance as the number of averaged samples increases. Therefore, we introduced an

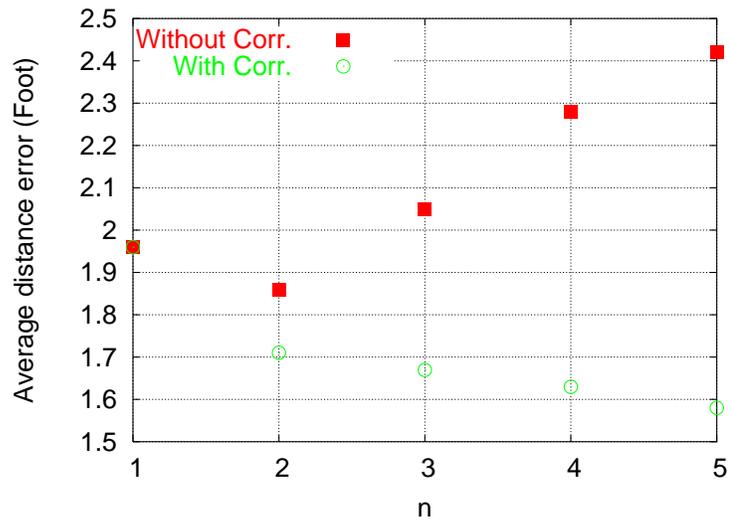


Figure 5.2: Average distance error with and without taking correlation into account for the first testbed.

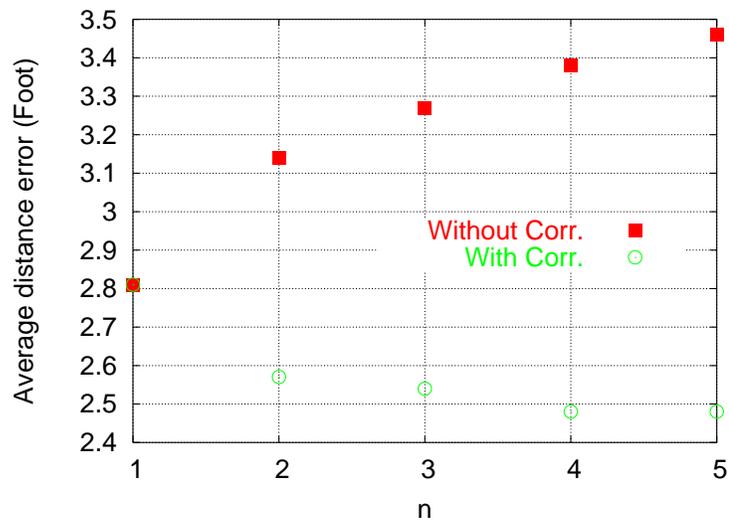


Figure 5.3: Average distance error with and without taking correlation into account for the second testbed.

autoregressive model for handling the correlation between samples from the same access point. Based on this model, we presented a technique to take multiple samples from an access point into account to enhance the accuracy. Using the modified technique, the system average accuracy is enhanced by more than 19% for the first testbed and 11% for the second testbed using five signal strength samples.

Chapter 6

Continuous-Space Estimation

In our discussion in previous chapters, we assume a discrete-space estimation where the system returns a single location from the set of location in the radio map. To increase the system accuracy, the *Horus* system uses two techniques to obtain a location estimate in the continuous space.

In this chapter, we discuss the two techniques and show their effect on the performance of the *Horus* system.

6.1 Technique 1: Center of Mass of the Top Candidate Locations

The first technique is based on treating each location in the radio map as an object in the physical space whose weight is equal to the normalized probability¹ assigned by the discrete-space estimation process. We then obtain the center of mass of the N objects with the largest mass, where N is a parameter to the system.

More formally, let $p(x)$ be the probability of a location $x \in \mathbb{X}$, i.e. the radio map, and let $\bar{\mathbb{X}}$ be the set of locations in the radio map *ordered* in a descending order

¹The normalization is used to ensure that the sum of the probabilities of all locations equals one.

according to the normalized probability. The center of mass technique estimates the current location x as:

$$x = \frac{\sum_{i=1}^{\min(N, \|\bar{\mathbb{X}}\|)} p(i) * \bar{\mathbb{X}}(i)}{\sum_{i=1}^{\min(N, \|\bar{\mathbb{X}}\|)} p(i)} \quad (6.1)$$

where $\bar{\mathbb{X}}(i)$ is the i^{th} element of $\bar{\mathbb{X}}$

Note that the estimated location x need not to be one of the radio map locations. Algorithm 4 shows the details of the center of mass technique.

Algorithm 4 $x = \text{Mass_GetLocation}(s, \mathbb{X}, RM, N)$

Input:

- s : Measured signal strength vector ($s = (s_1, \dots, s_k)$).
- \mathbb{X} : Radio map locations.
- RM : Radio map.
- N : Number of locations to use in the algorithm.

Output:

Estimated location x .

- 1: **for** $l \in \mathbb{X}$ **do**
 - 2: $p(l) \leftarrow$ probability of location l obtained using the discrete-space estimator.
 - 3: **end for**
 - 4: sort p in a descending order and sort \mathbb{X} accordingly. Call the sorted location list $\bar{\mathbb{X}}$.
 - 5: $x \leftarrow \frac{\sum_{i=1}^{\min(N, \|\bar{\mathbb{X}}\|)} p(i) * \bar{\mathbb{X}}(i)}{\sum_{i=1}^{\min(N, \|\bar{\mathbb{X}}\|)} p(i)}$
-

Intuitively, the higher the probability of a location, the closer the estimated position should be to this location. For example, consider the case in Figure 6.1. The figure shows the simple case where the radio map consists of two locations: A and B . As-

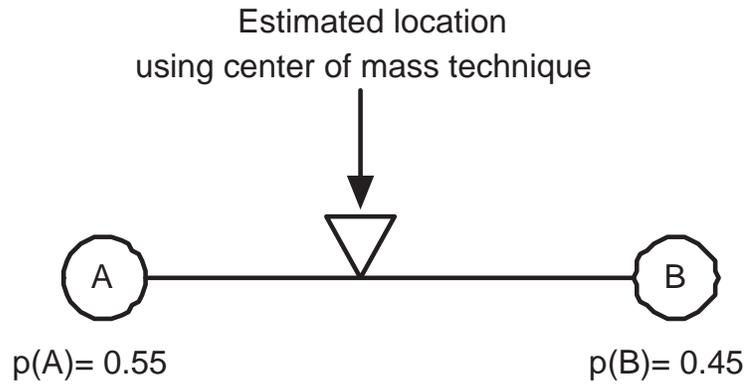


Figure 6.1: An example of the center of mass technique. Here location A has probability of 0.55 and location B has a probability of 0.45. The center of mass technique will return the estimated location in the middle of the two locations, slightly biased to location A . The discrete-space estimator will return location A as the location estimate.

sume that the user is standing in the middle between the two locations. In this case, the signal strength vector obtained from this location will be different from the signal strength vector expected at either location A or location B . Let's assume that for the given signal strength vector the discrete-space estimation process assigns a normalized probability of 0.55 for location A and 0.45 for location B . In this case, the discrete-space estimation process will return location A as the estimated location. However, the technique proposed in this section would return a location estimate in the middle of the two locations, slightly biased towards location A , which should be a more accurate estimate to the actual location than location A . We study the performance of this technique compared to the discrete-space estimator in Section 6.3.

6.2 Technique 2: Time-Averaging in the Physical Space

The second technique uses a time-average window to smooth the resulting location estimate. The technique obtains the location estimate by averaging the last W locations estimates obtained by either the discrete-space estimator or the continuous-space estimator discussed in the previous section.

More formally, given a stream of location estimates x_1, x_2, \dots, x_t , the technique estimates the current location \bar{x}_t at time t as:

$$\bar{x}_t = \frac{1}{\min(W, t)} \cdot \sum_{i=t-\min(W,t)+1}^t x_i \quad (6.2)$$

Algorithm 5 shows the details of the time-averaging of mass technique.

6.3 Experimental Evaluation

In this section, we compare the performance of the two techniques for obtaining a continuous-space location estimate. For the time-averaging technique, we choose to use the discrete-space estimator, as the estimation method that the technique uses, in order to show how it improves the basic technique.

6.3.1 Center of Mass Technique

Figures 6.2 and 6.3 show the effect of increasing the parameter N (number of locations to interpolate between) on the performance of the center of mass technique for the two testbeds. Note that the special case of $N = 1$ is equivalent to the discrete-space estimator output. The figures show that the performance of the *Horus* system is enhanced by more than 13% for the first testbed and more than 6% for the second testbed for $N = 6$.

Algorithm 5 $x = \text{TAvg_GetLocation}(s, \mathbb{X}, RM, W, Q)$

Input:

s : Measured signal strength vector ($s = (s_1, \dots, s_k)$).

\mathbb{X} : Radio map locations.

RM : Radio map.

W : Size of the averaging window.

Q : A queue of size W to hold past location estimates. Initially empty

Output:

Estimated location x .

- 1: $tx \leftarrow \text{GetLocation}(s, \mathbb{X}, RM)$. {obtain location estimate normally}
 - 2: **if** $Q.\text{IsFull}()$ **then**
 - 3: $Q.\text{remove}()$. {keep the last W estimates only}
 - 4: **end if**
 - 5: $Q.\text{insert}(tx)$.
 - 6: $x \leftarrow Q.\text{average}()$. {average of all elements in Q }
-

The figures also show that there is an optimal value for the parameter N around $N = 6$.

6.3.2 Time-averaging Technique

Figures 6.4 and 6.5 show the effect of increasing the parameter W (size of the averaging window) on the performance of the time-averaging technique for the two testbeds. The figures show that the larger the averaging window, the better the accuracy. The performance of the *Horus* system is enhanced by more than 24% for the first testbed and more than 15% for the second testbed for $W = 10$.

6.4 Summary

In this chapter, we described two techniques for allowing continuous-space estimation: the *Center of Mass* technique and the *Time-Averaging* technique. Using the *Center of Mass* technique, the performance of the *Horus* system is enhanced by more than 13% for the first testbed and more than 6% for the second testbed compared to the basic technique. The *Time-Averaging* technique enhances the performance of the *Horus* system by more than 24% for the first testbed and more than 15% for the second testbed. The two techniques are independent and can be applied together.

