Position determination using multiple wireless interfaces

DANIEL HASSELLÖF

KTH Information and
Communication Technology

Master of Science Thesis
Stockholm, Sweden 2008

COS/CCS 2008-08

Position determination using multiple wireless interfaces

DANIEL HASSELLÖF

Master's Thesis
Examiner: Gerald Maguire

28 April 2008

This thesis is dedicated to my parents, whose unconditional support made it possible.

Abstract

This Master's thesis studies different ways of exploiting the signal strength measurements from wireless interfaces for position determination. Difficulties include handling the fluctuating observations and their sensitivity to obstruction. We list important factors to take into account before describing a new system based on location fingerprinting and capable of integrating observations from multiple wireless interfaces.

Compared to typical fingerprinting solutions, the training time is an order of magnitude shorter, but the location resolution is limited to locations of particular interest. In an office environment, the proposed solution determines the location correctly 80 percent of the time with sufficient precision for being used with context-aware services. In an open space environment, an incorrect location is reported 42 percent of the time.

Sammanfattning

Det här exjobbet studerar olika sätt att använda signalstyrka från trådlösa gränssnitt för positionsbestämning. Några av svårigheterna ligger i att hantera observationernas fluktuationer och deras känslighet för obstruktion. De viktigaste faktorerna att ta hänsyn till tas upp innan ett nytt system beskrivs. Det är baserat på positionsigenkänning (location fingerprinting) och kan dra nytta av observationer från flera olika trådlösa gränssnitt.

Jämfört med vanliga metoder för positionsigenkänning är träningstiden en storleksordning kortare, men positionsupplösningen är begränsad till ett visst antal positioner av särskilt värde. I en kontorsmiljö klarar den föreslagna lösningen att korrekt bestämma positionen i 80 procent av fallen med tillräckligt hög noggrannhet för att användas till kontextmedvetna tjänster (context-aware services). I en öppen rumslösning ger lösningen en felaktig position i 42 procent av fallen.

Acknowledgments

I would like to thank Ambient Services group leader F. Bataille and his team for giving me the opportunity to carry out this project at Alcatel Research and Innovation in Paris, France.

For providing constructive feedback throughout the different parts of the project, I would like to thank the extremely competent Professor G. Q. Maguire Jr, my academic supervisor and examiner at KTH.

Daniel Hassellöf
Stockholm, 4 December 2006

Contents

Chapter 1

Introduction

1.1 Goal

The objective of this thesis is to study how the wireless interfaces of any mobile device can be used in order to determine its location. The paper deals with how multiple wireless interfaces can be integrated in order to unambiguously provide better location information.

The analysis of different potential solutions was expected to result in a recommendation on which approach is the most suitable, considering the given constraints. In addition, a proof of concept was to be developed, to provide a prototype, proof of concept implementation.

1.2 Scope

The scope of this thesis is limited to how the signal strength measurements of wireless interfaces can be exploited for position determination. This includes an entire framework, with its core components being the treatment of the measurements and the algorithm resulting in a location reference.

No attention is given to the hardware design of the mobile device or the surrounding infrastructure. Even though the intention is to use the location information for context-aware services, no more than only an overview will be given to this concept (see section 2.2).

1.3 Method overview

One could distinguish four main phases of the project. They should each result in a conclusion, and are hereinafter briefly presented. These are:

1. Studying available radio localization technologies based on Wireless LAN, Bluetooth, RFID, Zigbee, etc. (see chapter 6)

2. Developing a new system capable of integrating all available location information to provide unambiguous location information. (see chapter 10)

3. Evaluate the performance of the implemented system. (see chapter 11)

Chapter 2

Need for location information

2.1 Location information in a wider perspective

Localization technology is something our everyday life depends on. One of the most common applications is navigation, be it on water, in the sky, or on the land. All navigation is based on the objective of orienting the user.

Wireless localization is often seen a specialized case of localization, particularly in robotics. Determining the position of a mobile robot using its various sensors is a well-studied problem, and it has been described as the most fundamental problem of building an autonomous robot [3].

2.2 Location based services

2.2.1 Direct location usage

When studying location based services, often referred to as LBS, one can distinguish two categories, which differs in the way the location information is exploited. In many applications, finding the position of a device is the actual objective. Then the location information is directly used. A common usage is to find something in the proximity of the user, typically a nomadic friend/colleague or a fix point of interest.

2.2.2 Indirect location usage

The location information can also be exploited in a more indirect way by services that are said to be context-aware. The position determination system developed in this thesis will serve this second service type, therefor it is of more interest.

There are several reasons for the existences of context-aware services. One is that the amount of available information has been growing constantly for decades and the same is assumed to be true for the number of communication services. Due to the current limitations of mobile terminals, accessing a service is normally much harder than on a PC. Consequently, every additional click needed to access a service has been estimated to reduce service profit by 50 percent. [16]

The ambition of context aware services is, among other things, to address this dilemma thus enhancing the usefulness of mobile services. This can be achieved by offering the right services at the right time and by adapting them to best suit the personal situation of the user. [16] The selection and adaptation is done with the help of certain context information, which in [17] is defined as *any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.* According to [15] there are three important dimensions of context: *where* you are, *who* you are, and *what* resources are nearby. This thesis is dedicated to the first of these.

2.3 Location based services in practice

This section explains in a more in a more concrete and practical way how wireless position determination technology can be used. The intention is not to give a thorough description of potential applications, but to indicate the need for and benefits of location determination.

The possible uses for localization are uncountable. For example in the context of mobile computing, administrators might want to track laptop users for security reasons and the user could appreciate getting help finding the nearest printer or office. [1]

Hospitals are also looking at creating special wrist tags for particular patients, such as those suffering from Alzheimer's disease. The location service would then monitor the patient and warn if he or she wanders outside a given area. [34]

Another possible application of location services is guided tours. Today, for example in the case of a museum, a common solution is to provide the visitors with a handheld device capable of presenting information related to the piece of art being viewed. Typically, the user must type in an identity code for the correct information to appear. A localization enabled device would permit the position to automatically be determined, allowing the user to walk freely around the museum while getting relevant information presented. (see use case in section 2.3.3)

In addition to facilitate the visitor experience, a location system would allow the museum to collect statistics about visitor behavior. This data could be uploaded when the user hands in their device. A WLAN based location system would allow the location information to be continuously sent to a server. This way the visitors could be recommended a certain tour path in order to avoid congestion. One could also imagine a more advanced service where two-way communication allows the visitor to interact with an extended resource database or even a human guide.

Another solution has been deployed as described in [67], where a tour guide robot takes advantage of localization services for its navigation.

Here follows some use cases to clarify the benefits of position determination:

2.3.1 Use case: Entering new territory

One of the situations when location related information would come in handy is when a mobile device user enters unknown territory, for example a new part of an office building. The following sequence of events could take place.

Assumption: device is wireless enabled and is running location determination software.

1. A location determination server notices the presence of a previously unknown MAC address, belonging to the mobile device.

2. Pushed by the location determination server, the user is invited to download information related to the area.

3. Accepting the proposal, the information is loaded and used to determine his or her position.

4. The user can then take advantage of information helping him to find the place in the building that he or she is interested in visiting. The device could guide the user to various context related information such as nearby printers, meeting rooms and coffee machines.

2.3.2 Use case: Entering and exiting meeting room

1. A user with a location determination enabled device enters a room for a meeting about to start.

2. The room has been defined as a silent zone, and the user has previously accepted to respect silent zones. Therefore the device automatically activates silent mode or call redirection.

3. Exiting the meeting room, the device will automatically revert to its normal behavior.

2.3.3 Use case: Virtual tour guide

1. In a museum a visitor has been given a handheld device to provide information related to the different objects.

2. When approaching an object, the position of this device will be determined and the corresponding information loaded.

3. As soon as the visitor is satisfied he or she can simple leave the spot. This will stop the video or voice from being played and allow more appropriate general information to be presented, such as highlights of the museum or today's recommendation from the local cafe or restaurant.

2.3.4 Use case: Studying customer behavior

This use case is of interest to businesses, but would surely not be appreciated by all mobile device users due to the violation of their privacy.

1. A customer enters a commercial area, for example a supermarket.

2. As soon as his handheld device has been spotted by the wireless infrastructure, it will be tracked when moving around the premises.

3. By analyzing customer behavior one could improve the product organization and the positioning of entrances and exits.

4. When the customer passes the cashier, one could link the path taken to the list of products bought. Combined this information could take the studying of customer behavior even further than it is today.

For this scenario to be possible the customers would either by habit carry around their devices with the wireless interface active. It is possible, not to say probable, that this will be the case in a not so distant future. Otherwise, one could always motivate customers to participate voluntarily in the survey by compensating them or just notifying them about special offers.

2.4 Position determination requirements

What accuracy and level of confidence is needed for the position determination system? This is obviously application dependent.

For a system to be used for context-aware services, the precision needs to be good enough for a given service to trigger the desired event. Consider the use case in section 2.3.2. This probably requires that we must be able to tell when the user is near his or her desk, and when the user is in a meeting room next door, or across the corridor.

For the other applications previously described, the requirements are similar. In other words, it is unlikely that higher precision than 2-3 meters is needed. On the other hand, for people to trust the context-aware service, it is even more important that incorrect locations are not repeatedly determined. A false positive rate of 10-15 percent is probably enough to damage the user's confidence in a service.

How long can the position determination process take before an indication of the user's position must be given? For many applications, the user is not directly concerned with the context-aware services and the location information that it uses. Therefore, a delay of up 15 seconds is acceptable. Some applications may be more time constrained, whereas in others a delay of several minutes will go unnoticed. In all cases, it is probably desirable to delay the result if this gives a more certain location, to maintain the user's trust. A larger set of measurements is likely to result in a more representative result since occasional deceptive observations will then be balanced out. Up to a few hours observation is likely to have an improving effect on the result.

Chapter 3

Criteria for localization technologies

This chapter describes the most important factors to take into account when choosing how to perform position determination. These factors are supposed to cover various aspects, such as the type of wireless interface to use and how to exploit it for position determination.

3.1 Coverage

When deploying a localization system coverage is obviously important for its usability. This coverage can be expressed primarily in terms of two factors:

1. functionality for non line-of-sight localization, and

2. coverage area (generally characterized by a range).

Considering the first of the two factors, some technologies, such as infra-red, imply limitations which can depending on the circumstances greatly reduce its usability. In contrast, radio frequency (RF) based technologies generally functions also when the access point is in non line-of-sight, which is an advantage of wireless LAN and Bluetooth. Nevertheless, obstacles obstruct RF signals and render localization more difficult.

The limitation related to the distance between transmitter and receiver (given an emitted signal power), is applicable to all technologies. In the case of wireless LAN the indoor range can be estimated to 35 meters. The limitation is generally the receiver and not the transmitter, since the transmitter's emitter power is limited by local regulations. In infrastructure mode wireless LANs the receiver characteristics of the mobile terminal will be decisive, assuming the mobile terminal is used to receive signals from multiple access points.

3.2 Privacy

3.2.1 Passive localization

When a device is tracked without actively taking part in the localization process it is called passive localization. The localization process is then performed by the surrounding devices or infrastructure, and the tracked device need not be aware of being tracked. All it must do to be tracked is to transmit. [5]

The obvious application of passive localization is to determine user position(s) within a wireless network. This information could be used to provide high level statistics about user movement. Note, however, that with a similar approach anyone with physical access to the building can deploy an *ad hoc* network of snoopers and effectively locate all mobile agents that reveal their location due to communicating via the wireless network. As shown in [11], attempts to avoid being successfully tracked by modulating transmission power for each packet is not sufficient. However, in order to attain low probability of detection, interception and exploitation (LPD, LPI, LPE) more sophisticated methods must be used [21].

From a client perspective, it is obviously undesirable to be passively localized, because the user's possibility to keep his position private is very limited. Passive localization is generally possible unless (a) the mobile nodes do not ever transmit, which greatly limits their functionality, or (b) the nodes use LPD, LPI and LPE techniques, which is generally not cost effective except in very special situations (such as special operations warfare). Thus to ensure user acceptance it is best that

1. the user can know when they are being tracked,
2. the user have a feeling that they can control the use of this information, and that
3. the user's location information is protected by legal mechanisms - as technical protection mechanisms are generally not viable.

3.2.2 User privacy

There are two sides to location privacy. The user may want to protect his or her location privacy by not revealing his or her position. On the other hand, an administrator may want to track network users to tune network performance or to detect intruders. Sometimes such security precautions are also appreciated by the users, which makes it a paradox.

Either way, Swedish law greatly limits the use of location information. For example, a cellular operator may use this to help tune the network, but can only use it to identify individual locations under a court order, or when the user makes a call to a particular set of numbers (such as emergency number 112).

In order to preserve the user's location privacy, the location determination code should be run on the user's terminal to provide the infrastructure with as little information as possible. [11] The ideal case is that the client is simply a passive listener.

This is the case of GPS. The client simply collects enough information to calculate its position autonomously, i.e. without revealing its location by transmitting.

It must be stated that this is (normally) not the case in wireless networks, since most services depend on two-way communication with other network nodes. Also in the case of localization, with a commercial WLAN interface, it has proved difficult to be a completely passive listener. (as will be described next)

Even while using promiscuous mode it is stated in [1] that the only packets guaranteed to be received are those sent from the access point the network card is currently associated with. In their case, the solution was to broadcast a probe request. Access points reply to these packets with a probe response, allowing the client to collect packets from all access points within range in order to determine its position. Note that probing of access points might also be needed because they do not necessarily broadcast their existence.

However, in the case of an office environment we can expect that there will be significant amount of traffic due to the sharing of access points via a number of users. As it is only necessary that there be traffic to receive in order to conduct measurements, it might be feasible to do client side localization if the client know the location of the access points in advance, simply based upon passive listening. This thesis does not pursue this possibility, but rather examines the case when the mobile device is both sending and receiving traffic.

To sum up, in the context of location privacy the location determination should be performed on the mobile device to the extent possible in order to protect the user's privacy. Note that either way, the access points know which user devices are registered where, hence using SNMP the network operator could already find out roughly where each user is (assuming that the users are not randomizing their MAC addresses).

3.3 Infrastructure

As previously indicated, the intention is to take advantage of the wireless infrastructure providing connectivity to the user. Thus some of the wireless access points will be deployed while keeping in mind the needs of the localization system.

However, in the majority of the cases the infrastructure will already be in place, thus any necessary modifications such as moving or adding an access point or changing the parameters of an access point will imply additional costs. Problems could also be caused by firewalls or restrictive policies imposed by network administrators. [77] In this thesis we will assume that the user does have access to the network, thus we ignore the questions of access control, authentication etc.

In summary, minimizing the constraints imposed on the infrastructure will result in easier deployment of a localization system. Thus, the solution should be as independent as possible of the infrastructure, preferably gathering measurements and performing calculations autonomously to determine the device's location. Although, it is obviously a great advantage if the access point infrastructure is already in place.

3.4 Terminal availability

A prerequisite for this position determination system is that it should be supported by off-the-shelf (OTS) hardware. The intention is that it should be available not only to professionals provided (by their employers) with special terminals, but also on more typical clients.

3.5 Terminal limitation

Mobile devices are constrained due to limited battery power and limited computational resources. In practice, this inhibits our ambition to protect the privacy of the user, as minimizing terminal computations and power consumption causes parts of the location determination process to move to the fixed infrastructure and consequently reveals information about the client.

Notice in particular, as explained in [100], that the 802.11 interface of a handheld device is likely to consume a major part of its battery resources. In an office environment it is reasonable to assume that the wireless interface will be used for regular communication an important work part of the time. Therefore the extra power used strictly for localization is likely to remain insignificant in such an environment.

3.6 Scalability

For location determination of wireless devices there are two types of scalability. First, we must consider the number of terminals that simultaneously can perform localization. This suggests that a scalable solution should be independent of the fixed infrastructure, as otherwise the load (caused by the localization) would increase with the number of clients.

Second, a good solution should permit the addition of new access points in order to increase the area covered by the system or its capacity with little extra work. Again, the less the dependence on the infrastructure the smoother the scaling of the system will be. None-the-less the additional work will be non-negligible, in particular if the system is based on an empirical model requiring the new coverage to be sampled or an existing area to be re-sampled [49] [50].

3.7 Fault tolerance

In case of an access point failing, how will the system perform? Consider a system based on client terminals that perform the majority of the work themselves (querying its surroundings and calculating its position), a non-responding access point will result in the absence of one response. Hopefully this results in only a reduction of accuracy rather than a completely misleading location.

However, methods based on signatures or signal strengths might not be able to resolve the location to a single area, as there could now be multiple possible locations for the mobile given the remaining unconstrained data. Anyhow, because detection range is likely to be greater than communication range, most infrastructures will have more than the minimum number of access points to enable a mobile device to determine its location (particularly in the indoor setting which this thesis is concerned with).

3.8 Terminal installation and initiation

As previously stated, as much as possible of the location determination process should be performed at the client side. Nevertheless, the user effort necessary must be kept at an absolute minimum as not to impede its use, thus allowing even a novice to utilize it.

The installation of the localization software and its initiation prior usage should therefore be largely automated. The speed of the actual location determination must also be considered. Depending on the type of service, few users will be patient enough to wait a full minute for their position to be determined. However, a possible scenario is that the user can take advantage of rather coarse location estimation quickly, while a few seconds later getting a refined location (if desired).

3.9 Training needs

As discussed in section 6 different approaches to localization differ greatly in how much time must be spent on training before the position determination can take place. Depending on the needed precision, it may or may not be worth spending extra time on training.

The amount of time available (or necessary) for training is a very important aspect when evaluating a localization system.

3.10 Integration of environment changes

Though unwanted, the localization system will eventually face a change in the environment where it is operating. Perhaps an access point or a major obstacle is moved. Somehow these changes will have to be dealt with in order for the localization system to maintain its performance.

Ideally, the system is capable of automatically recalibrating itself continually. To summarize, a localization system should require minimal human effort while adapting to changes in the environment.

3.11 Global localization

When the client is initially unaware of its position and wishes to determine its absolute location in relation to the world, this is called *global localization*. This is obviously more challenging than local localization, but is nevertheless relevant for this thesis, since we can not assume to know where or when the client started his or her terminal.

3.12 Multi radio mechanisms

As previously indicated, the localization system discussed in this paper is intended to utilize location information collected from multiple wireless technologies. This data fusion, as well as the easy integration of new technologies, must consequently be considered throughout the entire design.

3.13 Precision versus confidence versus certainty

When evaluating the outcome of a localization system, in other words the provided location, there are two main aspects to consider: precision and confidence. Even though one might regard the two as being similar, it is essential to understand the difference between them.

Precision can be used to indicate the error of a determined position, in other words, the difference between the calculated and real position. However, the precision is unlikely to be constant over a series of localizations, which is why a confidence interval is often used: x,y$\pm$z meters 80 percent of the time.

