Enhancements to the RADAR User Location and Tracking System

Paramvir Bahl and Venkata N. Padmanabhan
Microsoft Research

Anand Balachandran
University of California at San Diego

February 2000

Technical Report
MSR-TR-2000-12

Microsoft Research

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Enhancements to the RADAR User Location and Tracking System

Paramvir Bahl and Venkata N. Padmanabhan
Microsoft Research
{bahl,padmanab}@microsoft.com

Anand Balachandran
University of California at San Diego
anandb@cs.ucsd.edu

Abstract

We address the problem of locating users inside buildings using a radio-frequency (RF) wireless LAN. A previous paper presented the basic design and a limited evaluation of a user-location system we have developed. In this paper, we analyze shortcomings of the basic system, and develop and evaluate solutions to address these shortcomings. Additionally, we describe several new enhancements, including a novel access point-based environmental profiling scheme, and a Viterbi-like algorithm for continuous user tracking and disambiguation of candidate user locations. Using extensive data collected from our deployment, we evaluate our system's performance over multiple wireless LAN technologies and in different buildings on our campus. We also discuss significant practical issues that arise in implementing such a system. Our techniques are implemented purely in software and are easily deployable over a standard wireless LAN.

Keywords: *wireless networks, nomadic computing, location-awareness, measurement, implementation experience*

1 Introduction

The proliferation of lightweight, portable computing devices and high-speed wireless local-area networks has enabled users to remain connected while moving about inside buildings. This emerging paradigm has spurred a lot of interest in applications and services that are a function of the mobile user's physical location. The goal here is to enable the user to interact effectively with his or her physical surroundings. Examples of such interactions include: printing a document on the closest printer, locating a mobile user, displaying a map of the immediate surroundings, and guiding a user inside a building. As the surroundings change, so does the computing that happens. The interaction between computing and location may also be less direct. For instance, when in the boss' office, pre-fetch facts and figures on business performance and project-status for ready access, but while in the cafeteria, turn on the sports score ticker.

The granularity of location information needed varies from one application to another. For example, locating a nearby printer requires fairly coarse-grained location information whereas locating a book in a library would require fine-grained information [1]. In general, the amount of precision desired dictates the cost and complexity of the location-determination system.

To the best of our knowledge, previous research on in-building location systems has generally resulted in creation of specialized hardware and technologies that are severely limited. As an example, systems that use infrared (IR) wireless technology have been reported in [2], [3], [4], [5], and [6]. The limited range of an IR network, which facilitates user location, is a handicap in providing ubiquitous coverage. To overcome this problem, a few researchers have developed RF-tag based location systems as well [7], [8]. Unfortunately, these systems, like their IR counterparts, are often built for the sole purpose of determining users location, i.e., they do not provide any data networking services. Furthermore, they all use specialized hardware and are generally cost prohibitive. The trade-off between deployment cost and perceived value of these systems has not been compelling enough for large-scale adoption.

We have developed a system, called RADAR, which avoids many of the limitations of the previous systems. Ours is a software-only system built over an off-the-shelf RF wireless local area network. A wireless LAN is typically deployed to provide tether-less data networking capability to mobile hosts. The location-aware services enabled by RADAR complement this already useful data networking capability of RF wireless LANs. This makes a wireless LAN more valuable and, in our opinion, increases the chances of large-scale deployment.

RADAR is built over an IEEE 802.11 standard-based [9] RF wireless LAN deployed in our building. Our initial results are encouraging; with a high probability, RADAR is able to locate the user to within a few meters of his or her actual location.

In this paper, we analyze shortcomings of the basic RADAR system (discussed in detail in [10]) and develop and evaluate solutions that address these shortcomings. We describe significant new enhancements that we have made. Specifically, we describe a novel access point[1]-based *environmental profiling* scheme that takes into account the reality that RF signals are significantly impacted by changes in the environment (e.g., change in number of people and obstructions in the building, change in temperature, etc.

[11]). Our scheme mitigates some of the problems caused by environmental fluctuations and improves the overall efficiency of the system. We also describe a novel Viterbi-like algorithm that we have developed for *continuous user tracking*. This algorithm takes into account, in real-time, the mobility pattern of the user helps disambiguate between candidate user locations guessed by the basic system. Our initial work did not analyze the case of buildings with multiple floors. In this paper, we consider the effect of floors and describe a design that compensates for it. Using extensive data collected from our multiple deployments, we show that our system works well over multiple different wireless LAN technologies and in different buildings on our campus. Our techniques are implemented purely in software and are easily deployable over a standard wireless LAN.

The rest of this paper is organized as follows: In Section 2, we survey related work in the field of location-determination. In Section 3, we describe the basic RADAR system. In Section 4, we describe our experimental testbed. In Section 5, we discuss the base system performance. In Section 6, we focus on the new enhancements we have made to the basic system. In Section 7, we discuss some practical issues that arise in implementing and deploying a system such as ours. We present our conclusions in Section 8 and pointers to future work in Section 9.

2 Related Work

Related work in the area of location and tracking system falls into the following four broad categories: (1) IR-based systems (2) indoor RF-based systems (3) wide-area cellular-based systems, and (4) everything-else, e.g. ultrasound, magnetic fields, etc.

The seminal work in IR-based location systems is the *Active Badge* system reported in [2], [6]. In this system, a badge worn by a person emits a unique IR signal every 10 seconds. Sensors placed at known positions within a building pick up the unique identifiers and relay these to the location manager software. While this system provides accurate location information, it suffers from several drawbacks: (a) it scales poorly due to the limited range of IR, (b) it incurs significant installation and maintenance costs, and (c) it performs poorly in the presence of direct sunlight, which is likely to be a problem in rooms with windows.

Another system based on IR technology is described in [3]. IR transmitters are attached to the ceiling at known positions in the building. An optical sensor on a head-mounted unit senses the IR beacons, which enables the system software to determine the user's location. This system suffers from similar drawbacks as the Active Badge system.

Recently, we learned about an RF-based location-determination system called the *Duress Alarm Location System* (DALS) [7]. This system uses RF signal strengths to determine user location in a manner similar to our basic system. However, this system differs significantly from our enhanced system, which we describe in this paper. Also, DALS is different from our basic system in that it (a) depends on specialized hardware (b) requires infrastructure deployment over and above a wireless data network, (c) does not take into consideration the effect of the user's body orientation on RF signals, which our study shows can be significant, and (d) does not take RF propagation into account.