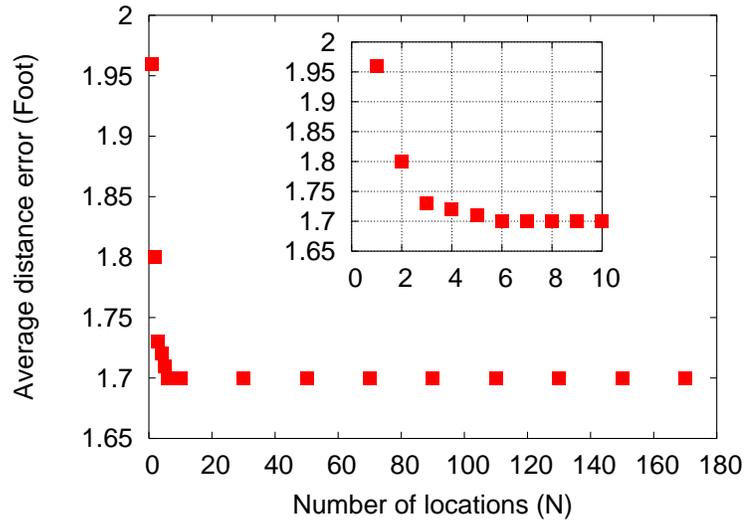


Figure 6.2: Average distance error using the center of mass technique for the first testbed. The sub-figure shows the same curve for $N = 0 - 10$.

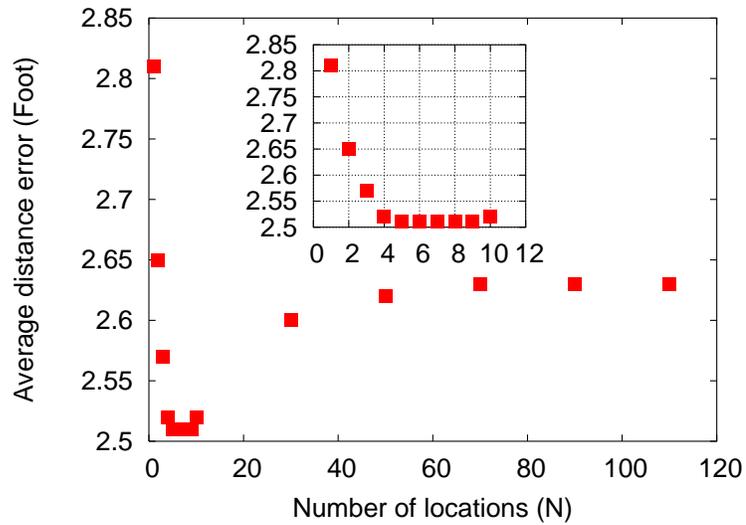


Figure 6.3: Average distance error using the center of mass technique for the second testbed. The sub-figure shows the same curve for $N = 0 - 12$.

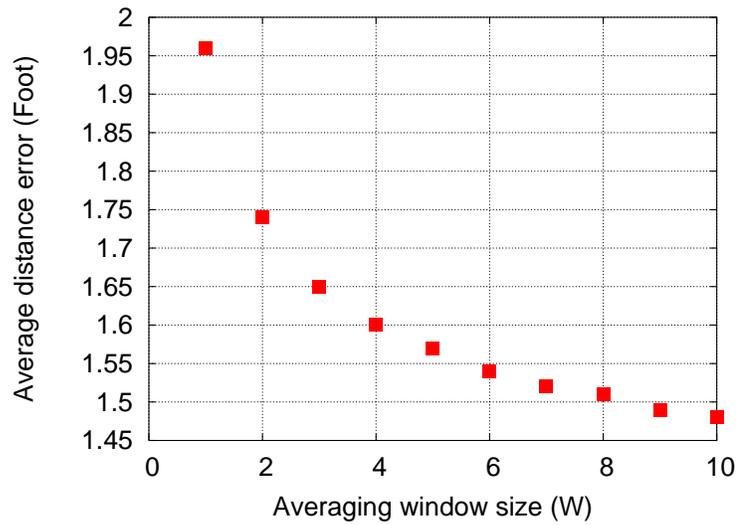


Figure 6.4: Average distance error using the time-averaging technique for the first testbed. The sub-figure shows the same curve for $W = 0 - 10$.

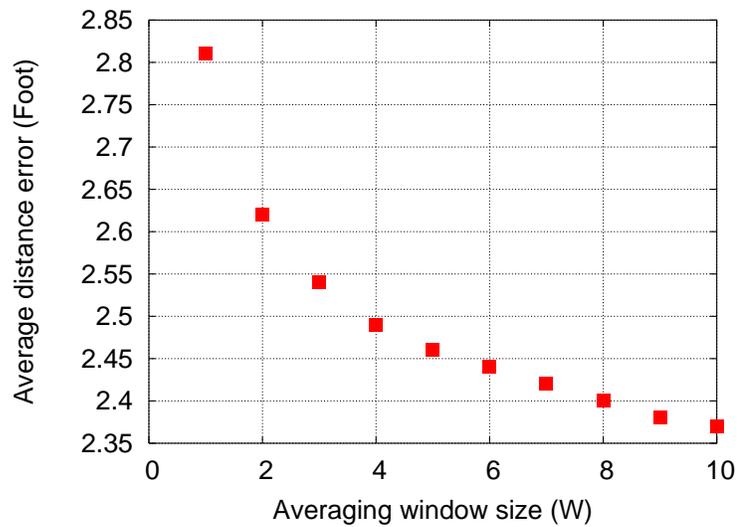


Figure 6.5: Average distance error using the time-averaging technique for the second testbed. The sub-figure shows the same curve for $W = 0 - 10$.

Chapter 7

Radio-Map Locations Clustering

This chapter describes the technique used by the *Horus* system to reduce the computational requirements of the location determination algorithm. This is based on clustering radio map locations. We present two radio map location clustering techniques: the *Explicit Clustering* technique and the *Implicit Clustering* technique.

7.1 Introduction

We define a *cluster* as a set of locations sharing a common set of access points. We call this common set of access points the *cluster key*. The problem can be stated as: Given a location x , we want to determine the cluster to which x belongs. The noisy characteristics of the wireless channel described in Chapter 3 make clustering a challenging problem because the number of access points covering a location varies with time.

We present two approaches:

- *Explicit clustering*: Here the system must determine the clusters during the offline training phase as a separate step.
- *Implicit clustering*: Here, no special processing is performed in the offline phase.

However, during the location determination phase, the system performs clustering implicitly.

The details of the algorithms are given next.

7.2 Explicit Clustering

One way to do clustering is to group locations according to the access points that cover them. i.e. two locations x_1 and x_2 are placed in the same cluster *iff* the set of access points covering these locations are identical. However, this approach for clustering has problems when applied in a real environment. As shown in Figure 3.5, an access point may be missing from some of the samples and, therefore, using the entire set of access points that cover a location for clustering may fail to find the correct cluster due to the missing access point.

Instead, we use a subset of this set containing only q elements and the problem becomes: Given a number q , we want to put all the locations that share q access points in one cluster. Therefore, we have 2 sub-problems:

1. How to determine the value of q ?
2. Which q access points to choose for clustering?

For the first sub-problem, we need to choose q such that all locations are covered by at least q access points most of the time. This factor lessens the effect of variability in the number of access points with time. This suggests that the value of q should be less than or equal to the minimum number of access points covering any location in the radio map. Moreover, we need a value for q that distributes locations evenly between the clusters to reduce the required computations. Experiments showing the effect of the parameter q on performance are given in Section 7.5.

The solution for sub-problem 2 is related to the solution of sub-problem 1. If the number of access points covering a location is varying with time, which access points should we choose? Intuitively, we should choose the access points that appear most of the time in the samples. Figure 3.5 suggests that we should choose to use the q access points with the largest signal strength values at each location.

During the data analysis we found that, at some locations, the order of the access points with the largest signal strength values changes when the signal strength values from these access points are near to each other. Therefore, we choose to treat the q access points as a set and not as an ordered tuple. For example, if $q = 2$ and the two access points with the largest and second largest signal strength value at location x_1 are (AP_1, AP_2) respectively, and (AP_2, AP_1) for another location x_2 , then we place location x_1 and location x_2 in the same cluster regardless of the order of the access points.

To summarize, for a given location x , we use the set of the q strongest access points covering this location to determine the cluster to which it belongs. Therefore, the cluster key in the *Explicit Clustering* approach is the set of the q access points used to group the locations in this cluster.

We call the modified location determination algorithm, that uses the *Explicit Clustering* technique, the *Joint Clustering* algorithm (Algorithm 6). The algorithm is identical to the previously described *Horus* algorithm with the exception of reducing the radio-map space to a single cluster.

Algorithm 6 $x = \text{JC_GetLocation}(s, \mathbb{X}, RM, Cluster, q)$

Input:

s : Measured signal strength vector ($s = (s_1, \dots, s_k)$).

RM : Radio map.

$Cluster$: Clustering function where $Cluster(k)$ is the set of radio-map locations whose key is k .

q : Number of access points to use in clustering.

Output:

Estimated location x .

1: $OrderedS \leftarrow s$ sorted in a descending order.

2: $CandidateList \leftarrow Cluster(OrderedS(1..q))$

3: $x \leftarrow GetLocation(s, CandidateList, RM)$.

{Get the candidate location using standard algorithms}

7.3 Implicit Clustering

One can look at the clustering problem from a different viewpoint. Each access point defines a subset of the radio map locations that are covered by this access point. These locations can be viewed as a cluster of locations whose key is the access point covering the locations in this cluster. If during the location determination phase we use the access points incrementally, one after the other, then starting with the first access point, we restrict our search space to the locations covered by this access point. The second access point chooses only the locations in the range of the first access point and covered by the second access point and so on, leading to a multi-level clustering process.

Notice that no preprocessing is required in the offline training phase. During the online phase, a location x belongs to a cluster whose key is access point a if there is information about access point a at location x in the radio map.

We call the modified location determination algorithm, that uses the *Implicit Clustering* technique, the *Incremental Triangulation* algorithm (Algorithm 7). The algorithm works as follows. Given a sequence of observations from each access point, we start by sorting the access points in descending order according to the average received signal strength. For the first access point, the one with the strongest average signal strength, we calculate the probability of each location in the radio map set given the observation sequence from this access point alone. This gives us a set of candidate locations (locations that have non-zero probability). If the probability of the most probable location is “significantly” higher (according to a threshold) than the probability of the second most probable location, we return the most probable location as our location estimate, after consulting only one access point. If this is not the case, we go to the next access point in the sorted access point list. For this access point, we repeat the same process again, but only for the set of candidate locations obtained from the

first access point.

