Wireless Networks 11, 189–204, 2005

Robotics-Based Location Sensing Using Wireless Ethernet *

ANDREW M. LADD, KOSTAS E. BEKRIS, ALGIS RUDYS **, LYDIA E. KAVRAKI and DAN S. WALLACH
Department of Computer Science, Rice University, 6100 Main St., MS 132, Houston, TX 77005-1892, USA

Abstract. A key subproblem in the construction of location-aware systems is the determination of the position of a mobile device. This article describes the design, implementation and analysis of a system for determining position inside a building from measured RF signal strengths of packets on an IEEE 802.11b wireless Ethernet network. Previous approaches to location-awareness with RF signals have been severely hampered by non-Gaussian signals, noise, and complex correlations due to multi-path effects, interference and absorption. The design of our system begins with the observation that determining position from complex, noisy and non-Gaussian signals is a well-studied problem in the field of robotics. Using only off-the-shelf hardware, we achieve robust position estimation to within a meter in our experimental context and after adequate training of our system. We can also coarsely determine our orientation and can track our position as we move. Our results show that we can localize a stationary device to within 1.5 meters over 80% of the time and track a moving device to within 1 meter over 50% of the time. Both localization and tracking run in real-time. By applying recent advances in probabilistic inference of position and sensor fusion from noisy signals, we show that the RF emissions from base stations as measured by off-the-shelf wireless Ethernet cards are sufficiently rich in information to permit a mobile device to reliably track its location.

Keywords: wireless networks, 802.11, mobile systems, localization, probabilistic analysis

1. Introduction

There has been great progress in wireless communications over the last decade, causing the available mobile tools and the emerging mobile applications to become more sophisticated. At the same time, wireless networking is becoming a critical component of networking infrastructure. Wireless technology enables mobility which, in turn, creates a need for location-aware applications. The recent interest in location sensing for network applications and the growing need for large-scale commercial deployment of such systems has brought network researchers up against a fundamental and well-studied problem in the field of robotics: determination of physical position using uncertain sensors, or localization.

Robot localization is the process of maintaining an ongoing estimate of a robot's location with respect to its environment, given a representation of this environment and some sensing ability within the environment. Localization has been described as "the most fundamental problem to providing a mobile robot with autonomous capabilities" [9]. We can treat the wireless device like a mobile robot for purposes of localization. If there is no a priori estimate of the robot's location, the problem is referred to as *global localization*, a particularly challenging case of localization. This is the type of problem we are discussing: the device has no information on where it is before it starts communicating with the network.

* An earlier version of this article was published as A.M. Ladd, K.E. Bekris, A. Rudys, G. Marceau, L.E. Kavraki and D.S. Wallach, Robotics-based location sensing using wireless Ethernet, at *the 8th ACM International Conference on Mobile Computing and Networking (MOBICOM)* (September 2002, Atlanta, GA).

** Corresponding author.
E-mail: arudys@cs.rice.edu

Many mobile devices and many buildings, both commercial and residential, are already equipped with off-the-shelf IEEE 802.11b wireless Ethernet, a popular and inexpensive technology. Furthermore, most wireless Ethernet devices already measure signal strength of received packets as part of their standard operation and the signal strength varies noticeably as the distance and obstacles between wireless nodes change. If a reliable localization system could be developed using only this technology, then localization services can be provided using these existing platforms. Similarly, in robotics applications, wireless Ethernet localization can be combined with other sensors to improve the robustness of location estimates.

The development of efficient and accurate location-support systems for indoor environments, which would also have the potential of being widely available, is a challenging task. The limitations usually stem from the harsh nature of the signal and the sensors with which one has to work. Indoor environments affect the propagation of wave signals in non-trivial ways, causing severe multi-path effects, dead-spots, noise and interference [6]. These effects make it infeasible to construct a simple and accurate model of the signal's propagation in the space. A location support system has to overcome the high uncertainty due to the behavior of the indoor wireless channels but at the same time it should keep the cost and the complexity of large-scale deployment as small as possible. The remainder of this section discusses various motivations and applications for localization and introduces our approach.

1.1. Motivation

Location-awareness. In the wireless world many desirable applications require location-awareness. For example, government initiatives require that cellular phone providers should develop a way to locate any phone that makes an emergency call [14]. Location-awareness is also a prerequisite for many applications which require context-awareness. The context of an application refers to the information that is part of the application's operating environment, and typically includes information such as location, activity of people, and the state of other devices [22].

In outdoor settings, GPS [35] has been used in many commercial applications, as in the case of locating automobiles. Despite the widespread availability of GPS technology, though, most GPS devices cannot reliably receive GPS signals indoors. While some devices can receive GPS signals indoors, these devices are limited to providing only coarse-grained location information [49].

An indoor system must use different sensors, such as infrared (IR), sonar, vision, or radio (RF), to accurately infer position of a mobile device. Location-aware applications based on these sensors could enable users to discover resources in their physical proximity, such as active maps of their surroundings and adaptive interfaces to the user's location [22]. Specific applications of such a system vary from tracking a guard's position in a penitentiary institution [8] to hospitals, where equipment and people must be efficiently located [52]. These applications can also be useful in large office environments, where the loss of valuable equipment such as laptop computers has become a serious problem and locating resources such as printers takes time and disrupts other activities.

Wireless security. We are also interested in the utility of a location support system over an existing wireless network for security applications. A principal difference between wired networks and wireless networks is that physical security is no longer sufficient to ensure the security of the network. In addition, in a wireless network, the location of an intruder is considerably more difficult to determine versus a traditional wired network where cables can be traced to their source. However, a mobile device which is transmitting on a wireless Ethernet network is leaking its position. This information can be used to locate an intruder who makes no deliberate effort to decorrelate her signal from her position. Although an intruder could possibly hide by using expensive directional antennas, off-the-shelf hardware is less conspicuous and more readily-available.

Mobile robotics. Many mobile robot platforms make extensive use of wireless networking to communicate with off-line computing resources, other robots, and various user-interface devices. Since the advent of inexpensive wireless networking, many mobile robots have been equipped with 802.11b wireless Ethernet. In many applications, a sensor from which position can be inferred directly without the computational overhead of image processing or the material expense of laser range-finders is of great use. Many robotics applications would benefit from being able to use wireless Ethernet for both sensing and communication. For example, exploration, map-building and navigation with low-cost wheeled robots could be readily achieved using wireless Ethernet and sonar.

1.2. Our approach

In this article, we describe a system that achieves robust indoor localization using only RF signal strength as measured by an IEEE 802.11b wireless Ethernet card communicating with standard base stations. Since the required equipment for a wireless Ethernet network is usually already present in the workspace serving communication purposes, this reduces the cost of providing localization services in an indoor environment. This also reduces the complexity for the user of a mobile device who wishes to take advantage of this localization service. To achieve our goal, we have adapted the most successful available approaches from robotics-based localization, notably the explicit manipulation of noise distributions and the modeling of position as a probability distribution.

Our method for localizing a mobile station is divided in two phases. Initially, there is a training phase, where a sensor map of the environment is built by sampling the space and gathering data at various predefined checkpoints of the indoor environment. Later, the operator of a mobile computer walks in the same workspace and the system locates and tracks the operator's position. Our system currently assumes that the environment remains consistent from training to localization. In particular, we assume that people are minimally present when we attempt to localize, since people absorb signal.

Section 2 presents the algorithms and methodology for our localization system. The results of our experiments are reported in section 3 and a discussion of our work is presented in section 4. In section 5, we discuss related work in the fields of location-aware computing and robot localization.

2. Methodology

In this section, we discuss our methodology for determining a user's location using wireless network signal strength. We begin by discussing the platform and environment we considered. We then discuss RF signal propagation and describe some problems with devising a signal attenuation model for wireless Ethernet. Finally, we discuss our algorithms for determining the user's location.

2.1. Experimental setup

Hardware. Our experiments were conducted by a human operator carrying a Hewlett Packard OmniBook 6000 laptop with a PCMCIA LinkSys wireless Ethernet card. This particular card uses the Intersil Prism2 chipset. We modified the standard Linux kernel driver for this card to support a number of new functionalities, including the scanning and recording of hardware MAC addresses and signal strengths of packets and the automatic scanning of base stations.

We needed a constant source of signal from all base stations for optimum results. Unfortunately, this meant we could not simply be a passive observer. While we could put the network interface adapter into promiscuous mode and listen to all packets being transmitted by base stations, this can only guarantee a stream of packets from one base station: the one with which the card is currently associated. While base stations do send out beacon packets several times a second, the hardware drivers we used did not give us access to this signal.

Instead, we were forced to use the base station probe facility of 802.11 [27]. Client nodes can broadcast a probe request packet on a wireless network. Base stations that receive such a request respond with a probe response packet. The client then collects these packets and, judging by the strengths of the incoming signals, can determine the closest base station to which to connect. We analyze these signal strengths to determine our location relative to the base stations.

A given base station can appear anywhere between zero and four times in the packets the firmware returned to us. For each packet, we get an eight-bit reading representing the signal strength. This value is computed by the network card, and we have no way of determining or affecting how it is calculated. Unless the sender is very close to the receiver, signals in the top half of this range rarely occur. Certain other signal strengths simply never occur. The lowest order bit tends to be very noisy. When compared to other sensors, such as sonar, this signal is very thin: at most 5 usable bits of signal per packet.

Building geometry. We operated on the third floor of Duncan Hall at Rice University, in the four hallways shown in figure 1. The two longer hallways (hallways 1 and 2) measure 32.5 meters, and the two shorter hallways (hallways 3 and 4) measure 17 meters. Hallway 1 has a base station near one end, and hallway 2 has a base station really close to the middle. Hallways 3 and 4 are notable in that they are open above and either partially (in the case of hallway 4) or totally (in the case of hallway 3) open on the sides.