In this paper, due to the characteristics of the position outcome to be explained later, we use *certainty* to indicate the likelihood of being at one of a number of predefined locations.

Chapter 4

Position determination hardware

Outdoor location information can be acquired using the widely used Global Positioning System (GPS). But what about indoors, where there are numerous interesting applications? This chapter introduces the reader to approaches to indoor localization.

4.1 Dedicated technologies

Several location determination systems have been developed based on sonar [22] [23], tactile [27], infrared [24] [25], visual [26], and acoustic technology [28] [29]. Each of these techniques is described briefly below.

4.1.1 Visibility

This approach associates the client with a given reference point when the client is within visual range of the reference point. Some solutions assume one reference point being visible at a time, whereas other takes into account multiple targets to define a potential subspace for the device's location. The reference points are normally placed so that only one at a time is visible to the client. Obviously the resulting answer defines the device's location to be within a certain volume of space, rather than an exact position. [4]

4.1.2 Image analysis

Several systems have successfully been developed based on image analysis methods. Apart from needing suitable image sensors, this type of solution demands a large amount of processing, which makes it less suitable than some other techniques. [96] However, real-time tracking of visual targets has been done with an HP iPAQ.

4.1.3 Dead-reckoning

The simplest technique for indoor localization is dead-reckoning. That is, we limit the number of possible entrance points, and starting from one of these we use an accelerometer to estimate the movement of the client. With this technique the estimation errors will accumulate until we are able to localize the device according to some external reference point. This also constrains the client to being initiated outside this entrance point. [1] For PDAs and cell phones that are actually rarely turned off this is not a major limitation. Nevertheless, to locate the entrance points it also requires the chosen interface to be enabled.

4.1.4 Multi-lateralization

By measuring the distance or angle to several reference points we can determine the position of the client. One way of measuring the distances is to measure the time of signal arrival, which obviously demands time to be precise and the same throughout the system. This can be done by sending time stamped frames to the mobile or receiving frames from the mobile. The final position can be found using multi-lateralization between the different parts. This way of determining the location of a client is used in GPS, which is the most common method of localization for outdoor usage [4]. For more information on using triangulation for indoor localization, see section 4.2.

4.1.5 Summary dedicated technology

Some of the more successful position determination systems include Cricket [15], Active Badge [19], ORL ultrasonic location system [2], and the SNU indoor navigation system [20]. The best provide fine grained resolution (down to a few centimeters), while being relatively inexpensive to manufacture and consuming little power. [4] In addition, some solutions have been scalable both in terms of the number of clients supported and the area to be covered, and the impact on the system it was monitoring was minimal. [2]

What most of these methods have in common is that they have been implemented using hardware which has been integrated explicitly for location determination. Unfortunately, this contradicts the criterion in section 3.4 concerning terminal availability. The idea is that location based services should be available to users with non-specialized terminals, i.e. without requiring additional hardware.

While many terminals do include IR it is generally the case that today's infrastructures rarely have IR transceivers, accept in a very limited number of locations. Similarly, since most handsets have audio input/output and support high sampling rates, one could use an acoustic system operated at frequencies above human hearing to locate handsets. Considering only terminal availability, IR or acoustic solutions would be very suitable. On the other hand, the infrastructure is unlikely to be available as required by the criteria in section 3.3.

Despite the clear advantages of the above solutions, their technology can not be considered appropriate for the purposes of this thesis.

4.2 GPS derivatives

The success of the Global Positioning System (GPS) for outdoor navigation - whether it for ships, airplanes, other vehicles, or an individual walking - has been enormous. It is very effective over wide and open areas.

Unfortunately, for indoor and urban usage, it turns out to be less successful because buildings, trees, vehicles, etc. will block the line-of-sight signal or cause the radio signals to be reflected. This can result in an incorrect position or no position at all.

Still, a potential solution is to extend the functionality of GPS in order to be able to take advantage of the existing infrastructure, both in terms of satellites and available terminals. This concept aims to place ground-based transmitters called pseudolites sending GPS like signals to complement or even replace the GPS constellation entirely. Such systems have been developed with success, both for outdoor and indoor usage. [13]

Basing an indoor localization system on GPS technology seems appealing due to the advantages of combining worldwide satellite based and indoor pseudolite based localization while utilizing a common mechanism. There are indeed GPS receivers that support using pseudolites in parallel with satellites, however there are several technical and legal limitations [14].

Unfortunately, in order to take advantage of this type of indoor position determination, one will not be able to take advantage of the existing infrastructure hardware. Instead one must deploy pseudolites to sufficiently cover the domain. In addition, the question of the immediate availability of client and infrastructure hardware remains to be answered, as of fall 2006. Finally, the power consumption of the GPS receiver is not negligible, which will reduce the operating time. [9]

Together, these disadvantages limit solutions based on GPS, hence they will not be studied further in this thesis. Nonetheless, they represent an interesting alternative for the near future and are actively being researched by others.

4.3 Wireless position determination

One of the most popular standards for indoor wireless communication is IEEE 802.11b wireless LAN (WLAN), currently deployed in numerous office buildings, museums, hospitals, shopping centra, academic and corporate campus, and so on. This evolution has been possible thanks to the relatively inexpensive hardware, due to high market volumes and product integration. Recent research in the field of localization has focused on using such off-the-shelf devices in order to exploit the opportunities offered by this widely used wireless technology.

Using this approach, personal data assistants (PDA), laptops, etc. with one or more integrated wireless interface that is (or are) for communication purposes, can be localized using the existing communication infrastructure. The advantage compared to deploying additional hardware for localization is obvious; very large number of clients can utilize the services and there is a tremendous cost saving for the service provider. [6]

Position determination based on common wireless technologies such as WLAN or Bluetooth seems to best meet the requirements of this thesis project. As a consequence they will henceforth be our focus when developing our localization system.

Chapter 5

Technical overview of wireless position determination

5.1 Functional mode

The different technologies used for wireless location determination can be categorized in two main approaches: those based on an infrastructure with fixed reference points and those without (ad hoc). As of today, the majority of the research and commercial solutions are using an infrastructure.

- Ad hoc solutions: Determining the location of a wireless terminal when there is no fixed infrastructure available has recently become a more and more an active area of research. As there are no fixed access points these systems can not be assumed to know anything about the surrounding environment. Therefore, they have to rely solely on radio propagation modeling of the received signals.

- Infrastructure based solutions: The most trivial way of locating a terminal is to say that it must be in range of the access point that it is currently associated with. [10] While this does localize the terminal, the encircled area can be quite large, hence we may look at methods for reducing this area to increase the location resolution to the desired precision.

5.2 Location determination

Most wireless location solutions consist of two subsequent phases. Together they result in an estimation of the device's location.

1. distance/angle estimation, and

2. distance/angle combining.

5.2.1 Phase one

Distance estimation is normally based on one of the following approaches:

- **Received signal strength (RSS):** Subtracting the received strength of the signal from the known transmit power will give us the power loss. This information can then be combined with an empirical or theoretical model to provide an estimate of the actual distance between the transmitter and receiver.
- **Time:** The time-of-arrival (ToA) or the time-difference-of-arrival (TDoA) can be translated into distance estimation by knowing the propagation speed of the signal. The signal in this approach can be radio frequency (RF), acoustic, infrared, or ultra-sound or a combination of these.
- **Angle:** Estimate the angle of arrival (AoA).

5.2.2 Phase two

When several distances or angles have been estimated as described above, they are together used in one or more combinations in the following manners:

- **Hyperbolic trilateration**: Locate the node by finding the intersection of at least three circles where the radius is the estimated distance. See figure 5.1.

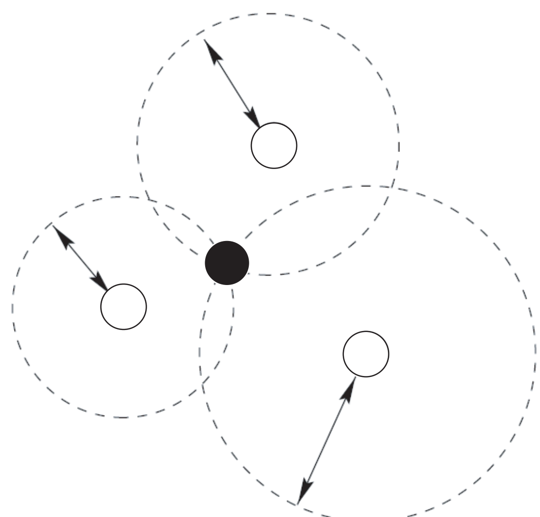

Figure 5.1. Position determination through trilateration

- **Triangulation**: If an angle has been estimated, use trigonometry to calculate the position of the node. See figure 5.2.

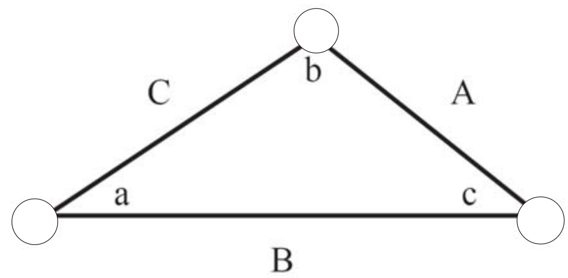

Figure 5.2. Position determination through triangulation

- **Maximum Likelihood (ML)**: Estimate the position by minimizing the difference between measured and estimated distances. See figure 5.3. [9]

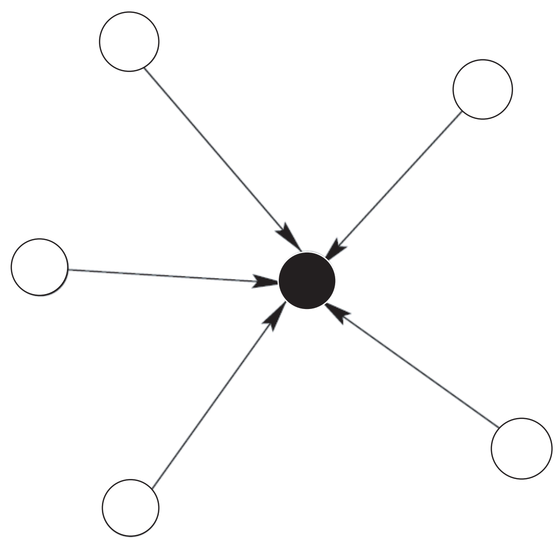

Figure 5.3. Position determination through maximum likelihood

5.3 Signal strength units

In the context of wireless signal strength measurements there are three common units used to represent the values: mW (milliwatts), dBm (dB-milliwatts), and RSSI (Received Signal Strength Indicator). The intention of this section is to clarify these expressions and their relation to each other.

mW indicates the amount of energy present in the signal. A typical value for a wireless access point or a wireless network card could be up to 100 mW output.

dBm is a logarithmic measurement of signal strength. The benefit is that the range will be sampled in a more convenient way, as the signal strength fades as the inverse of the square of the distance (assuming free space). A linear scale would provide less accuracy for a fixed number of bits.

$$dBm = 10log(mW)$$

RSSI is a signal strength unit defined as an 8-bit value in the IEEE 802.11 specification. However, in practice, each vendor has defined its own maximum RSSI value, so that fewer than the allowed 256 values are used. Notice that no general relation between the RSSI and the received energy can be defined, as it is intended to be used internally in the network card. However, for most major vendors there is a known relationship between RSSI and dBm. For example, to get the signal strength in dBm from an interface based on Atheros chip set 95 must be reduced from the RSSI. [97]

Chapter 6

Approaches based on wireless signal strength

This chapter describes two approaches for determining the position of a client. They are both based on the received wireless signal strength. Exploiting the signal strength is not the only way to determine the location, but doing so has the advantage of being applicable to several wireless communication technologies. It is also one of the more common approaches.

The two main approaches based on the received wireless signal strength are:

- Training, or
- Modeling

A detailed description of each is given in the following sections.

6.1 Fingerprinting

Fingerprinting is based on a number of reference points transmitting some signal, usually via radio frequency. As discussed in [8], [80], [81], [82], [83], [84], [85], [86], [87], [88], [89], [90] and [91], the set of signal strengths of surrounding access points can be used to characterize locations.

Before explaining further, let us start with a definition:
In the context of wireless localization, we define the fingerprint of a certain location as a list of all access points within range and their corresponding signal strengths. Together, these pairs constitute the identity of the position.

Position determination using fingerprinting is based on two phases.

1. Training
2. Position determination, based upon matching against the data derived during the training phase.

During a prior training phase, the area (or volume) will typically be sampled regularly in a two-dimensional (or three-dimensional) coordinate system. With an interval between samples of for example one meter, the fingerprint, or signal strength signature, of each of these sampled locations is determined.

During the actual position determination, the received signal strengths of the nearby access points will be matched against the set of sample measurements previously collected. The location of the fingerprint that most resembles the current values is the likely location of the device.

In order to quickly be able to search these fingerprints on a mobile device with limited resources, the format for storing them is obviously important.

6.2 Modeling

Like the previously described approach based on training, this method assumes that a number of surrounding reference points are regularly transmitting.

The aim of propagation modeling is to estimate what the signal will look like at a given distance from each reference point, given information about the transmitted signal strength and the surroundings.

Obviously, the more information about indoor geometry and materials which we take into account, the more accurate the propagation model will be. [8] Consequently, this will increase the computational requirements.

6.3 Alternative design

In the above solutions, the client device takes advantage of surrounding reference points in order to localize itself. Note that a inverse solution is also possible, where the client transmits a signal to be received by several fixed reference points [4]. In the case of location fingerprinting, the different signal strengths at the reference points must then be utilized to compute a signature in order to determine the position of the client.

This second alternative approach is more complex to implement. Though less common, it is used by several indoor position determination systems [31] [32] [33] [51].

6.4 Performance

6.4.1 Training

In general terms, performing location fingerprinting based on more samplings of the potential area where a device is to be localized will give more accurate results, especially when the signal propagation modeling fails due to environmental effects. However, in practice, localization entirely based on fingerprinting is rarely desirable because of the training that it requires.

Some implementations (previously referenced) based on fingerprinting have achieved a standard accuracy of 1-2 meters during decent conditions and after collecting samples with a corresponding resolution. One probably needs to spend at least 30 seconds at a each location to collect enough signal strength observations to get representative values. In other words, building a complete training set represents a none-negligible amount of work. It is clearly the case that for a large space automated techniques for sampling are desirable.

6.4.2 Modeling

Although numerous efforts have been made to model signal propagation, experiments in different indoor environments have shown varying results. Several office buildings have shown a good log-normal fit, but this turns out to be feasible only when the signals are line-of-sight. [1]

It has been shown that the result can be improved by using a Bayesian localization framework to probabilistically evaluate the location. With this technique we can expect a precision of about one meter in good conditions. [1]

Haeberlen et al. [5] proposed a solution that offers room- or region-level granularity. The main advantages is the system's minimal training needs, quick responses, support for dynamic environment (i.e. with moving people), and dynamic localization (i.e. moving terminals), and that it is based on off-the-shelf terminals. In this thesis we will focus on room level localization, but using one or more mechanisms to be described in the next chapter.

Chapter 7

Particle filter

This chapter explains the theoretical foundation on which the so called "particle filter" method is based, before describing how a particle filter can be used for localization.

7.1 Particle filter for position determination

We are interested in using a particle model to describe the potential location and changes in the location of a user's device. The idea is to evaluate the probability of a number a potential locations, each represented by a particle. The most likely location of the device can then be determined by combining the locations of the particles.

7.2 Bayes filters

We begin by describing a Bayes filter, of which the particle filter is a particular case.

A Bayes filter is a powerful probabilistic framework. As it is very general, the number of applications is quite large, but there are two main reasons for its use with localization [36]:

- Since no sensor is perfect, multiple sensors will not provide completely coherent measurement information. Representing and operating on uncertainty and conflicting hypotheses, as possible with a Bayes filter, is therefore a key task to exploit multiple measurements.

- Representing locations statistically in the way that a Bayes filter does enables a unified interface for location information. This allows sensor fusion to take place, i.e. combining sensor information from different sensor types.

In brief, a Bayes filter estimates a probability distribution over a state space based on possibly noisy observations. In the case of localization in its basic form, the different states correspond to possible locations, and the observations are the result of sensor measurements.

For each point in time, we define the *belief* as

$$Bel(x_t) = p(x_t|z_{1:t})$$

where x_t is a position and $z_1, z_2 ... z_t$ observations through time t. Thus the belief that a device is at a particular location is based upon the prior observations.

Roughly, the belief quantifies the probabilities of the different positions given all prior observations. This is called the posterior distribution [52] and the complexity of computing it is in general exponential due to the increased amount of data with time.

However, by assuming that each position x_t only depends on the previous position x_{t-1} the belief can efficiently be computed without information loss. [37] This assumption is generally referred to as the Markov assumption, which can be summarized as the following equation [59]:

$$P(x(t)|H, x(t-1), x(t-2), ...) = P(x(t)|H, x(t-1))$$

The process of updating a Bayes filter consists of two main steps:

1. Whenever a sensor provides a new observation, the first step called prediction is performed as follows:

   $$Belp(x_t) = \int p(x_t|x_{t-1}) Bel(x_{t-1})\, dx_{t-1}$$

   where $Bel(x_{t-1})$ is the previous belief or prior distribution [52], and $p(x_t|x_{t-1})$ is the motion model. The conditional probability describes the system dynamics. It defines, given the previous location x_{t-1}, where the object could be at time t.

2. Notice that so far the new observation z_t is not used, but in the second (correction) step it is:

   $$Bel(x_t) = \alpha_t p(z_t|x_t) Belp(x_t)$$

   where $Belp(x_t)$ is the prediction from above, α_t is a normalizing factor ensuring the sum over the entire posterior equals one, and $p(z_t|x_t)$ is the sensor model. [63]

   The sensor model is completely sensor dependent and should take into account its properties including position and error characteristics. Roughly, it quantifies the probabilities of current observation at the potential positions [36] and is often referred to as the likelihood distribution.

> Note that formally according to the theorem of Thomas Bayes, the left hand side of the equation should be divided by the evidence distribution, representing the likelihood of an observation. However, it is usually independent of the solution parameters we seek, and can hence be neglected. [52]

To recapitulate, here are the important distributions above.

$p(x_t\|Z_t)$	: the a *posteriori* density given the measurements
$p(x_t\|Z_{t-1})$	: the *a priori* density
$p(x_t\|x_{t-1})$	: the process density describing the system's dynamics
$p(x_t\|x_t)$	: the observation density

7.3 System dynamics

As previously indicated, the motion model should predict the likely position of an object, given what we know about its (just) previous state. In practice, the accuracy of prediction varies greatly between implementations. Some of the most common motion models are briefly described in the following sections.