This process of calculating the probabilities and determining the significance of the most probable location is repeated incrementally, for each access point in order, until the location can be estimated or all access points are consulted. In the latter case, the algorithm returns the most probable location in the candidate list that remains after consulting all the access points. We call our approach the *Incremental Triangulation* technique as we start by a set of candidate locations using the first access point and reduce this set using other access points iteratively. In contrast, the standard triangulation approach starts by an infinite number of locations on a circle and reduces this number to 2 points using another circle and finally reduces these two points to only one point using a third circle (assuming a perfect environment). However, typically this is done by solving a set of nonlinear equations and not in an iterative manner.

7.4 Discussion

Both clustering techniques reduce the search space and thus lead to a reduction of the computational cost of location determination techniques employing these clustering approaches. Moreover, using clustering helps in scaling the system to a larger coverage area.

The *Explicit Clustering* divides the radio-map space \mathbb{X} in a number of flat clusters. However, the *Implicit Clustering* tries to use the access points incrementally, one after the other, until it can estimate the location with certain accuracy leading to clustering at multi-levels and hence to a more reduced search space than the *Explicit Clustering* approach, and hence fewer number of operations, on the average, per sample. However, treating each access point incrementally, instead of using the joint distribution,

Algorithm 7 $x = \text{IT_GetLocation}(s, \mathbb{X}, RM, \text{Threshold})$

Input:

s : Measured signal strength vector ($s = (s_1, \dots, s_k)$).

\mathbb{X} : Radio map locations.

RM : Radio map.

Threshold : Stopping threshold.

Output:

Estimated location x .

- 1: $\text{OrderedS} \leftarrow s$ sorted in a descending order.
- 2: $\text{CandidateList} \leftarrow \mathbb{X}$
- 3: $\text{CurrentAP} \leftarrow 1$.
- 4: $\text{Done} \leftarrow \text{false}$.
- 5: **while** Not Done **do**
- 6: **for** $l \in \text{CandidateList}$ **do**
- 7: $\bar{s} \leftarrow \text{OrderedS}(1..\text{CurrentAP})$.
- 8: $p(l) \leftarrow$ probability of location l using
 $\text{GetLocation}(\bar{s}, \text{CandidateList}, RM)$.
 {Get the probability using standard algorithms}
- 9: **end for**
- 10: sort p in a descending order and sort CandidateList accordingly.
- 11: **if** $(\frac{p(\text{CandidateList}(1)) - p(\text{CandidateList}(2))}{p(\text{CandidateList}(1))} > \text{Threshold} \parallel \text{CurrentAP} == k)$
 then
- 12: $\text{Done} = \text{true}$
- 13: **else**
- 14: $\text{CurrentAP} = \text{CurrentAP} + 1$.
- 15: $\text{CandidateList} \leftarrow$ Elements with non-zero probability of CandidateList .
- 16: **end if**
- 17: **end while**
- 18: $x \leftarrow \text{CandidateList}(1)$

leads to the loss of some information and thus the accuracy of the *Implicit Clustering* is lower than the *Explicit Clustering* technique.

It is important to note that the clustering techniques presented in this chapter are general and can be used with all the current WLAN location determination systems.

7.5 Experimental Results

In this section, we show the performance evaluation of the Joint Clustering algorithm and the Incremental Triangulation algorithm compared to the basic *Horus* algorithm.

7.5.1 Joint Clustering Technique

Figures 7.1 and 7.2 show the effect of the parameter q (number of access points used in clustering) on the performance of the clustering process of the *Joint Clustering* technique for the two testbeds. For this experiment, we change the value of q from 1 to 6 and calculate the number of clusters, the average size of each cluster, and the standard deviation of the cluster size. From the figures we can see that as q increases, the number of clusters increases and the average size of each clusters decreases until we reach a point where the number of clusters decreases. For the standard deviation, the variation of the size of the clusters decreases until we reach a minimum value and it increases again. A small value for the standard deviation means that the sizes of the clusters are more uniform, which is a desirable property. The maximum value of the number of clusters can be explained by noting that as q increases more locations are differentiated into different clusters due to the addition of new access points. When q is increased past the critical value, different locations start to share the same access points, especially for locations close to each other, and clusters merge.

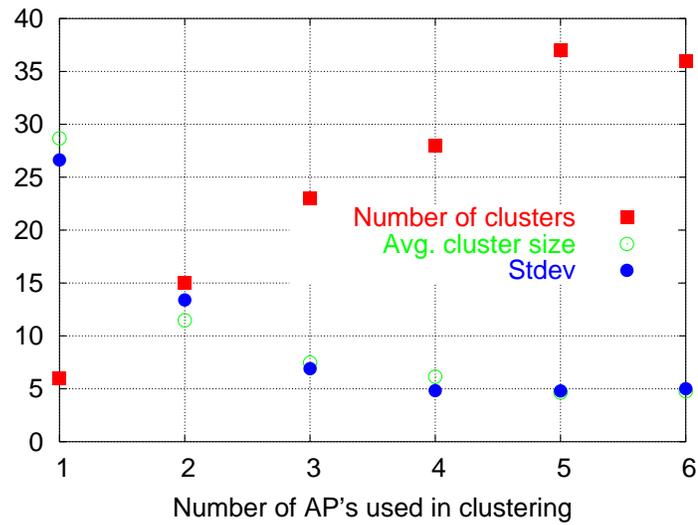


Figure 7.1: Effect of the parameter q on the performance of the clustering process for the first testbed.

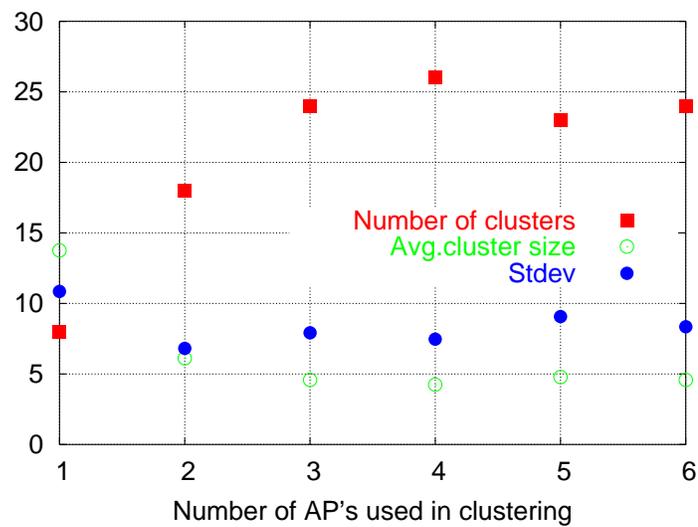


Figure 7.2: Effect of the parameter q on the performance of the clustering process for the second testbed.

Figures 7.3 shows the effect of the parameter q on the average distance error for the two testbeds. We can see that as q increases, the accuracy of the system slightly increases. Since, clustering reduces the search space, clustering can enhance the system accuracy.

Figures 7.4 shows the effect of the parameter q on the average number of operations per location estimate. As q increases, the average number of operations per location estimate decreases. For this range of q values, as q increases, the average cluster size decreases and hence the system uses a fewer number of operations per location estimates.

We use a value of $q = 4$ for the rest of the thesis.

7.5.2 Incremental Triangulation Technique

Figures 7.5 and 7.6 (the “No Clustering” case is not shown for clarity) shows the effect of the parameter *Threshold* on the performance of the two testbeds. For small values of the *Threshold* parameter, the decision is taken quickly after examining a small number of access points. As the threshold value increases, more access points are consulted to reach a decision. As the number of access points consulted increases, the number of operations per location estimate increases and the so does the accuracy.

For the rest of the paper, we set the value of the *Threshold* parameter to 0.1.

7.5.3 Discussion

The results show that using clustering reduces the average number of operations per location estimate by more than an order of magnitude. The *Joint Clustering* technique gives slightly better accuracy than the *Incremental Triangulation* technique. However, The average number of operations performed per location estimate for the *Incremental*

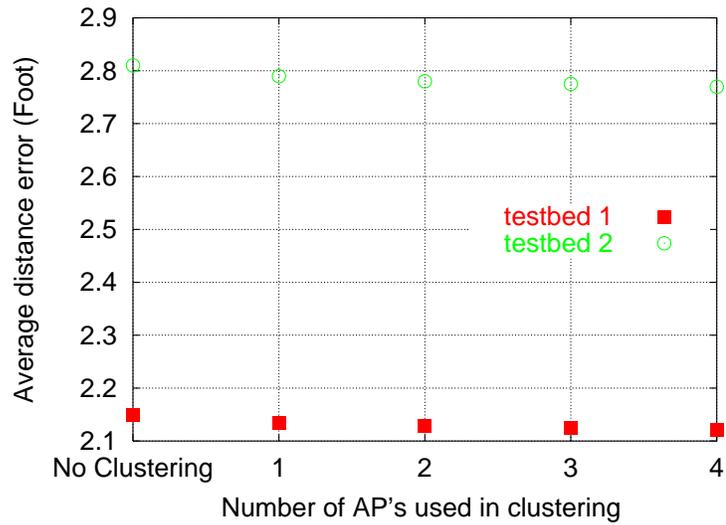


Figure 7.3: Effect of the parameter q on the average distance error for the two testbeds.

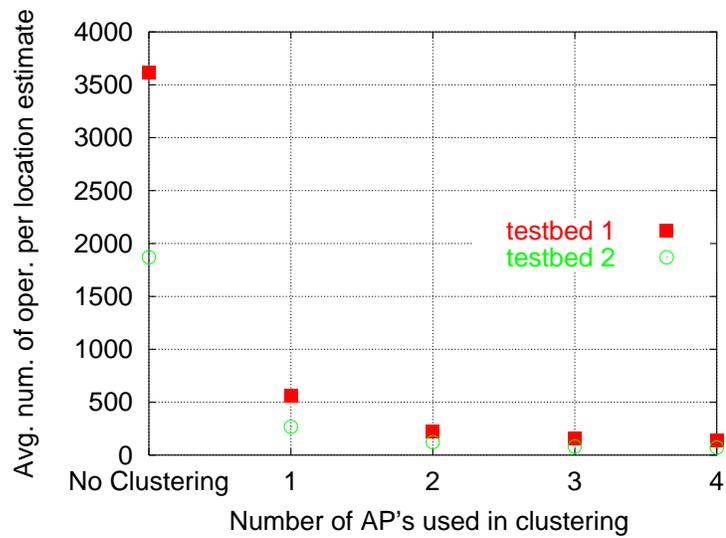


Figure 7.4: Effect of the parameter q on the average number of operations per location estimate for the two testbeds.