There were nine base stations distributed on this floor. Those within the area described by the map in figure 1 are marked with circles. The base stations were Apple AirPort base stations and were mounted between two and three meters off the ground. We made no changes to the locations of base station in our building, and had no input in their original placement; the base stations were laid out to service a production wireless network. We had a fairly precise map of the building that we had processed to mark off free space and obstacles. The pixel resolution was roughly six centimeters in this map.

2.2. RF signal propagation in wireless Ethernet

The IEEE 802.11b High-Rate 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 [10], so interference on a single frequency does not block the signal.

The main problem with this sensor is that an accurate prediction of the signal's strength in every position of the environment is a very complex and difficult task because the signal propagates in many unpredictable ways [37]. 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.

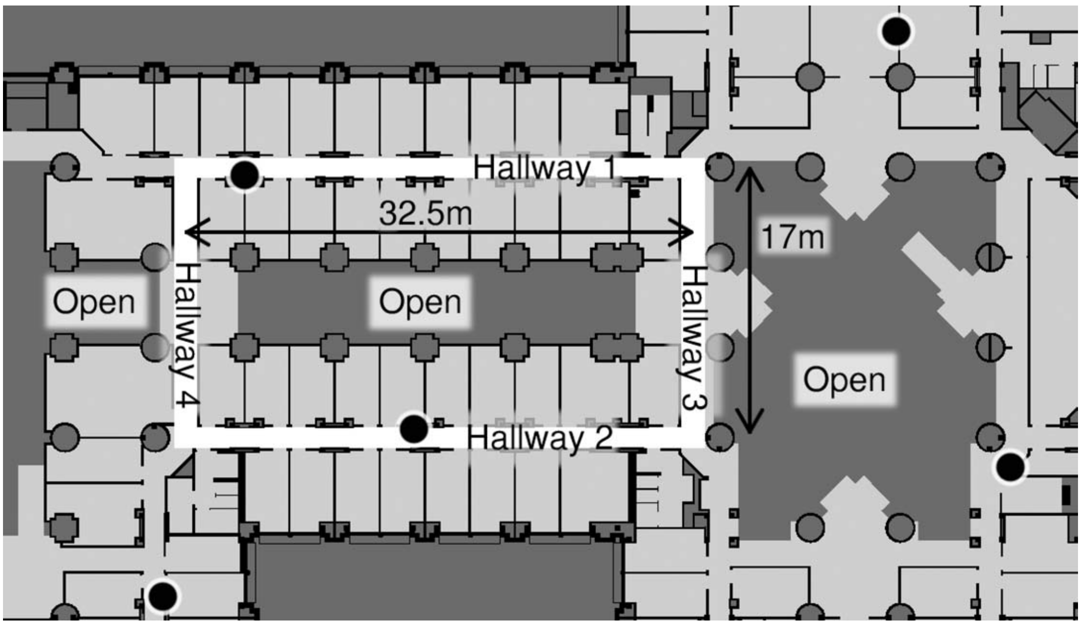

Figure 1. Map of the region of the Duncan Hall where we conducted our tests. Base stations are indicated by solid black circles on the map. Note that additional base stations outside of this region (including on other floors) were used in our experiments.

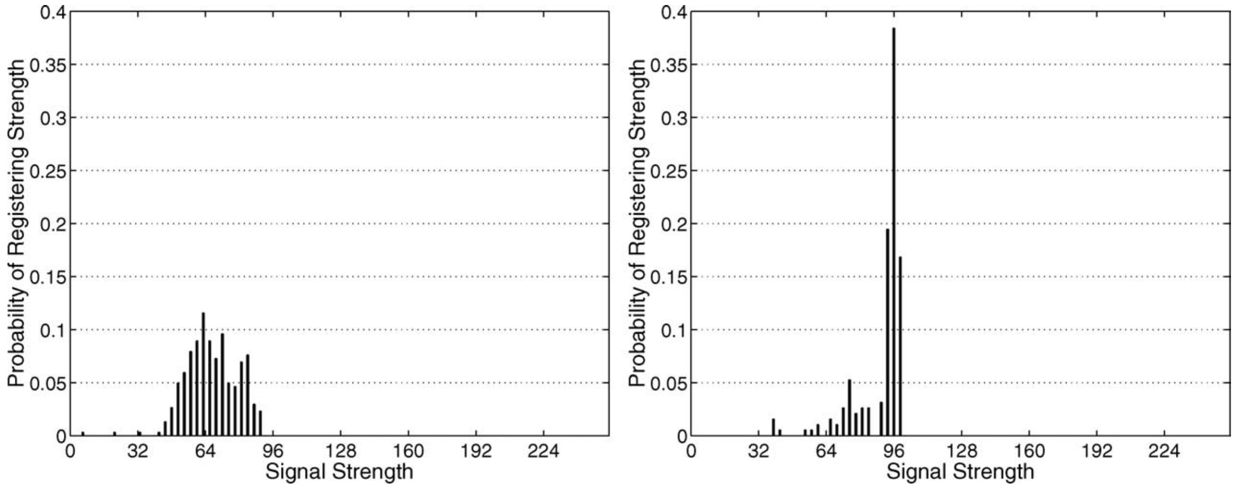

Figure 2. Samples of signal strength taken at the same positions facing opposite directions. The signal strength values indicated are the raw values returned by the network interface.

Changes in atmospheric conditions like air temperature can also affect the propagation of waves and the resulting signal strengths. Unfortunately, 2.4 GHz is absorbed by water, and consequently people will also absorb signal since human bodies are almost 70% water.

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 [10]. 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 [24]. The signal components arriving from indirect paths and the direct path, if this exists, combine and produce a distorted version of the transmitted signal.

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. However, signal profiles corresponding to spatial coupled locations are expected to be similar as the various external variables remain approximately the same over short distances [24]. The local average of the signal varies slowly with the displacement. These slow fluctuations depend mostly on environmental characteristics and are known as long-term fading.

While much effort has been made to model radio signal propagation and attenuation in indoor environments [21,23, 37], no single consistent model is available. During our initial experiments we took numerous measurements at various positions in our environment. Our objective was to try to see if the variables in the system could be captured with a simple theoretical model to minimize the training phase. We observed a number of interesting properties of RF signals in our environment.

Orientation matters. The authors of RADAR established a correlation between orientation and measured signal strength [2,3]. We also observed this. The laptop and the operator affect the signal in a measurable way. It is interesting to note that the presence of the operator affects signal strength and gives the omnidirectional signals some weakly directional properties. Typically the mean signal strength varies less than the statistical distribution of signal strengths. In figure 2, we give an example of two distributions sampled at the same points while facing in opposite directions.

Noise distribution non-Gaussian. The noise distributions at a fixed position were very heterogeneous as we varied the pose and base station that we sampled. In figure 3, we show two typical examples of the signal strength from two different base stations measured simultaneously at the same physical position. Several hundred samples were taken in about 45 seconds for these particular histograms. Notice that the first-order properties of these distributions differ greatly from each other. In general, we observed that distributions were asymmetric and had multiple modes. There was usually a dominant mode which often differed from the mean. We concluded that distributions were essentially non-Gaussian. Since the noise behavior is an extremely complex physical phenomenon and explicit histograms are fairly compact, we decided that it would be better to work directly with these distributions rather than reduce the data to average values.

We found it useful to postprocess the sampled distributions by applying a small window smoothing convolution, adding a very small uniform baseline distribution and then normalizing. This is done to try to artificially compensate for sampling errors and allow for a small probability of unexpected measurements in the Bayesian inference cal-

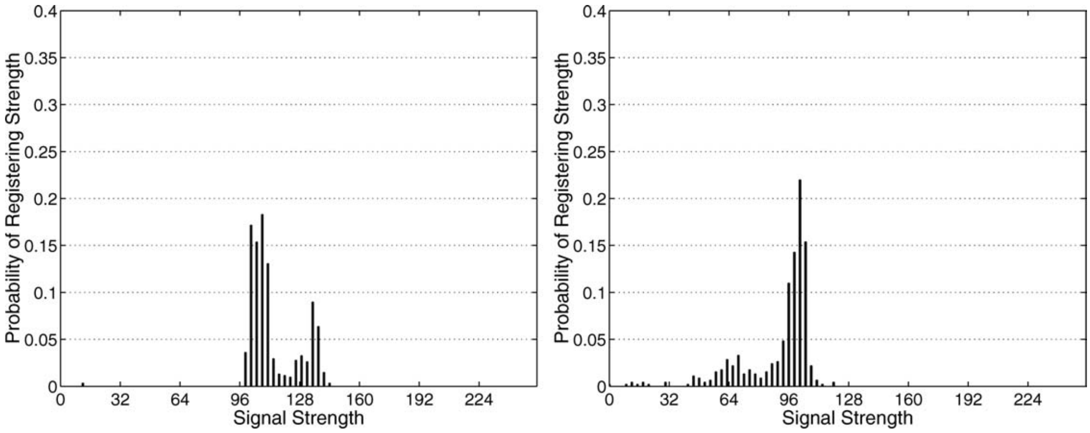

Figure 3. Samples of signal strength distributions of two different base stations, measured simultaneously from one location. The signal strength values indicated are the raw values returned by the network interface.

Table 1
Table of variables and expressions used to explain our Bayesian inference engine and the meanings of these variables.