7.3.1 Brownian motion model

One of the most basic motion models assumes that the object can travel in any direction at any time. In other words the direction of the movement is random. This may be false in practice for a person, but it avoids the need to calculate the acceleration and velocity in addition to the position. In the case of a wireless device being localized, this is even more relevant as no accelerometer is present in most devices. [1]

The Brownian motion model's relatively weak constraint on motion does not risk being violated by a person quickly changing his direction. [61] On the other hand, the disadvantage with such a motion model is that the hypothesis will quickly become spread out when there are no observations. [62] In practice, this motion model has turned out to be applicable in many different circumstances. [61]

7.3.2 First order motion model

As in [64], the motion can also be modeled as a first order (linear) approximation of the velocity, based on information obtained by comparing the particle's state from the last two time steps. This obviously assumes that the target's heading will remain unchanged, which is often not the case as people will turn corners and avoid obstacles. These sort of actions can not be represented with a first order model. [62] However, it provides a good motion model when sampling rate is high; after a turn, the user's device is likely to return to moving in a straight line.

[1] Note, however, that more and more vehicles are being equipped with accelerometers and gyroscopes to provide inertial navigation information.

7.3.3 Gaussian mixture motion model

A more sophisticated model for human motion is proposed in [65], which exploits the fact that human motion often engages in one of a small number of predefined actions, i.e. move forward, backward, left, right, or stop. It consists of a piecewise-linear Gaussian mixture, so that each cell holds a Gaussian representing the probability of a given movement.

7.3.4 Higher level motion model

Human motion is often characterized by a final destination of the movement. For example by incorporating information about where the object has previously moved in the present situation, this destination can be predicted.

7.4 Deterministic approach

Generally, if no prior information about the user motion is available and we believe that the likelihood distribution peaks around a particular value that maximizes both the posterior and the likelihood, there is no need for Bayes' probabilistic approach. Then, the most efficient way to solve the problem would be to use the Maximum-Likelihood (ML) method, for example chi square-minimization.

Even though ML is less taxing, the use of Bayes method is often advantageous due to the following reasons:

- If the likelihood distribution has more than one local maximum, ML will be sensitive to the initial solution parameters whereas Bayes will be nearly independent of them.

- ML ignores the probability of other solutions that might be equivalent within the data's resolution, whereas Bayes method provides much more information about the possible solutions by generating the actual posterior.

- ML generally adds more restrictions upon the type of parameters for which we can stably maximize the likelihood than does Bayes.

7.5 Belief representations

As previously indicated, Bayes-filter only provides an abstract concept for recursive state estimation. For its implementation there are several possible ways of representing the belief $Bel(x_t)$: Kalman filters [38] [39] [40], multi-hypothesis tracking [41], grid-based approaches [47] [48], topological approaches [43] [44] [45] [46], and finally the choice of this paper - particle filters. An explanation of why the particle filter has been chosen will be given in the following section.

7.6 Particle filters

In particle filters [42] belief is represented by a set of samples, or particles:

$$S = S_i | i = 1..N$$

The goal of the algorithm is to recursively compute these samples at each time-step. In analogy with the description of Bayes filter depicted above, there are two main phases. During the prediction phase samples are randomly drawn from the probability density $p(x|z)$. These samples can then be used to approximately reconstruct the density in order to determine the most probable current location.

Depending on the implementation the particle filter algorithm is also known as the bootstrap filter [54], the Monte Carlo filter [55], or the Condensation algorithm [56] [57].

As the properties of the probability density vary between the different belief representations, each have their advantages and disadvantages depending on the circumstances. [36] The main reasons why this paper is using a particle filter implementation are:

- It is able to represent multi-modal (arbitrary) distributions. In practice, this provides the ability to handle ambiguities, i.e. multiple probable locations, and ultimately to globally localize the terminal. This is not the case when using techniques based on the otherwise robust Kalman filter which is limited to representing unimodal belief. [53] Consequently, Kalman filters are mainly applied when tracking a device with a known initial position [59]. Then, the probability distribution will be unimodal and the Kalman filter very efficient [36].
- It can converge to the true posterior even in non-Gaussian and non-linear dynamic systems. [36]
- It drastically reduces the memory usage compared with grid-based methods. The latter can admittedly provide arbitrary accuracy when a fine grain resolution size is specified, resulting in perhaps the best location precision to date [59], but at the cost of a high computational complexity. [36]
- It is more accurate than Markov localization with a fixed cell size. This is due to the states representing samples not being discretized. [53]
- It performs well with multiple noisy and inaccurate sensors by integrating measurements over time. [36]
- It is a flexible method [36] and is relatively straightforward to implement. [53]
- It has successfully been further improved by adapting the sampling size through Kullback-Leibler distance (KLD) sampling. [60]

7.7 Particle location report

In order to determine the location of an object, the samples in the particle filter must somehow be combined to produce the actual location estimate. There are several different possible ways of doing this [66], of which the most common are described below.

7.7.1 Best sample

Use the sample having the highest importance factor as the estimated location, this can be found in linear time. The disadvantage is that the sample choice will result in a discretization error, of which the effect will vary depending on sample size and convergence.

$$x_k^j | b_k^j = max(b_k^i) : i = 1, 2, ...M$$

7.7.2 Weighted sample mean

Estimate the location as the mean position of all samples where each sample is weighted by its importance factor (as per the equation below). As with the previous method, it suffices to traverse the samples once.

$$x_k = \sum_{i=1}^{M} b_k^i x_k^i$$

However, this method will in certain circumstances report a location far from correct, due to the nature of particle filters. As these are able to represent multi-modal probability distributions, two centers apart with high particle density would typically result in an estimated location situated in-between the two likely locations of which one is most likely the correct.

7.7.3 Robust sample mean

This way of estimating the object's location is a combination of the previous two methods. Instead of calculating the mean of all samples only those positioned within in a certain distance from the best sample are taken into account. Overall this method would perform better than the previous methods, with the drawback that it is also the most computationally expensive.

Chapter 8

Wireless LAN

This section describes the fundamentals of using WLAN to provide signal strength measurements. Initially the specification of the underlying wireless technology is described. Then the fundamentals of signal strength measurements are explained. Finally, the architecture of the Linux and PocketPC platforms are described, indicating the application programming interfaces for getting the measurement samples.

8.1 IEEE 802.11 specification

The IEEE 802.11 standard [68] refers to a family of specifications developed by the IEEE [69]. It specifies wireless communication between a client and a base station (also called an access point), or between two clients.

In the initial standard which was accepted (in 1997) two methods were defined for communicating using the 2.4 GHz band: frequency hoping spread spectrum (FHSS) and direct sequence spread spectrum (DSSS). Advantages of spread spectrum technology is that it avoids narrow band interference at a single frequency from blocking the signal and it spreads the interference causes by WLAN communication over a wider band so as not to interfere with narrow band signals. The most commonly used version of the 802.11 specification is called 802.11b. It is often referred to as Wi-Fi, and is based on DSSS. From now on, this is the standard which should be referred to unless indicated otherwise.

8.1.1 Received signal strength indicator

The 802.11 wireless LAN medium access control (MAC) utilizes the services of the physical layer (PHY) of the 2.4 GHz DSSS system. It consists of two functions:

- A physical layer convergence function, which adapts the capabilities of the physical medium dependent (PMD) system to the PHY service. It must be supported by the physical layer convergence procedure (PLCP), which in turn provides a means of mapping the IEEE 802.11 MAC sublayer protocol data units (MPDUs) into a framing format for the sending and receiving of user data and management information using the PMD system.

- A PMD system, which defines the transmitting and receiving of data through a wireless medium (WM) which is based on DSSS.

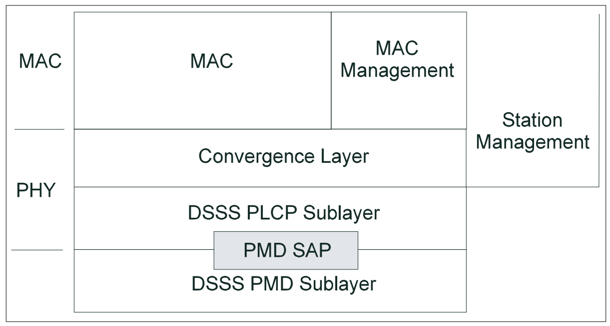

Figure 8.1. PMD layer reference model. Source: [30]

The two sub layers (PLCP and PMD) of the DSSS are depicted above, showing their relation to the entire DSSS physical layer (DSSS_PHY). As indicated, the DSSS_PMD sublayer provides service to the PLCP for the DSSS_PHY. The PMD SAP specifies a list of parameters. One is the PMD_RSSI which is of particular interest to us.

Parameter	Associate primitive	Value
DATA	PHY-DATA.request PHY-DATA.indicate	Octet value: X'00'–X'FF'
TXVECTOR	PHY-DATA.request	A set of parameters
RXVECTOR	PHY-DATA.indicate	A set of parameters
TXD_UNIT	PMD_DATA.request	One(1), Zero(0): DBPSK dibit combinations 00,01,11,10: DQPSK
RXD_UNIT	PMD_DATA.indicate	One(1), Zero(0): DBPSK dibit combinations 00,01,11,10: DQPSK
RF_STATE	PMD_TXE.request	Receive, Transmit
ANT_STATE	PMD_ANTSEL.indicate PMD_ANTSEL.request	1 to 256
TXPWR_LEVEL	PHY-TXSTART	0, 1, 2, 3 (max of 4 levels)
RATE	PMD_RATE.indicate PMD_RATE.request	X'0A' for 1 Mbit/s DBPSK X'14' for 2 Mbit/s DQPSK
RSSI	PMD_RSSI.indicate	0–8 bits of RSSI
SQ	PMD_SQ.indicate	0–8 bits of SQ

Figure 8.2. List of parameters for the PMD primitives. Source: [30]

The PMD_RSSI.indicate primitive is optional and returns the received signal strength indicator (RSSI). It is the result of an RF energy measurement by the physical layer receiver and is generated when the transceiver is in the receive state. The PMD then generates this RSSI value and continuously updates it - passing it to the PLCP. It is represented by a value of up to 8 bits allowing 256 levels of signal strength to be discriminated.

A related parameter is ED_THRESHOLD, which indicates the threshold of the received energy which is considered sufficient to decode incoming symbols. If PMD_RSSI is greater than ED_THRESHOLD, this is indicated when the PMD_ED parameter is ENABLED, otherwise it is indicated as DISABLED.

8.2 Accessing the WLAN signal strength measurements

8.2.1 Prerequisites

First of all, a wireless adapter must be installed in the computer (in this case, a HP nc6120) which is to be used with the test bed. The operating system used was the Fedora Core 4 distribution of Linux. I updated the firmware and the driver for a Intel® PRO/Wireless 2200BG adapter before installing the latest version of the Linux Wireless Extensions (see the following section).

8.2.2 Linux architecture

Under the Linux operating system five layers can be distinguished between the network interface card (NIC) and the system when performing the position determination.

At the lowest level, the network interface card (NIC) driver can be found. The driver chosen for the Intel PRO/Wireless 2200BG is called IPW2200 [71] and was chosen due to its support of Wireless Extensions and the fact that it is open source.

Wireless Extensions (WE) [72] is a generic API which, when supported by the driver, exposes the wireless configuration and interface statistics to user space. To be available it must be compiled with the Linux kernel. Traditionally, the method to access driver parameters in Unix is to set and get parameters of a device via *ioctl* calls. WE adds a new set of ioctls, allowing the parameters to be changed in a generic way - without restarting the driver. This API is to a great part defined in /usr/include/linux/wireless.h.

Iwlib defines a set of subroutines which are common to the Wireless Tools (WT) [73]. These are a set of user tools for manipulating the NIC by taking advantage of WE. I have used some of these routines for performing the scan for access points and getting the RSSI measurements.

Spotter.c defines a way of performing active probing for access points by using the WE. This code is derived from the Placelab [73] project to facilitate scanning for access points.

The Java Native Interface (JNI) [70] layer provides the link between the lower levels implemented in C and the rest of the code implemented in Java. Recall that the ambition was to implement as much as possible of the localization system in Java, in order to provide platform independence to the extent possible. The JNI is implemented in ScanLib.

8.2.3 PocketPC architecture

In the PDA version of the implementation the architecture is simpler, with three main layers.

The NIC driver used in Windows CE supports the Network Driver Interface Specification (NDIS) [74].

In order to access the driver a set of methods derived from Placelab [73] are used. These are defined in WiFiSpotterImpl.c. In a manner similar to under Linux, the native code is accessed from Java via JNI [70]. This adaptation layer is defined in com_alcatel_moba_ctxserv_core_common_WiFiSpotter.c, which is also derived from Placelab [73].

8.3 Signal propagation

As previously indicated, the IEEE 802.11b standard [30] uses frequencies in the license-free 2.4 GHz band. The position determination system developed here is based on the signal strength of nearby access points; hence a good understanding of signal propagation is crucial.

Moreover, understanding signal propagation is not only important when developing a position determination system. The ability to predict the signal strength provided by access points in a WLAN is also of interest to implement services such as transmission power control (so far typically not implemented for IEEE 802.11), and interference prediction and minimization. In addition, this knowledge can help when deploying access points to enable them to be positioned as suitably as possible.

Unfortunately, WLAN signal propagation turns out to be rather unpredictable. First, the signal decrease is non-linear with distance [1]. Second, structures inside the building will cause the radio waves to be reflected, refracted, scattered, and absorbed. [7] Notice that a single person can alter the signal strength by as much as -3.5 dBm [58]. The fact that signals can take many different paths results in a phenomenon called multi-path fading, which means that the signal will often reach the receiver by several paths with multiple different amplitudes and phases [11], hence the resulting noise will be non-Gaussian and some of the signals may actually cancel out others. Third, other 2.4 GHz equipment can interfere with the 802.11b signal, such as microwave ovens, Bluetooth (R) devices, and American 2.4 GHz cordless phones. [1].

The standard way of predicting the signal strength across an environment is by means of a path loss model, which can for example be found empirically [75]. Simple models base the attenuation solely on the distance from the access point. More complex models taking into account additional information about the environment can also be applied, with the obvious inconvenience of needing to enter into the system details of the materials and locations of walls and other obstacles. [76]

8.4 Determining the signal strength

Due to the fact that a rather small change in estimated received signal strength of a distant WLAN host could result in a rather big error in the distance estimation, the received signal strength must be determined as accurately as possible. This section will focus on describing the method used in order to find the most appropriate way of doing this. The different methods will then be evaluated in different circumstances.

8.4.1 Hardware

Unlike several other projects related to signal strength determination which use pole mounted antennas and spectrum analyzers, I decided to only use standard off-the-shelf hardware throughout all my experiments and evaluation. The reason is that I believe this will make the results more representative and hence more applicable in the final application.

The possible differences are that signal strength measurements provided by standard devices may be noisier, and that the transmission and reception patterns of consumer access points and network cards will be neither exactly the same nor omni-directional.

8.4.2 Reference signal strength

In order to find and evaluate the most appropriate ways of determining the signal strength one must first define what signal strength is considered as "correct". I initially define this reference signal strength as the most frequent value after a series of measurements. Several papers, such as [79], have drawn the conclusion that the signal strength distribution is Gaussian.

The test bed used to determine the reference signal strength consisted of a wall mounted Netgear WG-302 wireless access point. To collect the measurements, a Linux based laptop was put on a stool at a distance of five meters from the access point so that the height difference between the two was 1.9 meters. To avoid the risk of small data sets, with the attendant risk of not being correct, I therefore collected a larger number of measurements before drawing any conclusions from my experimentation.

A rather high scan frequency was required to observe changes as quickly as possible. At the same time, the delay had to be long enough for a *new* signal strength measurement to be reported by the WLAN interface's driver. This lead to using a delay between scans of 200 milliseconds, this value was consequently used in all subsequent measurements.

Figure 8.3 presents these experiment results. The experiment lasted 67 minutes during which 20021 signal strength measurements were collected. The propagation path between the two nodes was for the most of the part line-of-sight (LOS). However, the environment was highly dynamic with many people, and occasionally even major furniture items, crossing the path.

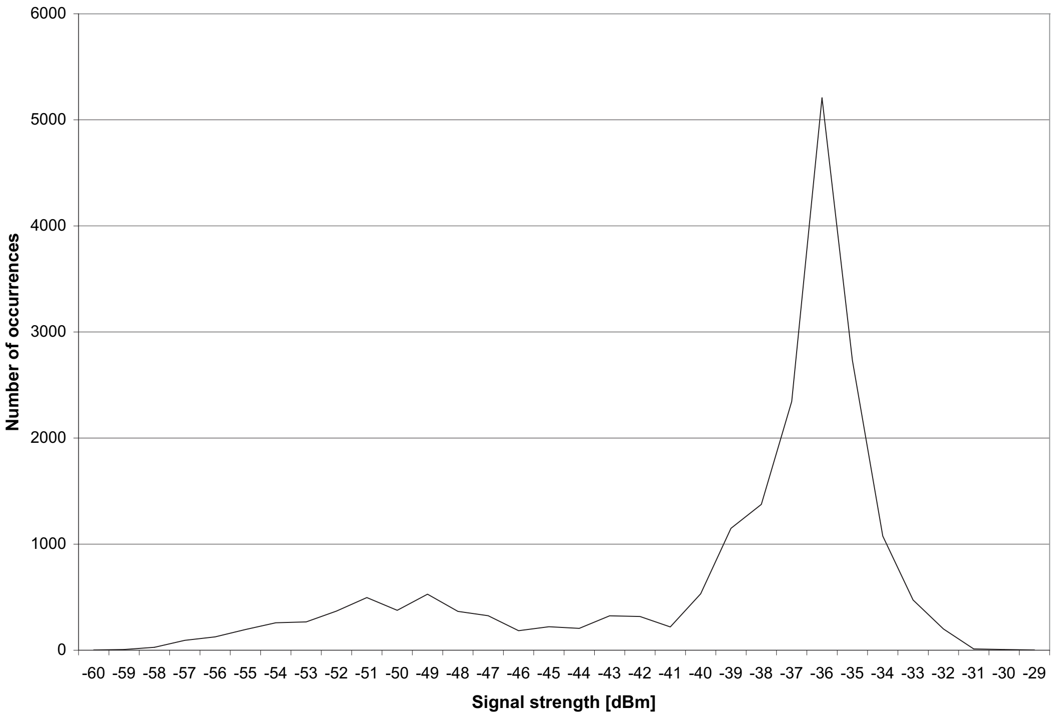

Figure 8.3. Signal strength histogram: 20 021 observations for a given access point.

The signal strength distribution is clearly spread out over an interval between -29 and -60 dBm, not counting two outliers at -67 dBm. However, one value is noticeably more frequent with over 25 percent of the samples having this same value: -36 dBm. This will consequently be considered as the reference signal strength for this location.

8.4.3 Sliding window signal strength

If it was always possible to collect thousands of signal strength measurements before reporting the most frequently occurring value, the task would be simpler than in a real-world application where the available amount of time for measurements is more limited. The objective is to be able to correctly estimate the signal strength after the shortest possible amount of time.