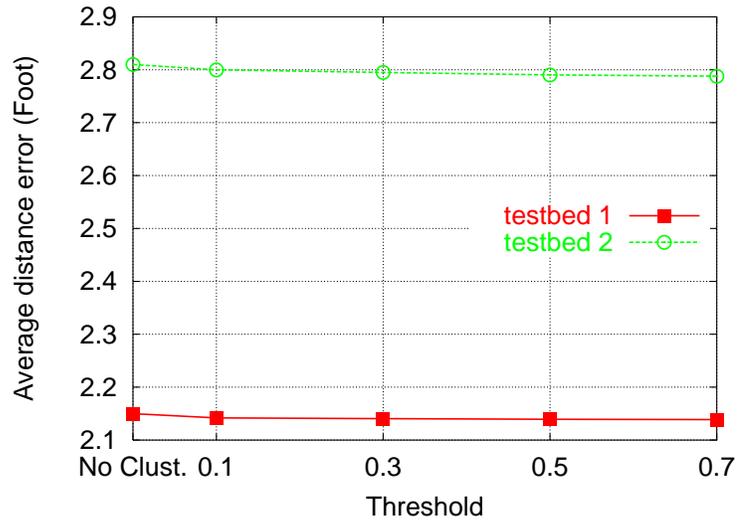


Figure 7.5: Effect of the parameter *Threshold* on the average distance error for the two testbeds.

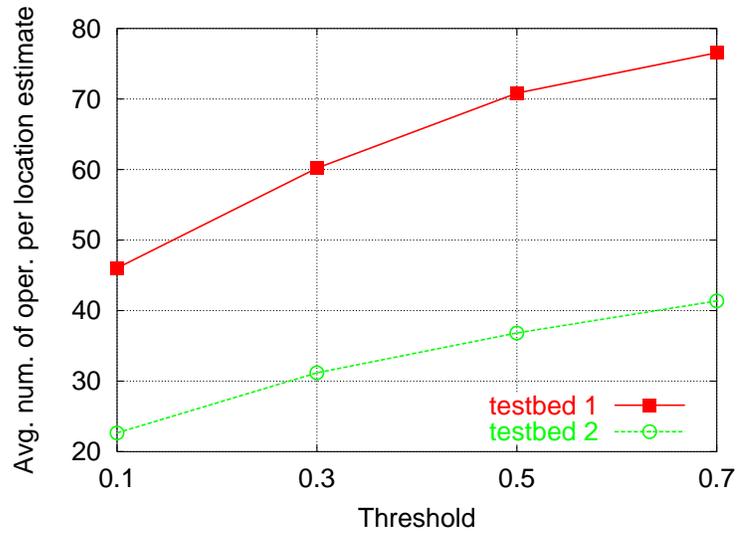


Figure 7.6: Effect of the parameter *Threshold* on the average number of operations per location estimate for the two testbeds.

Triangulation technique is lower than the corresponding number of the *Joint Clustering* technique. Therefore we have a tradeoff here, if one is interested in accuracy more than power consumption then the *Joint Clustering* technique is the one to use. If on the other hand the power consumption is the key factor then one should choose the *Incremental Triangulation* technique as it leads to less computations and hence better power consumption.

7.6 Summary

In this chapter, we introduced clustering of radio map locations as an approach to reduce the computational requirements of the location determination algorithms and achieve scalability. We describe two general approaches to clustering: *Explicit Clustering* and *Implicit Clustering*. The results show that using clustering reduces the average number of operations per location estimate by more than an order of magnitude. The *Explicit Clustering* technique gives slightly better accuracy than the *Implicit Clustering* technique. However, The average number of operations performed per location estimate for the *Implicit clustering* technique is much lower than the corresponding number of the *Explicit Clustering* technique. We also show that clustering enhances the accuracy as the search space is reduced by the clustering process.

Chapter 8

Small-Scale Compensation

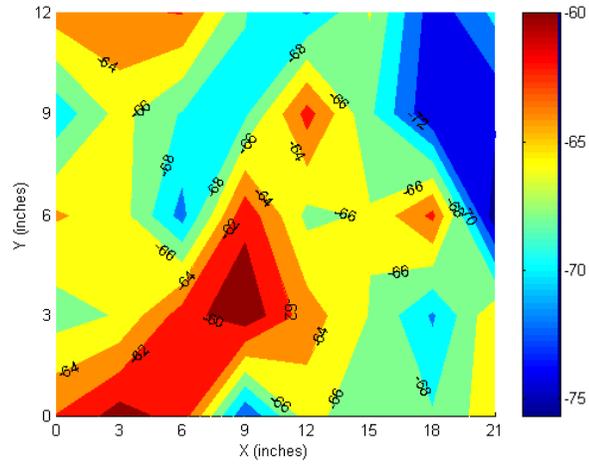
This chapter describes how the *Horus* system handles small scale variations. We start by describing an experiment to show the effect of small-scale variations and then present the perturbation technique, used by the *Horus* system, to identify and overcome these variations.

8.1 Small-Scale Variations

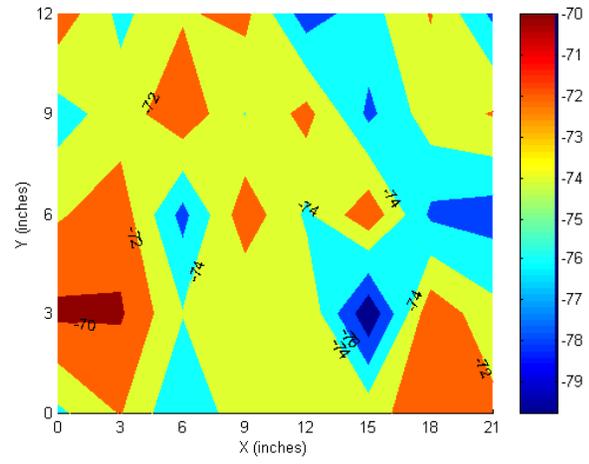
We performed an experiment to measure the signal strength from different access points in an area of 12×21 inches (approximately 1×2 feet). Figure 8.1 shows the signal strength contours for different access points. We can note two things from this figure.

1. The signal strength can vary by as much as 10 dBm in a distance as small as 3 inches.
2. The degree of variation depends on the average signal strength: The higher the average signal strength, the higher the variations.

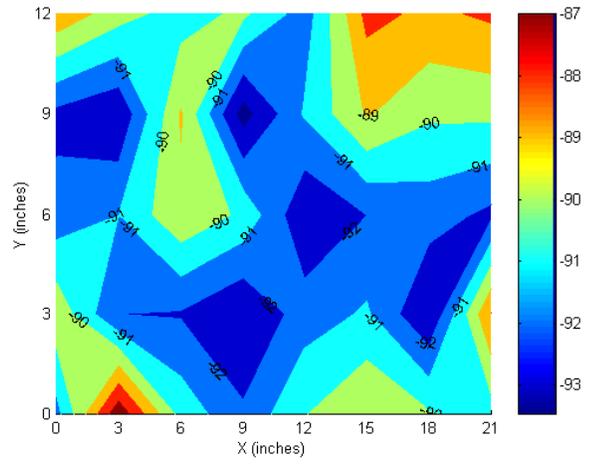
Dealing with small-scale variations is challenging and *none* of the current systems handles it. Since the selected radio map locations are typically placed few feet apart,



(a) AP 1



(b) AP 2



(c) AP 3

Figure 8.1: Small-scale variations: signal strength contours for different access points.

to limit the radio map size, the radio map does not capture small-scale variations. This contributes significantly to the estimation errors in the current systems. One way to handle small-scale variations is to increase the granularity of the resulting radio map, which is not practical in terms in the size of the radio map and the time required to build this radio map. In the next section, we present a new technique to handle small-scale variations.

8.2 The Perturbation Technique

We propose a new technique, the Perturbation technique, to handle the small-scale variations. The technique is based on two sub-functions:

1. Detecting the effect of small-scale variations.
2. Compensating for small-scale variations.

8.2.1 Detecting Small Scale Variations

In order to detect small-scale variations, the *Horus* system uses the heuristic that a user location cannot change faster than his movement rate in a certain amount of time. To do that, the system calculates the estimated location using the standard radio map and the original inference algorithm. The system then calculates the distance between the estimated location and the previous user location. If this distance is above a threshold, based on the user movement rate and estimation frequency, the system detects that there are small-scale variations affecting the signal strength.

8.2.2 Compensating for Small Scale Variations

To compensate for these small-scale variations, the system perturbs the received vector entries, and re-estimates the location. The nearest location to the previous user location is chosen as the final location estimate. For example, if the received signal strength vector is (s_1, s_2, \dots, s_k) , the system perturbs this vector to obtain the set of vectors: $(s_1(1 + e), s_2(1 + e), \dots, s_k(1 + e))$, where $e \in \{-d, 0, d\}$ is the percentage by which to perturb the signal strength (0 for no perturbation).

Typically, three access points ($k = 3$) are sufficient to obtain good accuracy, so the number of perturbed vectors is limited. Moreover, the measurements show that the small-scale variations depend heavily on the strength of the signal strength received. This means that perturbing the component corresponding to the strongest access point can obtain sufficient results. The value of d is chosen to be relative to the received signal strength. The effect of the number of access points to perturb and the value of d on accuracy is described in the next section.

8.3 Experimental Evaluation

To test the performance of the perturbation technique, we performed a set of experiments. The *threshold* parameter t , used to detect small-scale variations, is dependent on both the user speed and location update rate. For the purpose of this paper, we set this threshold to 15 feet for a location update every 2 seconds (maximum user speed of 7.5 feet per second).

Figure 8.2 shows the effect of changing the perturbation percentage (d , which is the amount by which to perturb each access point) on average error. We can see from this figure that the best value for the perturbation percentage is 4% for the first testbed

and 5% for the second testbed. We use these values for the rest of this section.

Figure 8.3 shows the effect of increasing the number of perturbed access points on the average distance error. The access points chosen at a location are the strongest access points in the set of access points that cover that location. The figure shows that perturbation technique is not sensitive to the number of access points. This means that perturbing one access point only is sufficient to enhance the performance.

Figures 8.4 and 8.5 show the effect of using the perturbation technique on the basic *Horus* system for the two testbeds. The perturbation technique enhances the average distance error by more than 25% for the first testbed and more than 21% for the second testbed. Moreover, the worst-case error is enhanced by more than 30% for the two testbeds.

Figure 8.6 shows the effect of changing the number of access points used in the perturbation technique on the average number of operations per location estimates for the two testbeds. Although the average number of operations per location estimates increases significantly for a large number of access points, *perturbing only one access point* leads to the desired accuracy with very low computational overhead (Figure 8.3).