Another interesting indoor RF system is the 3D-iD RF tag system built by PinPoint Corporation [8]. Antennas planted around a facility emit RF signals at 2.4 GHz. Tags, acting like RF mirrors, transmit a response signal at 5.8 GHz along with an identification code. Various antennas receive the signal, and send the results to cell controllers, which triangulate the reflections to determine the tag's whereabouts. The system's locating ability varies depending on the number of antennas installed in an area but the best advertised resolution is 10 feet. The cost of an entire system is quite high. Once again Pinpoint's system differs from our system in that (a) it requires specialized hardware to do location determination, (b) they use signal-processing techniques that are significantly different from ours, and (c) their system does not include high-speed data networking capability.

The Daedalus project [17], a briefly mentions a wireless LAN based system for location estimation. This is a very coarse-grained user location system. Access points (APs) transmit beacons augmented with their physical coordinates. A mobile host estimates its location to be the same as that of the AP to which it is attached. Consequently, the accuracy of the system is limited by the (possibly large) cell size

In the wide-area cellular arena, several location determination systems have recently been proposed [13]. The technological alternatives for locating cellular telephones involve measuring the signal attenuation, the angle of arrival (AOA), and/or the time difference of arrival (TDOA). While these systems have been found to be promising in outdoor environments, their effectiveness in indoor environments is limited by the multiple reflections suffered by the RF signal, and the inability of off-the-shelf and inexpensive hardware to provide fine-grain time synchronization.

[1] An access point is a bridge between the wired and wireless networks.

Systems based on the Global Positioning System (GPS) [12], [13], [14] while very useful outdoors, are ineffective indoors because buildings block GPS transmissions.

Researchers have also built systems using alternative technologies. One uses pulsed DC magnetic fields [15] to determine user orientation while another uses ultrasound signals [16] to determine user location. While these technologies and systems are very interesting, they generally suffer the same drawbacks as their IR and RF-tag counterparts. Their specialized hardware is generally targeted at niche markets, tending to make the system cost prohibitive, range limited, and unsuitable for large-scale deployment.

Our work differs from previous work in that we tackle the problem of people location and tracking using the widely available RF-based wireless LANs. With data networking speeds of up to 11 Mbps [18], wireless LANs have gained rapid acceptance and are widely being deployed in offices, schools, homes etc. Besides the existing wireless LAN our system does not require any additional hardware and can be enabled using purely software means.

These points are clarified in the following section.

3 The RADAR System

The RADAR system is built on a deployment of off-the-shelf wireless LAN technology. Access points (or base stations) are located so as to provide overlapping coverage in the area of interest[2]. The mobile user carries with her a communications device equipped with a wireless LAN card capable of bi-directional communication with the access points.

The fundamental idea in RADAR is that in an RF network, the energy level or signal strength (*SS*) of a packet is a function of the receiver's (mobile user's) location. Consequently, it provides a means for inferring the user's position. There is a clear trend in SS as a user walks about the building. Not surprisingly, the signal received at the mobile is strongest when the receiver is close to the AP and weakest when it is far away. This strong trend, observed for all neighboring APs independently, is exploited by the system to estimate the mobile's location.

With this as motivation, RADAR takes the following approach to location determination. A *Radio Map* of the building is created. A Radio Map is a database of locations in the building and the observed (or estimated) signal strength of the beacons emanating from the APs as recorded at these locations. So, for example, an entry in the Radio Map may look like $(x, y, z, ss_{i\ (i=1..n)})$ where (x, y, z) is the physical coordinates of the location where the signal is recorded and ss_i is the signal strength of the beacon signal emanating from the i^{th} AP. For the basic system we used three APs (i.e. $n = 3$) and one floor, consequently, the Radio Map contained entries that looked like (x, y, ss_1, ss_2, ss_3)

The major part of the effort in deploying RADAR goes into creating the Radio Map of the building. We evaluated two approaches for this purpose.

The first method for creating a Radio Map involves a mobile user walking to different locations in the building, preferably close to one another, and explicitly measuring and recording at each location, both the physical coordinates (using a floor layout map as reference) and the signal strength of the beacon packets from each of the APs within range.

The second method to construct a Radio Map involves computing and recording the signal strength from all neighboring APs using a mathematical model of indoor RF signal propagation. We have developed a simple yet fairly accurate model that accommodates different building layouts while taking into account both free-space path loss and attenuation due to obstructions (e.g., walls) between the transmitter and the receiver.

To locate the position of the mobile user in real-time, the mobile measures the signal strength of each of the APs within range. It then searches through the Radio Map database to determine the signal strength tuple that best matches the signal strengths it has measured. The system estimates the location associated with the best-matching signal strength tuple, to be the location of the mobile.

The specific search technique we developed is called *nearest neighbor(s) in signal space (NNSS).* The *NNSS* algorithm computes the *Euclidean distance* (in signal space) between each SS tuple in the Radio Map (ss_1,ss_2,ss_3) and the measured *SS* tuple (ss'_1,ss'_2,ss'_3). It then picks the SS tuple that minimizes the distance in signal space and declares the corresponding physical coordinates as its estimate of the user's location.

One variant of the basic NNSS algorithm is *NNSS-AVG.* The intuition here is that in case there is more than one SS tuple in the Radio Map that is "close" to measured SS tuple, there is little reason to pick just the closest one and discard others that are almost as close. So the NNSS-AVG algorithm picks a small number of closely matching tuples and averages their physical location to obtain an estimate of the user's location. Often this results in an estimate that is better than any individual tuple.

Of the two approaches to building a Radio Map, the first, which involves explicit measurement of RF signal strength at several different locations, performs better than the second, which is based on computing signal strength at the various locations using an indoor RF propagation model. The *error distance*, which we define to be the Euclidean (physical) distance between the true location and the estimated location of the user,

[2] Having a network that is designed to provide overlapping coverage has an added bonus as it improves the system's performance and adds protection against downtime in the event of AP failure [28]

has a median of 2 to 3 meters, about the size of a typical office room in the building.

A more detailed discussion of the basic RADAR system and its performance appears in [10]. Subsequent to that paper, we have deployed a second RADAR testbed. There are significant differences in the hardware and technology of the two systems. In the next section, we describe both our testbeds, which are used for the experiments reported in this paper.

4 The RADAR Testbeds

In this section we describe in detail two significantly different deployments of RADAR based on different wireless hardware.