Others have chosen to calculate the current signal strength directly as a function of the previous measurement. Such an approach will generally not be used in this paper. Instead the intention is to calculate the signal strength in a way that the impact of a temporary change in signal strength will be minimized. Such a short term change would typically be a drop in signal strength due to for example an obstacle impeding the RF signal when an object moves in the space between transmitter and receiver.

This paper's approach for determining the most likely signal strength is to maintain a sliding window of the most recent measurements reported by the network card. The intention is that at any time one could then extract the probable signal strength value by applying a well chosen function to this history vector. Several such functions have been implemented. See section 11.3 for more information on these functions and their performance.

The length of the this history vector is an important choice since it will heavily influence the responsiveness and accuracy of the application. A longer history will naturally smooth out temporal variations more efficiently. On the other hand, this will increase the delay between changes in the momentary signal strength and the change in the reported signal strength.

The choice of history length is related to the type of applications in which the reported signal strength will be used. Unfortunately this makes the implementation dependent on the application. Unless a value is determined as generally appropriate, a parameter for the history length will have to be set for each application.

Chapter 9

Bluetooth

Bluetooth was originally conceived by developers at L. M. Ericsson in 1994. The name is derived from the name of the Danish king Harald Bluetooth (Blåtand) who unified the warriors of Norway, Sweden and Denmark. In 1998, a Bluetooth Special Interest Group (SIG) was formed with the participation of other companies to commercialize the technology. [102] [110]

9.1 Motivation

Bluetooth has several obvious disadvantages when compared with WLAN, hence one might ask why Bluetooth is of interest for a localization system. One of the key reasons is that the power from the most commonly used Class 2 devices, is limited to four dBm which gives an operational range of 10 meters. [98] Thus the trivial method of location determination described in section 5.1 may actually be useful, since if you are near enough to hear another class 2 Bluetooth device you must be quite near it.

Class	Maximum range [meter]
1	1
2	10
3	100

Table 9.1. Operating range for the device classes.

Despite the limited range, Bluetooth possesses some characteristics which makes it interesting for a wide range of handheld products, the type of devices for which position determination is needed (within the scope of this thesis). Two of the most important characteristics are that Bluetooth interfaces consume little power and the transceivers are relatively cheap. [100] Together, these two factors allow its ubiquitous usage and "always-on" presence in everyday devices such as mobile phones and PDAs, in contrast to WLAN devices which require more power.

For a position determination system intended for large number of users, the availability of Bluetooth devices is a valuable characteristic. At the end of 2005, more than 272 million units were on the market, and the number is steadily growing by more than nine million units per week globally [99].

9.2 Specification

Bluetooth [101], with the intention of replacing short cables, is a short-range wireless protocol operating in the license free 2.4 GHz spectrum. In order to ensure robustness in noisy environments, Bluetooth divides the band into 79 channels and frequency hops across them up to 1600 times per second, so that the signal is time-division multiplexed with slots of 625 microseconds. In a connection between two devices one of them is always defined to be "master", and the "slave" has to synchronize and adapt its frequency hoping according to it.

Up to version 1.1 of the Bluetooth specification [107] a connection with another device must be established in order to measure its received signal strength. This is an obvious limitation as multiple signal strengths must be determined for the localization algorithm to function, especially since many Bluetooth chipsets only support a single connection being established at any time. [100]

With Bluetooth specification version 1.2 [108] there were several improvements. One was the option of getting the signal strength of discovered devices without actually connecting to them. This is obviously a great advantage for signal strength based localization. However, another feature was also added which could make localization a challenge: Adaptive Frequency Hopping (AFH). AFH was introduced in order to minimize the interference with other devices using the 2.4 GHz ISM band, such as WLAN. Through version 1.2, Bluetooth actually violated the US spectrum regulations (FCC Part 15) in the 2.4 GHz band which requires that secondary users such as WLAN and Bluetooth listen before transmitting and if there is someone else using the channel the device must not transmit. AFH can be described as a non-collaborative mechanism functioning by intelligently choosing and hopping over the occupied channels [111].

9.3 Scanning

To discover other devices in the neighborhood a Bluetooth device performs a device inquiry by sending a predefined sequence of bits while hopping through the channels pseudo-randomly. During the process of detecting nearby devices (such as Bluetooth beacons) the device does not reveal its identity. On the other hand, for a connection to take place, the two devices must exchange identities before being paired. [106]

9.4 Distance estimation

Due to its fundamental advantages over version 1.1, devices compliant with version 1.2 are preferred for use in position determination. These newer devices support providing the received signal strength while scanning its surroundings. This should be contrasted with the functionality of the previous version which required establishing a connection to the other device in order for the signal strength value to be determined.

Unfortunately, none of the currently available PDAs support both WLAN and Bluetooth version 1.2 or later[1]. As a consequence, the possibilities of using the earlier Bluetooth version 1.1 were further studied.

Devices compliant with Bluetooth version 1.1 provide a way of directly accessing information about the signal strength. The Bluetooth specification defines a Golden Receive Power Rank (GRPR) which is the received power level in dBm considered to be ideal. The corresponding RSSI value is an 8-bit signed integer which can be discovered separately for each ACL connection by using the Host Controller Interface (HCI) command Read_RSSI.

A value of zero indicates that the signal strength is in-between the lower and the upper thresholds, a negative value indicates signal strength below the lower threshold and a positive value indicates signal strength above the upper threshold. The Bluetooth definition of RSSI is in fact rather loose, and the upper and lower thresholds of the GRPR must be defined individually for each Bluetooth device model. [107]

Partially as a consequence of the weak RSSI definition, it has been found [105] that signal strength was poorly correlated with distance, and that the RSSI value does not vary uniformly with the actual radio frequency (RF) power level [112].

A related command, the Read_Link_Quality, determines the quality of a Bluetooth connection with another device. In the case of a relatively noise free connection in the absence of obstructions, this parameter should be inversely related to the distance between the two nodes. In fact, in a localization system [106] with D-Link DBT-120 USB Bluetooth devices, it was found that link quality was more closely related to distance than RSSI.

9.4.1 Discussion

In the context of a solution intended to be platform independent, it should also be noted that neither of RSSI nor Link Quality parameters are accessible through the Java Specification Request for Bluetooth communication (JSR-82) [112]. However, this support could be added through a Java Native Interface as in [113].

Finally, one must consider the practical functionality of localization based on signal strength measurements from version 1.1 of the Bluetooth specification. Even though the specification allows up to three simultaneous SCO and seven simulta-

[1]The author had access to for example Qtek 9090, i-mate JASJAR, HP iPAQ 6340 and Treo 650, of which none supported WLAN and Bluetooth version 1.2

neous ACL connections, typical consumer devices only support a single SCO and three ACL connections (or even less for many lightweight devices) [100].

Consequently, getting the link information for surrounding devices would require all the ongoing connections to be halted in order for the scanning to take place. First, nearby devices must be discovered, something which in practice has been shown to often take five seconds, or even ten seconds in some cases [106]. Afterwards, a connection must be established to each device discovered in order to get the RSSI.

Altogether, the complications related to accessing the Bluetooth signal strength and the lack of measurement accuracy greatly limits its usability for the application of position determination.

9.5 Bluetooth related work

Compared to the number of localization systems based on WLAN technologies, rather few are based on Bluetooth. Atlantis [103] reports an accuracy of approximately seven meters. Another solution [100], does not provide detailed information about their localization accuracy, but concludes that Bluetooth is not suited for high resolution, low latency applications due to specification and hardware limitations. In [104] statically positioned Bluetooth devices are used to constantly scan for other Bluetooth devices. Detected devices were then entered into a central database which was used to track the location of all moving Bluetooth devices.

Chapter 10

Fingerprinting implementation

In the context of this paper, two ways of determining the position of a device have been implemented. Both assume the use of wireless signal strength as the principle input and are based on the previously described techniques.

As previously indicated, these two approaches are fundamentally different in their way of exploiting the wireless signal strength. One aims to be deployed in minimal time, providing position information limited to certain locations. The second solution requires more training but could result in better position resolution.

10.1 System description

10.1.1 Potential position description

My approach to this problem was to take only as many samples as needed to determine if the user's device is in one of a small number of locations. The idea was to reduce the number of sampled points to the actual positions that we are interested in being able to report, hence limiting the localization resolution, but minimizing the training needs. In an office environment this could mean taking one sample at each desk, to be able to determine that a user is near his or her desk, along with a few other strategic positions such as meeting rooms, printer areas, etcetera.

Note that, so far, nothing has been said about which wireless technology the collected signal strengths are taken from. This is because the method has the advantage of being applicable to all wireless technologies able to report the received signal strength. Despite the wide applicability, the discussion in this paper will focus on only two technologies - Wireless LAN and Bluetooth.

10.1.2 Location evaluation

Similar to the standard way of performing location fingerprinting, the approach presented in this paper collects data about potential user device locations. The signal strengths of currently available access points are compared with the values registered during the training phase, these are subsequently used in order to determine the location which best corresponds to the measurements.

10.1.3 Bluetooth presence

One could also consider a slightly simpler implementation where the simple presence or absence of an access point can be used to characterize a position. Due to the limitations of Bluetooth scanning described elsewhere in this paper, this approach could be applied to devices with earlier Bluetooth versions.

In this case, this binary observation will not contribute to the precision of the localization, but rather increase the confidence in a location report, as some positions can be ruled out as impossible due to the presence or absence of a given Bluetooth access point.

10.1.4 Discussion

The essential difference between the proposed solution and standard solutions based on fingerprinting is the number of signal strength signatures used as reference locations. This difference results in several advantages. To explain, a concrete scenario will be used to exemplify these differences.

Take, for example, a company with 3000 employees where each has a 10 square meter office, then the company is likely to have total area of about 60000 square meters (assuming additional space for hallways, conference rooms, etc.). For a typical position determination system based on location fingerprinting, the distance between samples could be 1 meter.

During the training phase 60000 samples will have to be collected, which corresponds to a considerable amount of work. If 30 seconds is spent on each location, just collecting the training data will require 500 hours. An automated way of performing the training would therefore obviously be preferred, but as the availability of commercial solutions to automate the data collection is very limited this is not necessarily feasible.

If only 4 signals are stored at each location and each access point is identified by 4 bytes, the training data of the above samples would be about $4*4*60000 = 960000$ bytes, which is just below 1 Mb. This assumes using the 24 bottom bits of the MAC plus a byte to identify the vendor (assuming there are less than 256 different access point vendors at a site, which is most likely the case).

With the same assumptions, what would be the effect of using the solution proposed in this thesis? To begin with, the number of samples would be reduced to only 3500, assuming one sample is taken at each employee's desk and in addition at 500 other locations of particular interest. Comparing with 60000, the training is now more easily manageable, and it could probably even be performed by the employees themselves.

For storage, the training set was kept using all 6 bytes of the MAC (this is the case of the solution implemented by the author). The storage would then require

$4*6*3500 = 84000$ bytes, which is an order of magnitude smaller than if a complete sampling grid would have been used.

It is true that handheld devices have increasing capabilities - and that either of the above data sets are particularly big. However, two things must be remembered. First, when performing the actual position determination on the client device, the idea is for the data set for a given area to be downloaded on request, for example automatically when the user enters a new building. This means that a smaller data set will result in quicker download of new training data (although one could also consider downloading only a subset of the data).

Second, the localization algorithm is intended to be run as a background process. Depending on the computational complexity of the algorithm the position determination will require more or less computational resources. Since the proposed algorithm takes linear time in relation to the number of locations and uses hash tables to store the data structures, no significant difference in execution time can be expected.

10.2 Implementation design

10.2.1 Prerequisite

As previously described, my software for position determination will be part of a system providing context dependent services. The existing system provides, among other things, a framework for sending messages between the different clients and a context server.[1]

In brief, clients report changes in their context to the server. For example, location reports keep the server updated about the locations of participating clients. Meanwhile, clients can define contexts for which certain events should be triggered. The server maintains the status of a client's current context and the triggers defined. This server is also responsible for performing the necessary calculations in order to send messages as a part of these context-aware services.

A prerequisite, for my work, is to integrate the position determination software that I have developed with the software architecture described above. Nevertheless, the localization software must be autonomous and independent enough to allow future evolution of the software.

With the addition of the localization software, the total number of classes and packages in the Java project grew to about twice the original size. In order to maintain good software design, the architecture was therefore restructured. The new design will be described below.

[1]The framework and server software used during this thesis were proprietary. However, Alcatel participated in several national and European projects (such as Mobilife [93], SPICE, S4ALL and Safari) in the context of which [94] and [95] were published. These describe some of the framework.

10.2.2 Client applications

In the context of this project, three different clients have been developed:

- Positions - for a training phase
- Position Scanner - to determine the current position of a device
- Particle Filter Scanner

The first two clients are a part of the position determination method based on finger printing. Positions is used to define the fingerprint for points of interest. Position Scanner is used to determine current position, based on the values recorded by Positions.

Particle Filter Scanner exploits the particle filter based method for position determination.

10.2.3 Shared files

Many of the files developed for this project are used by more than one of the three applications above. The Java project is organized so that these classes are separated from the client software.

10.2.4 Java class overview

Here follows a brief presentation of the most essential Java classes in this position determination project. To give an overview, only the most essential functionality will be mentioned here.

- **Access Point**
  Represents a wireless access point, including its SSID, position, and signal strength characteristics. Methods include computing the distance from this access point given a signal strength, and vice verse.
- **Registered Access Points**
  Maintains a set of Access Points. Provides methods for adding and retrieving an access point with a given SSID from this set.
- **Measurement Set**
  Contains a history of signal strength measurements with a certain type, such as Bluetooth or Wi-Fi. Provides methods for adding new observations and for estimating the most probable signal strength.
- **Potential Position**
  Represents for the fingerprinting based application a possible location. It defines the location's name and/or coordinates and contains a set of all access points within reach. With each access point is associated its signal strength

at this location. Provides methods for getting and setting the signal strength for a given access point.

- **Position Collector**
  Helper class used to create or update a given Potential Position with signal strength values collected over a certain time.

- **Poller**
  This class manages the scanning for nearby access points. To do this it provides methods that can be used independent of the wireless signal type. For each combination of platform and hardware type, one need to implement the core scan functionality, which is hardware dependent.

  The class provides a method for registering a signal strength observation for a given access point. Another method allows enumerating all registered access points as well as an estimation of their signal strength.

- **Position Determinator**
  Given a set of Potential Positions and a set of access points associated with their signal strengths, this class calculates which of the Potential Positions is most likely to represent current location.

10.2.5 Functional overview

- **Training**
  Register points of interest. In other words, define for each location a Potential Position with a Measurement Set for each access point within range.

- **Localization**

1. For each of the access points within range, determine the differences between current and the registered signal strengths for all Potential Positions.

2. Find the Potential Position with the smallest sum of the calculated differences. This is the most likely location since its measurements correspond the most to current measurements.

10.2.6 Single radio scanning

To limit the complexity, early versions of the software only supported scanning using one radio technology. To facilitate reading this report, this section is also based on use of only a single type of radio.

Fundamental to the position determination software is obviously the collection of up to date signal strength measurements concerning nearby access points. This is the responsibility of a separate thread, having its core functionality defined in Poller.

As indicated by its name, the thread will on a regular basis poll the lower level drivers to get the latest scanning result. We refer to this result as an observation. Registering the observation consists of adding it to the Measurement Set with its associated SSID. On request, the thread should provide an estimate of the signal strength for a given access point.

10.2.7 Access point information storage and access: implementation

Implementing the storage and access of access points in an efficient way is of great importance. If the interval between the scans is 200 milliseconds and 15 access points responds on average to a scan, then an average of 75 observations will be added every second. The time required to add these measurements is not significant, but it is still the most frequent operation.

A hash table is used as a data structure, resulting in several advantages. First, using the SSID as key allows easily eliminating duplicates. When adding a new observation one checks for the existence of this SSID, which can be done in constant time, before either instantiating a new Measurement Set or adding the observation to an existing observation set. Second, given a SSID, its signal strength estimate can be accessed in constant time when requested.

For the localization method based on fingerprinting one must be able to iterate over all registered access points. This was implemented using two methods, the first initiates an enumeration of the hash table keys, and the second returns the next key in the sequence. One can then iterate through the access points by getting the next key before accessing the related signal strength.

10.2.8 Access point validation

Early tests using 802.11b showed that devices not flagged as a master were regularly being registered. This is naturally undesirable for position determination which wants to use the idea of fixed devices as reference points. As a consequence a validation phase was introduced, filtering out access points without the master flag set. To function properly, this assumes mobile devices, i.e. laptops etc., are not configured to be master.

The validation of access points was later taken even further, after some access points were shown to be noticeably less reliable than others, for example those belonging to a different floor in the building. To be able to fully control which access points should be taken into account, a configuration file can be created. At program startup, the list is loaded and henceforth used for validation if desired.

This is an obvious limitation since a floor change requires a program restart. Future versions of the software should support dynamically loading a new set of valid access points. The load could for example be activated when none of the scan responses is above a certain limit.

10.2.9 Position determination

Assuming the fundamental functionality of getting a signal strength estimation for a given access point is in place, how could this information be used to determine the device location according to an approach based on fingerprinting?

To begin with, a number of Potential Positions must be registered. This is done by keeping the device stationary for a certain time at locations of particular interest, allowing it to collect and store signal strength estimations for all access points within range.

When training is done, the actual position determination can take place. Determining the current location consists of evaluating, for every Potential Position, the divergence between the trained signal strength observation and the most recent estimate - for each access point currently responding. This first step results in a two dimensional data set whose width and height correspond to the number of potential positions and the number of access points.

The most likely location is the one corresponding to the Potential Position with the smallest sum of the access points' absolute differences. If this result is not a singleton, the first implementation chose arbitrarily a location from this set (something which was later improved). We refer to this Potential Position as the Primary result.

Note that during the tests performed in the context of this thesis, all available access points were used. One would probably want to limit the search at each location to for example the 4 access points, and consequently reducing the size of the data set and the number of iterations in the localization algorithm. If the access points to take into account are well chosen this should not reduce the confidence of the result, since all of the access points unlikely well characterize the location.

10.3 Features and improvements

10.3.1 Consider the number of access points

The initial solution tends to discriminate in favor of locations with fewer responses, which is natural since more terms will result in a bigger sum. The algorithm was consequently improved to take into consideration the number of access points for each location.

10.3.2 Estimate location certainty

The knowledge of the system is limited to the potential locations that have been registered, and the result will also be limited by this set of potential locations. To make up for this, the certainty of a particular result is estimated before considering it as the most probable location.

The certainty estimation has two criteria:

- The total divergence for the primary location must not exceed a certain limit.
- The difference between the total divergences for the primary and secondary results must exceed a certain percentage of the primary result.

In other words, this check prevents the algorithm from returning a result where the actual and expected signal strengths differ significantly, or where the primary results does not enough outperform the secondary.