8.4 Summary

In this chapter, we showed that dealing with small-scale variations is challenging and *none* of the current WLAN location determination systems handles it. Since the selected radio map locations are typically placed few feet apart, to limit the radio map size, the radio map does not capture small-scale variations. This contributes significantly to the estimation errors in the current systems. We proposed the *Perturbation*

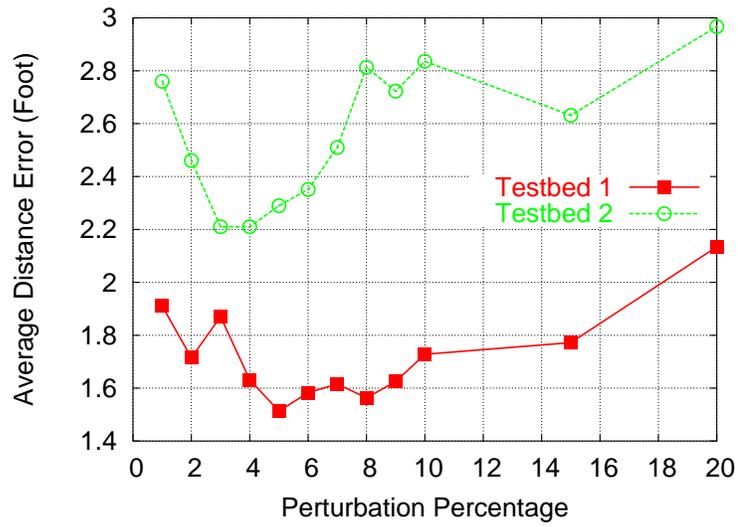


Figure 8.2: Effect of changing the perturbation percentage on average distance error.

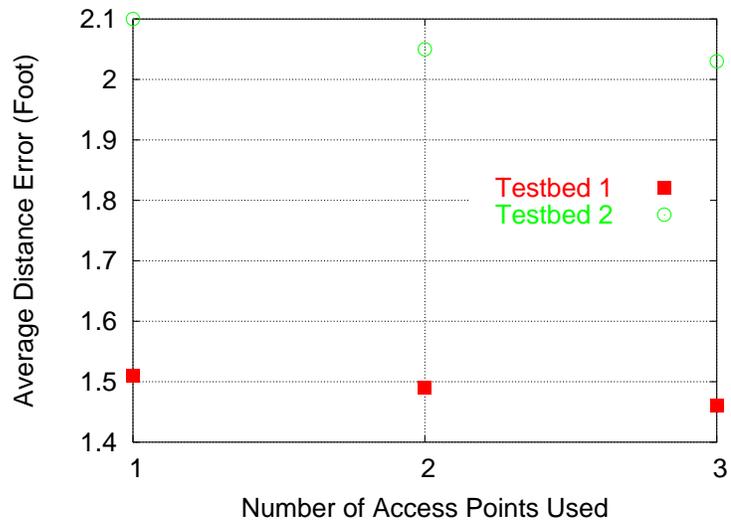


Figure 8.3: Effect of changing the number of perturbed access points on average distance error.

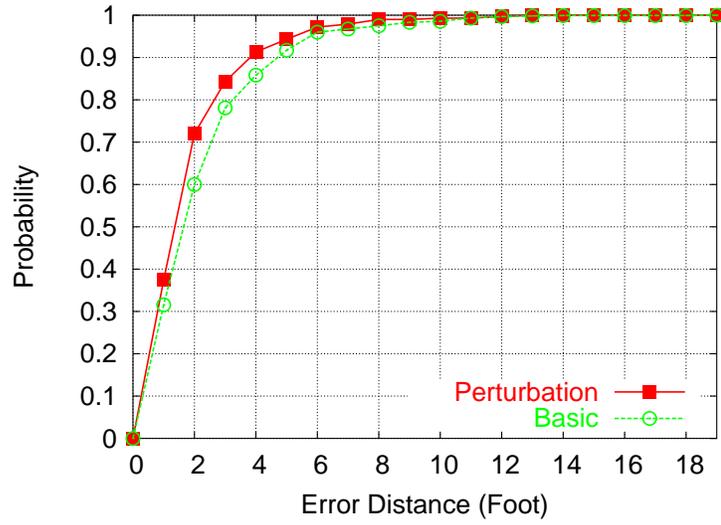


Figure 8.4: CDF for the distance error for the first testbed.

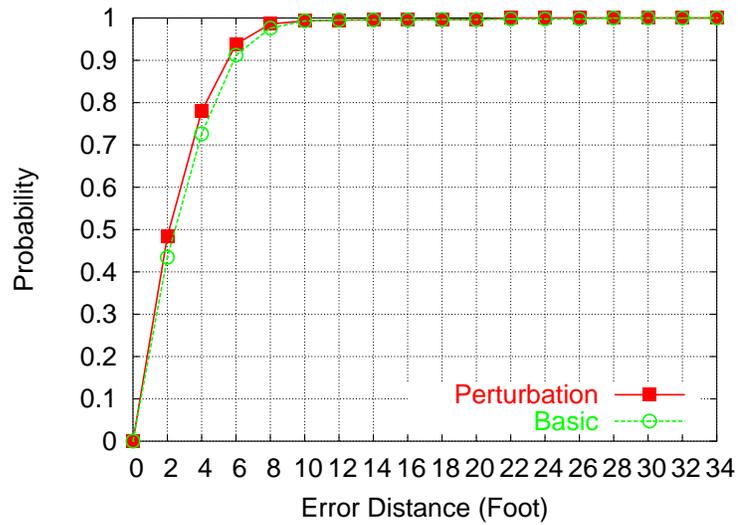


Figure 8.5: CDF for the distance error for the second testbed.

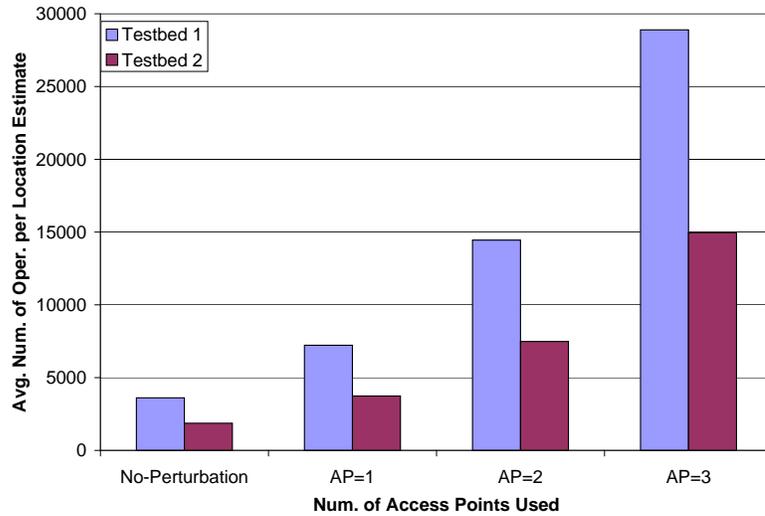


Figure 8.6: Effect of changing the number of perturbed access points on computational requirements.

technique for handling small-scale variations. The perturbation technique enhances the average distance error by more than 25% for the first testbed and more than 21% for the second testbed. Moreover, the worst-case error is enhanced by more than 30% for the two testbeds.

Chapter 9

Performance Evaluation

In this chapter, we compare the performance of the *Horus* system to the performance of the *Radar* system [22]. We also study how sensitive the *Horus* system is to the training time and the distance between the radio map locations. Finally, we show the result of applying the techniques of the *Horus* system to the *Radar* system.

9.1 Comparison with the Radar System

In the previous chapters, we studied the effect of each component of the *Horus* system separately on the performance. In this section, we compare the performance of the full *Horus* system, to the performance of the *Radar* system. We use the parametric distribution technique and the *Incremental Triangulation* technique for the clustering component as it reduces the computational requirements significantly with minimal impact on accuracy. Tables 9.1 and 9.2 show the values of different parameters for the two testbeds.

Figures 9.1 and 9.2 show the performance of the *Horus* system and the *Radar* system for the two testbeds (the curve for the *Radar* system is truncated). Tables 9.3 and 9.4 summarize the results for the two testbeds. The tables show that the *Horus*

Table 9.1: Estimation parameters for the first testbed

Parameter	Value
Correlation Degree (α)	0.9
Number of samples to average (n)	10
Number of locations to use in interpolation (N)	6
Averaging window (W)	10
Threshold	0.1

Table 9.2: Estimation parameters for the second testbed

Parameter	Value
Correlation Degree (α)	0.7
Number of samples to average (n)	10
Number of locations to use in interpolation (N)	6
Averaging window (W)	10
Threshold	0.1

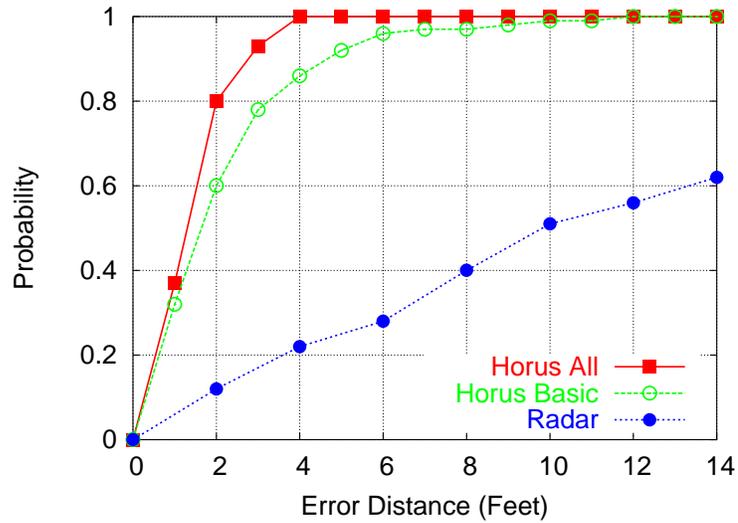


Figure 9.1: Performance of the *Horus* system and the *Radar* system for the first testbed.

Table 9.3: Comparison of the *Horus* system and the *Radar* system for the first testbed

	Median	Avg	Stdev	Max
<i>Horus</i> (all components)	1.28	1.38	0.95	4.00
<i>Horus</i> (basic)	1.60	2.16	2.09	18.08
<i>Radar</i>	9.74	13.15	10.71	57.67

system is better than the *Radar* system by more than 11 feet, on the average, for the first testbed and more than 9 feet, on the average, for the second testbed. Moreover, the tail for the distribution of the distance error for the *Horus* system is significantly better than that of the *Radar* system. Comparing the basic *Horus* system to the system after including all components shows that the average error is increased by more than 36% for the first testbed and more than 25% for the second testbed. The worst case error is enhanced by more than 78% for the first testbed and more than 70% for the second testbed. These results shows the effectiveness of the proposed techniques.