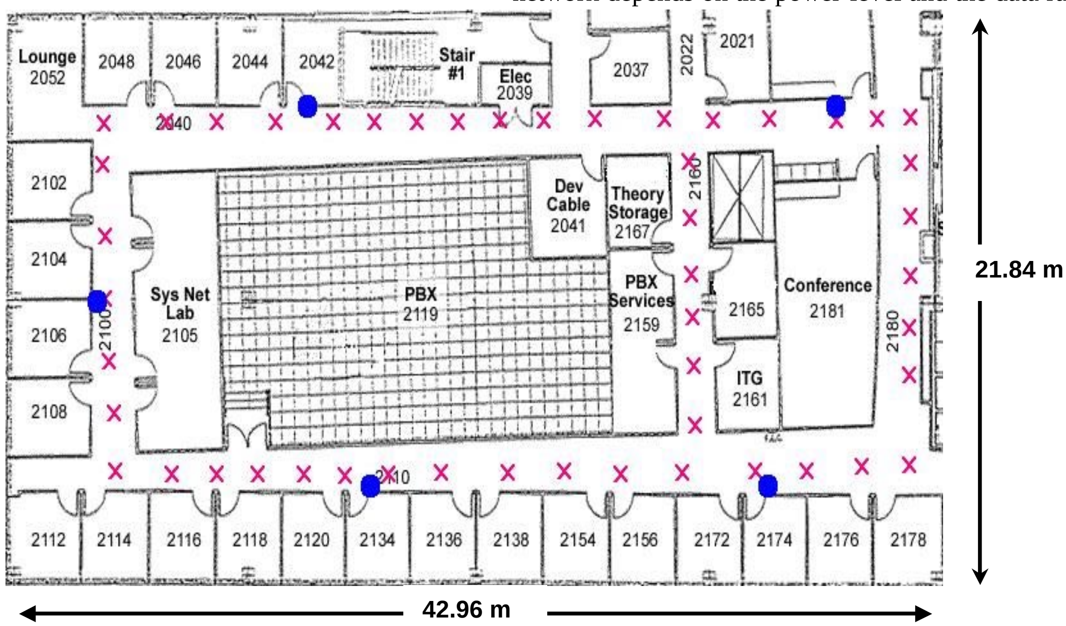

Figure 1: Map of the floor where the new experiments were conducted. The crosses denote the locations were signal strength from beacon packets was recorded. The filled dots show the locations of the 5 access points.

Our first testbed, which we used to build the basic system, is deployed on the second floor of a 3-storey building. The dimensions of the floor are 43.5 m by 22.5 m, an area of 980 sq. m (10500 sq. ft.), which includes more than 50 rooms. Three APs cover the entire floor. Each AP and mobile host is equipped with a Digital RoamAbout™ network interface card, which is based on Lucent's 2 Mbps proprietary WaveLAN™ RF LAN technology. The APs are attached to a Pentium-based PCs running FreeBSD 3.0 while the mobile hosts are Pentium-based laptop computers running Microsoft Windows 95. The network operates in the 2.4 GHz license-free ISM (Industrial, Scientific and Medical) band and has a range of 200 m, 50 m, and 25 m, respectively, for open, semi-open, and closed office environments [27].

Our second (newer) testbed is deployed on the second floor of a 4-storey building. The layout of the floor and the placement of the APs are shown in Figure 1. Five wall-mounted APs provide overlapping coverage in the portion of the floor where the experiments were carried out. In contrast to the first testbed, the new testbed is built over a standards-based state-of-the-art wireless LAN from Aironet Communications Inc. Specifically, we use Aironet's 4800 series of products, which includes the AP4800™ APs and the PC4800™ wireless network interface cards [28]. Like RoamAbout™, this RF hardware also operates in the 2.4 GHz ISM band. However, it has different medium access control (MAC) and different physical (PHY) Layer. The 4800 is a multi-rate, direct sequence spread spectrum, IEEE 802.11b network [9]. It supports raw data rates of 1, 2, 5.5 and 11 Mbps and power levels of 5, 20, 50 and 100mW. The range of the network depends on the power-level and the data rate at which it is operating. Table 1 compares the two deployments of our system.

Table 1: Highlights of our testbeds

	Testbed 1	**Testbed 2**
Hardware	Digital Equip. Corp. WaveLAN	Aironet Wireless Inc 4800 series
MAC	CSMA/CA [27]	IEEE 802.11b [28]
Modulation	Spread-spectrum DQPSK	Spread-spectrum CCK
Output Power	50 mW	100 mW
Data Rate	2 Mbps	11 Mbps
Number of APs	3	5
Floor Dimensions	43.2 m x 22.5 m	42.9 m x 21.8 m
OS	FreeBSD 3.0	Windows 2000

In the following section we analyze the performance of our basic system across the two deployments.

5 Basic System Performance

We evaluate the base performance of our system by feeding the signal strength tuples recorded at known locations of the user into the NNSS algorithm (Section 3) and comparing the guessed location with the true location. This experiment simulates the case where we are trying to locate a static user. We quantify performance using the error distance. For much of our discussion, we focus on a single floor of the building, so the Euclidean distance is computed in two dimensions. (We discuss the effect of multiple floors in Section 6.3)

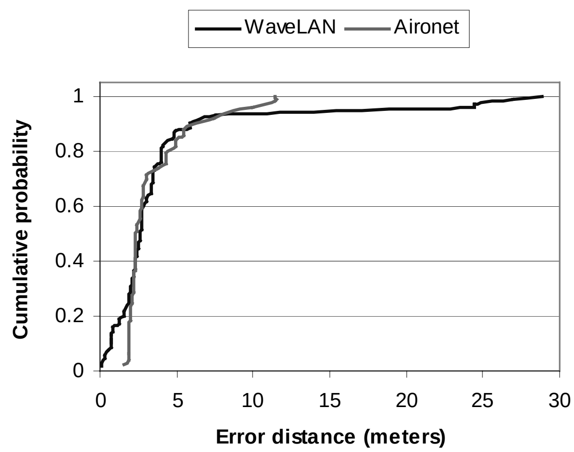

Figure 2: CDF of the Error Distance

Figure 2 plots the cumulative distribution function (CDF) of the error distance for the two deployments of our system. Since the WaveLAN™ deployment only had 3 APs, we consider only 3 APs for the Aironet deployment too. We observe that CDFs for both cases match well for the most part. The median error distance and the 90th percentile of the error distance for WaveLAN™ are 2.65 m and 5.93 m, respectively, while the corresponding values for Aironet are 2.37 m and 5.97 m. However, the tail of the CDF is much longer for WaveLAN™, implying that there are instances where the error distance is very large.