Symbol	Meaning
n	Number of states
m	Number of possible observations
N	Number of distinct base stations that are ever encountered
i, α	Indexes into the state space
j	Index into the observation space
β	Index into the set of base stations
π	A probability vector over the states
π_i	Entry in the probability vector for state i That is, π_i is the probability that the current state is i
π'	Next probability vector. Generally, $\pi' = f(\pi)$ for some function f
π'_i	Entry in the probability vector π' for state i
S	The set of all states, $S = \{s_1, \dots, s_n\}$
s_i	The ith state, $s_i = (x_i, y_i, \theta_i)$
x_i	The x-coordinate of state i
y_i	The y-coordinate of state i
θ_i	The facing direction (angle) of state i
O	The set of all possible observations, $O = \{o_1, \dots, o_m\}$
o_j	The jth possible observation, $o_j = \langle k, f_1, \dots, f_N, (b_1, \lambda_1), \dots, (b_k, \lambda_k) \rangle$
k	Number of individual signal strength measurements in the current observation
ρ	Index into the set of measurements in the current observation
f_β	Number of measurements in the current observation corresponding to base station β
b_ρ	Base station corresponding to measurement ρ
λ_ρ	Signal strength corresponding to measurement ρ
$Pr(o_j \mid s_i)$	The probability of seeing observation o_j while in state s_i
$Pr(f_\beta \mid s_i)$	The probability of an observation taken at state s_i containing f_β measurements of base station β
$Pr(\lambda_\rho \mid b_\rho, s_i)$	The probability of an observation taken at state s_i containing a measurement of base station b_ρ with signal strength λ_ρ

culations that follow. These corrections produced minor but noticeable improvements in the precision of the calculations.

2.3. A Bayesian inference algorithm

Possibly the most powerful family of global localization algorithms to date is based on Bayesian inference, including Markov localization [17,28] and Monte Carlo localization [15,48]. Many examples of Bayesian approaches to localization exist in the robotics literature [46]. These algorithms estimate posterior distributions over robot poses which are approximated by piecewise constant functions instead of Gaussians, enabling them to represent highly multi-modal distributions. In this way, they can be applied in the case of sensors that have non-Gaussian noise distributions, such as our signal strength sensor. We use a Bayesian algorithm to implement localization using wireless Ethernet.

We model the world as a finite space $S = \{s_1, \dots, s_n\}$ of *states* with a finite *observation* space $O = \{o_1, \dots, o_m\}$. The *sensor model* is some learned or predicted model of the conditional probabilities of seeing some observation o_j at state s_i, in other words $Pr(o_j \mid s_i)$. A *state vector* π is a probability vector (distribution) over the various states. An explanation of the meanings of these and all the other variables used in this section can be found in table 1.

Position is represented as a probability distribution over the states. The inference calculation starts with a prior estimate of our state π. After making an observation o_j, we can calculate the individual conditional probabilities π_i' for each state s_i where $1 \leqslant i \leqslant n$ using Bayes' rule,

$$\pi_i' = \frac{\pi_i Pr(o_j|s_i)}{\sum_{\alpha=1}^{n}(\pi_\alpha Pr(o_j|s_\alpha))}. \tag{1}$$

We then combine these individual π_i's into a new estimate of our state, π'. We can now select a representative point from the resulting distribution: our position estimation.

This is a simple principle on which probabilistic inference schemes are built. To implement our system, however, we need to make several design decisions. We first choose appropriate state and observation spaces. This involves deciding on a sampling granularity for both spaces. We then learn the conditional probability distributions for plugging into the formula above.

2.3.1. Our model

Our initial experiments and literature search indicated that a priori models of RF signal propagation would be difficult to set up without some on-site training. After verifying that simple assumptions such as fitting analytic curves and surfaces to the means and Gaussians or other simple distributions to the variances provide poor fits to sampled data, we opted for the simpler, more robust scheme of sampling the conditional probabilities directly. The reasoning for this is discussed further in section 2.1.

We begin by defining our state space. We choose a set of points on the map, each tuple (x, y, θ) a location and orientation on the floor of Duncan Hall where our experiments took place. There is no indication that adding an additional parameter for three-dimensional localization would be any harder, although we did no experiments to verify this. Our state space S consists of a set of n points

$$S = \{s_1 = (x_1, y_1, \theta_1), \ldots, s_n = (x_n, y_n, \theta_n)\}. \tag{2}$$

Each observation in our observation space consists of the measurements that occurred in a single scan from our base station scanner. Each base station scan returns a set of k base station signal strength measurements. Each base station can appear in the scan up to four times. We represent each observation as a vector

$$o = \langle k, f_1, \ldots, f_N, (b_1, \lambda_1), \ldots, (b_k, \lambda_k)\rangle, \tag{3}$$

where k is the total number of base station signal strength measurements, N is the total number of unique base stations represented, f_β is the frequency count for the βth base station, b_ρ represents the base station in the ρth measurement and λ_ρ is the signal strength of that measurement. An explanation of the meanings of these and all the other variables used in this section can be found in table 1.

In the training phase, at each point s_i, we take a number of observations. For each base station we build two histograms for that point. The first is the distribution of frequency counts over the sampled observations. The second is a distribution of observed signal strengths. These histograms encode two conditional probabilities. $Pr(f_\beta|s_i)$ is the probability that the frequency count for the βth base station is f_β when we are at state s_i. $Pr(\lambda_\rho|b_\rho, s_i)$ is the probability that a measurement appears for base station b_ρ with signal strength λ_ρ at state s_i. By multiplying these conditional probabilities we obtain the conditional probability of receiving a particular observation. For each observation o_j, as defined in equation (3), we compute

$$Pr(o_j|s_i) = \left(\prod_{\beta=1}^{N} Pr(f_\beta|s_i)\right)\left(\prod_{\rho=1}^{k} Pr(\lambda_\rho|b_\rho, s_i)\right). \tag{4}$$

Once training is complete, we move on to localization. The exact calculation proceeds as follows: before each observation we choose our prior state distribution π as the uniform distribution. This is a common Bayesian assumption: we assume we are lost, so every position is equally likely. This provides a conservative estimate of our location; any attempt to bias this initial estimate may inhibit accurate localization right from the start. When we make the observation, we simply use Bayes' rule as defined in equation (1) to compute π', the probability distribution over the states. Then it is simply a matter of choosing appropriate candidate locations.

After trying several possible schemes, we decided to solve a global localization problem for each observation rather than keep a running estimate because each observation usually contains enough information to get a good guess of our position. The resulting stream of guesses can be combined in a post-processing step to create a more refined estimate of position. One such mechanism is described in section 2.4.

Although one observation is typically enough information to decide on one's position, errors in the training phase can lead to inaccuracy during localization. Significant causes of such error are subsampling and time-dependent phenomena. Subsampling can create a posteriori model of the noise as measured at that point. Certain measurements that occur rarely may never occur in the subsample. When the measurement occurs online, the hypothesis can be rejected entirely based on a conditional probability of zero for that position. We describe heuristics compensating for this difficulty in section 4.

2.3.2. Naïve averaging

To improve this estimate, we apply a simple post-processing technique. For each $1 \leqslant i \leqslant n$, where n is the number of states,

$$\pi_i'' = (\pi_i + u_1)(\pi_i' + u_2), \tag{5}$$

π is the prior distribution on position, π' is the probability distribution computed with Bayesian inference, as describes in section 2.3, π'' is the revised distribution, and u_1, u_2 are small constants introduced to prevent the distribution from equaling zero. The resulting distribution π'' needs to be normalized after this calculation.

Table 2
Table of variables and expressions used to explain our HMM sensor fusion mechanism and the meanings of these variables.

Symbol	Meaning
n	Number of states
m	Number of possible observations
S	The set of all states, $S = \{s_1, \ldots, s_n\}$
O	The set of all possible observations, $O = \{o_1, \ldots, o_m\}$. Each observation o represents a position estimate made by our Bayesian inference engine
π	A probability distribution over the state space, S. As above, π' represents the "next" distribution, which is to say, $\pi' = f(\pi)$ for some function f
λ	A conditional probability function $\lambda : S \times O \to [0, 1]$. λ is defined as $\lambda(s, o) = Pr(o\|s)$, the probability of observing o while in state s
A	A transition probability matrix. A represents the possibility that a random state change has occurred "hidden" from the observer

2.4. Sensor fusion

Sensor fusion is another important concept in robot localization. A broad definition of sensor fusion is the combination of multiple independent observations to obtain a more robust and precise estimate of the measured variables. This can be implemented in terms of integrating sensor readings over time or in the synthesis of measurements from multiple sensors. We use this technique to refine our initial location estimate. We implemented a filter which takes the output of the inference engine as a stream of timed observations and tries to stabilize the distribution by noting that a person carrying a laptop typically does not move very quickly. It also takes into account some probability of error on the part of the inference engine.

We model a moving operator trying to track her position as a hidden Markov model (HMM) [38,41]. We use a more finely discretized state space than the Bayesian inference engine and try to interpolate the operator's position out of the stream of measurements coming from the inference engine. We chose this finer discretization after observing that a naïve averaging of the inference engine's output, described in section 2.3.2, produced results with twice the precision we expected for points where we had not taken any training samples.

For our purposes, an HMM is a set of states $S = \{s_1, \ldots, s_n\}$, a set of observations $O = \{o_1, \ldots, o_m\}$, a conditional probability function $\lambda : S \times O \to [0, 1]$, and a transition probability matrix A. Each state and each observation is a point (x, y, θ). An observation represents a position estimate made by our Bayesian inference engine. That is, the set of observations for the HMM is the same as the set of states for the Bayesian inference engine. An explanation of the meanings of these and all the other variables used in this section can be found in table 2.

