This is a preliminary version of an article published in
Proc. of the Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom), pp. 145 – 150,
New York, NY, USA, March 2007
by Thomas King, Thomas Haenselmann, Stephan Kopf, Wolfgang Effelsberg

Overhearing the Wireless Interface for 802.11-based Positioning Systems

Thomas King, Thomas Haenselmann, Stephan Kopf, Wolfgang Effelsberg
{king,haenselmann,kopf,effelsberg}@informatik.uni-mannheim.de
Department of Computer Science, University of Mannheim

Abstract

Not only the communication capabilities of 802.11, but also the capability to determine the position of mobile devices make 802.11 highly appealing for many application areas. Typically, a mobile device that wants to identify its position regularly performs active or passive scans to obtain the signal strength measurements of neighboring access points. However, so far, no investigations are known to have been launched into how regular scanning affects concurrent data transmissions from an end-user point of view. In this paper, we explore how common data communication is affected while actively or passively scanning at the same time. Furthermore, we present a novel scan scheme called Monitor Sniffing*. Monitor Sniffing exploits the fact that 802.11 operates on overlapping channels by overhearing the wireless interface. We have implemented our Monitor Sniffing algorithm using commodity 802.11g hardware, and we demonstrate that it does not disturb concurrent data communication.*

1. Introduction

During recent years we have seen considerable improvements in downsizing computer hardware and in increasing the capacity of rechargeable batteries, as well as the advent of wireless networks for the mass markets. These technologies allowed the manufacturers to build mobile devices that have a similar performance as desktop computers had several years ago. The benefit of mobile devices can be leveraged by so-called *location-based services*: Applications that act differently depending on the location of the user or, even better, proactively offer location-dependent information to the user, are currently a hot topic in research, and are considered to be a promising market.

Nowadays, the *Global Positioning System* (GPS) [10] is the predominant outdoor positioning system. Whereas GPS works well in many outdoor scenarios, it suffers from obstacles such as skyscrapers creating shielded street canyons or walls and ceilings blocking the radio signals indoors. To this end, a large number of research projects conceived novel positioning techniques (e.g. [17], [20], and [4]). However, all these systems either require specialized hardware or show poor positioning accuracy.

Many recent research activities focus on *IEEE* 802.11-based positioning because almost everywhere, especially in occupied areas of developed countries, 802.11 network infrastructure is available for data communication[1]. Universities, offices and many private homes utilize 802.11 networks to get rid of wires. As a reaction to the proliferation of 802.11, almost all modern mobile devices ranging from smartphones to laptops, are shipped with built-in 802.11 network interfaces. 802.11 networks are not only used in indoor scenarios; even outdoors, many universities and coffee stop owners support nomadic users.

Another key argument for 802.11-based positioning systems is that 802.11 hardware can be used in dual mode: For data communication and for measuring the signal strength of neighboring access points as a prerequisite for positioning systems. For instance, the *GUIDE* tourist guide [6] and *PlaceLab* [14] are two representatives for outdoor positioning systems that utilize 802.11 for data transmissions as well as position determination. Many indoor positioning systems such as *RADAR* [2], *HORUS* [22] and *COMPASS* [13] require signal strength readings and communication capabilities provided by 802.11.

In 802.11 the typical way of measuring the signal strength of access points within communication range is to perform active or passive scans. So far, it was unknown what happens to the data communication capabilities of mobile devices if signal strength measurements in form of active or passive scans are preformed concurrently. This work fills the gap by investigating how throughput and round trip delay suffer from different scan techniques and intervals.

Based on these results we conclude that active and passive scanning are inappropriate for positioning systems because they require too much time to gather information about neighboring access points and hence produce large communication dropouts. To overcome this problem, we propose a novel scanning technique called *Monitor Sniffing*. Monitor Sniffing exploits the fact that 802.11 uses overlapping channels by overhearing the wireless interface. Over-

[1] http://www.wigle.net

hearing allows the mobile device to listen to frames from adjacent channels while concurrently staying on the channel used for data communication with the access point it is associated with. This is usefull because from [16] and [5] we know that in many populated areas on average 2.4 access points are in communication range. We have implemented Monitor Sniffing using commodity 802.11g hardware and show that this scan scheme does not disrupt concurrent data transmissions and produces faster scanning results. Finally, our approach is compliant with the existing 802.11 standard and requires only a software update on the client side to get ready for use.