The reason for the long tail in the WaveLAN™ deployment is due to a phenomenon we term as *aliasing*. Two points that are far apart physically may be very close together in signal space. Such aliasing can happen because of the complex indoor propagation environment. The signal strength at a point close to an AP may be similar to another point that is far away simply because of an obstruction (such as a wall) that attenuates the signal received at the former point while the latter point receives an unobstructed signal. Whether aliasing occurs and how commonplace it is essentially a function of the building layout and the placement of APs. We discuss a technique we have developed to alleviate the effects of aliasing in Section 6.1.

The results presented the following subsections are a sampling from several experiments we conducted in both the WaveLAN and the Aironet testbeds.

5.1 Effect of the Number of Access Points

The number of APs determines the dimension of the signal strength tuples that the NNSS algorithm operates on. A larger number of APs with overlapping coverage may make the NNSS search more accurate albeit at the cost of a larger hardware deployment. To quantify the benefit, if any, of increasing the number of APs, we varied their number from 1 through 5 in the Aironet deployment. The mean, median, and 90th percentile of the error distance are plotted in Figure 3.

The main observation is that while there is a significant benefit to going from 1 AP to 2 APs and again from 2 APs to 3 APs, there is little benefit in going beyond 3 APs. The inherent noise in the signal strength imposes a limit on how accurately location can be inferred using the NNSS technique, no matter how many APs provide coverage in a region. For instance, the signal strength can vary by a few dBm even while the receiver is stationary.

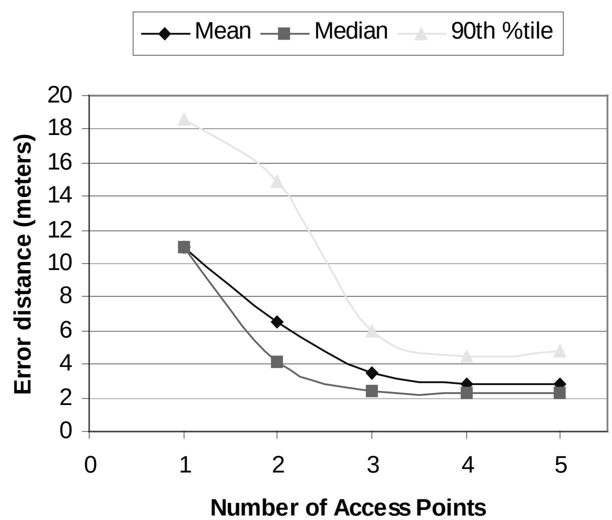

Figure 3: Impact of the number of APs on the error distance.

6 Enhancements to the Basic System

In this section we discuss some of the shortcomings of the basic system, describe the enhancements we have made to overcome these shortcomings, and present a performance evaluation of these enhancements.

6.1 Continuous User Tracking

The analysis in Section 5.1 focused on taking a static snapshot of the strength of the signals from multiple APs recorded at a mobile host and using this information to guess the location of the static user. The NNSS algorithm used for this purpose does not consider location information (or, to be more precise, guesses of user location) from the past.

The idea behind continuous user tracking is precisely to use information from the past to come up with a better guess of a user's location. The intuition is that since physical constraints preclude a user from "jumping about" over large distances at random, the user's location at a given time instant is likely to be near that at the previous time instant. So, by tracking the user continuously, we complement signal strength information with the physical contiguity constraint to potentially improve the accuracy of location determination.

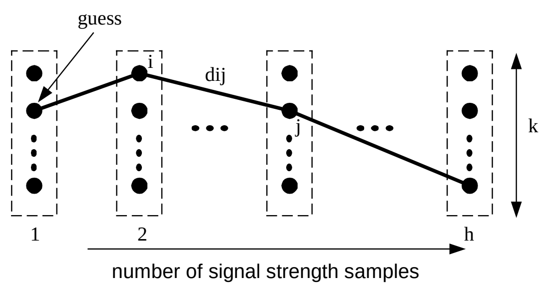

Figure 4: A depiction of the state maintained by the Viterbi-like continuous tracking algorithm. The shortest path is shown in bold. The location corresponding to the mid-point of the path is guessed to be the user's location. The weight of an edge between vertices i and j is d_{ij}, the Eucledian distance between the corresponding locations.

A beneficial side effect of continuous user tracking is that the problem of aliasing may be alleviated. Suppose that two physically distant points A and B are so close together in signal space (due to aliasing) that the NNSS algorithm is unable to disambiguate between the two. If the location of the user was determined unambiguously a little while earlier, then we can pick between guesses A and B by using this unambiguous guess from the recent past together with the physical contiguity constraint.

The physical contiguity constraint has been employed elsewhere in the context of wireless networks. An example is the determination of a user's trajectory (for instance, while he/she is driving down a highway) to enable anticipation of handoffs in cellular telephone networks [25].

6.1.1 Viterbi-like Algorithm

Our algorithm for continuous user tracking operates as follows. Each time a signal strength tuple is obtained by the mobile host, an NNSS search is done to determine the *k* nearest neighbors in signal space (*k*-NNSS), i.e., the *k* best guesses of the user's location. A history of depth *h* of such *k*-NNSS sets is maintained. The collection of these *h* *k*-NNSS sets can be viewed as a graph as depicted in Figure 4. There are edges only between vertices contained within consecutive sets. Each edge is assigned a weight to model the likelihood of the user transitioning (in successive time instances) between the locations represented by the two endpoints of the edge. The larger the weight is, the less likely is the transition. We use a very simple metric — the Euclidean distance between the two physical locations — as the weight.

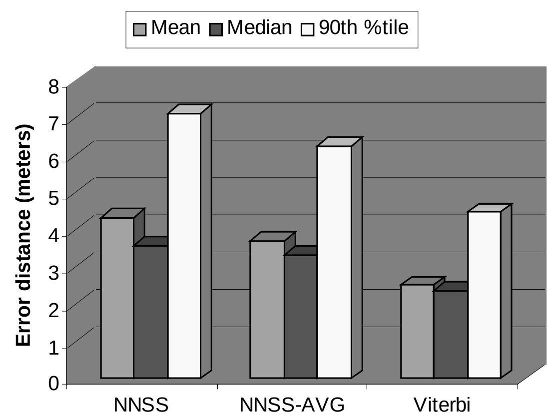

Figure 5: Performance of the various algorithms in tracking a user who is walking.