These limits were predefined according to the required confidence of a result and as a function of the environment. One could also imagine a more dynamic solution where the limit is computed from the combined variance of the access points within range of the location.

10.3.3 Measurement evaluation

As indicated the idea of a Measurement Set was to maintain a data structure representing the history of signal strength observations for access points. It is implemented as a circular list of a predefined size to which every new observation is added. When requesting the signal strength for an access point, a function is applied to the history to estimate this value.

10.3.4 Eliminating old measurements

What happens if an access point is correctly registered, but subsequently stops being relevant, i.e. when the device leaves its range? As it will no longer respond to scanning attempts, its signal strength measurement will no longer be updated, and the resulting estimate risks being outdated.

The proposed solution to the above problem consists of associating a time stamp with each Measurement Set. To keep track of when the last observation took place, adding a new observation results in a timestamp being added to this measurement. When an estimate of the signal strength is requested, the timestamp is compared to the current time in order to determine if it is valid or not.

As described above, the current implementation produces a single result. For future versions, one could consider a result based upon weighting the signal strength estimate according to the timestamped observations.

10.3.5 Out of range handling

When an access point is not in range there are two possibilities:

1. the access point is unknown, or
2. the access point is known, but is no longer considered to be valid.

To avoid using this invalid result one method is to simply ignore it. A better solution, however, is to exploit this so called *negative* information that indicates the absence of a specific access point.

To handle the out of range case, a constant is used to indicate its signal strength: NON_RESPONDING_RSS. Moreover, this constant is not only used to indicate the signal strength of access points that are out of range. The reason is that the signal strength measurement was found to be more unreliable in border regions, often fluctuating between responding and not. Therefore, a global signal strength threshold was introduced, causing all measurements below a certain level to be considered as out of range.

10.3.6 Different ways of evaluating the history

As previously indicated, a function of choice will be applied to the observation history of the Measurement Set in order to evaluate these measurements and to compute an estimate of the signal strength. Multiple such functions were developed for two reasons. First, the most accurate function had to be found. Second, once multiple observation types were supported there was a need to adapt the evaluation method according to the type of observation.

- **Average**
  As the name says, calculates an average of all measurements in this set.

- **Min Average**
  Calculates the average of the X strongest measurements, where X can be for example five.

- **Max Hit Rate**
  Estimates the signal strength based on the frequency of the different values. Does not return the value with the greatest frequency, but the first value with a frequency above a certain predefined limit, starting from the strongest values.

- **Binary value**
  Result is either an indication of the presence or the absence of the given access point.

10.3.7 Importance factor

So far, when combining the position determination results for the different wireless interfaces they have been treated as having equal importance. However, one should ask if their level of confidence is the same or not.

The position determination system was extended to be able to assign an importance factor to the signal strength of an access point. In other words, each time the difference between the expected and the actual value is calculated, the corresponding weight must be applied.

As proposed, one way of weighting the result of an observation is to consider its type. For example one could assign a higher importance factor to a result derived from an 802.11b observation than from a Bluetooth observation.

A more dynamic way of weighting the results is to consider how much the signal strength of a given access point varies over time. For example, if the variance of an access point is high, it could be considered with little importance. One could also compare the variance of the expected and actual signal strength measurements, and use it evaluate the likelihood of a potential position - in a way similar to what is done for the signal strength. This was no further investigated.

10.3.8 Missing scan responses

Increasing the scan frequency, performing up to four scans per second, will typically result in an improved result since fluctuations are less likely to incorrectly bias the result. In the case of a 802.11b interface in a laptop, the cost of a scan is low. As the initial tests were conducted using WLAN, a rather high scan frequency was used.

At a later stage, when testing the position determination using a Pocket PC the limitation of such a handheld device became evident. Tests showed that the contextual services linked to a certain location were for some reason regularly not triggered. This occurred even though the user device had correctly been able to determine and locally report its position.

The source of the problem was that the wireless interface during the scanning periods became unresponsive to handling other requests. It therefore failed to send its new location to the contextual services server.

For the problem to be confirmed, a workstation equipped with a wireless interface was used to test the responsiveness of the handheld device using the Ping command. The distance between the two devices was about 1 meter, and they were using ad hoc mode as to have a direct connection and eliminate any possible influence of a busy access point.

Here follows an extract of the result of pinging the PDA every 1000 ms.

```
Reply from 192.168.0.4: bytes=32 time=95ms TTL=64
Reply from 192.168.0.4: bytes=32 time=79ms TTL=64
Reply from 192.168.0.4: bytes=32 time=111ms TTL=64
Reply from 192.168.0.4: bytes=32 time=119ms TTL=64
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.0.4: bytes=32 time=107ms TTL=64
Request timed out.
Request timed out.
Reply from 192.168.0.4: bytes=32 time=70ms TTL=64
Reply from 192.168.0.4: bytes=32 time=107ms TTL=64
Reply from 192.168.0.4: bytes=32 time=110ms TTL=64
Reply from 192.168.0.4: bytes=32 time=98ms TTL=64
Request timed out.
Reply from 192.168.0.4: bytes=32 time=110ms TTL=64
```

```
Request timed out.
Request timed out.
Reply from 192.168.0.4: bytes=32 time=80ms TTL=64
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.0.4: bytes=32 time=107ms TTL=64
Request timed out.
Reply from 192.168.0.4: bytes=32 time=80ms TTL=64
Reply from 192.168.0.4: bytes=32 time=80ms TTL=64
Reply from 192.168.0.4: bytes=32 time=109ms TTL=64
Request timed out.
Reply from 192.168.0.4: bytes=32 time=111ms TTL=64
Request timed out.
Reply from 192.168.0.4: bytes=32 time=105ms TTL=64
Request timed out.
Request timed out.
Request timed out.
Reply from 192.168.0.4: bytes=32 time=111ms TTL=64
Reply from 192.168.0.4: bytes=32 time=95ms TTL=64
```

Each line shows the result of an ICMP echo request, which represents a rather light load, but is still a good indicator of accessibility. While irregular, one can still distinguish periods where the device has more frequently failed to reply, even though the conditions must be considered nearly ideal, considering the distance between the devices. With the delay between each ICMP request set to 1 second (default), one can nearly distinguish that the scan interval was set to 10 seconds, which it was.

To be able to better evaluate the result, a corresponding experiment was performed with a laptop instead of the PocketPC. In a similar way, a direct connection was used with the two devices in ad hoc mode. The laptop revealed no such problem, in other words, all requests were replied to. In fact, this was also the result when halting the scanning thread on the handheld device, i.e. all the ICMP echo requests were replied to.

The conclusion is that the handheld device is not able to perform the scanning in parallel with other tasks. The reason is probably the different models for thread execution. The kernel is non-preemptive, and if the scanning is controlled by the CPU rather than a micro controller on the WLAN card (as in the case of a workstation WLAN card), the scanning thread may not give up the CPU. This will stop other tasks from being executed, even those to send and reply to ICMP echo requests.

Note that in the case of an access point the mobile device could explicitly indicate that it is sleeping while scanning other channels - thus avoid missing so many packets as the packets could be buffered at the access point. Moreover, since the link is using another frequency during this scan the radio would need to be set back to the channel to be used to receive the ICMP echo request in order to hear it.

10.3.9 Adaptive scan delay

To address this problem, an adaptive scan delay was introduced, in comparison to the previous delay that was static. The scan delay ought to increase for stationary devices and decrease for faster moving devices since more observations are needed for the later to determine its new position. The problem in the adaptive case is to determine the velocity of the device, in order to set the time for collecting new data to be sufficiently short to learn of the new position, but not so short as to take too many resources from other tasks.

To keep the solution simple, the adaptive scan delay was limited to two modes - one for moving and one for not moving. A device is likely to be moving if the change in signal strength for the surrounding access points is large. Note that the change in signal strength could also be due to changes in the environment, which would result in additional scans. To be able to track the estimate for a given access point, a variable containing the previous signal strength estimate is stored as a reference. When determining the new scan delay, the difference between the previous and current estimate for all access points is calculated before summing up the absolute differences. If this sum exceeds a predefined difference, the device is considered to be moving. This threshold was empirically set to 3.0 and did not seem sensitive.

When the device is judged not to be moving, the device software enters sleep mode with a minimum scan rate. Otherwise the standard scan rate with shorter delay is used. A delay of 5 seconds during sleep mode was found appropriate.

The scan delay will be reevaluated in two cases:

- at every scan when in sleep mode, or
- at every x:th scan when in standard mode, where x is a number that was experimentally determined to 5.

10.4 Multi radio scanning

So far, in this part of the report, we have assumed the use of one wireless interface. Nevertheless the objective was always to allow multiple types of radios to be used to collect observations. This part of the thesis explains how to add new wireless interfaces to the localization system and how their observations are exploited. This is based on the assumption that the interface returns a numeric signal strength observation.

An initial implementation evaluated the different position estimates in a separate process for each wireless interface, before joining these different results in order to draw a conclusion. The architecture was soon enhanced to facilitate the integration of new observation types.

The idea is for the localization algorithm to function in a transparent way for the different interfaces. In other words, it should not need to take into consideration whether a particular signal strength estimate originates from the Poller of a Bluetooth or a IEEE 802.11b interface.

The proposed solution unitizes a shared data structure for the different observation types, so that a Bluetooth and a 802.11b observation can each contribute to a Measurement Set. The Position Collector and the Position Determinator programs can therefore iterate over the scan result independent of the observation types. During the position determination it will look for a Measurement Set with the given identity, and if the access point[2] is known, expected and actual signal strength for this type will be compared.

In order to add a new observation type, it is sufficient to implement a new thread the main Poller class. The new (sub) class must be instantiated and started as a new thread to allow simultaneous scanning.

Depending on the new observation, one could improve the position evaluation by adjusting how the expected and the actual signal strength estimates are compared for a given observation type.

[2]Note that *access point* refers to any type of reference point, such as 802.11b or a Bluetooth beacon.

Chapter 11

Evaluation

11.1 Prerequisite

In order to more easily evaluate the performance of the position determination system, some utilities dedicated to testing have been developed.

11.1.1 Simulated Poller

A new type of Poller was implemented without being related to a particular wireless interface. Instead, it is capable of sequentially reading signal strength observations from a file before reporting them, and in this way simulate the scanning of one or multiple interfaces in a device.

The observations file defines a signal strength trace. Each line in the file corresponds to an observation and has the following syntax:

```
<scan number>;<access point id>;<signal strength>;<observation type>
```

For example, the following observation belongs to the 22nd scan and reports that a device (access point) with a MAC of 00:12:a9:08:79:52 is of type 1 (defined to be 802.11b) and has a signal strength of -79 dBm:

```
22;00:12:a9:08:79:52;-79;1
```

Note that the scan number is used as a timestamp of the observation. As we know the interval between scans we can then determine when the observation was reported in relation to the first scan.

11.1.2 Trace Writer

To be able to create an observation trace a new method was needed in the main Poller class. When this feature is activated, it writes observations to a file one by one (independent of their type) as they are added by one of the Poller threads, hence creating a trace file to be read by the Simulated Poller.

11.1.3 Analyzer

This utility was written with the objective to help analyze signal strength variations. Just like the Simulated Poller, the Analyzer takes its input from an observation trace. It processes each observation and stores them in a spreadsheet compatible format grouped by their identity, thus making it possible to evaluate the evolution of the signal strength over time, or more exactly, as it changes from scan to scan.

The Analyzer was subsequently improved to also note the absence of signal strength observations for a certain access point and relate this to a particular scan. The reason is obviously to avoid having the observations shifted in time, which was the case in the first version of the program.

11.1.4 Trace Creator

As described above, the main Poller class was augmented with a method to create a signal strength trace while performing live scanning. Using the Trace Creator, a trace file could be synthetically generated.

The functionality consists of reading a set of Potential Positions and based on the registered signal strength estimates for this position generate trace entries. The resulting trace corresponds to a walk between the locations as if the movement between the different locations was instant and as if all the measurements were perfectly reproducible.

Note that this lack of realism, in theory, have no effect on the static performance of the position determination. However, it will speed up the convergence since the observations corresponding to the reference point will be matched identically.

To make the observation trace more realistic, Trace Creator has a configurable mechanism to distort the observations. It does so by applying a Gaussian random variable to the recorded signal strength measurement before reporting it.

11.2 Signal strength variations

This section studies how the signal strength of a device varies over time.

11.2.1 Overview

The test bed is the same as the one used for determining the reference signal strength in Section 8.4.2.

For the experiment to be as representative as possible, three different cases will be evaluated:

1. Line-of-sight (LOS)

2. Non-line-of-sight, with a metal door permanently obstructing the signal (NLOS)

3. LOS with occasional obstruction (Mixed)

For each case, 50 scans will be performed.[1]

11.2.2 Test 1-3

Figure 11.1-11.3 depicts the frequency of the different signal strength observations for the three scenarios.

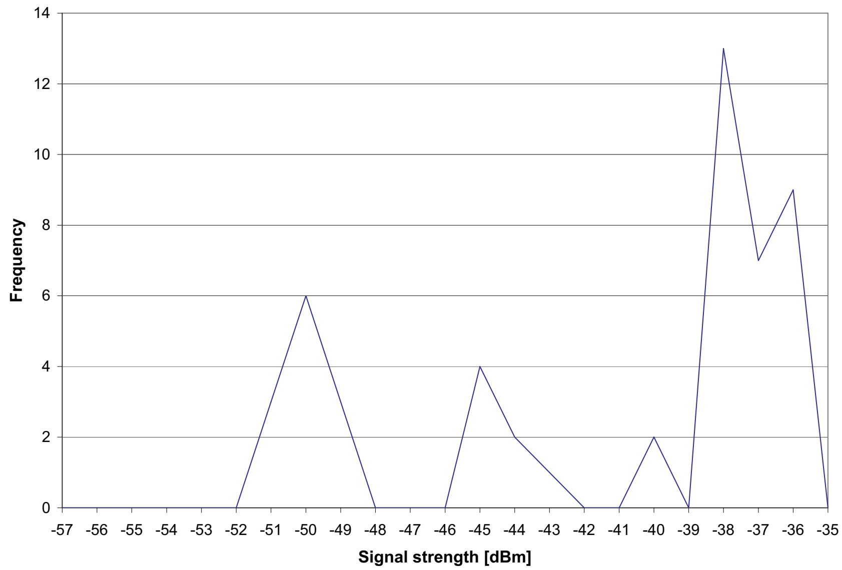

Figure 11.1. Line-of-sight measurements: showing the number of observations per signal strength.

11.2.3 Analysis

There is clearly a big difference between the minimum and the maximum signal strength that have been registered. For example, in the line-of-sight case the weakest signal was -51 dBm and the strongest -36 dBm, which is a difference of 15 dBm.

We also note a significant difference between the three scenarios. The number of distribution peaks as well as their location vary greatly.

Note that these tests were performed several times to assure that these measurement observations are representative. There were differences between the peak signal strengths and the signal strength scatterness, but they had all similar characteristics.

The following table summarizes the three tests using the standard deviation.

[1] Why 50 samples were used will be explained in the following section.

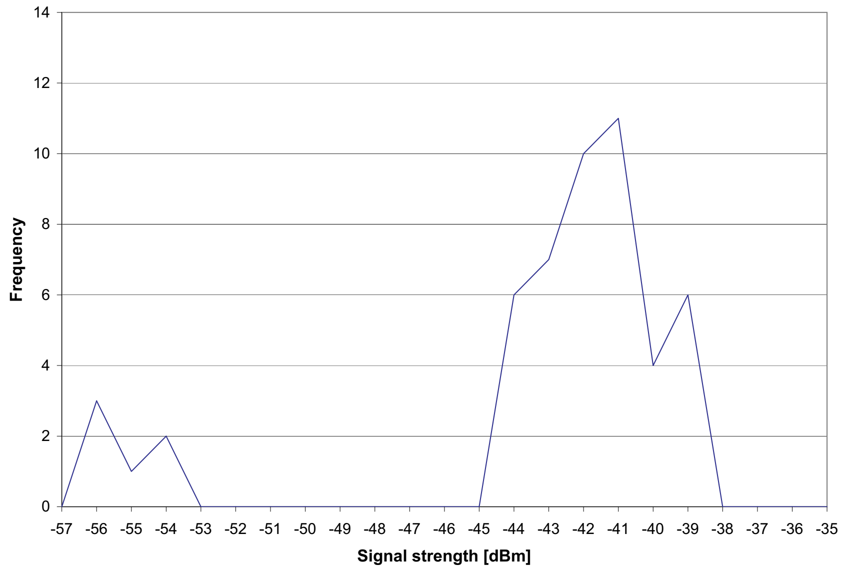

Figure 11.2. Non-line-of-sight measurements: showing the number of observations per signal strength.

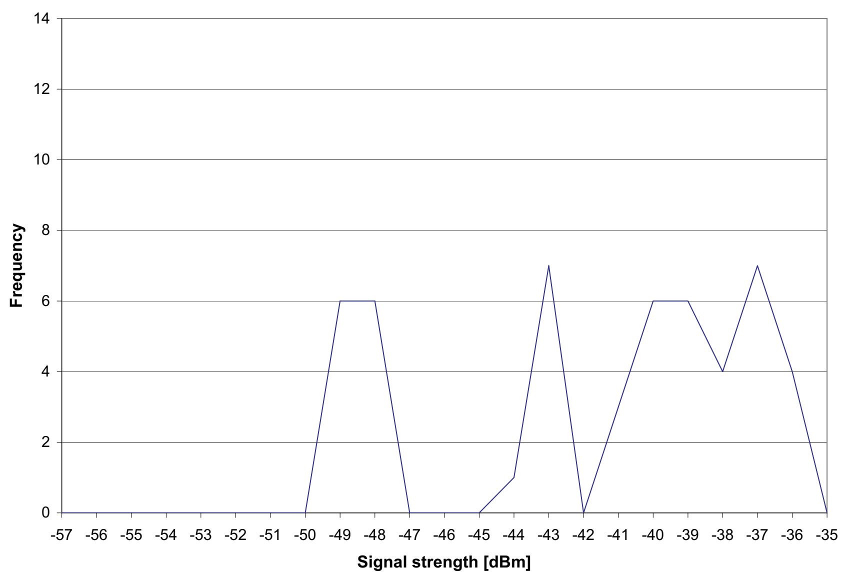

Figure 11.3. Measurements for line-of-sight with occasional obstruction: showing the number of observations per signal strength.

	LOS	NLOS	Mixed
Std deviation [dBm]	5.6	4.7	4.5

Table 11.1. Test 1-3: Standard deviation

11.3 Sliding window signal strength extraction methods

Several ways of extracting the most correct signal strength from the observation history have been implemented. The difficulty is illustrated by the great variations as shown in the previous section. This section evaluates the performance of the proposed methods.

11.3.1 Overview

We use the three scenarios and measurement data from Section 11.2. In order to evaluate the different cases and methods, we compare with the reference value that was determined (for this location) in Section 8.4.2.