The rest of the paper is organized as follows. Section 2 introduces active and passive scanning and discusses how these approaches affect the communication capabilities of mobile devices. In Section 3, we propose Monitor Sniffing, a scan scheme especially designed for 802.11-based positioning systems. Section 4 presents the relevant related work. Finally, we conclude the paper and give directions for future work in Section 5.

2. Active vs. Passive Scanning

In this section, we discuss two techniques to discover neighboring access points and to measure their signal strength, namely *active scanning* and *passive scanning*, as described by the IEEE 802.11 standard [9]. After introducing these two techniques, we investigate the effects of concurrent scanning on common data transmissions.

2.1. Functional Principles

IEEE 802.11 subdivides the radio spectrum into a set of channels. The number of available channels depends on where 802.11 is used and which sub-specification of the 802.11 physical layer is selected. For instance, in the United States, only 11 channels are allowed for 802.11b and 802.11g, whereas 13 channels can be used in Europe. In contrast, the commercially less successful 802.11a defines 12 channels, however, in some countries the radio spectrum of 802.11a is still assigned to other purposes, today.

A wireless network interface usually listens to one channel at a given time. So, if a mobile station wants to get to know all the access points in communication range, it has to tune its wireless network interface to each channel, one after another, and perform a scan.

IEEE 802.11 defines two scanning techniques: Active scanning and passive scanning. The former approach requires a bi-directional communication initiated by the mobile station. For the latter approach, the mobile station passively listens for management frames send out by access points. The details of these techniques are discussed in the following two sections.

For the remainder of this paper, we focus on the infrastructure mode of 802.11 because this is the typical scenario in the field of positioning systems. We mainly focus on 802.11g because it is the one most frequently used. However, our results are also applicable to 802.11b. Furthermore, our scenario is located in Europe and 802.11 operates on 13 channels.

Active Scanning A mobile device follows the subsequent procedure for each channel to perform an active scan: It tunes the wireless interface to the particular channel. Depending on the network card used, switching the channel requires 5 to 19 milliseconds [18]. After that, it uses the 802.11 medium access procedure to gain access to the channel and sends a so-called *Probe Request* frame. It waits for a certain time, and if no frame is received it proceeds to the next channel. If a frame is received, the mobile device processes any so-called *Probe Response* frame received within a certain amount of time.

Access points are supposed to reply to a Probe Request frame with a Probe Response frame. By examining received Probe Responses, a mobile device is able to recognize neighbouring access points and their signal strength.

The IEEE 802.11 standard does not define default values for these timers, however, [1] and [19] empirically studied the values used by wireless network card manufactures. In total, the exact time required to perform an active scan can vary significantly based on the number of available access points and hardware capabilities. In our measurements, we found that at most 20 milliseconds are required to scan one channel. In total, an active scan over all channels takes less than 260 milliseconds to complete.

Passive Scanning Passive scanning has been introduced to reduce the workload of mobile devices and hence save battery power. While scanning passively, a device listens to each channel and waits for a given period of time. If an access point is assigned to a particular channel, the mobile station should receive a so-called *Beacon* frame. Every access point broadcasts Beacon frames on a regular basis to maintain the network. Beacons usually contain the same information as Probe Response frames, such as supported data rates, supported extended data rates, and the name of the network. By examining the received Beacon frames, a mobile device is able to recognize access points within communication range and their signal strength.

Access points usually broadcast a Beacon packet every 100 milliseconds which means that a mobile station should stay on a particular channel at least for the same period to make sure not to loose a Beacon from an unknown access point. In total, a passive scan requires at least 1.3 seconds to be completed. Once again, note that this configurable value is not defined in the IEEE 802.11 standard.

2.2. Effects on Communication Performance

Among other positioning systems, 802.11-based positioning systems rely on a steady stream of signal strength measurements to determine the position of the user, especially if they are running in tracking mode. This means, that active or passive scans are executed at a high rate (e.g., every 0.5 seconds) and hence the network card is quite busy with scanning. In this subsection, we investigate how active and passive scanning affects regular data transmission in terms of throughput and round trip time.

Experimental Environment To achieve interpretable results we simplified our scenario: Only one mobile device communicates with one access point. In a more complex scenario with additional mobile devices, throughput and round trip time may even be worse and more volatile.