The transition probability matrix semantics describe how the system being modeled evolves with time. In this case, it describes how a person travels through the state space. If π is a probability distribution over S, then $\pi' = A\pi$ is the probability distribution after some discrete time step. The idea is that the random state change occurs "hidden" from the observer. We generate the transitional probability matrix A using a relatively simple heuristic, that people do not travel too fast or change directions too frequently.

The observation function λ has semantics identical to observation in the Bayesian inference of position. $\lambda(s, o) = Pr(o|s)$, the probability of observing o while at s. The conditional probability function λ is also defined using a relatively simple heuristic; smaller distances from an observation to a given state lead to higher probabilities of making that observation at that state. As each observation arrives, λ is used to update the probability of being in a given state in S, and then A is used to transition states. If λ accurately models the behavior of the inference engine and A accurately models the behavior of a person transitioning from state to state, the sensor fusion will have superior results to Bayesian inference alone.

3. Results

In this section we describe several experiments which try to objectively measure the precision and reliability of our system. We first present the results for static localization. We then describe the results for user tracking using sensor fusion.

Our system was trained by taking samples at various points in the world, as discussed in section 2.3.1. The amount of data taken at each point is varied adaptively according to a simple heuristic which measures the rate of convergence to a stable distribution. Once the sampled distribution at each visible base station had converged beyond a threshold, we halt the process. This allowed us to adaptively determine how much sampling is necessary as a function of variation in the signal. In our case, usual sampling times ranged from ten seconds to about a minute.

3.1. Static localization in a hallway

This subsection describes experiments executed in hallway 1 of our test area (see figure 1), which was sampled in two different orientations at every 1.5 meters. The purpose of this was to test the precision of the Bayesian inference localizer. Timed tests occurred at various positions and at both orientations in the hallway and bulk statistics were calculated.

The training data was taken by two different operators, with each operator training the localizer in one of the two directions. All experiments were executed by a third operator. The purpose of this was to demonstrate a degree of operator-independence.

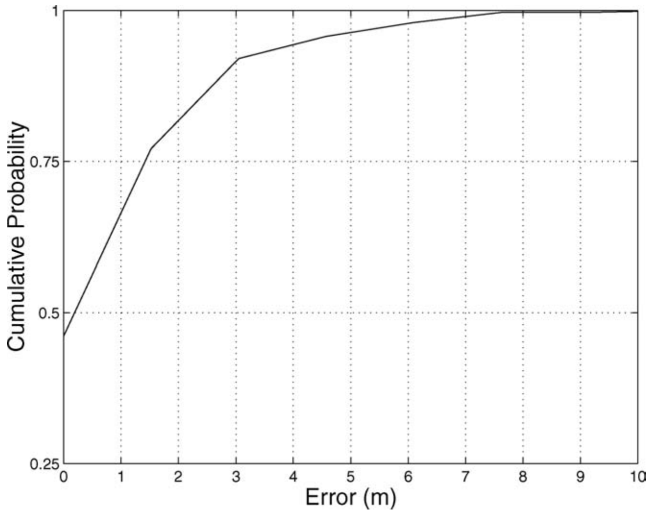

Figure 4. Bulk cumulative error distribution for 1307 packets over 22 poses in hallway 1 in our test area (see figure 1, the building map) localized using the position of maximum probability as calculated by direct application of Bayes' rule.

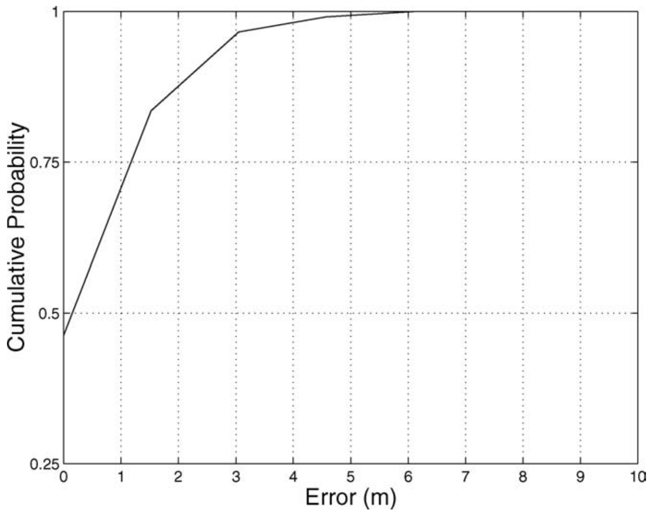

Figure 5. Bulk cumulative error distribution for 1465 packets over 22 poses in hallway 1 in our test area (see figure 1, the building map) localized using the position of maximum probability as calculated by merging distributions over a one second window.

Basic Bayesian inference. Using the algorithm described in section 2.3, we measured a total of 1307 packets over both orientations on 11 different positions. The positions were spread every 3 meters to be exhaustive. The algorithm reported positions back discretized to 1.5 meters. In figure 4, we show the cumulative probability of obtaining error less than a given distance. We have observed that error is at most 1.5 meters with probability 0.77.

Simple averaging improves results. In the second experiment, we post-processed the probability distributions computed by Bayesian inference as described in section 2.3.2. This simple calculation improved our results significantly and is usable as a tracker. Our results are summarized in figure 5. The measured error was at most 1.5 meters with probability 0.83. This is an 8% improvement over the raw filter. As a tracker, we observed that it lagged behind the actual position and we attempted to improve our results by using more sophisticated methods described in section 2.4.

Operator bias. The above experiments were run with one person operating the laptop to generate training data and a different person operating the laptop to localize based on that training data. Our experiments indicate that operator bias introduced in training does not cause localization results to become unstable.

3.2. Experiments with tracking

We attempted to improve these results by implementing a more sophisticated sensor fusion based on a hidden Markov model (HMM), as described in section 2.4. We then walked round-trips of the four hallways in our test area, shown on the map in figure 1, tracking our current position and recording the output of both the static localization as described in section 3.1 and the sensor fusion. The results are shown in figures 6–9.

For hallways 1 and 2, sensor fusion increased by 44% and 40%, respectively, the probability of error less than one meter. The traces show that while static localization is good at tracking, sensor fusion improves the results by effectively ignoring outliers. See figures 6 and 7 for the results on these hallways.

The results for hallway 4 were somewhat more disappointing. The probability of error less that one meter was increased by a scant 8%. Sensor fusion loosely tracked actual movement, but the signal from the static localizer was too noisy to allow for the level of accuracy achieved on hallways 1 and 2. We attribute this noise to the fact that the hallway is open above, and there are no base stations installed along the hallway. See figure 8 for the results on this hallway.

The worst result was on hallway 3, which is entirely open above, and overlooks a three-story atrium. The probability of error less than one meter actually went down by 10% as a result of sensor fusion. As seen in figure 9, the static localizer for the most part tended to choose either an endpoint or one of two particular points in the middle of the hallway. This was caused in part by the fact that this hallway is exposed to a large open area, diluting the signal. In addition, no base stations are installed close to or in line with the hallway. The result is that the signal strength from base stations does not vary much from one point on the hallway to another.

Note that the conditional probability function and transition probability matrix we used to initialize the hidden Markov model were generated based on Gaussian distributions. While these were good fits for hallways 1 and 2, they failed to model the noisiness of the static localizer on hallways 3 and 4. A conditional probability function trained to the actual points would likely provide better results.

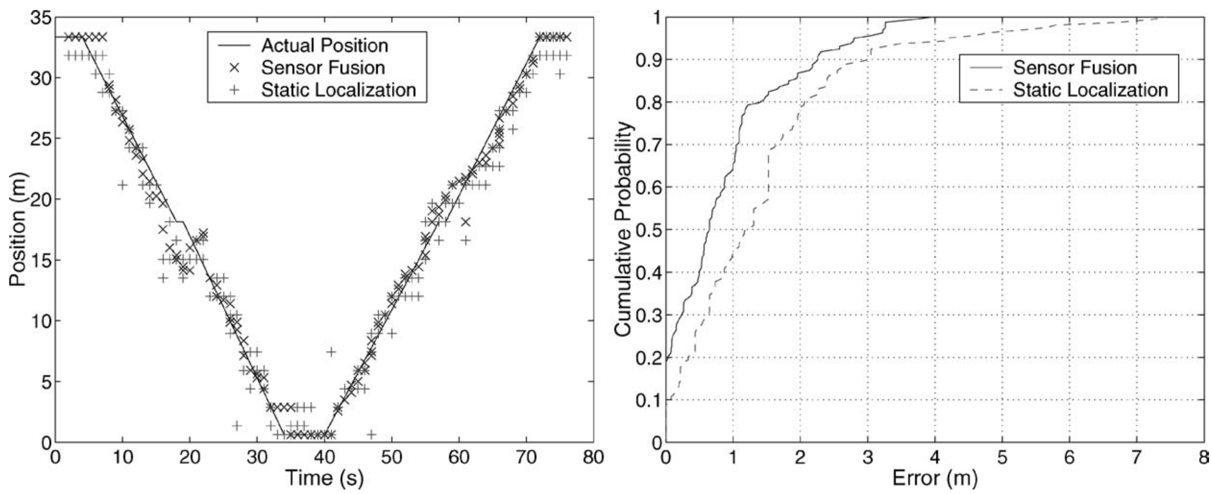

Figure 6. Tracking a round-trip walk of hallway 1 in our test area (see figure 1, the building map). Measured error for the track, shown on the right graph, is at most one meter with probability 0.64, an improvement of 45% over static localization. This improvement is illustrated in the actual tracking performance, shown in the left graph.