Each time the history vector is updated with the addition of the most recent *k*-NNSS set (and the deletion of the oldest set), the shortest path between the vertices in the oldest and the newest sets is computed. This shortest path can be viewed as representing the "most likely" trajectory of the mobile user. (This is similar to the Viterbi algorithm [29] in communication theory used by a receiver to determine the most likely message to have been transmitted over a noisy channel.) Once the shortest path is determined, we guess the user's location to be the point at the start of the path (Figure 4). This procedure captures the physical contiguity constraint, but it also implies that there is a lag of *h* signal strength samples between when a user is at a location and when the system guesses the user's location.

6.1.2 Performance of Viterbi-like Algorithm

To evaluate the effectiveness of the Viterbi-like continuous tracking algorithm, we gathered signal strength data at 1-second intervals while the user was walking in the WaveLAN testbed. We also recorded the precise location of the user as a function of time. Then we used the signal strength data from the walk in conjunction with the pre-computed Radio Map of the building to try and reconstruct the user's trajectory. We evaluated three different algorithms: NNSS, NNSS-AVG (where the physical locations of the 3 nearest neighbors in signal space are averaged to obtain an estimate of location), and Viterbi (with $k = 3$ and $h = 6$). The mean, median, and 90th percentile of the error distance are shown in Figure 5.

The main observation is that the Viterbi-like algorithm significantly outperforms both NNSS and NNSS-AVG. For instance, the median error distance for NNSS (3.59 m) and NNSS-AVG (3.32 m) are 51% and 40% worse, respectively, compared to Viterbi (2.37 m). Also, the significant reduction in the 90% percentile of the error distance for Viterbi compared to the other two algorithms indicates that the long tail caused by aliasing (Section 5.1) has been shortened. This underscores the importance of continuous tracking of the user.

In our analysis, we found that a history depth of $h = 6$ was optimal. This rather small history depth implies that the time lag caused by the Viterbi algorithm is likely to be quite small.

6.2 Profiling the Environment

In a previous section we had mentioned that data collection for building a Radio Map is necessary when the environment changes. In this section we discuss this issue in greater depth.

6.2.1 Problem Description

The radio frequency environment is a hostile environment for signal strength-based location systems. This is because signal propagation is dominated by reflections, diffractions, and scattering of radio waves caused by structures within the building

[11]. The transmitted signal generally reaches the receiver via multiple paths (termed the *multipath* phenomenon). Multipath causes fluctuations in the received signal envelope and phase, and the signal components arriving from indirect and direct paths combine to produce a distorted version of the transmitted signal. Multipath within buildings is strongly influenced by the layout of the building, the construction material used, and the number of people in the building.

As the number of people in the building varies, the propagation characteristics of RF signals change as well. This is because the human body is made up of water and water absorbs RF signals. In experiments we performed, a single human body may on an average change the signal strength by as much as 3.5 dBm.

At different times of the day, a different number of humans may be present in the building (due to meetings, etc.) causing the signal strength at the various locations in the building to vary considerably. As a consequence, a Radio Map created at any one particular time may not accurately reflect the environment at a different time. This can reduce the accuracy of the RADAR system considerably.

6.2.2 Solution

To take into account the changes in the environment, we decided to use multiple Radio Maps representing the different environmental conditions. We were then left with the problem of dynamically choosing the Radio Map that best represented the environment when determining the user's location at a given time.

We solved this problem by using our access points to calibrate the environment, exploiting the fact that the APs are at fixed and known locations. Conceptually, we use RADAR to determine the location of each AP, using beacons from neighboring APs, just as if it were a mobile user. For each AP, we measure the signal strength of beacons from neighboring APs and then use this to estimate its location using each of the available Radio Map databases. The Radio Map that results in an estimate that is closest to the known location of the target AP, for the majority of the APs, is then be used to determine the location of the user. Figure 6 illustrates an example in which AP4's location is determined using AP1, AP2, and AP3.

The algorithm we actually experimented with is simpler and is discussed in Section 6.2.3.1.

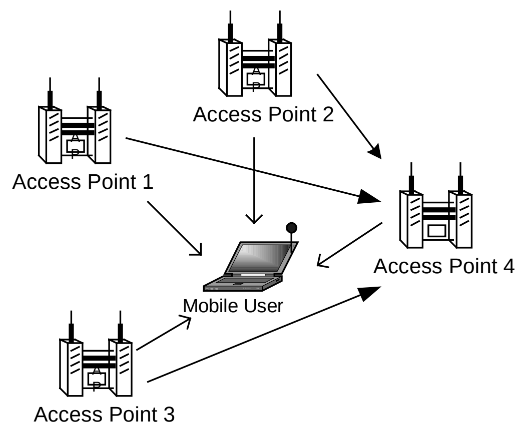

Figure 6: Access point-based environmental profiling: Beacon packets from neighboring APs are used to estimate (known) location of the target AP (AP4) using different Radio Maps.

6.2.3 Experimental Validation

Our evaluation of environmental profiling is in two parts. First, we investigate the feasibility of having the APs probe the environment accurately. Second, we evaluate the impact of environmental profiling on the accuracy of the location estimate.

6.2.3.1 Feasibility of Probing the Environment

The APs probe the environment to determine when there is a significant shift in the radio propagation environment, which would, in turn, cause the NNSS algorithm to switch to a different pre-computed Radio Map. We designed a simple algorithm for the APs to use to determine when and to which new (environmental) state to switch. Each AP records signal strength samples extracted from beacons and packets received from other APs within range. For each other AP, say AP_i, it computes the mean, m_i, of the received signal strength samples over a sliding window w samples. It uses m_i together with the pre-computed mean (μ_e) and standard deviation (σ_e) of the signal strength corresponding to each environmental state, e, to estimate the likelihood that the received signal strength samples are in conformance with that environmental state. We assume a Gaussian (Normal) distribution, $N(\mu_e,\sigma_e)$, for the signal strength and quantify the likelihood that the mean, m_i, conforms to the distribution using the probability density function (PDF) of the $N(\mu_e,\sigma_e)$ distribution. For each environmental state, e, the likelihood of match determined by each pair of APs are multiplied together to obtain an overall estimate of the likelihood of the environmental state being e. The environment e_{max} with the highest likelihood of match is then guessed to be the true environmental state.

Each time a new set of signal strength samples is received (such as from periodic beacons), the sliding window used for averaging is moved forward one step and the computation described above is repeated. Whenever the computation determines a state other than the current one to be the best match, RADAR transitions to the new state.