We used a *Fujitsu Siemens* Lifebook T4010 laptop running *Linux* kernel 2.6.16 and *Wireless Tools* 28pre13. We implemented passive scanning support into the ipw2200 1.1.3 network interface driver [11], so that we were able to use the built-in *Intel* PRO/Wireless 802.11b/g network card of the laptop.

A *Linksys / Cisco* WRT54GS access point assigned to channel 8 has been used to gain access to the local network of the *University of Mannheim*. The access point was running the *Alchemy* firmware version 1.0 and was configured for 802.11b/g mode with a beacon interval of 100 milliseconds and a Delivery Traffic Indication Map every 10^{th} Beacon. The distance between the laptop and the access point was approximately 3 meters, and during the measurements a 54 MBit/s link between the laptop and the access point was established.

We conducted data transmission measurements with *iperf* 2.0.2[2] to gauge throughput and round trip time. For this, we used an iperf server within the local network and an iperf client running on the laptop. The iperf server was connected to the access point via a 100 MBit/s switched Ethernet, so that the wireless link was the only bottleneck. Iperf was configured to measure the throughput every 0.5 seconds and to transmit data for 60 seconds. For all the graphs presented in this paper, we carried out the experiments at least three times and selected the result showing the highest throughput.

Experimental Results Throughput and round trip delay are the main objectives and are first measured without any scanning at all to get a reference. Based on this reference, throughput and round trip time are quantified for various scan intervals and different scan schemes. The relation between the maximum throughput and the throughput achievable for a particular scan interval gives a well-balanced estimate on how data communication is affected by scanning. Additionally, the round trip time measurements indicate how interactive communication is strained by concurrent scans.

In this paper, due to page restrictions we focus only on TCP, because TCP is the most frequently used transport protocol, and its flow control algorithms might be confused about communication dropouts caused by frequent scanning operations. For an in-depth analysis of how TCP and UDP traffic is affected by current scanning we refer to [12].

To get a reference value of how much data can be transferred over a 802.11g link, we invoked iperf in TCP mode and sampled the throughput and round trip delay for 60 seconds. Our measurements show, on average, a throughput of 17.1 MBit/s and a round trip time of 39.3 milliseconds; the standard deviation of the measurements is 768 KBit/s for throughput and 11.82 milliseconds for delay.

Figure 1(a) shows average throughput and average round trip time as well as standard deviations for both measures in the active scanning scenario. An active scan interval of 0.3 seconds results in 547 KBit/s and a round trip time of 1.7 seconds. However, with an active scan interval of 1 second, 10.4 MBit/s are achievable. In other words, more than 60 percent of the maximal achievable throughput can be obtained. The round trip delay also shows relatively stable values around 90 milliseconds with a standard deviation of nearly 60 milliseconds, rendering network conditions well enough to allow meaningful communication.

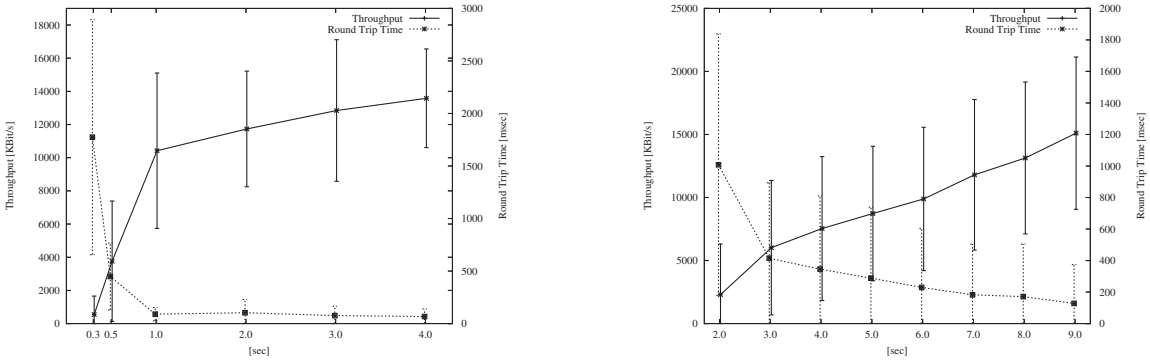
With a passive scan interval of 2 seconds TCP achieves less than 2.2 MBit/s throughput and an average round trip delay of more than 1 second (see Figure 1(b)). To obtain network conditions that make interactive communication feasible, a scan interval of 7 or more seconds is required. The average round trip delay drops below 190 milliseconds and the average troughput is about 11 MBit/s.