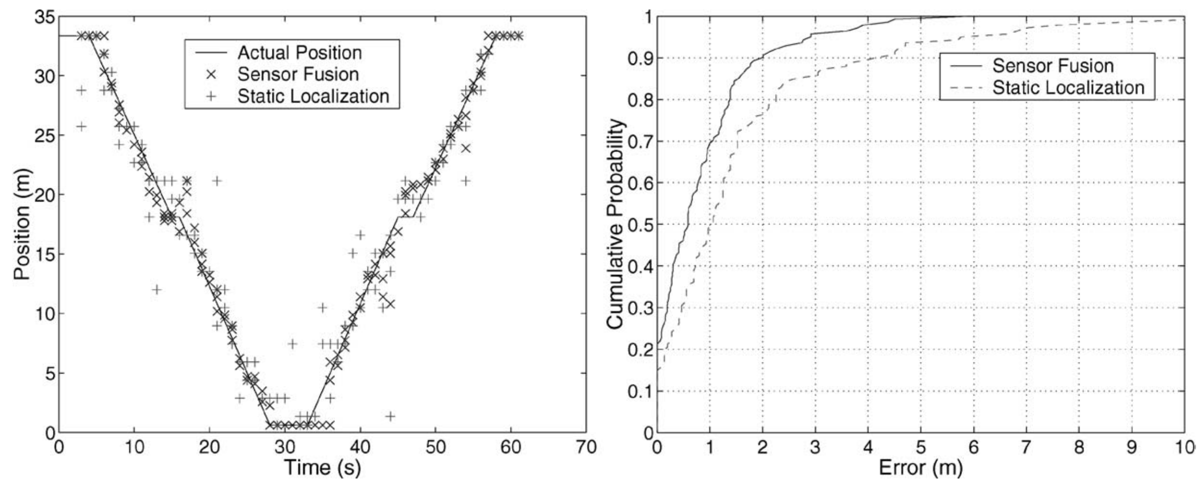

Figure 7. Tracking a round-trip walk of hallway 2 in our test area (see figure 1, the building map). Measured error for the track, shown on the right graph, is at most one meter with probability 0.7, an improvement of 40% over static localization. This improvement is illustrated in the actual tracking performance, shown in the left graph.

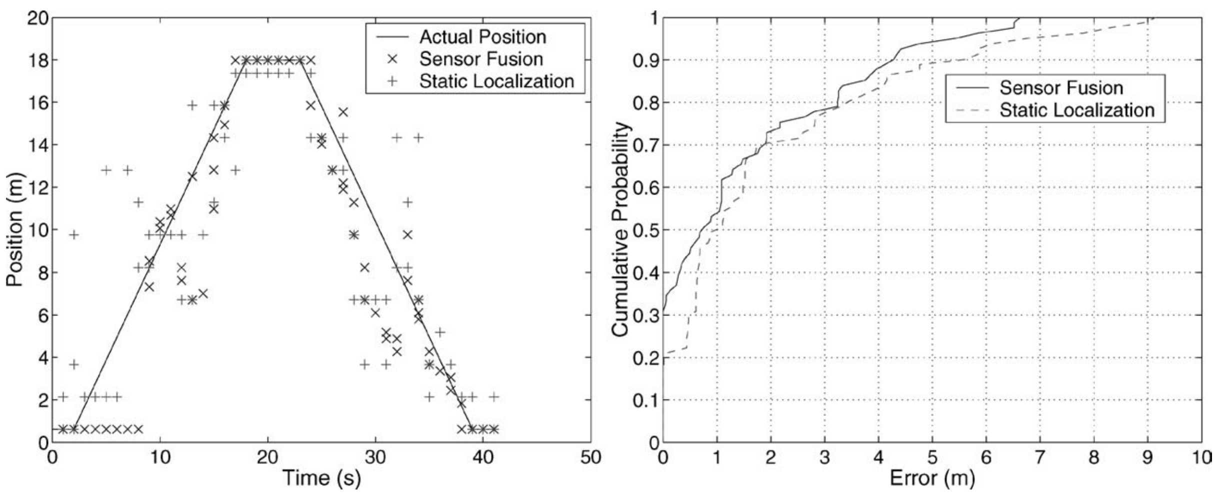

Figure 8. Tracking a round-trip walk of hallway 4 in our test area (see figure 1, the building map). While sensor fusion provided some improvement, it was not significant. As shown in the left graph, when static localization was significantly off, so was sensor fusion, but when static localization appears to track actual movement, sensor fusion is surprisingly accurate despite the noise.

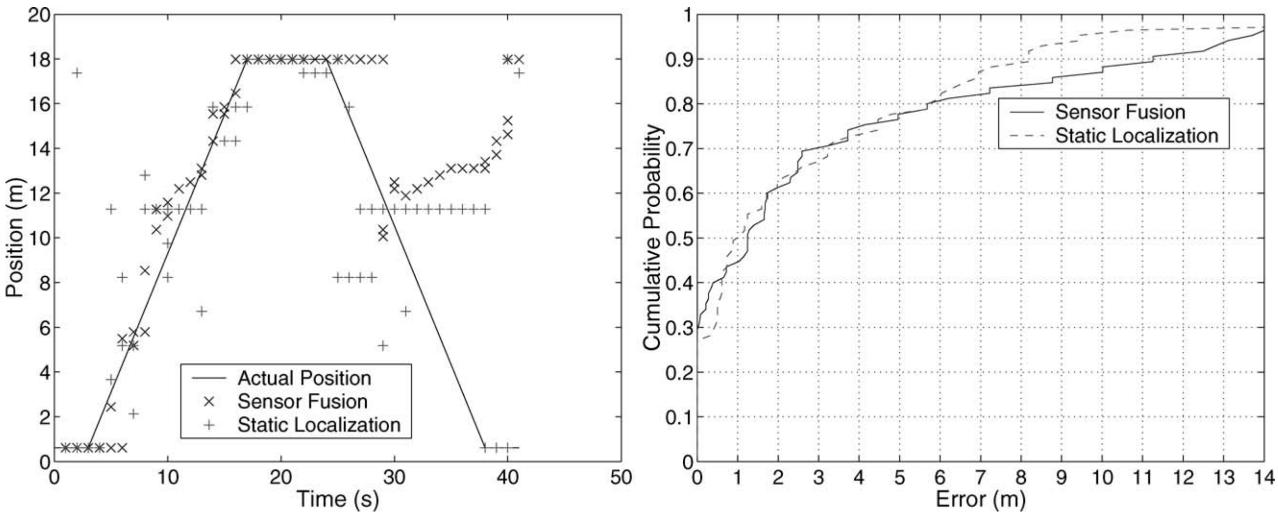

Figure 9. Tracking a round-trip walk of hallway 3 in our test area (see figure 1, the building map). Sensor fusion did not provide a significant improvement in error, and at times increased error, as shown in the right graph. However, as shown in the left graph, the raw data was already extremely noisy in this case.

4. Discussion

The probabilistic robotics-based location-support method with RF-signals that has been described in this article efficiently reports and tracks the two dimensional position and orientation of a mobile wireless device in an indoor environment. While this is not the first application of probabilistic techniques to the field of location-aware computing, it is one of the first application of such techniques for wireless computing in an indoor environment with commodity hardware. This and the rigorous application of state-of-the-art techniques borrowed from robot localization are the main contributions of this article. Our work provides a strong indication that localization can be achieved with widely available and inexpensive 802.11b wireless Ethernet hardware. This section will discuss some advantages and disadvantages of our techniques.

4.1. Advantages

Accuracy. The accuracy of RF based localization is substantially improved in our experimental setup over the reported resolution and accuracy of similar previous efforts. RADAR [2,3] exhibited a median resolution in the range of 2–3 meters. Our results indicate that we can get a resolution of less than 1.5 meters with an accuracy 83% given suitable base station layout. At a coarse resolution, we are very reliable. This is because noise texture varies significantly over relatively large distances, especially when there are intervening obstacles. Inside a room, there are ambiguities in sensing that lead to error. In all of our experiments, we never observed coarse granularity errors except at corners and doorways where the operator is transitioning from one area to another. Our sensor fusion can improve precision while tracking a moving object by interpolating between sampled points and taking advantage of spatial continuity assumptions to probabilistically reject outliers.

Real-time results. Both the static localization and the tracking systems provide real-time results. This means our system is useful in cases where timing is essential, such as providing accurate directions, locating a malicious user, or activating local resources. Some tracking systems [2] require a delay of several seconds to generate a reading.

Orientation. Our method explicitly tries to solve for orientation. This is necessary since as we and others [2,3] have observed orientation is a factor in observed signal strength. In fact, our experiments show that orientation can be coarsely determined by signal strength variations which shows the correlation is often highly non-trivial. By explicitly modeling position and direction, we greatly improve static localization and sensor fusion although orientation determination tended to be much noisier than position. This allows us to overcome difficulties that weakened the applicability of the results of RADAR. However, the variations in signal due to orientation are not sufficiently large to ever obtain more than a coarse estimate of direction.

Cost and complexity. The advantage of using wireless Ethernet RF signals for localization is that the sensor doubles as a communication device. The infrastructure for such networks already exists in many real-world environments and consequently, for many mobile devices, this sort of localization can be implemented as a software-only solution. This is an attractive option for a number of real-world applications.

Extensibility and scalability. The methods we use are very general and experiments with a variety of robot localization applications have proven the approach very adaptable. In particular, the framework can be used with other sensors. For example, by using ultrasound sensors such as those used in Cricket [39], we estimate that we could increase our precision to the order of twenty centimeters. This increase in precision is alluded to by the authors of Cricket as a point of future work when they suggest employing Kalman filters [39,40].