Figure 9.3 shows the average number of operations required per location estimate

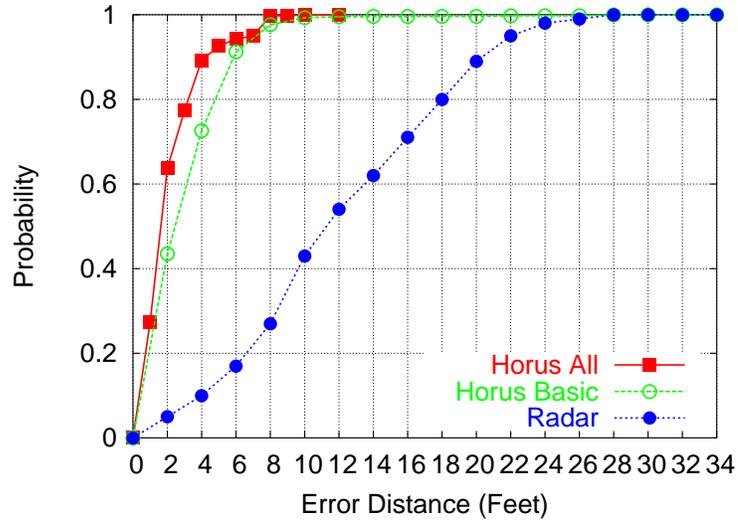


Figure 9.2: Performance of the *Horus* system and the *Radar* system for the second testbed.

Table 9.4: Comparison of the *Horus* system and the *Radar* system for the second testbed

	Median	Avg	Stdev	Max
<i>Horus</i> (all components)	1.68	2.10	1.74	9.49
<i>Horus</i> (basic)	2.36	2.81	2.54	32.52
<i>Radar</i>	11.18	11.86	6.05	31.72

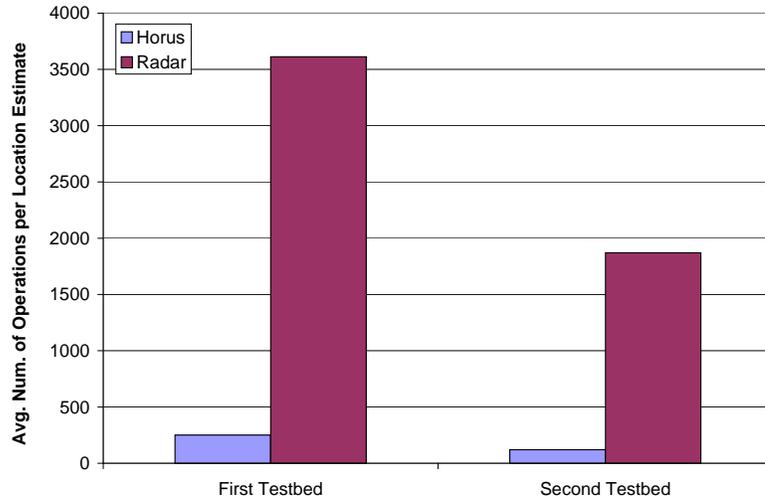


Figure 9.3: Average number of operations per location estimate for the *Horus* system and the *Radar* system for the two testbeds.

for the two systems. The figure shows that the *Horus* system leads to more than an order of magnitude savings in the number of operations required per location estimate compared to the *Radar* system.

9.2 Effect of the Training Parameters

In this section, we study how sensitive the *Horus* system is to the training time and to the radio-map spacing.

9.2.1 Training time

Figures 9.4 and 9.5 show the effect of the training time on accuracy. The error bars represent one standard deviation. The figures show that the *Horus* accuracy is enhanced as we increase the training time till we reach a stable value. A training time of 15 seconds is enough to obtain the desired accuracy for both testbeds.

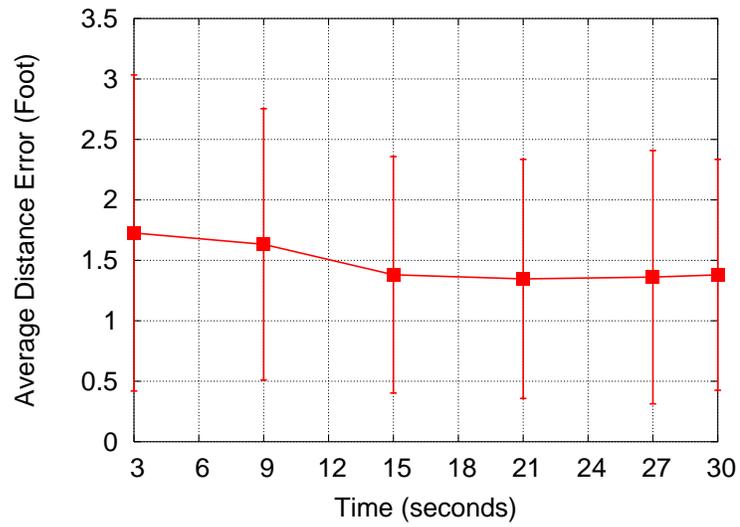


Figure 9.4: Effect of the training time on the performance of the *Horus* system for the first testbed.

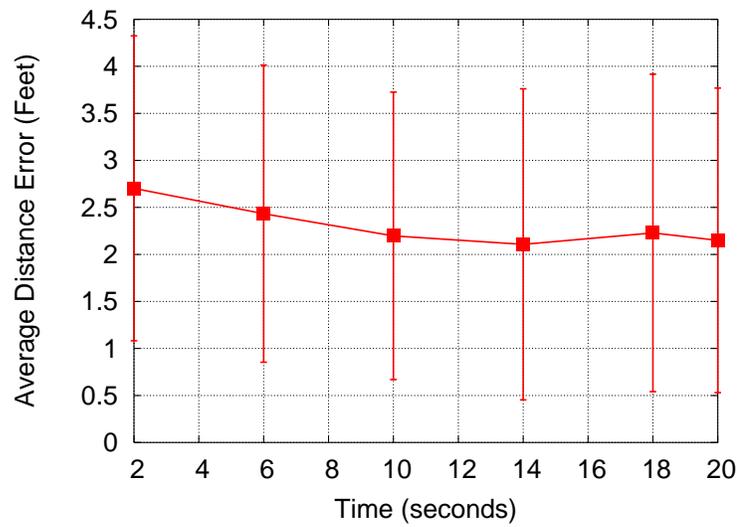


Figure 9.5: Effect of the training time on the performance of the *Horus* system for the second testbed.

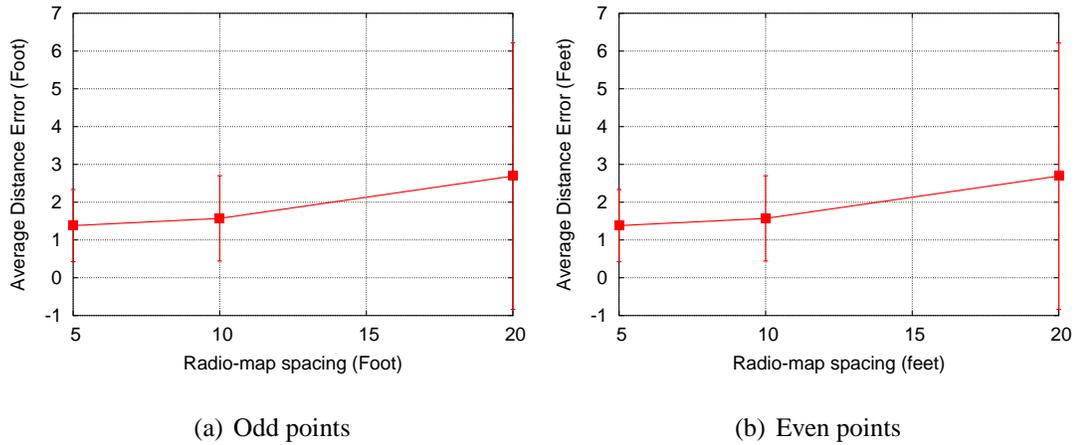


Figure 9.6: Effect of the radio map spacing on the performance of the *Horus* system for the first testbed.

9.2.2 Radio-map spacing

To study the effect of the distance between radio-map locations on the *Horus* system performance, we selected subsets of the radio-map locations. For each selected subset, we test the performance of the *Horus* system using this set and its complement. Figures 9.6 and 9.7 show the results. The error bars represent one standard deviation.

The results show that as the radio-map spacing increases, the accuracy decreases and the standard deviation increases. The average distance error can increase by as much as 100% (Figure 9.6.b). However, for a radio-map spacing of 10 feet, the *Horus* system maintains its high accuracy. The reason is that the *continuous-space estimator* component of the system allows us to compensate for the increased radio-map spacing.

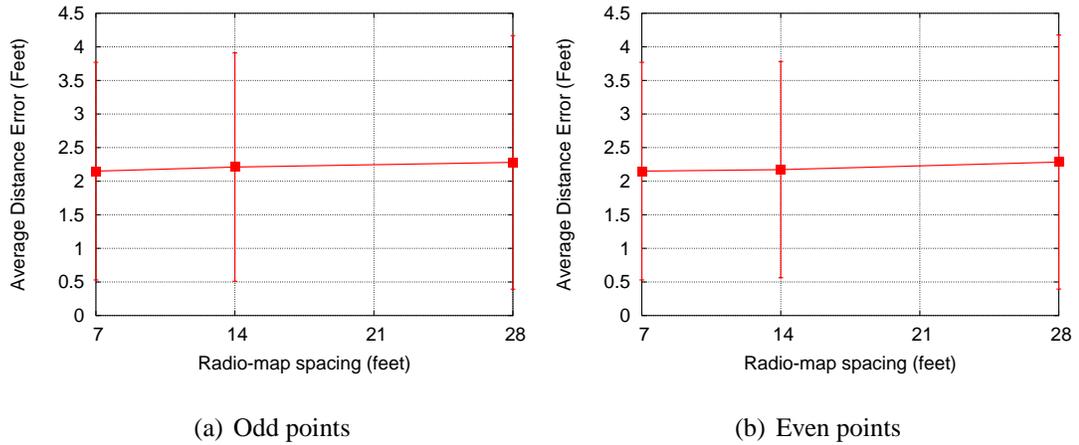


Figure 9.7: Effect of the radio map spacing on the performance of the *Horus* system for the second testbed.

9.3 Applying *Horus* Techniques to the *Radar* System

Figures 9.8 and 9.9 compare the performance of the original *Radar* system and the system with the enhancement discussed in this thesis (without correlation handling) for the two testbeds. The figures show that the average distance error is enhanced by more than 58% for the first testbed and more than 54% percent for the second testbed. Moreover, the worst case error is decreased by more than 76% for the first testbed and more than 48% for the second testbed.