3. Overhearing the Wireless Interface

An important feature of a good scan scheme for positioning systems is its ability to minimize client service disruption as well as to deliver a high rate of signal strength measurements of access points within communication range. Note that service disruption incorporates communication dropouts, declined throughput and increased delays from an end-user point of view. Unfortunately, these two requirements are mutually exclusive. Consequently, any scanning approach needs to balance the trade-off between these two requirements.

In the previous section, we already presented how active and passive scanning balance these features. Both approaches mainly focus on a high rate of signal strength measurements and care less about client service disruptions. To address a scan scheme that handles the client service disruption with care, we present a novel scan approach called

[2] http://dast.nlanr.net/Projects/Iperf



(a) Active scans every 0.3, 0.5, 1.0, . . ., 4.0 seconds.

(b) Passive scans every 2.0, . . ., 9.0 seconds.

Figure 1. TCP throughput and round trip time while concurrently performing scans.

Monitor Sniffing. Monitor Sniffing exploits overlapping channels of 802.11 to cut down client service disruptions while at the same time delivering a high rate of signal strength measurements of at least a subset of access points within communication range.

3.1. Monitor Sniffing

Our Monitor Sniffing scan scheme works as follows:

(i): We configure the wireless network card to work in monitor mode while associating it at the same time with a given access point for data communication. Monitor mode means that the network card driver does not block management frames such as Beacons, Probe Requests and Probe Responses. Instead, the network card driver forwards management frames to the network interface socket, so that these frames can be captured by a program.
In the past, a wireless network interface could not be used to monitor the wireless channel while at the same time transmit data. For instance, the *Lucent* Orinoco Silver network card works only in monitor mode or in data communication mode, but not in both modes at the same time. Recently, wireless network cards are available that support both modes concurrently (e.g., cards based on the *Intel* Pro/Wireless 2200BG or *Atheros* AR5xxx chipsets).

(ii): We overhear the wireless network interface by switching it into promiscuous mode.

(iii): We add a filter to the network interface socket, so that only Beacons get through.

(iv): We examine any Beacon we receive and offer the signal strength as well as the MAC address of access points to the application layer.

The following sections discuss the concept of overlapping channels used by 802.11 and present an experiment that investigates how this concept can be exploited by overhearing the wireless interface to meet our needs.

3.2. Overlapping Channels

The IEEE 802.11b/g standards utilize the frequency spectrum between 2.4 and 2.5 Ghz. This spectrum is subdivided into 13 overlapping channels whose center frequencies are 5 Mhz apart while each channel has a spread of 22 Mhz around the center frequency. For instance, channel 6 spreads from 2.426 to 2.448 Mhz while channel 4, 5, 7, and 8 also utilize parts of this spectrum by having their center at 2.427 Mhz, 2.432 Mhz, 2.442 Mhz, and 2.447 Mhz, respectively. As a result, a transmission on one channel becomes detectable on an adjacent one. While overlapping signals from neighboring channels are undesirable for undisturbed data transmission it is preferable for positioning systems if the signals can be decoded.

Broadcasts such as Beacons or Probe Requests are usually sent with a data rate of 2 MBit/s in 802.11b/g networks. For this data rate, a *Direct Sequence Spread Spectrum* (DSSS) encoding is used. In general, DSSS spreads a signal over a wider frequency band which means in our case that the signal is spread over a complete channel. Therefore, transmissions encoded with DSSS are well protected against interference, such as noise. Interference tends to take the form of relatively narrow pulses and hence destroys only a part of the spread signal. Even if only parts of a DSSS encoded signal can be heard, it is often still feasible to decode the signal. This is beneficial for our Monitor Sniffing approach, because it tries to decode signals from adjacent channels.

3.3. Experimentation in a Real Environment

We carried out an experiment to investigate how well frames from adjacent channels can be decoded. The experimental environment as well as the results are described in the subsequent sections.

Experimental Environment In addition to the experimental environment already described in Section 2.2, we set

up four *Netgear* WG102, three *D-Link* 700AP, three *Cisco / Linksys* WRT54G, and two *enterasys* RBT-4102-EUR access points to cover all 802.11 channels. The access points were placed on a table in our lab, and the distance between the laptop and the access points was approximately 5 meters.