We believe that localization with wireless Ethernet signal strengths scales well into much larger arenas than our experimental test-bed with the caveat that the layout of base stations should be non-pathological. Our evidence for this comes from robot localization and the experimental observation that, at room granularity, signal strength distributions differ greatly.

The particular algorithms we present do not scale if used verbatim. The computational cost of localization in the algorithms we present grows as a linear (Bayesian) or quadratic (sensor fusion) function of the number of possible poses. The vectors and matrices involved however are almost always very sparse. The typical approach in larger cases is to proceed by Monte Carlo (MC) integration of the conditional probability distributions [48]. The computational efficiency of MC is validated by the successful implementation of these algorithms for mobile robots with severely restricted computational power such as the Sony AIBO robot [36]. Finally, the set of visible base stations can provide a coarse estimate of location, reducing the search space.

4.2. Disadvantages

Environment dependence. Every localization system is hampered by a dependence on the environment it is executed in. In our case, we noticed that some of the areas we tested, notably hallway 3, provided lower accuracy than other areas. The placement of the base stations, the materials in the building, and the building's geometry can affect the difficulty of localizing at a given point. A more worrisome challenge is the variation induced by people absorbing RF signals and other dynamic effects. When working with 2.4 GHz RF signals both static and dynamic environmental conditions can be difficult to predict and have complex behaviors. We believe that continued research on heuristics for coping with these problems either by judicious placement of base stations or by improvements in the localization algorithm can produce usable results for many applications even in the face of such environmental flux.

Training. The complexity of indoor RF signal propagation is avoided by building a sensor map. The time spent training these maps is a limitation of all localization approaches using a sampling technique for generating maps. As it is, maps were built by marking the workspace and taking measurements at each point. Further automation might be necessary to facilitate deployment of an approach in this spirit. In mobile robotics, map building and exploration for such localization approaches is an important area of research. By augmenting the operator with some extra sensors, for example an accelerometer and magnetic compass to use for dead reckoning, a walk around the building could be used together with a mapping algorithm [47] to automate training further.

Privacy and security. It has been claimed in previous works, such as Cricket [39,40], that a location support system can be implemented in such a way as to localize a user only if she is willing to be localized. This assertion, though, breaks down if the mobile device is not passive, for example, if it is using an active localization scheme or is using wireless networking to communicate. This raises issues of anonymity, privacy, and security. Third-party observers using conventional hardware could conceivably determine the position of a mobile device broadcasting on a wireless Ethernet network without the device's knowledge or permission. Likewise, a network administrator could use the network to track users by having the base stations monitor observed signal strengths.

4.3. Future work

This work can be extended in a number of different directions. Most directly, we could expand the experimental area, possibly considering multiple floors and significant amounts of area within rooms. There are also a number of algorithmic aspects of mobile node location tracking that could be explored. Finally, we should apply recent results from robotics research, such as adaptive Monte Carlo localization [19].

Compensating for dynamic occlusion in robotics localization is a studied problem but is also quite difficult. Many approaches try to predict some variables describing dynamic state. For example, a tour-guide museum robot needs to model the motion of people in the museum to avoid collisions [4]. Multi-robot, collaborative localization is another branch of localization research [16]. Much of the work in this area is relevant to collaborative localization in an ad hoc wireless network. This is a fascinating problem which mixes issues in protocol design and communication with uncertainty and localization. Relative and differential techniques may be of use in combating variations that occur due to environmental effects. For example, landmark based navigation operates using only the angle of deflection to some feature of the environment [1]. This technique could be applied to mobile devices using directional antennae to detect the angle of deflection to base stations. The pursuit-evasion problem in robotics is the problem of capturing an active evader under various sensing and environmental constraints. In location-aware security for wireless networks, studying how to intercept a moving intruder under various assumptions about sensing could be an interesting and challenging application of this problem.

Another area of robotics where this work might be useful is SLAM (Simultaneous Localization and Mapping) [7,33]. This technique involves using short range sensors to build a small part of a map while global sensors are used to place that small part in the larger context. Since wireless interfaces are prevalent on mobile robots, signal strength would be an attractive candidate for the global sensor.

5. Related work

5.1. Location aware computing

Many other systems have been built to support indoor localization. These systems vary in many parameters, such as the sensors, the cost, the required hardware, the infrastructure and the resolution in time and space [25].

The AT&T Cambridge Laboratory's Active Badge location system [50] and the more recent Active Bat system [51] are two of the first systems in the field. Active Badge uses diffuse IR technology while Active Bat uses an ultrasound time-of-flight technique to provide accurate physical positioning. Users and objects have to carry Active Bat tags, emitting an ultrasonic pulse to a grid of ceiling-mounted receivers and a simultaneous "reset" signal over a radio link. Each ceiling sensor measures the time interval from reset to ultrasonic pulse arrival and computes its distance from the Bat.

The Cricket Location Support System [39,40] also uses ultrasound emitters and embeds low-cost receivers in the object being located. Cricket uses additional radio frequency signals to synchronize time measurements and to distinguish ultrasound signals that are a result of multi-path effects. The main localization techniques that are employed in Cricket are based on triangulation relative to the beacons. Cricket trades accuracy for simpler hardware and infrastructure. It does not require a grid of ceiling sensors with fixed locations as in the Active Bat system but returns an estimation of the user's position with a possible error of a four foot by four foot region, while the Active Bat has an accuracy of nine centimeters. Both of these systems provide excellent localization primitives by employing specialized hardware.

Computer vision has also been used in location support systems. Microsoft Research's Easy Living uses stereo-vision cameras to measure three-dimensional position in a home environment [29]. Camera-based approaches are expensive in terms of hardware infrastructure due to the cost of the camera and the computational overhead of image processing.

RF-based systems. The RADAR system [2,3] uses only a wireless networking signal, employing nearest neighbor heuristics and other pattern recognition techniques for localization. The authors report localization accuracy of about 3 meters of their actual position with about fifty percent probability. They also discuss the problems of localizing in the face of multiple floors and changing environmental conditions, as well as tracking of moving users. While our work has similar design goals to RADAR, we take a very different algorithmic approach, using a probabilistic technique popular in many robotics applications.

The PinPoint location system [52] is similar to RADAR, but uses expensive, proprietary base station and tag hardware to measure radio time of flight. PinPoint's accuracy is roughly 1–3 meters. In the SpotOn system [26], special tags use radio signal attenuation to estimate distance between tags. The aim in SpotOn is to localize wireless devices relative to one another, rather than to fixed base stations, allowing for *ad-hoc localization*. Ad-hoc localization is also needed to provide localization for sensor networks [12,43]. The probabilistic framework we are proposing could also be applied in the case of ad-hoc location sensing.

A number of systems have been built using probabilistic techniques to determine location based on RF signal strength for cellular telephone systems. Liu et al. [34] use Markov modeling and Kalman filtering to predict when a mobile node will cross cell boundaries. Yamamoto et al. [53] use Bayesian analysis to determine the absolute location of a mobile node.

Since the paper was submitted and accepted a number of new papers on wireless location-sensing have been published. Roos et al. [42] implemented a similar system and got similar localization results. They are also the first to compare taking a Gaussian fit of signal strength to using the full histogram of signal strength, although they came to no definite conclusion on this. Nibble [5] is a system that uses a Bayesian network to estimate a device's location. Tao et al. [45] explored variations in hardware and transmission power, and addressed the symmetry of localizing a laptop by measuring the signal intensity of packets transmitted from a mobile device as received by a base station versus packets transmitted by a base station and received by the device. Clustering techniques have also been applied to the problem of location determination [54]. Krumm and Platt [30] introduced a number of techniques for simplifying the process of training a location-sensing system, including localizing based on topological regions (e.g., rooms) rather than grid coordinates. Finally, Haeberlen et al. [20] deployed within an entire office building a wireless location-sensing system which required little training and could tolerate variations in sensing hardware and changes in the environment.

RF signal attenuation. Much effort has been made to model radio signal propagation in an indoor environment [21,37]. Different experiments in the literature have arrived at different distributions. Although each result may be justifiable for a certain set of conditions that govern a certain set of measurements, a consistent model that would give a signal strength distribution under a diversified set of conditions is unavailable. However, experiments with 12000 impulse response profiles in two office buildings have shown good log-normal fit [23]. A general empirical model [21] for indoor propagation of radio signals can be expressed as

$$PL(d) = PL(d_0) + 10n \log\left(\frac{d}{d_0}\right) + X_\sigma,$$

where $PL(d)$ is the path loss in dB at distance d, $PL(d_0)$ is the known path loss at the reference distance d_0, n denotes the exponent depending on the propagation environment and X_σ is the variable representing uncertainty of the model. We note that decibels are a log-scale.

Based on this general formulation, many empirical models have been derived in the field of indoor propagation modeling in the wireless community. Parameter n is very sensitive to the propagation environment, like the type of the construction material and type of the interior [37], limiting the value of these models.

5.2. Robot localization

Robot localization is a well-studied problem in robotics. Robot localization is the process of maintaining an ongoing estimate of a robot's location with respect to its environment, given a representation of this environment and some sensing ability within the environment. The importance of this problem in the context of building reliable robot systems cannot be overstated; determining the pose (position and orientation) of the robot from physical sensors is often referred to as "the most fundamental problem to providing a mobile robot with autonomous capabilities" [9]. In our case, we can consider any wireless device as a mobile robot.

If there is no a priori estimate of the robot's location, the problem is referred to as *global localization*, which is a particularly challenging case of localization. This is the type of problem we discussed. We have no information where the wireless device is before it starts communicating with the network's base stations. Furthermore, there is the need to refine the estimate of the device's pose continuously. This task is known as *pose maintenance*.