The duration of each scenario was 50 scans. This value was chosen while having in mind that it is reasonable for the entire observation history to be replaced after 10 seconds, which with 5 scans per second corresponds to 50 scans. The value of 10 seconds was selected for a good balance between responsiveness and certainty.

11.3.2 Test 4

- Setup

  First, the average of the history was calculated.

- Result

	LOS	NLOS	Mixed
Signal strength [dBm]	-41.4	-43.2	-41.6

Table 11.2. Test 4: signal strength extraction: average

11.3.3 Test 5

- Setup

  After having noticed that the signal strength measurements focused on the most frequent values, an average based on only the strongest values were computed. The number of values to be taken into account can be varied. By studying the observation distributions in section 11.2, it seems that 4 values is suitable in order to neither let strong measurement outliers nor a weak average influence the estimate.

- Result

	LOS	NLOS	Mixed
Signal strength [dBm]	-36.0	-39.0	-36.0

Table 11.3. Test 5: signal strength extraction: limited average

11.3.4 Test 6

- Setup

  Next, a quite different approach was evaluated, based on using the most frequent signal strength value.

  After noticing that the signal strength distribution quite often was multimodal, for example due to an obstacle temporarily obstructing the signal, the algorithm was improved to report the value corresponding to the mode with the strongest observations. To find it, a frequency threshold was introduced, and the first signal strength (starting from the highest observed signal strength) with a frequency above this threshold is the estimate.

  Visually studying the frequency graphs for the three scenarios it seemed a frequency of 6.5 would give the desired estimate. This corresponds to 13 percent of the total number of measurements, and was used for the test. Note that this threshold is the most appropriate for the these three scenarios, but probably not for all other scenarios. The result is also sensitive to changing the threshold, since a small increase or decrease may result in a different frequency peak meeting the requirement.

- Result

	LOS	NLOS	Mixed
Signal strength [dBm]	-36.0	-41.0	-37.0

Table 11.4. Test 6: signal strength extraction: frequency

11.3.5 Analysis

The following table summarizes the result of the methods for extracting an estimate of the signal strength. For each method applied to the observation history, and each of the three scenarios, it shows (with one significant decimal) the divergence from the reference value that was determined to be -36 dBm in section 8.4.2.

	LOS	NLOS	Mixed
Average [dBm]	-5.4	-7.2	-5.6
Limited average [dBm]	0	-3	0
Frequency [dBm]	0	-5	-1

Table 11.5. Summary of signal strength extraction functions, showing the difference from the reference value -36 dBm.

Assuming that the measurements are distributed in a similar way as during the determination of the reference signal strength, the average method will report a value higher than the reference, which it also did during the tests based on 50 observations. Unsurprisingly, in the non-LOS situation there is a noticeable difference between the estimate and the reference value, because the observations are noticeably weaker than the reference value.

The method based on the maximal frequency performs better than the average method, and succeeds in reporting one of the stronger values among the collected measurements even though a large number of observations were weaker.

In the three settings, the most correct result is provided by the method of calculating the average over a limited data set - Limited average. In the case of non-LOS the majority of the observation measurements, with weaker signal strength, will be ignored, and the value reported will be based on fewer but stronger observed signal strengths.

Its ability to report a close to correct estimate becomes clear in the case where a temporary drop in signal strength occurs, for example when a walking person momentarily obstructs the signal propagation. In many real case scenarios this is likely to happen regularly, so being able to handle it properly will likely result in a better estimate of the user's location. If the obstacle is not present for too long, uninfluenced and stronger signal strength measurements will prevent the obstructed values from influencing the resulting value.

11.3.6 Discussion

One might ask why it is not a good idea to simply use the strongest signal strength as the most correct. This method would indeed have performed flawlessly on the data collections above. However, as proved by the much larger data collection depicted in 8.3, occasional measurements will undoubtedly appear that are much stronger than the reference value, for example due to the superposition of RF waveforms. By taking the average of only the strongest measurements the effect of outlying values will be minimized.

The three scenarios above have been chosen to in best way cover possible scenarios in practice. However, they obviously do not represent all possible scenarios; hence it is difficult to draw any general conclusions about its performance. Note that performance refers to consistency rather than to accuracy.

11.4 Seven points of interest

11.4.1 Overview

For all tests in this section, the following steps were followed

1. Seven points of interest were selected in the test area
2. A Potential Position was registered for each of the seven locations
3. The Trace Creator was then run using these Potential Positions, to simulate a device moving from one position to the next with a given interval.

4. The location determination algorithm was applied

The seven points of interest, or potential positions, were defined in a two dimensional space with a distance of about three meters in-between most of them. Each location was marked and numbered in order to be able to reproduce a given test. Their fingerprints were defined by sampling measurements during 20 seconds. Note that with the proposed solution for location fingerprinting, this training phase took only 4 minutes.

To test the performance of the system the localization algorithm was applied to the trace by using the Simulated Poller. In order to simulate the variability of observations, the configuration of the Trace Creator was altered. In addition, the localization parameters were varied to see the impact of each one of them.

11.4.2 Test 7

- Configuration:

Applied variance	History size	Selected function
None	35	Limited average

Table 11.6. Test 7: configuration

A signal strength evaluation method of Limited average means that the average was calculated on a set of the strongest observations that had been reported.

- Result: see Table 11.7

- Comment

The first test shows the basic performance of the position determination algorithm.

The first column in the table shows the actual location of the device at a given location update, which is indicated by the second column. With 10 scans per update we conclude that a total of 380 scans took place, and that the device spent 50 scans at each location.

We can see the location which has been determined by the algorithm in the third column. Comparing it with the first column we see that it is been shifted a number of scans. This delay between the actual and the estimated location is natural considering how the locations are evaluated.

Note that following the first four updates no location estimate could be provided. This is an intentional feature to avoid returning a location based on insufficient information.

Real location	Update	Estimated location
1	1	-
1	2	-
1	3	-
1	4	-
1	5	1
2	6	1
2	7	1
2	8	1
2	9	1
2	10	1
3	11	2
3	12	2
3	13	2
3	14	2
3	15	2
4	16	3
4	17	3
4	18	4
4	19	4
4	20	4
5	21	4
5	22	4
5	23	5
5	24	5
5	25	5
6	26	5
6	27	5
6	28	6
6	29	6
6	30	6
7	31	6
7	32	6
7	33	6
7	34	6
7	35	6
	36	6
	37	7
	38	7

Table 11.7. Test 7: position determination result

11.4.3 Test 8

- Configuration:

Applied variance	History size	Selected function
± 3	variable	Limited average

Table 11.8. Test 8: configuration

A variance of ± 3 indicates that the observation values have been distorted by adding a Gaussian with variance three.

- Result: see Table 11.9

- Comment

  This test indicates two things. First, it shows the impact when changing the size of the history in each Measurement Set. We see that a smaller history results in a quicker response when it comes to finding that the location has changed. With a history size of 50 it took 7 location updates to notice the move from location 1 to location 2. This should be compared with 3 updates when the history size is set to 5 or 10.

  Second, there are occasional incorrect estimations for location 5. With a history size of 50, this location never occurs. With a history size of 40 or 30 it occurs at the correct location, but in addition the location is incorrectly determined. Notice that for smaller history sizies location 5 is correctly found.

11.4.4 Test 9

- Configuration:

  The variance of $+6$ indicates that every observation has been added a Gaussian with the variance of six after getting its absolute value.

- Result: see Table 11.11

- Comment

  The test result shows no significant overall change when comparing with Test 8 in section 11.4.3. However, we notice the difficulty of correctly determining current position for a very small history size of 5.

11.4.5 Test 10

- Configuration:

  The different between this simulation and that in section 11.4.4 is that the function applied to the observation history is changed to a simple average calculation.

		Estimated location History size					
Real location	Update	50	40	30	20	10	5
1	1	-	-	-	-	-	-
1	2	-	-	-	-	-	-
1	3	-	-	-	-	1	1
1	4	-	-	-	1	1	1
1	5	-	1	1	1	1	1
2	6	1	1	1	1	1	1
2	7	1	1	1	1	1	1
2	8	1	1	1	1	2	2
2	9	1	1	1	2	2	2
2	10	1	1	2	2	2	2
3	11	1	2	2	2	2	2
3	12	2	2	2	2	2	2
3	13	2	2	2	2	2	3
3	14	2	2	2	2	3	3
3	15	2	2	2	3	3	3
4	16	2	2	3	3	3	3
4	17	2	3	3	3	3	3
4	18	4	4	4	4	4	4
4	19	4	4	4	4	4	4
4	20	4	4	4	4	4	4
5	21	4	4	4	4	4	4
5	22	4	4	4	4	4	4
5	23	6	6	5	4	5	5
5	24	6	6	6	5	5	5
5	25	6	6	6	5	5	5
6	26	6	6	5	5	5	5
6	27	6	5	5	5	5	5
6	28	6	6	6	5	6	6
6	29	6	6	6	6	6	6
6	30	6	6	6	6	6	6
7	31	6	6	6	6	6	6
7	32	6	6	6	6	6	6
7	33	6	6	6	6	6	6
7	34	6	6	6	6	7	7
7	35	6	6	6	7	7	7
	36	6	6	7	7	7	7
	37	6	7	7	7	7	7
	38	7	7	7	7	7	7

Table 11.9. Test 8: position determination result

Applied variance	History size	Selected function
+6	variable	min average

Table 11.10. Test 9: configuration

- Result: see Table 11.13
- Comment

  This test proved difficult for the localization algorithm. We can see that multiple incorrect locations are repeatedly proposed when the algorithm changes back and forth between locations.

11.4.6 Analysis

From Test 7 in 11.4.2 we can conclude that the algorithm performs well under these ideal conditions. There is a noticeable delay between a position change and it being reflected in the estimated location. However, Test 8 in 11.4.3 showed that the delay can be significantly reduced by decreasing the history size while maintaining certainty of the proposed location.

In 11.4.3 an interesting phenomenon occurs. We can see that location 4 was determined at the 18th update no matter what size the history was set to. The last update for which 4 was found was the 22nd scan, apart from one exception. Why did the decreased history size have no impact on location 4?

The explanation can be found by studying the trace diagram shown in Appendix A.1. It reveals that several access points gave a significant difference in signal strength between location 3 and 4, and 4 and 5. Because of these differences the algorithm managed to distinguish and determine location 4 within an interval of sufficiently few scans for the result to appear at the same location update.

The perhaps most interesting result can be derived by comparing the results in Test 9 and Test 10. Recall that the only difference between the them is in which way the estimated signal strength is calculated. The result of Test 9 is clearly much better than of Test 10, even though the difficulty of the two tests is identical, as the circumstances are the same. We draw the conclusion that the Limited average function outperforms the pure Average function not only in the case of a simple signal strength comparison but also in the case of actual position determination.

11.5 Visual position determination in an office environment

11.5.1 Overview

This test aims at testing the position determination in an office environment depicted in Figure B.1. The test scenario is very authentic, as it consisted of walking

		Estimated location					
		History size					
Real location	Update	50	40	30	20	10	5
1	1	-	-	-	-	-	-
1	2	-	-	-	-	-	-
1	3	-	-	-	-	1	1
1	4	-	-	-	1	1	1
1	5	-	1	1	1	1	1
2	6	1	1	1	1	1	1
2	7	1	1	1	1	1	1
2	8	1	1	1	1	2	2
2	9	1	1	2	2	2	2
2	10	1	2	2	2	2	2
3	11	2	2	2	2	2	2
3	12	2	2	2	2	2	2
3	13	2	2	2	2	3	3
3	14	2	2	2	3	3	3
3	15	2	2	3	3	3	3
4	16	2	3	3	3	3	3
4	17	3	3	3	3	3	3
4	18	3	3	3	3	4	3
4	19	3	3	3	4	4	3
4	20	3	4	4	4	4	4
5	21	4	4	4	4	4	3
5	22	4	4	4	4	4	3
5	23	4	4	4	4	7	5
5	24	5	5	5	5	5	5
5	25	5	5	5	5	5	5
6	26	5	5	5	5	5	5
6	27	5	5	5	5	5	5
6	28	5	5	5	5	6	6
6	29	6	6	6	6	6	6
6	30	6	6	6	6	6	6
7	31	6	6	6	6	6	6
7	32	6	6	6	6	6	6
7	33	6	6	6	6	6	7
7	34	6	6	6	6	7	7
7	35	6	6	6	7	7	7
	36	6	6	7	7	7	7
	37	6	7	7	7	7	7
	38	7	7	7	7	7	7

Table 11.11. Test 9: position determination result

Applied variance	History size	Selected function
+6	variable	average

Table 11.12. Test 10: configuration

around the floor among moving persons. Of the 7 access points used, 3 were mounted in a line in the main corridor (vertical on the figure).

1. I registered 16 Potential Positions - in the corridor and in the rooms along it - several placed with a distance of 3 meters in-between.

2. I then linked the position to a location on the map.

3. I then tested the system by moving between the different rooms in walking speed and noting if the correct location was determined.

First, tests were done using a notebook with an integrated 802.11b interface. The values of the minimum certainty parameter were set to 0.29 and 15 respectively (see 10.3.2 for more information on this parameter). The observation history was set to 35, the delay between scans was 200 milliseconds, and the delay between location updates was 1 second.

In a second phase of the test, a Bluetooth dongle was mounted in each room. As previously explained, the device discovery implementation only supported reporting the presence (or absence) of a given device.

11.5.2 Test 11

A scenario with 16 rooms were repeatedly executed in order not for temporary variations to impact the result. Most of the part, 12 or 13 were correctly determined depending on the circumstances. The variance of observations were higher than expected. No tour resulted in less than 11 or more than 15 correct locations. In other words, the certainty was about 80 percent.

When arriving at a new location, it typically took the system less than 3 seconds for the location to be updated. This means that early update cycles after the arrival did not result in a new location. Some locations in the corridor were normally updated within a second or even just before the location was reached.

11.5.3 Analysis

The locations used in this test correspond to the needs of position determination for context-aware services. As a correct location was reported 80 percent of the time with a confidence of 85 percent, the algorithm has in this test environment succeeded in meeting the requirements defined in section 2.4.

		Estimated location History size					
Real location	Update	50	40	30	20	10	5
1	1	-	-	-	-	-	-
1	2	-	-	-	-	-	-
1	3	-	-	-	-	1	1
1	4	-	-	-	1	1	1
1	5	-	-	1	1	1	1
2	6	1	1	1	1	1	1
2	7	1	1	1	1	1	1
2	8	2	2	2	2	2	2
2	9	2	2	2	2	2	2
2	10	2	2	2	2	2	2
3	11	2	2	2	2	2	2
3	12	2	2	2	2	2	2
3	13	3	3	3	3	3	3
3	14	3	3	3	3	3	3
3	15	3	3	3	3	3	3
4	16	3	3	3	3	3	3
4	17	3	3	3	3	3	3
4	18	3	3	3	3	3	4
4	19	3	3	3	3	3	3
4	20	3	3	4	3	4	4
5	21	3	3	3	4	3	3
5	22	3	3	3	3	3	3
5	23	7	7	7	7	7	7
5	24	3	3	3	3	5	5
5	25	3	3	5	5	5	5
6	26	3	5	5	5	5	5
6	27	5	5	5	5	5	5
6	28	5	5	5	5	4	5
6	29	5	5	4	4	6	6
6	30	4	4	6	6	6	6
7	31	4	6	6	6	6	6
7	32	6	6	6	6	6	6
7	33	6	6	6	6	6	6
7	34	6	6	7	7	7	7
7	35	7	7	7	7	7	7
	36	7	7	7	7	7	7
	37	7	7	7	7	7	7
	38	7	7	7	7	7	7

Table 11.13. Test 10: position determination result

Several times the desk at the opposite side of the corridor from the correct desk was indicated as the device location. The symmetry of the corridor structure clearly was a problem for this solution[2].

It turned out that the Bluetooth dongles could rule out some potential locations due to the absence of the Bluetooth dongle. However, when the distance between two Bluetooth devices was short enough for their signals to intersect the addition of Bluetooth dongles had no impact on the correct determination of a location.

11.6 Position determination in open space environment

11.6.1 Overview

This test aims at analyzing the performance of the position determination system in an open space environment.

The following steps were performed for the test.

1. Use the 7 registered locations from the test in 11.4.
2. Record a trace when walking between the registered locations.
3. Apply the localization algorithm on the trace using the 7 registered locations as potential positions.

A scan delay of 200 milliseconds was used and the size of the observation history was set to 20.

The live trace used for this test was recorded and can be found in Figure A.2 in the appendix.

11.6.2 Test 12

Table 11.14 shows the expected and the reported location over the 38 location updates.

11.6.3 Analysis

As shown, the position determination algorithm repeatedly reported an incorrect location. In fact, assuming the new location should be reported after a few scans delay, 15 of the updates have resulted in an incorrect location. This corresponds to 42 percent of the updates, which is not acceptable.

To get a better result, the size of the observation history was changed. However, this did not improve the result.

Consequently, we conclude that in this test environment, the location fingerprinting algorithm has failed to meet the requirements. The conclusion is that

[2] As in many real case scenarios, the access points had not been deployed with the intention of being used for position determination, and it was not in the power of the author to break the symmetry by adding another access point.

Real location	Update	Estimated location
1	1	-
1	2	-
1	3	1
1	4	1
1	5	1
2	6	1
2	7	1
2	8	1
2	9	4
2	10	4
3	11	1
3	12	1
3	13	1
3	14	7
3	15	4
4	16	3
4	17	3
4	18	7
4	19	3
4	20	4
5	21	4
5	22	4
5	23	4
5	24	4
5	25	3
6	26	7
6	27	7
6	28	7
6	29	5
6	30	5
7	31	4
7	32	4
7	33	4
7	34	7
7	35	7
	36	7
	37	7
	38	7

Table 11.14. Position determination in an open space environment

in an open space environment, the signal strengths of different locations are not discriminant enough to identify the location.

Chapter 12

Modified approach

As previously indicated, the position determination algorithm developed in this thesis took advantage of all available access point information in order to determine the current location. This means that each access point will equally influence the result location, independently of whether it actually contributes to distinguishing the location or not.

A modified approach is hereinafter considered. It consists of limiting the match process to the access points which best characterize a location. Here follows a general description of the method, before applying it to position determination.

12.1 Discriminant analysis

Condsider the outcome Y of a test which is thought to depend on parameters $x_1, x_2, ..., x_m$. Specifically, let Y represent the group belonging of an object with characteristics given by a set of observations named X. Generally, discriminant analysis is used to determine which variable X (or variables) discriminate between two or more groups.

Statistically, it consists of analysing whether there is a significant difference between groups with regard to the mean of one or several variables.

In the two-group case, discriminant analysis is analogues to multiple regression. They both aim at fitting a linear equation of the following type:

$$GroupY = a + b_1x_1 + b_2x_2 + ... + b_mx_m$$

where a is a constant and b_1 through b_m are regression coefficients. The larger a regression coefficient is the more it contributes to discriminating this group.