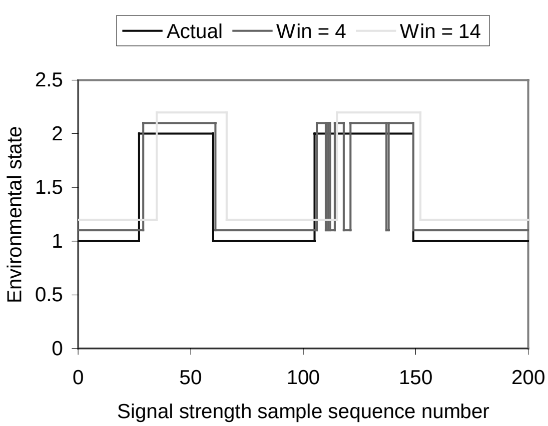

Figure 7: The transitions, both actual and inferred, between two different environmental states.

We conducted a simple experiment to evaluate this algorithm. We temporarily placed a pair of laptops in our campus cafeteria. We place one laptop in one corner and ran a program that periodically broadcast 4-byte UDP packets. We placed the second laptop at a different corner and recorded the signal strength from broadcast packets. We performed the experiment during two periods in the day – one at lunch time, between 11:40 a.m. and 12:20 p.m. when there are many people in the cafeteria (the *busy* period) and the other near the end of the business day, between 4:00 and 4:40 p.m. when there are very few people in the cafeteria (the *lean* period). We used the first half of each trace to estimate the μ and σ for the corresponding environment. This information is summarized in Table 2. As we would expect, the greater concentration of crowds during the busy hour results in a smaller mean but a greater degree of variability in the signal strength.

	Mean	Std. Deviation
Busy hour	46.07	2.41
Non-busy hour	50.05	1.19

Table 2: Characteristics of the received signal strength in two different environments.

We used the second half of the two traces to create an artificial signal strength trace by splicing together snippets from the two different environments alternately. This artificial trace had a total of 4 state transitions, as shown by the curve marked "Actual" in Figure 7. We show how our environment state inference algorithm performs for two sizes of the averaging window w — 4 and 14. We observe that with w = 4, there are several false transitions because the inference process maintains little history. On the other hand, with w = 14, the inferred transitions track the actual transitions well, but with a significant lag.

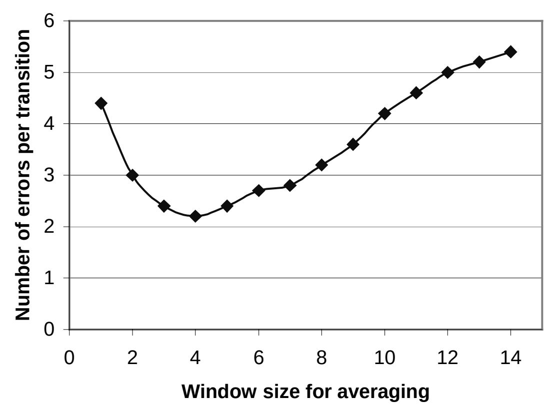

Figure 8: The error in inferring the environmental state as a function of the window size used for averaging signal strength samples.

The accuracy of the inference algorithm is impacted both by the presence of false transitions and the time lag between the actual and the inferred transitions. We quantify both of these using a single metric, namely the number of time instances when the inferred state was different from the true state. We normalize the number of such errors by dividing by the number of actual state transitions in the duration of the artificial trace. This is plotted in Figure 8. We see that w = 4 minimizes the degree to which state inference is erroneous, despite suffering from a large number of false transitions.

In summary, this simple experiment suggests that it is indeed feasible to quickly infer significant changes in the environment using our inference algorithm.

6.2.3.2 Impact of Environmental Profiling on the Accuracy of Location Estimation

We now investigate how important it actually is to infer the correct environmental state and feed in the corresponding Radio Map to the NNSS algorithm. In other words, does using the correct Radio Map have a significant impact on the accuracy of location determination?

Our office building, in which the Aironet network with 5 APs is deployed, is a spacious and rather sparsely populated building. As such there is not much of a variation in the RF environment over time. In contrast, we would expect a place such as a shopping mall to undergo significant shifts in the environment as crowds gather and dissipate. We simulated such variations in our environment by introducing artificial obstructions.

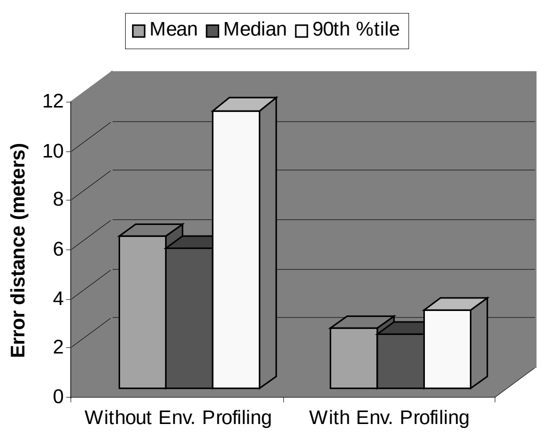

Figure 9: Performance of the NNSS algorithm with and without environmental profiling.

For the specific experiment described here, we constructed two different Radio Maps — one during normal operation ("non-busy" hour) and another when 2 of the 5 APs had barriers placed right next to them ("busy" hour). We took signal strength samples from the "busy" hour and inferred user location using the NNSS algorithm. We did this in two ways — using the Radio Map constructed during the "non-busy" hour (this simulates the case where environmental profiling is *not* performed) and using the map from the "busy" hour (i.e., using the correct map determined via environmental profiling).

Figure 9 shows the results. We observe that using the incorrect map results in far worse performance than when environmental profiling is used to pick the correct map. For instance, the 90th percentile of the error distance is 11.29 m in the former case compared to 3.16 meters in the latter case. This significant difference underscores the value of doing environmental profiling when there are significant shifts in the radio propagation environment.

6.3 The Effect of Multiple Floors

Our analysis thus far has been in the context of RADAR deployed on a single floor of a building. A logical follow-on question we would like to answer is how well RADAR performs in a building with a wireless LAN deployed on multiple floors.

Our building has 4 floors. We picked five points each on all except the ground floor, with the same (x,y) coordinates on each floor. This gave us a total of 15 points. We placed 3 access points on one of the floors and measured the beacon signal strength at each of the 15 points. We observed that the floor caused an attenuation of at least 9 to 10 dBm between points directly above or below one another. For points with different (x,y) coordinates on different floors, the attenuation was even larger. Despite the physical proximity between points on adjacent floors, signal aliasing between a point on a floor and the corresponding point on an adjacent floor is unlikely because the floor acts as a significant barrier to signal propagation.