Sensor-based localization is based on the premise that we use sensor data in conjunction with the representation of the environment to produce a refined position estimate, such that this estimate is more likely to predict the true positions. By sensor, we mean any device which can measure attributes of the environment in a way that can be correlated to position. Typical sensors that are deployed in robotics are IR transmitters, ultrasound or laser proximity sensors and cameras.

Sensor fusion is another important notion in robot localization. A broad definition of sensor fusion is the combination of multiple independent observations to obtain a more robust and precise estimate of the measured variables. This can be implemented in terms of integrating sensor readings over time or in the synthesis of measurements from multiple sensors. Most of the recent work in robot localization has been in improving and implementing sensor fusion for many systems.

Much progress has been made in developing localization techniques since the problem first appeared in the literature. Dead reckoning can be used for pose maintenance, but requires some initial knowledge of location. Some of the simplest methods for global localization include landmark-based localization and triangulation. Probabilistic techniques, such as Kalman filtering, and later, Bayesian analysis, were developed to address flaws in these systems. Finally, for when a grid-based map is inappropriate to the application or environment, topological approaches have been developed.

Dead reckoning. Perhaps the simplest approach to the pose maintenance task is to keep track of how far the robot moves in each direction and then to sum these motions to produce a net displacement that can be added to an initial position estimate. Keeping track of how much one moves by observing internal parameters without reference to the external world is known as *dead reckoning* and is usually implemented with an odometer. However, with each odometer reading, some error is added to the absolute pose estimate. If only dead reckoning is used for position estimation, these errors are accumulated. Long-term localization must make reference to the external world for position correction. This involves the use of sensory data for recalibrating a robot's sense of its own location with the environment. In some circumstances, such as the case of a wireless device that a person is moving around in space, we have no analogue of odometry.

Triangulation. Distance to known landmarks is frequently used to determine pose as this can be computed with cameras, laser range-finders, IR transmitters, sonar and other commonly used sensors. A naïve approach is to take three distance measurements and triangulate position. This works when the sensors are reliable and relatively noise-free but leaves several problems unaddressed. When the sensors are noisy, the calculations for triangulation become unstable for many positions and landmark arrangements and lead to significant loss of precision. Typically, multiple measurements are merged over time to try to compensate for this, however, some care must be taken in choosing the method of merging or poor results will be obtained [13]. In some cases where the sensors are fairly reliable and have simple noise distributions, direct triangulation or triangulation with differential windowing can produce excellent results. Noisy sensors, however, complicate triangulation adding uncertainty to the results. GPS [35] is one of the best-known and most used triangulation-based sensors.

Kalman filter. In 1987, Smith and Cheeseman introduced the use of Kalman filters to the problem of determining position [44]. Many systems in robot localization, since then, have been based on Kalman filtering [11,18,32]. The robot's pose estimation is maintained as a Gaussian distribution in $\mathbb{R}^2 \times S$ and sensor data from dead reckoning and landmark observations is fused to obtain a new position distribution. This method is provably optimal when all distributions are Gaussian but typically fails when this assumption breaks down. Extended Kalman filters address this problem by linearizing the system. In practice, obtaining linearizations for many sensing systems is difficult and errors can propagate very quickly through the system.

Bayesian approaches. Possibly the most powerful family of global localization algorithms to date is based on Bayesian inference, in particular Markov localization [17,28] and Monte Carlo localization [15,48]. These are generalizations of the Kalman filter. These algorithms estimate posterior distributions over robot poses which are approximated by piecewise constant functions instead of Gaussians, enabling them to represent highly multi-modal distributions. In this way, they can be applied in the case of sensors that have non-Gaussian noise distributions. The accuracy of the results, however, is limited by the resolution of the approximation. Due to the very complex nature of some sensors and usually also of the environment, many systems have difficulties modeling outliers and other artifacts. These difficulties can be addressed by sampling the distributions of the sensor signals in the target environment and using this directly as a model, as in the case of the sensor map we built in the first phase of our method. By explicitly integrating the conditional probability distributions, we can obtain precise approximations of the robot's positional distribution. This approach is both computationally tractable and effective [48]. Many excellent examples of this method exist in the literature [46]. This is the approach we took in implementing localization using wireless Ethernet, as described in section 2. Recently, however, adaptive Monte Carlo localization [19] promises to improve the accuracy and stability of localization results.

Topological approaches. Typically the Bayesian approach is applied in the case when we have a grid-based representation of the environment. Another alternative for modeling the environment is with a topological map, represented as a generalized Voronoi graph [7]. Localization on the topological map is based on the fact that the robot automatically identifies nodes in the graph from geometric environmental information [31].

6. Conclusions

In this article, we provide strong evidence that reliable localization with wireless Ethernet can be achieved. In our experiments, we can measure and track position robustly with the first meter of error distributed within a standard deviation. We used the Intersil Prism2 chipset for our wireless Ethernet cards and Apple AirPort base stations, both readily available and inexpensive hardware. The building we operated in had fairly complicated geometry and the base stations were laid out more than a year before we began our work. The methods we employed were general methods from robotics and followed the Bayesian approach to localization. These methods were readily adaptable to the problem at hand and can be applied to other location problems that might arise in mobile computing.

Acknowledgements

The authors would like to thank Guillaume Marceau for his valuable suggestions and help in this project. They would also like to thank Moez Abdel-Gawad and Skye Schell for helping to take measurements. Moreover, the authors thank Dave Johnson for his advice and comments. Thanks also to Scott Crosby and to the anonymous MOBICOM'02 and IROS'02 reviewers for their comments.

Andrew Ladd is partially supported by FCAR and NSF-IRI-970228. Kostas Bekris is partially supported by NSF-IRI-970228. Algis Rudys and Dan Wallach are supported by NSF-CCR-9985332, a Texas ATP grant, and a gift from Schlumberger. Lydia Kavraki is supported by NSF Career Award IRI-970228, a Whitaker grant, a Texas ATP award, and a Sloan Fellowship.

References

[1] A.A. Argyros, K.E. Bekris and S.C. Orphanoudakis, Robot homing based on corner tracking in a sequence of a panoramic images, in: *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Vol. 2, Kauai, HI (December 2001) pp. 3–10.

[2] P. Bahl and V.N. Padmanabhan, Enhancements to the RADAR user location and tracking system. Technical Report MSR-TR-2000-12, Microsoft Research (February 2000).

[3] P. Bahl and V.N. Padmanabhan, RADAR: An in-building RF-based user location and tracking system, in: *Proc. of IEEE Infocom 2000*, Vol. 2, Tel Aviv, Israel (March 2000) pp. 775–784.

[4] W. Burgard, A. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner and S. Thrun, The interactive museum tour-guide robot, in: *Proc. of the 15th National Conference on Artificial Intelligence (AAAI-98)*, Madison, WI (July 1998), pp. 11–18. Outstanding paper award.

[5] P. Castro, P. Chiu, T. Kremenek and R.R. Muntz, A probabilistic room location service for wireless networked environments, in: *Proc. of the 3rd International Conference on Ubiquitous Computing (Ubicomp)*, Atlanta, GA (September 2001) pp. 18–24.

[6] A. Chakraborty, A distributed architecture for mobile, location-dependent applications, Master's thesis, Massachusetts Institute of Technology (May 2000).

[7] H. Choset and K. Nagatani, Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization, IEEE Transactions on Robotics and Automation 17(2) (2001) 125–137.

[8] T.W. Christ and P.A. Godwin, A prison guard duress alarm location system, in: *Proc. IEEE International Carnahan Conference on Security Technology* (October 1993) pp. 106–116.

[9] I. Cox, Blanche – an experiment in guidance and navigation of an autonomous robot vehicle, IEEE Transactions on Robotics and Automation 7(2) (1991) 193–204.

[10] T. Cutler, *Wireless Ethernet and How to Use It*, The Online Industrial Ethernet Book, Vol. 5 (1999).

[11] A.J. Davison and N. Kita, 3D simultaneous localization and map-building using active vision for a robot moving on undulating terrain, in: *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, Vol. 1, Kauai, HI (December 2001) pp. 384–391.

[12] L. Doherty, K.S.J. Pister and L.E. Ghaoui, Convex position estimation in wireless sensor networks, in: *Proc. of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies*, Vol. 3, Anchorage, AK (April 2001) pp. 1655–1663.

[13] G. Dudek and M. Jenkins, *Computational Principles of Mobile Robotics* (Cambridge University Press, 2000).

[14] Federal Communications Commission Report and Order 96-264: Revision of the commission's rules to ensure compatibility with Enhanced 911 emergency calling systems (July 1996), `http://www.fcc.gov/Bureaus/Wireless/Orders/1996/fcc96264.txt`

[15] D. Fox, W. Burgard, F. Dellaert and S. Thrun, Monte Carlo localization: Efficient position estimation for mobile robots, in: *Proc. of the 16th National Conference on Artificial Intelligence (AAAI-99)*, Orlando, FL (1999) pp. 343–349.

[16] D. Fox, W. Burgard, H. Kruppa and S. Thrun, A probabilistic approach to collaborative multi-robot localization, Autonomous Robots 8(3) (2000) 325–344.

[17] D. Fox, W. Burgard and S. Thrun, Markov localization for mobile robots in dynamic environments, Journal of Artificial Intelligence Research (JAIR) 11 (November 1999) 391–427.

[18] J. Guivant and E. Nebot, Optimization of the simultaneous localization and map building algorithm for real time implementation, Journal of Robotics Research 17(10) (2000) 565–583.