These results show the effectiveness of the techniques proposed in the thesis and that these techniques are general and can be applied to other WLAN location determination systems to enhance their accuracy.

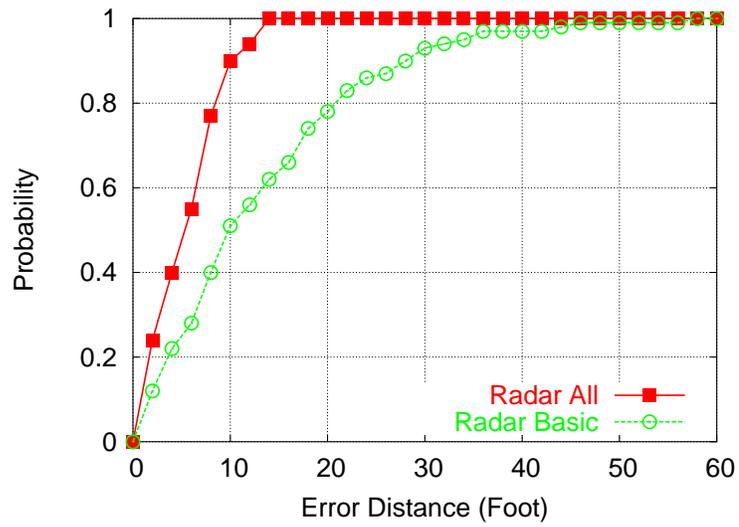


Figure 9.8: Effect of applying the *Horus* system’s techniques to the *Radar* system for the first testbed.

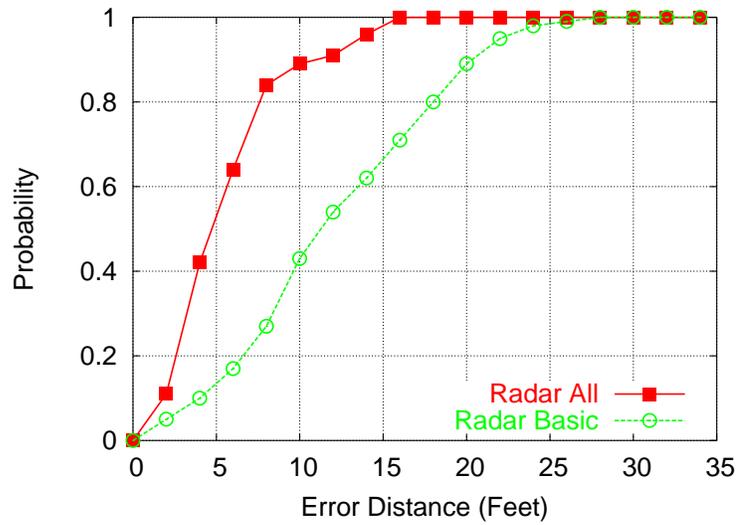


Figure 9.9: Effect of applying the *Horus* system’s techniques to the *Radar* system for the second testbed.

9.4 Summary

In this chapter, we compared the performance of the *Horus* system to the performance of the *Radar* system [22]. We showed that the accuracy of the *Horus* system is better than the *Radar* system by more than 11 feet, on the average, for the first testbed and more than 9 feet, on the average, for the second testbed. Moreover, the tail for the distribution of the distance error for the *Horus* system is significantly better than that of the *Radar* system. We also showed that the *Horus* system leads to more than an order of magnitude savings in the number of operations required per location estimate compared to the *Radar* system.

The experiments presented in this chapter show that a training time of 15 seconds per location is enough to construct the radio-map for the *Horus* system. The *continuous-space estimator* component of the system allows us to choose the radio-map locations as far as 10 feet while maintaining the high accuracy of the system.

The techniques presented in this thesis are general and can be applied to any of the current WLAN location determination systems. We showed the result of applying the techniques of the *Horus* system to the *Radar* system. The results show that the average distance error is enhanced by more than 58% for the first testbed and more than 54% percent for the second testbed. Moreover, the worst case error is decreased by more than 76% for the first testbed and more than 48% for the second testbed.

Chapter 10

Conclusions and Future Work

In this thesis, we presented the design of the *Horus* system: a WLAN-based location determination system. We laid out a taxonomy of the current research in the area of WLAN location determination systems and showed where the *Horus* system belongs in this taxonomy.

We approached the problem by identifying the various causes of variations in a wireless channel and developed techniques to overcome them. We also showed the various components of the system and how they interact.

The *Horus* system models the signal strength distributions received from access points using parametric and non-parametric distributions. By exploiting the distributions, the *Horus* system achieves accuracy better than the current WLAN location determination systems. The main advantage of the non-parametric technique is the efficiency of calculating the location estimate, while the parametric technique reduces the radio map size and smooths the distribution shape which leads to a slight performance advantage of the parametric technique over the non-parametric technique. We also showed analytically that these techniques are optimal among all discrete-space radio map-based WLAN location determination systems.

We showed that the correlation of the samples from the same access point can

be as high as 0.9. Experiments showed that under the independence assumption, as the number of averaged samples increases, the performance degrades. Therefore, we introduced an autoregressive model for handling the correlation between samples from the same access point. Based on this model, we presented a technique to take multiple samples from an access point into account to enhance the accuracy. Using the modified technique, the system average accuracy is enhanced by more than 19% for the first testbed and 11% for the second testbed using five signal strength samples.

For the proposed technique, as the number of samples used in the estimation process, n , increases the accuracy of the system is enhanced. However, a side effect of this increased accuracy is that the latency of calculating the location estimate increases. In general, we have a tradeoff between the accuracy required and the latency of location estimate. The higher the required accuracy, the higher n and the higher the latency to obtain the location estimate. This decision is dependent on the application in use.

Latency can be reduced by presenting the location estimate incrementally using one sample at a time. The system need not to wait till it acquires the n samples all at once. Instead, it can give a more accurate estimate of the location as more samples become available by reporting the estimated location given the partial samples it has. Other factors that affect the choice of the value of n are the user mobility rate and the sampling rate. The higher the user mobility rate or the sampling rate, the lower the value of n .

The basic *Horus* technique chooses the estimated location from the discrete set of radio map locations. We described two techniques for allowing continuous-space estimation: the *Center of Mass* technique and the *Time-Averaging* technique. Using the *Center of Mass* technique, the performance of the *Horus* system is enhanced by more than 13% for the first testbed and more than 6% for the second testbed compared

to the basic technique. The *Time-Averaging* technique enhances the performance of the *Horus* system by more than 24% for the first testbed and more than 15% for the second testbed. The two techniques are independent and can be applied together.

We introduced clustering of radio map locations as an approach to reduce the computational requirements of the location determination algorithms and achieve scalability. We describe two general approaches to clustering: *Explicit Clustering* and *Implicit Clustering*. The results show that using clustering reduces the average number of operations per location estimate by more than an order of magnitude. The *Explicit Clustering* technique gives slightly better accuracy than the *Implicit Clustering* technique. However, The average number of operations performed per location estimate for the *Implicit clustering* technique is much lower than the corresponding number of the *Explicit Clustering* technique. We also show that clustering enhances the accuracy as the search space is reduced by the clustering process.

Dealing with small-scale variations is challenging and *none* of the current systems handles it. Since the selected radio map locations are typically placed few feet apart, to limit the radio map size, the radio map does not capture small-scale variations. This contributes significantly to the estimation errors in the current systems. We proposed the *Perturbation* technique for handling small-scale variations. The perturbation technique enhances the average distance error by more than 25% for the first testbed and more than 21% for the second testbed. Moreover, the worst-case error is enhanced by more than 30% for the two testbeds.

We also compared the performance of the *Horus* system to the performance of the *Radar* system [22]. We showed that the *Horus* system is more accurate than the *Radar* system by more than 11 feet, on the average, for the first testbed and more than 9 feet, on the average, for the second testbed. Moreover, the tail for the distribution of the

distance error for the *Horus* system is significantly better than that of the *Radar* system. We also showed that the *Horus* system leads to more than an order of magnitude savings in number of operations required per location estimate compared to the *Radar* system.

The experiments presented in the thesis show that a training time of 15 seconds per location is enough to construct the radio-map information for the *Horus* system. The *continuous-space estimator* component of the system allows us to choose the radio-map locations as far as 10 feet while maintaining the high accuracy of the system.

The correlation handling, continuous-space estimator, clustering, and small-scale compensator modules are all applicable to any of the current WLAN location determination systems approaches. We show the result of applying the techniques of the *Horus* system to the *Radar* system. The results show that the average distance error is enhanced by more than 58% for the first testbed and more than 54% percent for the second testbed. Moreover, the worst case error is decreased by more than 76% for the first testbed and more than 48% for the second testbed.

The time-averaging technique can be considered a form of taking user history into account. However, more research on using the user history in location estimation and clustering is a future research direction. Another possible research direction is finding the optimal location for a set of access point to obtain the best accuracy. We are also looking at techniques to ensure user privacy even if the user is communicating with the system and the signal strength from his device can be recorded by the system. As part of our ongoing work we are experimenting with different clustering techniques. Automating the radio-map generation process is a possible research area. The *Horus* system provides an API for location-aware applications and services. We are looking at designing and developing applications and services over the *Horus* system. A

possible future extension is to dynamically change the system parameters based in the environment, such as changing the averaging window size as the user speed changes.

The *Horus* system achieves its goals of:

- High accuracy: through a probabilistic location determination technique, smoothing signal strength distributions by Gaussian approximation, using a continuous-space estimator, handling the high correlation between samples from the same access point, and the perturbation technique to handle small-scale variations.
- Low computational requirements: through the use of clustering techniques.
- Scalability in terms of the coverage area: through the use of clustering techniques.
- Scalability in terms of the number of users: through the distributed implementation on the mobile devices due to the low energy requirements of the algorithms.

These feature are unique to the *Horus* system and make its design and capabilities recognizable from all the current WLAN-based location determination systems. Moreover, the techniques presented in this thesis are applicable to other RF-technologies such as 802.11a, 802.11g, HiperLAN, and BlueTooth.