12.2 Classification

When the discriminant function has been determined the model can be adapted to only include the variables that most contribute to discriminating a group. A classification function can then be determined in the following form:

$$S_1 = c_1 + w_{i1}x_1 + w_{i2}x_2 + ... + w_{im}x_m$$

Here, the subscript i denotes the group, the subscripts 1,2,...,m denotes the m variables, c_i is a constant for the i:th group, w_{ij} is a weight for the j:th variable in the computation for the i:th group. During the classification process, a classification score S_1 is computed for each of the groups according to above by inserting the m observations.

12.3 Applied to position determination

To apply the above reasoning to position determination, let Y be a position which is characterized by a set of signal strength observations X. Each such observation corresponds to the signal strength of an access point and is said to be a feature of this location.

When performing the position determination, the signal strength observations are used with the classification functions in order to compute the likelihood of each position. This is done while taking into account the coefficient of each access point. The position with the highest score is the most likely position.

12.4 Advantages for position determination

This way of performing position determination brings two main advantages. First of all, we can expect a higher confidence in the result location. Second, and of less significance, the algorithm will faster be able to determine the most likely of a large set of potential locations, since the number of comparisons will be reduced.

Let us concentrate on the first advantage. Remember that all signal strength measurements come with a more or less important error (as explained in section 8.3). This means that if we return to a location with a known (previously determined) signal strength, we can not expect the signal strengths of the surrounding access points to correspond to the initial measurements. Comparing the expected and the actual signal strengths of all available access points, the algorithm risk not being able to determine the correct location since the signal strength of a large number of access points may be misleading.

In this improved approach to position determination the measurements will obviously be just as erroneous, but when limiting the match process to the sub set of the access points that best discriminate a location, it is more likely to be correctly evaluated.

In other words, only the access points with signal strengths that differentiate from other locations will be considered.

Chapter 13

Conclusion

Position determination is a service with numerous interesting applications. This paper focused on location determination to provide contextual services; instead of services based on the location itself. As a consequence, the precision of the location information is of less importance to the certainty of the location.

This chapter summarizes the conclusions from this project.

13.1 Summary

The following conclusions about wireless position determination are the result of this thesis project.

- Position determination based on wireless off-the-shelf products is promising due to their wide spread use. Unlike many other systems based on dedicated hardware, one can in many cases take advantage of existing access point infrastructure, which greatly reduces cost of deployment. In addition, services can potentially be used by the large number of devices already on the market.

- A solution for position determination based on location fingerprinting was chosen due to its simplicity. In it, locations are identified by a signature made up of the signal strengths of access points within range. During an initial training phase such fingerprints are registered, and when determining the location, present observations are matched against the training data.

- By limiting the sampling locations of the training to points of particular interest, the amount is reduced with more than an order of magnitude. Consequently, the time spent on training is drastically reduced, and the training data can easily be distributed. In addition, the match algorithm can with minimal time and resources determine sufficiently accurately the location some of the time.

- The wireless signals strength is rather unpredictable and sensitive to obstruction. In an environment with constant line-of-sight between the access point and a device, 50 samples were taken within a signal strength interval of 15 dBm. With a metal door in-between, the average fell about 2 dBm.

- A main difficulty is to estimate the signal strength that best reflects reality, given a very limited observation history. Computing the average of only a few of the strongest recent observations was found the best method.
- The proposed algorithm based on location fingerprinting also allows signal strength measurements from multiple wireless interfaces to be taken into account.
- Repeatedly testing the implemented algorithm in an office environment with 16 locations resulted in the correct desk being found in 80 percent of time.
- The ambiguity between two locations was successfully resolved by using a Bluetooth interface and integrating information about the presence of Bluetooth dongles. However, not enough tests were performed to draw any conclusion about using multiple wireless interfaces.
- Testing the algorithm in an open space environment with 7 potential positions showed that it in 42 percent of the cases failed to determine the correct location. We conclude that the signal strengths of the locations was not distinguishable for the proposed algorithm.
- Localization technologies based on dedicated or non-standard hardware have elsewhere shown very good performance. However, the author believes that methods based on off-the-shelf devices and wireless signal strength have a bright future due to its advantages.

13.2 Future work

Further testing must be done in order to evaluate how the proposed system performs while simultaneously using multiple wireless interfaces for position determination.

For a location determination system to be of commercial interest it must, as previously indicated, be usable on various devices. Something that has not been studied in this paper is the differences between these devices in terms of antenna output power and sensitivity. The author suspects that there are notable differences between devices. It would therefore be interesting to investigate the portability of signal strength based algorithms between different wireless devices.

Bibliography

[1] A. M. Ladd, K. E. Bekris, G. Marceau, A. Rudys, D. S. Wallach and L. E. Kavraki, "*Using Wireless Ethernet for Localization*, Conference on Intelligent Robots and Systems, 2002

[2] A. Ward, A. Jones and A. Hopper, *A new location technique for the active office*, IEEE Personal Communications 4(5):42-47, October 1997

[3] I. J. Cox, *An experiment in guidance and navigation of an autonomous robot vehicle*, IEEE Transactions on robotics and automation, 7(2):193-204, April, 1991

[4] N. B. Priyantha, *The cricket indoor location system*, Ph D thesis, MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA, 2005

[5] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, D. S. Wallach and L. E. Kavraki, *Practical robust localization over large-scale 802.11 wireless networks*, MobiCom '04, Sept. 26 - Oct. 1, 2004, Philadelphia, Pennsylvania, USA

[6] E. Elnahrawy, X. Li and R. P. Martin, *The Limits of Localization Using Signal Strength: A Comparative Study*, Department of Computer Science, Rutgers University, 2004

[7] H. Hashemi, *The indoor radio propagation channel*, Proc. of the IEEE, 81(7):943-968, July 1993

[8] B. Li, A. Dempster, C. Rizos and J. Barnes, *Hybrid Method for Localization Using WLAN*, Spatial Sciences Conference, Melbourne, Australia, 12-16 September, 2005

[9] A. Savvides, C. Han, M. B. Strivastava, *Dynamic FineGrained Localization in AdHoc Networks of Sensors*, International Conference on Mobile Computing and Networking, Proceedings of the 7th annual international conference on Mobile computing and networking, 166-179, Rome, Italy, 2001

[10] T. D. Hodes, R. H. Katz, E. Servan-Schreiber and L. A. Rowe, *Composable Ad-hoc Mobile Services for Universal Interaction*, Proceedings of the Third ACM/IEEE International Conference on Mobile Computing, Budapest, Hungary, September 1997

[11] M. A. Youssef, A. Agrawala and A. U. Shankar, *WLAN location determination via clustering and probability distributions*, PERCOM, Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003

[12] P. Tao, A. Rudys, A. M. Ladd and D. S. Wallach, *Wireless LAN Location-Sensing for Security Applications*, Proceedings of the Second ACM Workshop on Wireless Security, San Diego, CA, September 2003

[13] H. S. Cobb, *GPS pseudolites: theory, design, and applications*, Ph.D. dissertation, Department of Aeronautics and Astronautics, Stanford University, September 1997

[14] M. O. Kanli, *Limitations of Pseudolite Systems Using Off-The-Shelf GPS Receivers*, School of Surveying and Spatial Information Systems, University of New South Wales, Australia, December 2004

[15] N. B. Priyantha, *The Cricket Indoor Location System*, Ph. D. dissertation, Department of Electrical Engineering and Computer Science, MIT, June, 2005

[16] H. Stüttgen, *Context-aware applications: the next step for mobile services*, Network Laboratories, NEC Europe Ltd., 2004

[17] A. K. Dey, *Understanding and Using Context*, Personal and Ubiquitous Computing, Vol 5, No. 1, pp 4-7, 2001

[18] B. N. Schilit, N. Adams and R. Want, *Context-Aware Computing Applications*, IEEE Workshop on Mobile Computing Systems and Applications, December 8-9, 1994

[19] R. Want, A. Hopper, V. Falcão and J. Gibbons, *The active badge location system*, ACM Transactions on Information Systems (TOIS), 10(1):91-102, 1992

[20] C. Kee, H. Jun, D. Yun, B. Kim, Y. Kim, B.W. Parkinson, T. Langenstein, S. Pullen and J. Lee, *Development of indoor navigation system using asynchronous pseudolites*, 13th Int. Tech. Meeting of the Satellite Division of the U.S. Inst. of Navigation, Salt Lake City, Utah, 1038-1045, 19-22 September, 2000

[21] J. P. G. Sterbenz, R. Krishnan, R. R. Hain, A. W. Jackson, D. Levin, R. Ramanathan and J. Zao, *Survivable Mobile Wireless Networks: Issues, Challenges, and Research Directions*, WiSe'02, September 28, 2002, Atlanta, Georgia, USA

[22] B. Triggs, *Model-based sonar localization for mobile robots*, Robotics and Autonomous Systems, 1994, Vol 12, pp.173-186

[23] M. Drumheller, *Mobile robot localization using sonar*, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9(2):325–332, Mars 1987

[24] R. Want, A. Hopper, V. Falco, and J. Gibbons, *The Active Badge Location System*, ACM Transactions on Information Systems, vol. 10, no. 1, pp. 91-102, January 1992

[25] N. Adams, *An Infrared Network for Mobile Computers*, Proc. USENlX Mobile and Location-Independent Comp. Symp., Cambridge, MA, Aug. 2-3, 1993, pp. 41-51

[26] J. Krumm et al., *Multi-camera multi-person tracking for Easy Living*, in 3rd IEEE International Workshop on Visual Surveillance, Piscataway, NJ, 2000, pp. 3-10

[27] R. J. Orr and G. D. Abowd, *The Smart Floor: A Mechanism for Natural User Identification and Tracking*, in Conference on Human Factors in Computing Systems (CHI 2000), The Hague, Netherlands, April 2000, pp. 1-6

[28] S. T. Birchfield and D. K. Gillmor, *Fast Bayesian acoustic localization*, IEEE International Conference on Acoustics, Speech, and Signal Processing, Proceedings. (ICASSP '02), Volume 2, 2002, Page(s):1793 - 1796

[29] E. Mumolo, M. Nolich, G. Vercelli, *Algorithms and architectures for acoustic localization based on microphone array in service robotics*, IEEE International Conference on Robotics and Automation, Proceedings. ICRA '00, Volume 3, 24-28 April 2000, Page(s):2966 - 2971 vol.3

[30] IEEE-SA Standards Board, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band*, IEEE, 1999

[31] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, *The Anatomy of a Context-Aware Application*, In Proc. 5th ACM MOBICOM Conf., pages 59-68, Seattle, WA, August 1999

[32] A. Harter and A. Hopper, *A New Location Technique for the Active Office*, IEEE Personal Communications, 4(5):43-47, October 1997

[33] P. Bahl and V. Padmanabhan, *RADAR: An In-Building RF-based User Location and Tracking System*, In Proc. IEEE INFOCOM, pages 775-784, Tel-Aviv, Israel, March 2000.

[34] J. Cox, *Cisco releases Airespace WLAN location server*, NetworkWorld.com, 05/05/05

[35] T. Henderson, *Newbury Networks' WiFi Watchdog - an invisible fence to keep attack dogs away from your WLAN*, Network World Lab Alliance, Network World, 03/15/04

[36] D. Fox, J. Hightower, L. Liao, D. Schulz and G. Borriello, *Bayesian Filtering for Location Estimation*, Pervasive Computing, IEEE Volume 2, Issue 3, July-Sept. 2003 Page(s):24 - 33

[37] D. Fox, J. Hightower, H. Kauz, L. Liao and D. J. Patterson, *Bayesian Techniques for Location Estimation*, University of Washington, Department of Computer Science & Engineering, Seattle, WA, http://www.cs.washington.edu/ai/Mobile_Robotics/postscripts/bayes-location-ubicomp-03.pdf, accessed 17/10/2005

[38] A. Gelb, *Applied Optimal Estimation*, MIT Press, Cambridge, Massachusetts, 1974

[39] S. M. Bozic, *Digital and Kalman filtering*, Edward Arnold, second edtion, 1994

[40] P. Maybeck, *Stochastic Models, Estimation and Control*, volume 1, Academic, New York, 1979

[41] D. Schulz, W. Burgard, D. Fox and A. B. Cremers, *Tracking Multiple Moving Objects with a Mobile Robot*, Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001

[42] I. M. Rekleitis, *Cooperative localization and multi-robot exploration*, Ph D thesis, School of Computer Science, McGill University, Montreal, Quebec, Canada, February 2003

[43] W. Burgard, D. Fox, D. Hennig and T. Schmidt, *Estimating the absolute position of a mobile robot using position probability grids*, In Proc. of the Fourteenth National Conference on Artificial Intelligence (AAAI-96), pp. 896-901, 1996

[44] R. Simmons and S. Koenig, *Probabilistic robot navigation in partially observable environments*, in Proc. International Joint Conference on Artificial Intelligence, 1995

[45] L. P. Kaelbling, A. R. Cassandra and J. A. Kurien, *Acting under uncertainty: Discrete Bayesian models for mobile-robot navigation*, in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 1996

[46] S. Thrun, *Bayesian landmark learning for mobile robot localization*, Machine Learning, 1998

[47] W. Burgard, D. Fox, D. Hennig and T. Schmidt, *Estimating the absolute position of a mobile robot using position probability grids*, in Proc. of the Fourteenth National Conference on Artificial Intelligence (AAAI-96), pp. 896-901, 1996

[48] W. Burgard, A. Derr, D. Fox and A. B. Cremers, *Integrating global position estimation and position tracking for mobile robots: the Dynamic Markov Localization approach*, in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1998

[49] P. Krishnan, A. S. Krishnakumar, Wen-Hua Ju, C. Mallows, S. Ganu, *A System for LEASE: System for Location Estimation Assisted by Stationary Emitters for Indoor RF Wireless Networks*, Proceedings of IEEE Infocom 2004, Hong Kong, March 2004

[50] P. Bahl, V. N. Padmanabhan, A. Balachandran, *Enhancements to the RADAR User Location and Tracking System*, Microsoft Research Technical Report, February 2000

[51] S. Ganu, A. S. Krishnakumar and P. Krishnan, *Infrastructure-based location estimation in WLAN networks*, IEEE Wireless Communications and Networking Conference, Atlanta, Georgia, USA, March, 2004

[52] T. C. Black and W. J. Thompson, *Bayesian data analysis*, Computing in Science and Engineering, IEEE, July/August 2001

[53] F. Dellaert, D. Fox, W. Burgard and S. Thrun, *Monte Carlo Localization for Mobile Robots*, Computer Science Department, Carnegie Mellon University, Pittsburgh PA 15213, and zInstitute of Computer Science III, University of Bonn, D-53117 Bonn, 1999

[54] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, *Novel approach to nonlinear/non-Gaussian Bayesian state estimation*, IEE Procedings F, 140(2):107-113, 1993

[55] G. Kitagawa, *Monte Carlo filter and smoother for non-Gaussian nonlinear state space models*, J. Comput. Graph. Statist., vol. 5, pp. 1-25, 1996

[56] M. Isard and A. Blake, *Contour tracking by stochastic propagation of condensational density* in European Conference on Computer Vision, pp. 343-356, 1996

[57] M. Isard and A. Blake, *Condensation - conditional density propagation for visual tracking*, International Journal of Computer Vision, 29(1), pp. 5-28, 1998

[58] P. Bahl, V. N. Padmanabhan and A. Balachandran, *A software system for locating mobile users: Design, evaluation, and lessons*, MSR Technical Report, February 2000

[59] D. Fox, W. Burgard, F. Dellaert and S. Thrun, *Monte Carlo Localization: Efficient Position Estimation for Mobile Robots*, AAAI/IAAI, 1999

[60] D. Fox, *Adapting the Sample Size in Particle Filters Through KLD-sampling*, Department of Computer Science & Engineering, University of Washington, Seattle, WA 98195

[61] M. Montemerlo, S. Thrun and W. Whittaker, *Conditional Particle Filters for Simultaneous Mobile Robot Localization and People-Tracking*, Proceedings of the 2002 IEEE International Conference on Robotics & Automation, Washington DC, May 2002

[62] A. Bruce and G. Gordon, *Better Motion Prediction for People-Tracking*, In Proceedings of ICRA, 2004

[63] E. B. Meier and F. Ade, *Using the condensation algorithm to implement tracking for mobile robots* In Proc. of the Third European Workshop on Advanced Mobile Robots, Eurobot99, pages 73–80. IEEE, 1999

[64] S. Zhou, R. Chellappa and B. Moghaddam, *Appearance Tracking Using Adaptive Models in a Particle Filter*, Proc. of 6th Asian Conference on Computer Vision (ACCV), January 2004

[65] M. Koes, B. P. Sellner, B. Lisien, G. Gordon and F. Pfennig, *A learning algorithm for localizing people based on wireless signal strength that uses labeled and unlabeled data*, Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI), 2003

[66] E. Cuevas, D. Zaldivar and R. Rojas, *Particle filter in vision tracking*, Technical Report B 05-13, Freie Universität Berlin, Fachbereich Mathematik und Informatik, August 2005

[67] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. H ahnel, C. Rosenberg, N. Roy, J. Schulte and D. Schulz, *Probabilistic algorithms and the interactive museum tour-guide robot Minerva*, International Journal of Robotics Research, 19(11), 2000

[68] ANSI/IEEE Std 802.11, *Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Computer Society, 1999 Edition (R2003)

[69] Institute of Electrical and Electronics Engineers, *www.ieee.org*

[70] Sun Microsystems, *http://java.sun.com/j2se/1.4.2/docs/guide/jni/*, November 2005

[71] Sourceforge, *http://ipw2200.sourceforge.net/*, October 2005

[72] J. Tourrilhes, *http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.Extensions.html*, October 2005

[73] Placelab, *http://www.placelab.org/*, November 2005

[74] Microsoft, *http://www.microsoft.com/technet/prodtechnol/wce/maintain/ndis.mspx*, December 2005

[75] T. S. Rappaport, *Wireless communications - principles and practice*, Prentice Hall PTR, 1996

[76] H. Hashemi, *The indoor radio propagation channel*, Proceedings of IEE, 81(7):943-968, July 1993

[77] M. Brunato and R. Battiti, *Statistical Learning Theory for Location Fingerprinting in Wireless LANs*, Universita di Trento, Dipartimento di Informatica e Telecomunicazioni, Preprint submitted to Elsevier Science, 12 October 2004

[78] P. Gopalakrishnan, *Methods for Predicting the Throughput Characteristics of Rate-Adaptive Wireless LANs*, Master thesis, Rutgers University, 2004

[79] T. Locher, *Positioning with WLAN*, Distributed Computing Group, Swiss Federal Institute of Technology, Zurich, July 16, 2004