Based on our measurements, we conclude that RADAR would work well in a multi-floor environment. Of course, a Radio Map of all of the floors would have to be constructed, not just of one floor. Despite the similarity to the single-floor case, one significant difference is that aliasing, though unlikely, could be a lot more problematic in a multi-floor setting. For instance, it is one thing to have a print job sent to the wrong printer on the same floor but quite a bit worse to have it sent to a printer on the wrong floor! Our continuous tracking algorithm (Section 6.1) could be used to minimize the occurrence of this problem.

7 Implementation Insights

In this section we discuss some of practical issues we had to face in building the RADAR system.

7.1 Effect of Multiple Channels on RADAR

In designing our system we sidestepped one important issue that affects the deployment of RADAR. RADAR requires that the mobile node capture beacon packets from all the APs that it can hear. Since neighboring APs generally operate on different channels (a consequence of the classical frequency re-use requirement in cell-based networks [24]), the mobile has to enter an “active-scan” mode, in which it scans all channels[3]. In each channel it listens for beacon packets emanating from APs operating on that channel. The Aironet 4800 APs beacon every 100 milliseconds, and the DEC’s RoamAbout™ APs broadcast beacon packets every 200 milliseconds. The overhead of switching to a channel and waiting for a beacon can be significant.

[3] The 2.4 GHz ISM band is partitioned into 11 channels. Only 3 of these are non-overlapping. In our deployment we use only these three channels (Ch.1: 2412 Mhz, Ch 6: 2437 Mhz, and Ch. 11: 2462 MHz)

One way of avoiding the switching overhead is to have multiple APs on the same channel. This however, is not a desirable alternative since it increases the system cost and complicates network planning and management. An alternative approach is to carefully schedule the channel switching so that minimal time is spent waiting for beacon packets in the channel switched to.

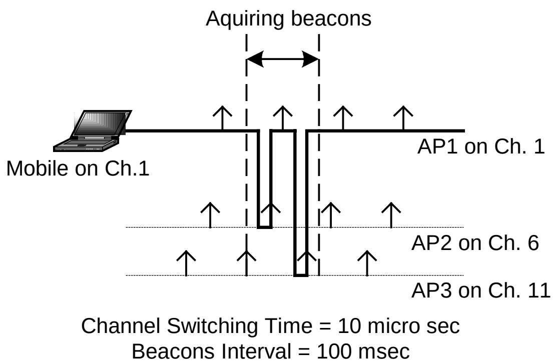

Figure 10: Mobile acquiring beacon packets from neighboring APs.

The idea is to synchronize the mobile with the APs and then exploit the fact that APs broadcast beacon packets periodically. If the mobile switches channels just before the beacon packet is expected, it minimizes the waiting period.

Figure 10 illustrates this concept graphically. A mobile operating on Channel 1, switches to Channel 2 and Channel 3 at appropriate times to grab the beacon packets. In between it switches back to Channel 1 to continue with its data transfer. For our Aironet hardware we found that the channel switching time was a steady 10 microseconds. The time to grab the signal strength information from the kernel to the user-level RADAR varies between 1 and 10 milliseconds depending on how heavily loaded the system is.

To measure the tightness with which the mobile can be synchronized to the AP, we measured the round trip propagation delay between the mobile and AP. We took two cases into account: (1) when the mobile is not transmitting any data to the AP and (2) when the mobile is busy with downloading streaming video over the net. We found that in the first case the round trip propagation delay was about 3 milliseconds while in the second case, it was about 15 milliseconds in the presence of intensive data transfer.

Well-established algorithms such as NTP [30] are able to synchronize nodes to within a few milliseconds. So the overhead of channel switching in RADAR can be reduced significantly[4].

[4] Unfortunately, as of this writing, due to a bug in the firmware, which we didn't have access to, we were unable to implement our channel switching algorithm on our deployed testbeds

7.2 Wireless is not Ethernet

Unfortunately, under the current implementation of most operating systems, a local area wireless network is generally treated as "just another network" and is exposed to the higher layer networking protocols, operating system and applications just as an Ethernet network. The interfaces designed within the context of wired Ethernet networks do not provide the necessary programming support or flexibility to support systems like RADAR and new applications such as location aware computing.

RADAR uses signal strength measurements to locate and track users. In order to extract the signal strength information from the beacons received by the mobile's wireless network interface card we had to extend Window's *Network Device Interface Specification (NDIS)* [26] to include a programming interface that exposes the wireless specific features of the underlying hardware.

In particular, we created WiLIB, a software library that provides user-level control of the underlying wireless hardware. In order to create WiLIB we extended NDIS in three areas: static queries, dynamic queries, and attribute setting, all within the context of a wireless device. In order to make NDIS extensions useful we had to extend Aironet's NDIS mini-port driver. An NDIS mini-port driver is a hardware device driver usually provided by the hardware vendor. We enhanced this hardware device driver to expose certain aspects of the hardware and the channel that were previously not available to the higher-level protocols and known only to the firmware. We were motivated to take a structured approach to creating these extensions so that in the future if hardware vendors decided to support our proposed NDIS wireless LAN extensions, researchers could write wireless-aware applications without having to worry about the characteristics of the underlying hardware, at least on Windows based machines. The complete discussion of WiLIB is beyond the scope of this paper, however we point out that to enable RADAR, we incorporated the following functionality in WiLIB and NDIS: (1) For every incoming packet the following information is retrievable by the software: signal strength, noise floor at the transmitter, noise floor at the receiver, and the MAC address of the transmitter. (2) A list of APs that a mobile node can hear beacons from can be obtained from the wireless network interface card (WNIC). This list includes all the above information for each of the APs, and (3) the WNIC can be configured to operate at a particular channel. This allows for promiscuous mode of operation where the WNIC can be programmed to gather beacon information from all neighboring APs.

8 Conclusions

In this paper, we discussed the problem of locating users inside buildings using a radio-frequency (RF) wireless LAN. Our goal is to leverage the existing wireless data communication infrastructure rather than require a specialized infrastructure solely for locating users. We identified limitations of the basic system that we had previously developed and presented new enhancements to the system to significantly improve performance. These include:

1. A continuous user tracking technique, based on a Viterbi-like algorithm, which improves the accuracy of user location by over 33%. It also helps alleviate problems due to signal aliasing.