BIBLIOGRAPHY

- [1] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster. The anatomy of a context-aware application. In *5th ACM MOBICOM*, August 1999.
- [2] G. Chen and D. Kotz. A survey of context-aware mobile computing research. Technical Report Dartmouth Computer Science Technical Report TR2000-381, 2000.
- [3] R. Want and B. Schilit. Expanding the horizons of location-aware computing. *IEEE Computer*, 34(8):31–34, August 2001.
- [4] K. Pahlavan, P. Krishnamurthy, and J. Beneat. Wideband radio propagation modeling for indoor geolocation applications. *IEEE Communications Magazine*, April 1998.
- [5] K. Pahlavan, X. Li, M. Ylianttila, R.S. Chana, and M. Latva-aho. An overview of wireless indoor geolocation techniques and systems. In *Mobile and Wireless Communications Networks*, pages 1–13, May 2000.
- [6] H.-W. Gellersen, A. Schmidt, and M. Beigl. Adding some smartness to devices and everyday things. In *IEEE Workshop on Mobile Computing Systems and Applications*, pages 3–10, December 2000.

- [7] Federal Communications Commission Report and Order 96-264. Revision of the commission's rules to ensure compatibility with enhanced 911 emergency call systems. Technical Report Docket No. 94-102, FCC, July 1996.
- [8] S. Tekinay. Special issue on wireless geolocation systems and services. *IEEE Communications Magazine*, April 1998.
- [9] P. Enge and P. Misra. Special issue on GPS: The global positioning system. *Proceedings of the IEEE*, pages 3–172, January 1999.
- [10] R. Want, A. Hopper, V. Falco, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 10(1):91–102, January 1992.
- [11] R. Azuma. Tracking requirements for augmented reality. *Communications of the ACM*, 36(7), July 1997.
- [12] J. Krumm et al. Multi-camera multi-person tracking for easy living. In *3rd IEEE Int'l Workshop on Visual Surveillance*, pages 3–10, Piscataway, NJ, 2000.
- [13] C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfindex: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.
- [14] H. Aoki, B. Schiele, and A. Pentland. Realtime personal positioning system for wearable computers. In *International Symposium on Wearable Computers ISWC'99*, pages 37–43, October 1999.
- [15] R. J. Orr and G. D. Abowd. The smart floor: A mechanism for natural user identification and tracking. In *Conference on Human Factors in Computing Systems (CHI 2000)*, pages 1–6, The Hague, Netherlands, April 2000.

- [16] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, October 1997.
- [17] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *6th ACM MOBICOM*, Boston, MA, August 2000.
- [18] M. Hazas and A. Ward. A novel broadband ultrasonic location system. In *Fourth International Conference on Ubiquitous Computing, UbiComp 2002*, 2002.
- [19] T. Christ and P. Godwin. A prison guard duress alarm location system. In *IEEE International Carnahan Conference on Security Technology*, March 2000.
- [20] J. Werb and C. Lanzl. Designing a positioning system for finding things and people indoors. *IEEE Spectrum*, 35(9):71–78, September 1998.
- [21] M. Wallbaum. Wheremops: An indoor geolocation system. In *The 13th IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications*, September 2002.
- [22] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *IEEE Infocom 2000*, 2000.
- [23] P. Bahl, V. N. Padmanabhan, and A. Balachandran. Enhancements to the RADAR user location and tracking system. Technical Report MSR-TR-00-12, Microsoft Research, February 2000.
- [24] P. Prasithsangaree, P. Krishnamurthy, and P. K. Chrysanthis. On indoor position location with wireless LANs, September 2002.

- [25] S. Saha, K. Chaudhuri, D. Sanghi, and P. Bhagwat. Location determination of a mobile device using IEEE 802.11b access point signals. In *IEEE Wireless Communications and Networking Conference 2003*, March 2003.
- [26] R. Battiti, T. L. Nhat, and A. Villani. Location-aware computing: a neural network model for determining location in wireless LANs. Technical Report Technical Report DIT-02-0083, 2002.
- [27] A. Smailagic, D. P. Siewiorek, J. Anhalt, D. Kogan, and Y. Wang. Location sensing and privacy in a context aware computing environment. *Pervasive Computing*, 2001.
- [28] M. Youssef and A. Agrawala. Handling samples correlation in the horus system. In *IEEE Infocom 2004*, 2004.
- [29] M. Youssef, A. Agrawala, and A. U. Shankar. WLAN location determination via clustering and probability distributions. In *IEEE PerCom 2003*, March 2003.
- [30] M. Youssef and A. Agrawala. Small-scale compensation for WLAN location determination systems. In *IEEE WCNC 2003*, March 2003.
- [31] P. Castro, P. Chiu, T. Kremenek, and R. Muntz. A probabilistic location service for wireless network environments. *Ubiquitous Computing 2001*, September 2001.
- [32] P. Castro and R. Muntz. Managing context for smart spaces. *IEEE Personal Communications*, October 2000.
- [33] 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*, 9(3), July 2002.

- [34] T. Roos, P. Myllymaki, and H. Tirri. A statistical modeling approach to location estimation. *IEEE Transactions on Mobile Computing*, 1(1):59–69, January-March 2002.
- [35] A. M. Ladd, K. Bekris, A. Rudys, G. Marceau, L. E. Kavasaki, and D. S. Wallach. Robotics-based location sensing using wireless ethernet. In *8th ACM MOBILCOM*, Atlanta, GA, September 2002.
- [36] P. Tao, A. Rudys, A. M. Ladd, and D. S. Wallach. Wireless LAN location-sensing for security applications. In *Proceedings of the ACM Workshop on Wireless Security*, San Diego, CA, 2003.
- [37] M. Youssef, A. Agrawala, A. U. Shankar, and Sam H. Noh. A probabilistic clustering-based indoor location determination system. Technical Report UMIACS-TR 2002-30 and CS-TR 4350, University of Maryland, College Park, March 2002.
- [38] M. Youssef and A. Agrawala. On the optimality of WLAN location determination systems. Technical Report UMIACS-TR 2003-29 and CS-TR 4459, University of Maryland, College Park, March 2003.
- [39] M. Youssef and A. Agrawala. Handling samples correlation in the horus system. Technical Report UMIACS-TR 2003-75 and CS-TR 4506, University of Maryland, College Park, March 2003.
- [40] M. Berna, B. Lisien, B. Sellner, G. Gordon, F. Pfenning, and S. Thrun. A learning algorithm for localizing people based on wireless signal strength that uses labeled and unlabeled data. In *8th International Joint Conference on Artificial Intelligence (IJCAI'03)*, Acapulco, Mexico, August 2003.

- [41] G. Shreve and D. Kell. A precision location network using ultra wideband WLAN radios, 2001.
- [42] M. Gruteser and D. Grunwald. Enhancing location privacy in wireless lan through disposable interface identifiers: A quantitative analysis. In *First ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots*, September 2003.
- [43] The Institute of Electrical and Inc. Electronics Engineers. IEEE standard 802.11 - wireless LAN medium access control (MAC) and physical layer (PHY) specifications. 1999.
- [44] W. Stallings. *Wireless Communications and Networks*. Prentice Hall, first edition, 2002.
- [45] A. Neskovic, N. Nescovic, and G. Paunovic. Modern approaches in modeling of mobile radio systems propagation environment. *IEEE Communications Surveys*, (Third Quarter), 2000.
- [46] T. S. Rappaport. *Wireless Communications Principles and Practice*. Pearson Education, second edition, 2002.
- [47] H. Hashemi. The indoor radio propagation channel. *Proceedings of the IEEE*, 81(7):943–968, July 1993.
- [48] <http://www.ascension-tech.com>.
- [49] E. Prigge and J. How. Signal architecture for a distributed magnetic local-positioning system. In *The first IEEE International Conference on Sensors*, June 2002.

- [50] J. Krumm, G. Cermak, and E. Horvitz. Rightspot: A novel sense of location for a smart personal object. In *Fifth International Conference on Ubiquitous Computing*, October 2003.
- [51] N. Bulusu, J. Heidemann, D. Estrin, and T. Tran. Self-configuring localization systems: Design and experimental evaluation. Technical Report 8, University of California, Los Angeles, Center for Embedded Networked Computing, September 2002. Accepted to appear, ACM TOCS.
- [52] N. Bulusu, D. Estrin, L. Girod, and J. Heidemann. Scalable coordination for wireless sensor networks: Self-configuring localization systems. In *International Symposium on Communication Theory and Applications (ISCTA)*, 2001.
- [53] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low-cost outdoor localization for very small devices. *IEEE Personal Communications*, 7(5):28–34, October 2000.
- [54] S. Capkun, M. Hamdi, and J. P. Hubaux. GPS-free positioning in mobile ad-hoc networks. *Cluster Computing Journal*, 5(2), April 2002.
- [55] S. Capkun, M. Hamdi, and J.-P. Hubaux. GPS-free positioning in mobile adhoc networks. In *Hawaii International Conference on System Sciences (HICSS-34)*, 2001.
- [56] L. Doherty, K. Pister, and L. El Ghaoui. Convex position estimation in wireless sensor networks. In *IEEE Infocom 2001*, April 2001.
- [57] P. Goel, S. Roumeliotis, and G. Sukhatme. Robust localization using relative and absolute position estimates. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1999.

- [58] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *9th ACM International Conference on Mobile Computing and Networking (Mobicom)*, September 2003.
- [59] D. Niculescu and B. Nath. Ad-hoc positioning system. In *IEEE GlobeCom*, November 2001.
- [60] N. Patwari and A. O. Hero III. Using proximity and quantized RSS for sensor localization in wireless networks. In *2nd International ACM Workshop on Wireless Sensor Networks and Applications (WSNA)*, September 2003.
- [61] N. Patwari, A. O. Hero III, M. Perkins, N. S. Correal, and R. J. O’Dea. Relative location estimation in wireless sensor networks. *IEEE Transactions on Signal Processing and Special Issue on Signal Processing in Networks*, 51(8):2137–2148, August 2003.
- [62] Henrik R Lundgren, David Lundberg, Erik Nordstrm, and Christian F. Tschudin. A large-scale testbed for reproducible ad hoc protocol evaluations. In *IEEE WCNC 2002*, March 2002.
- [63] Song Li, Gang Zhao, and Lin Liao. User location service over an 802.11 ad-hoc network, 2002.
- [64] T. D. Hodes, R. H. Katz, E. S. Schreiber, and L. Rowe. Composable ad hoc mobile services for universal interaction. In *3rd ACM MOBICOM*, pages 1–12, September 1997.
- [65] X. Li, K. Pahlavan, M. Latva-aho, and M. Ylianttila. Comparison of indoor geolocation methods in DSSS and OFDM wireless LAN systems. In *IEEE VTC’2000*, September 2000.

- [66] <http://www.orinocowireless.com>.
- [67] <http://www.cs.umd.edu/users/moustafa/downloads.html>.
- [68] http://www.hpl.hp.com/personal/jean_tourrilhes/.
- [69] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel. *Time Series Analysis: Forecasting and Control*. Prentice Hall, third edition, 1994.