[80] Tsung-Nan Lin and Po-Chiang Lin, *Performance comparison of indoor positioning techniques based on location fingerprinting in wireless networks*, Wireless Networks, Communications and Mobile Computing, 2005 International Conference on Volume 2, 13-16 June 2005 Page(s):1569 - 1574 vol.2,

[81] A. Taheri, A. Singh and A. Emmanuel, *Location fingerprinting on infrastructure 802.11 wireless local area networks (WLANs) using Locus*, Local Computer Networks, 2004. 29th Annual IEEE International Conference on, 16-18 Nov. 2004 Page(s):676 - 683,

[82] K. Kaemarungsi and P. Krishnamurthy, *Properties of indoor received signal strength for WLAN location fingerprinting*, Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on, 22-26 Aug. 2004 Page(s):14 - 23

[83] K. Kaemarungsi and P. Krishnamurthy, *Modeling of indoor positioning systems based on location fingerprinting*, INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies Volume 2, 7-11 March 2004 Page(s):1012 - 1022 vol.2,

[84] B. Li, Y. Wang, H. K. Lee, A. Dempster and C. Rizos, *Method for yielding a database of location fingerprints in WLAN*, Communications, IEE Proceedings-Volume 152, Issue 5, 7 Oct. 2005 Page(s):580 - 586,

[85] A. Kushki, K. N. Plataniotis, A. N. Venetsanopoulos and C. S. Regazzoni, *Radio Map Fusion for Indoor Positioning in Wireless Local Area Networks*, Information Fusion, 2005 7th International Conference on, Volume 2, 25-28 July 2005 Page(s):1311 - 1318

[86] V. Lang and C. Gu, *A Locating Method for WLAN based Location Service*, e-Business Engineering, 2005. ICEBE 2005. IEEE International Conference on, 12-18 Oct. 2005 Page(s):427 - 432

[87] K. Kaemarungsi, *Efficient Design of Indoor Positioning Systems Based on Location Fingerprinting*, Wireless Networks, Communications and Mobile Computing, 2005 International Conference on, Volume 1, 13-16 June 2005 Page(s):181 - 186

[88] J. Kwon, B. Dundar and P. Varaiya, *Hybrid algorithm for indoor positioning using wireless LAN*, Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th, Volume 7, 26-29 Sept. 2004 Page(s):4625 - 4629 Vol. 7

[89] Rong-Hong Jan and Y. R. Lee, *An indoor geolocation system for wireless LANs*, Parallel Processing Workshops, 2003. Proceedings. 2003 International Conference on, 6-9 Oct. 2003 Page(s):29 - 34

[90] P. Prasithsangaree, P. Krishnamurthy and P. Chrysanthis, *On indoor position location with wireless LANs*, Personal, Indoor and Mobile Radio Communications, 2002. The 13th IEEE International Symposium on, Volume 2, 15-18 Sept. 2002 Page(s):720 - 724 vol.2

[91] H. Koshima and J. Hoshen, *Personal locator services emerge*, Spectrum, IEEE, Volume 37, Issue 2, Feb. 2000 Page(s):41 - 48

[92] T. Pering, Y. Agarwal, K. R. Gupta and Roy Want, *CoolSpots: reducing the power consumption of wireless mobile devices with multiple radio interfaces*, Proceedings of the 4th International Conference on Mobile Systems, Applications, and Services (MobiSys 2006), Uppsala, Sweden, June 19-22, 2006

[93] Mobilife, *http://www.ist-mobilife.org*, Januari and December 2006

[94] P. Floréen, M. Przybilski, P. Nurmi, J. Koolwaaij, A. Tarlano, M. Wagner, M. Luther, F. Bataille, M. Boussard, B. Mrohs and S. Lau, *Towards a Context Management Framework for MobiLife*, IST Mobile and Communications Summit, 2005, Dresden, Germany.

[95] B. Mrohs, S. Steglich, M. Klemettinen, J. T. Salo, A. Aftelak, C. Cordier and F. Carrez, *MobiLife Service Infrastructure and SPICE Architecture Principles*, Vehicular Technology Conference (VTC) Fall 2006, 2006, September 25-28, Montreal, Canada.

[96] D. Wagner and D. Schmalstieg, *First Steps Towards Handheld Augmented Reality*", Seventh IEEE International Symposium on Wearable Computers (ISWC'03), p. 127

[97] P. Gopalakrishnan, P. Spasojevic, L. Greenstein, I. Seskar, *A method for predicting the throughput characteristics of rate-adaptive wireless LANs*, Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th, Volume 6, 26-29 Sept. 2004 Page(s):4528 - 4532 Vol. 6

[98] Bluetooth, *http://www.bluetooth.com/Bluetooth/Learn/Basics/*, February 2006

[99] C. Taylor and N. Mawston, *Bluetooth Market Doubles: CSR Still Gaining Momentum*, Strategy Analytics, 15 Dec 2005

[100] A. Madhavapeddy and A. Tse, *A Study of Bluetooth Propagation Using Accurate Indoor Location Mapping*, Lecture Notes in Computer Science, Vol. 3660 (2005), pp. 105-122.

[101] Bluetooth Special Interest Group, *Bluetooth core specification*, *https://www.bluetooth.org/spec*, Mars 2006

[102] The official Bluetooth membership site, *http://www.bluetooth.org*, December 2005

[103] J. Y. Ye, *Atlantis: Location Based Services with Bluetooth*, Department of Computer Science, Brown University, 2005

[104] G. Anastasi, R. Bandelloni, M. Conti, F. Delmastro, E. Gregori and G. Mainetto, *Experimenting an Indoor Bluetooth-Based Positioning Service* ICDSW'03: 23rd International Conference on Distributed Computing Systems Workshpos 2003, 480-483.

[105] M. Nilsson and J. Hallberg, *Positioning with Bluetooth, IrDA, and RFID*, Master's Thesis, Luleå University of Technology, 2002

[106] A. Huang and L. Rudolph, *A Privacy Conscious Bluetooth Infrastructure for Location Aware Computing*, Massachusetts Institute of Technology, 2004

[107] Bluetooth Special Interest Group, *Bluetooth Profile - Specification of the Bluetooth system*, Version 1.1, February, 2001

[108] Bluetooth Special Interest Group, *Bluetooth Profile - Specification of the Bluetooth system*, Version 1.2, November, 2003

[109] IEEE 802.15 WPAN Task Group 1 (TG1), *http://www.ieee802.org/15/pub/TG1.html*, January 2006

[110] A. Jones, *Bluetooth In-Depth*, CWRU Apple Campus Representative, mac.cwru.edu/files/presentations/bluetooth/bt_depth.ppt, February 2006

[111] IEEE P802.15 Working Group for Wireless Personal Area Networks, *Clause 14.3 Adaptive Frequency Hopping*, Jul 13, 2001

[112] JCP - The Java Community Process Program, JSR-82: Java APIs for Bluetooth, *http://jcp.org/en/jsr/detail?id=82*, Mars 2006

[113] C. di Flora, M. Ficco, S. Russo, V. Vecchio, *Indoor and outdoor location based services for portable wireless devices*, 25th IEEE International Conference on Distributed Computing Systems Workshops, 2005

[114] A. Kotanen, M. Hännikäinen, H. Leppäkoski, T. D. Hämäläinen, *Experiments on Local Positioning with Bluetooth*, Proceedings of the International Conference on Information Technology: Computers and Communications (ITCC'03)

Appendix A

Signal strength traces

A.1 Test 7-10 trace analysis

A.2 Position determination in open space environment

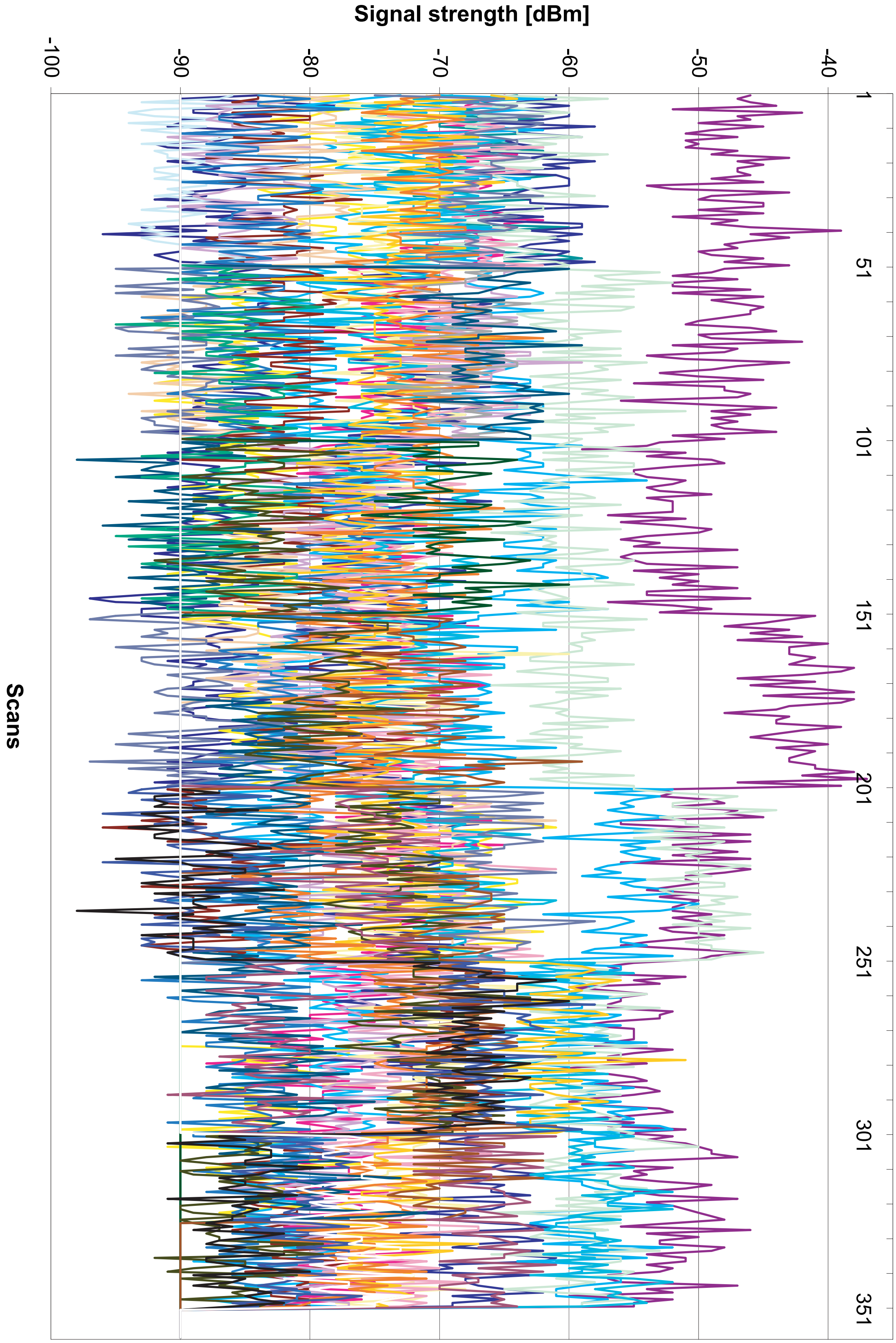

Figure A.1. Generated signal strength trace for 32 access points over 350 scans.

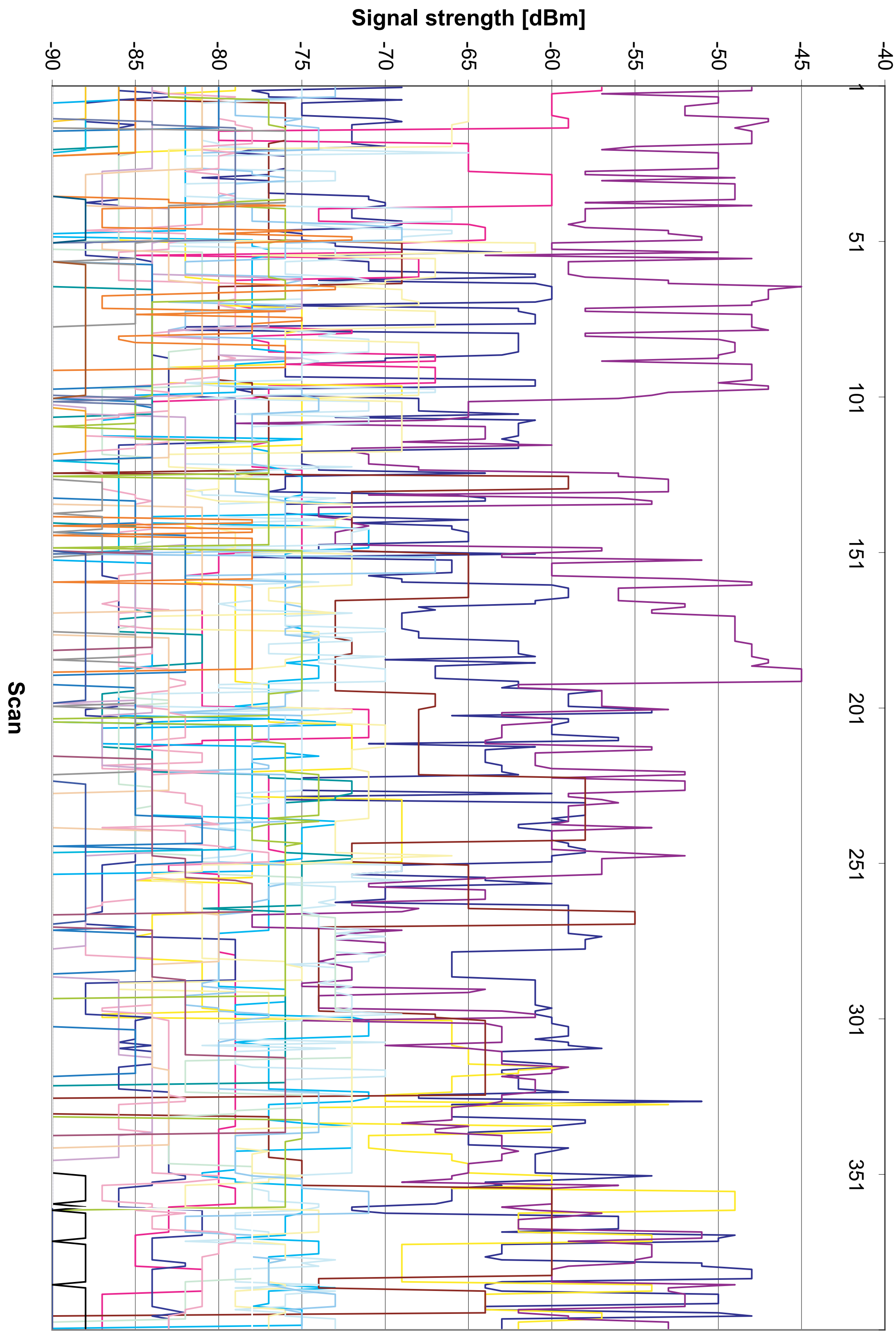

Figure A.2. Live signal strength trace for 32 access points over 400 scans.

Appendix B

Office floor

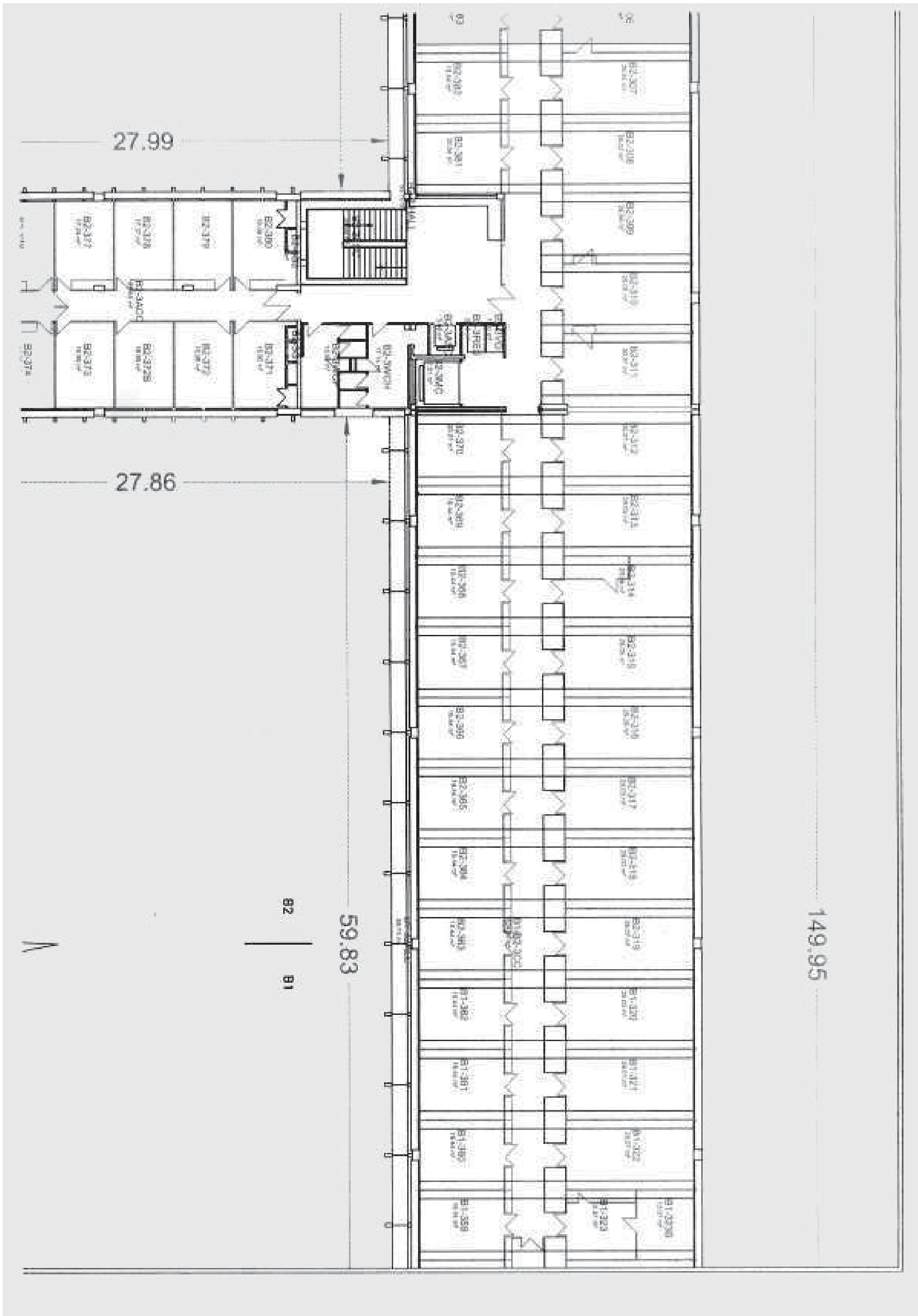

Figure B.1. The office floor where part of the testing took place.

COS/CCS 2008-08