[19] J.-S. Gutmann and D. Fox, An experimental comparison of localization methods continued, in: *Proc. of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 1, Lausanne, Switzerland (September 2002) pp. 454–459.

[20] A. Haeberlen, E. Flannery, A.M. Ladd, A. Rudys, D.S. Wallach and L.E. Kavraki, Practical robust location over large-scale 802.11 wireless networks, in: *Proc. of the 10th ACM International Conference on Mobile Computing and Networking (MOBICOM)*, Philadelphia, PA (September 2004) pp. 70–84.

[21] P. Harley, Short distance attenuation measurements at 900 MHz and 1.8 GHz using low antenna heights for microcells, IEEE Journal on Selected Areas in Communications (JSAC) 7(1) (1989) 5–11.

[22] A. Harter, A. Hopper, P. Steggles, A. Ward and P. Webster, The anatomy of a context-aware application, in: *Proc. of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 1999)*, Seattle, WA, (August 1999) pp. 59–68.

[23] H. Hashemi, Impulse response modeling of indoor radio propagation channels, IEEE Journal on Selected Areas in Communications (JSAC) 11 (1993) 967–978.

[24] H. Hashemi, The indoor radio propagation channel, Proceedings of the IEEE 81(7) (1993) 943–968.

[25] J. Hightower and G. Borriello, Location systems for ubiquitous computing, IEEE Computer 34(8) (2001) 57–66.

[26] J. Hightower, R. Want and G. Borriello, SpotON: An indoor 3D location sensing technology based on RF signal strength, Technical Report UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA (February 2000).

[27] Institute of Electrical and Electronics Engineers, Inc., ANSI/IEEE Standard 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (1999).

[28] K. Konolige and K. Chou, Markov localization using correlation, in: *Proc. of the 17th International Joint Conference on Artificial Intelligence (IJCAI)*, Seattle, WA (August 1999) pp. 1154–1159.

[29] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale and S. Shafer, Multi-camera multi-person tracking for EasyLiving, in: *Proc. of the 3rd IEEE International Workshop on Visual Surveillance*, Dublin (July 2000) pp. 3–10.

[30] J. Krumm and J. Platt, Minimizing calibration effort for an indoor 802.11 device location measurement system, Technical Report MSR-TR-2003-82, Microsoft Research, Seattle, WA (November 2003).

[31] B. Kuipers and Y.T. Byan, A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations, Journal on Robotics and Automatic Systems 8 (1991) 47–63.

[32] J.J. Leonard and H.F. Durrant-Whyte, Mobile robot localization by tracking geometric beacons, IEEE Transactions on Robotics and Automation 7(3) (1991) 376–382.

[33] J.J. Leonard and H.F. Durrant-Whyte, Simultaneous map building and localization for an autonomous mobile robot, in: *Proc. of the IEEE/RSJ International Workshop on Intelligent Robots and Systems (IROS)*, Vol. 3, Osaka, Japan (November 1991) pp. 1442–1447.

[34] T. Liu, P. Bahl and I. Chlamtac, Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks, IEEE Journal on Selected Areas in Communications 16(6) (1998) 922–936.

[35] T. Logsdon, *Understanding the Navstar: GPS, GIS and IVHS*, 2nd ed. (Van Nostrand-Reinhold, New York, 1995).

[36] G. Marceau, The McGill's RedDogs legged league system, in: *Proc. of Robocup 2000* (2000) pp. 627–630.

[37] A. Neskovic, N. Neskovic and G. Paunovic, Modern approaches in modeling of mobile radio systems propagation environment, IEEE Communications Surveys and Tutorials 3(3) (3rd Quarter 2000) pp. 2–12.

[38] A.B. Poritz, Hidden Markov models: a guided tour, in: *Proc. of the International Conference on Acoustics, Speech, and Signal Processing*, Vol. 1, New York (April 1988) pp. 7–13.

[39] N. Priyantha, A. Chakraborty and H. Balakrishman, The Cricket location support system, in: *Proc. of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 2000)*, Boston, MA (August 2000) pp. 32–43.

[40] N. Priyantha, A. Miu, H. Balakrishman and S. Teller, The Cricket compass for context-aware mobile applications, in: *Proc. of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM 2001)*, Rome, Italy (July 2001) pp. 1–14.

[41] L.R. Rabiner and B.-H. Juang, An introduction to hidden Markov models, IEEE ASSP Magazine 3(1) (1986) 4–16.

[42] T. Roos, P. Myllymaki, H. Tirri, P. Misikangas and J. Sievanan, A probabilistic approach to WLAN user location estimation, International Journal of Wireless Information Networks 9(3) (2002) 155–164.

[43] A. Savvides, C.-C. Han and M.B. Strivastava, Dynamic fine-grained localization in ad-hoc networks of sensors, in: *Proc. of the 7th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Rome, Italy (July 2001) pp. 166–179.

[44] R. Smith and P. Cheeseman, On the representation of spatial uncertainty, Journal of Robotics Research 5(4) (1987) 56–68.

[45] P. Tao, A. Rudys, A.M. Ladd and D.S. Wallach, Wireless LAN location-sensing for security applications, in: *Proc. of the ACM Workshop on Wireless Security*, San Diego, CA (September 2003) pp. 11–20.

[46] S. Thrun, Probabilistic algorithms in robotics, AI Magazine 21(4) (2000) 93–109.

[47] S. Thrun, W. Burgard and D. Fox, A probabilistic approach to concurrent mapping and localization for mobile robots, Machine Learning 31(1–3) (1998) 29–53.

[48] S. Thrun, D. Fox, W. Burgard and F. Dellaert, Robust Monte Carlo localization for mobile robots, Artificial Intelligence 101 (2000) 99–141.

[49] F. van Diggelen and C. Abraham, Indoor GPS technology, in: *Proc. of CTIA Wireless-Agenda*, Dallas, TX (May 2001).

[50] R. Want, A. Hopper, V. Falco and J. Gibbons, The Active Badge location system, ACM Transactions on Information Systems 10 (1992) 91–102.

[51] A. Ward, A. Jones and A. Hopper, A new location technique for the active office, IEEE Personal Communications 4(5) (1997) 42–47.

[52] J. Werb and C. Lanzl, Designing a positioning system for finding things and people indoors, IEEE Spectrum 35(9) (1998) 71–78.

[53] R. Yamamoto, H. Matsutani, H. Matsuki, T. Oono and H. Ohtsuka, Position location technologies using signal strength in cellular systems, in: *Proc. of the 53rd IEEE Vehicular Technology Conference*, Vol. 4 (May 2001) pp. 2570–2574.

[54] M. Youseff, A. Agrawala and A.U. Shankar, WLAN location determination via clustering and probability distributions, in: *Proc. of the IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Forth Worth, TX (March 2003) pp. 143–150.

Andrew Ladd is a Ph.D. candidate in computer science at Rice University. He is studying algorithmic robotics under Lydia Kavraki. His research interests include motion planning with dynamics, fast computational geometry algorithms, uncertainty models for robotics and approximate space representation. He received his B.S. in joint honors mathematics and computer science at McGill University in 2000 and an M.S. in computer science from Rice in 2003.
E-mail: aladd@cs.rice.edu

Kostas Bekris received the B.S. degree from the Computer Science Department in the University of Crete, Greece in 2001. He is currently a Ph.D. student in the Computer Science Department in Rice University, Houston, TX. His research interests include motion planning, mobile robots and sensor planning.
E-mail: bekris@cs.rice.edu

Algis Rudys is a graduate student at Rice University pursuing a Ph.D. in computer science. He is studying computer security under Dan S. Wallach. Algis primary research area is mobile code security. He received a B.S. in computer science in 2000, and a M.S. in computer science 2002, both at Rice University.
E-mail: arudys@cs.rice.edu

Lydia Kavraki is an Associate Professor of Computer Science and Bioengineering at Rice University. She completed her B.S. degree in computer science in Greece and obtained her M.S. and Ph.D. degrees in computer science from Stanford University. Kavraki's research interests are in physical algorithms and their applications in robotics (motion planning, assembly planning, flexible object manipulation) and bioinformatics (modeling of receptor-ligand interactions, computer-assisted drug design, and proteomics). Kavraki received the National Science Foundation CAREER award in 1997, a Sloan Fellowship in 2000 and the Association for Computing Machinery Grace Murray Hopper Award in 2000. She has also received the Early Academic Career Award from the IEEE Society on Robotics and Automation. Kavraki is the editor of the book "Robotics: The Algorithmic Perspective" and has authored more than sixty scientific papers published in refereed journals and conferences. She has served in the program committees of several conferences (RECOMB, ICRA, IROS, IJCAI, AAAI, WAER, SoCG), co-chaired the 1998 International Workshop on Algorithmic Foundations of Robotics and is an Associate Editor for the IEEE Robotics Magazine. Kavraki was included in the list of Top 100 Young Innovators of the MIT Technology Review Magazine in 2002 and in the list of Top 10 Young Scientists by the Popular Science Magazine in 2002.
E-mail: kavraki@cs.rice.edu
WWW: http://www.cs.rice.edu/ kavraki

Dan S. Wallach is an Assistant Professor of computer science at Rice University. Wallach studies security mechanisms for controlling malicious mobile code; he helped design Netscape's stack inspection architecture, now standard in systems from Sun, Microsoft, and other vendors. More recently, he has been studying security issues that occur in peer-to-peer networking systems, which need to be robust, even when some nodes in the network might want to take advantage of the network, getting more resources out than they put in. He also has an interest in finding all the holes in systems that seem inherently doomed to failure, including music copy protection systems and paperless electronic voting systems.
E-mail: dwallach@cs.rice.edu