2. An environment profiling technique, which makes the system resilient to variations in the radio propagation environment caused by crowds, weather, etc. and which can improve the accuracy by over a factor of 3.

3. Extension of the basic NNSS algorithm to a 3-dimensional space, i.e., to multiple floors in a building.

We have discussed details of our deployment of the system, the implementation insights we gained in the process, and our evaluation of the system's performance.

9 Discussions

Our results from deploying RADAR in two different buildings using two different wireless LAN technologies have been very encouraging. However, as we noted earlier, the environment in our building doesn't change much. A big question is how well RADAR would perform in a "real-world" setting, such as a shopping mall, where the constant movement of a varying number of people changes the RF propagation environment considerably. To answer to this question, we are in the process of deploying a wireless LAN network, in a local mall near our campus. We will test our system in this mall by lending users laptops and wireless network cards encouraging them to use the location-based applications that are enabled by RADAR.

In order to perform effectively, RADAR requires that beacons from at least three APs be heard at all physical locations. This would require APs to be deployed close enough together that their areas of coverage overlap significantly. Such a deployment might be expensive. To address this problem, we are toying with the idea of a "light" AP. The sole purpose of a light AP would be to augment radio coverage in regions where the APs of the wireless LAN do not provide overlapping coverage. A light AP will only transmit beacons periodically. It would not have any data networking functionality, which would make it inexpensive. While a light AP does constitute infrastructure over and beyond that needed for data networking, it uses the same RF technology as the data networking thereby obviating the need for specialized hardware on the mobile hosts.

References

[1] G. J. Nelson, "Context-Aware and Location Systems," Ph.D. Theses, Cambridge University, U.K., January 1998

[2] R. Want, A. Hopper, V. Falcao, J. Gibbons. "The Active Badge Location System," *ACM Transactions on Information Systems*, Vol. 10, No. 1, January 1992, pp 91-102. (ORL Technical Report 92.1)

[3] R. Azuma, "Tracking Requirements for Augmented Reality," *Communications of the ACM*, Vol. 36, No. 7, pp: 50-51, July 1993

[4] N. Adams, R. Gold, B. N. Schilit, M. Tso, and R. Want. An Infrared Network for Mobile Computers. In *Proceedings of the USENIX Symposium on Mobile and Location-independent Computing*, pages 41-52, Cambridge, MA, August 1993. USENIX Association.

[5] F. Bennett and A. Harter. "Low Bandwidth Infra-Red Networks and Protocols for Mobile Communicating Devices," *Oracle Research Lab (ORL) Technical Report* 93.5.

[6] A. Ward, A. Jones, and A. Hopper. "A New Location Technique for the Active Office," *IEEE Personal Communications*, Vol. 4, No. 5, October 1997, pp 42-47. (ORL Technical Report 97.10)

[7] T. W. Christ, P. A. Godwin, "A Prison Guard Duress Alarm Location System", Proc. IEEE International Carnahan Conference on Security Technology, October 1993

[8] J. Werb, C. Lanzl, "Designing a Positioning system for Finding Things and People indoors," IEEE Spectrum, (September 1998): 71-78 (also see http://www.pinpointco.com)

[9] IEEE Std. 802-11.1997, IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification. Approved 26 June 1997.

[10] P. Bahl and V. N. Padmanabhan, "RADAR: An RF-Based In-Building User Location and Tracking System," Proc. IEEE Infocom, March 2000

[11] H. Hashemi, "The Indoor Radio Propagation Channel," *Proceedings of the IEEE*, Vol. 81, No. 7, pages 943-968 July 1993

[12] P. Enge, and P. Misra, "Special Issue on GPS: The Global positioning System," *Proceedings of the IEEE*, pp. 3-172, January 1999

[13] S. Tekinay, "Wireless Geolocation Systems and Services," Special Issue of the *IEEE Communications Magazine*, April 1998

[14] M. Moeglein, and N. Crasner, "An Introduction to SnapTrack™ Server-Aided GPS Technology," available from: http://www.snaptrack.com/pdf/ion.pdf

[15] Ascension Technology Corporation, http://www.ascension-tech.com

[16] A. Ward, "Sensor-driven Computing," Ph. D. Thesis, Cambridge University, U.K., (May, 1999)

[17] T. D. Hodes, R. H. Katz, E. S. Schreiber, and L. Rowe, "Composable Ad Hoc Mobile Services for Universal Interaction," *MobiCom '97 Proceedings*, pp: 1-12, September 1997

[18] R. Nee, et. al., "New High-Rate Wireless LAN Standards," *IEEE Communications Magazine*, Vol. 37, no. 12, pp: 82-88, December 1999

[19] R. Jain, The Art of Computer Systems Performance Analysis, *John Wiley*, 1991

[20] T. Liu, P. Bahl and I. Chlamtac, "Mobility Modeling, Location Tracking, and Trajectory Prediction in Wireless Networks", *IEEE JSAC*, Vol. 16, No. 6, pp. 922-936, August 1998

[21] H. Maass, "Location-Aware Mobile Applications based on Directory Services," *MobiCom '97*, pp. 23-33, September 1997

[22] S. Y. Seidel and T. S. Rapport, "914 MHz path loss prediction Model for Indoor Wireless Communications in Multi-floored buildings," *IEEE Trans. on Antennas & Propagation*, February 1992

[23] M. Brown. Supporting User Mobility. In *Proceedings of the IFIP'96 Conference, Mobile Communications*, Canberra, Australia, September 1996. (ORL Technical Report 96.7)

[24] T. S. Rapport, Wireless Communications – Principles and Practice, *IEEE Press*, 1996

[25] T. Liu, P. Bahl, "Mobility Modeling, Location Tracking, and Trajectory Prediction in Wireless ATM Networks", *IEEE JSAC*, Vol. 16, No. 6, pp. 922-936, August 1998

[26] P G. Viscarola and W. A. Mason, *Windows NT Device Driver Development*, OSR Open System Resources, 1999

[27] "Digital RoamAbout 915/2400 DS/PC Card and ISA Network Adapter: Installation and Configuration," *Digital Equipment Corporation*, April 1996

[28] "Developer's Reference Manual: PC4500/PC4800 PC card Wireless LAN Adapter," *Aironet Wireless Communications Inc*. 1999

[29] L. W. Couch, "Digital and Analog Communication Systems," Fourth Edition

[30] D. L. Mills, "Network Time Protocol (Version 3): Specification, Implementation, and Analysis," RFC-1305, March 1992