We implemented the Monitor Sniffing approach to overhear the wireless network card and to collect Beacon frames from the network interface socket. Additionally, we enhanced this implementation to associate the network interface with a given access point for a given amount of time and to log every received Beacon (e.g., signal strength and MAC address of the originating access point).

Experimental Results For each channel, we collected the Beacons that were received during 60 seconds. Given a beacon interval of 100 milliseconds, an access point is supposed to emit 600 Beacons during a measurement cycle. Table 1 shows how many Beacons were received from which access point while the network interface switched through the channels. Note that we name the access points *A, B, . . ., M* corresponding to the channels they are assigned to (e.g., access point *A* is assigned to channel 1, the access point with name *M* is assigned to channel 13).

From the table we see that only a few Beacons are lost from the access point the network interface is associated with. The number of received Beacons ranges from 394 (see channel 6) to 582 (see channel 11). The reasons for this are a high interference rate due to the large number of access points located in close proximity as well as a high number of Beacon collisions. On the MAC layer, Beacons are broadcasted and hence no acknowledgments are used causing a colliding Beacon not to be retransmitted, and therefore to be missed.

Furthermore, we see that Beacons from neighboring channels can be heard. For instance, if the wireless network interface is assigned to channel 8 it only receives about 35 percent of the Beacons the access point of channel 9 is broadcasting. In contrast, if the wireless interface is tuned to channel 9 it is able to decode more than 87 percent of the Beacons access point *H* is emitting.

The variations are even larger if we look at the channels 10 Mhz away from the channel the wireless network interface is assigned to. For example, if the network card listens on channel 5 it is only capable to decode nearly 7 percent of the Beacons access point *G* is broadcasting. This is in contrast to channel 11 where nearly 85 percent of all Beacons broadcasted on channel 13 can be decoded.

Sometimes it is still feasible to decode Beacons from channels three steps away. As listed in Table 1, we see that it works for ten channels and only in three cases it is not possible to decode frames from channels three steps away (see channel 6, 9, and 10). If we calculate the percentage of received Beacons we see that there is a large variation. For example, only 1 percent of all broadcasted Beacons from access point *L* can be decoded if the network interface is tuned to channel 9. In contrast, nearly 64 percent of all Beacons from channel 1 can be decoded if the interface listens to channel 4. The reason for this is that channels three steps away do not directly overlap, instead, only side lobes as radiation of the main signal lobe are detectable. It is pretty difficult for the wireless interface to detect these low-powered signals and correctly decode them.

Furthermore, we also investigated how well the reception power of Beacons transmitted on adjacent channels can be measured. However, due to page restrictions we obmitted the analysis in this paper and refer the interested reader to [12].

4. Related Work

Several previously published studies investigate throughput and delay on 802.11. Xylomenos and Polyzos [21] explore the throughput of UDP and TCP achievable with several early 802.11 hardware devices. Their research focuses on throughput limitations caused by software implementation issues. The researchers recommend changes in the implementations of network protocols as well as in drivers. Duchamp and Reynolds measure throughput while varying the distance between a mobile device and an access point [7]. In [3], Bing measures delay and throughput for two early 802.11 network interfaces in a lab environment. A performance degradation is observed by Heusse et al. [8] if some mobile devices use a lower bit rate than the other devices. The authors analyzed the problem theoretically as well as empirically and derived a simple expression for the useful throughput. Compared to our work, all these approaches do not consider scanning at all and use rather outdated 802.11 or 802.11b hardware.

Scanning has been investigated mainly in the field of handover and roaming optimizations. Ishwar Ramani et al. [18] proposed *SyncScan*, a technique that tries to reduce the time required to perform an active scan by synchronizing the time when access points transmit Beacons. In their solution, access points are synchronized and broadcast Beacons in a pre-decided order. Therefore, a mobile device is able to predict the time when a neighboring access point will broadcast a Beacon by listing to a Beacon of the access point it is associated with.

A client side solution is proposed in [15]. The authors state that in typical urban and enterprise environments the access point density is fairly high, so that a mobile device often faces the opportunity of handover within the currently used channel. This approach is comparable to our Monitor Sniffing technique, however, we additionally exploit the fact that 802.11 defines overlapping channels and hence a mobile device is able to receive frames from access points assigned to neighboring channels.

Access Points	Channels 1	2	3	4	5	6	7	8	9	10	11	12	13
A	517	542	207	386	0	0	0	0	0	0	0	0	0
B	405	554	409	300	21	0	0	0	0	0	0	0	0
C	182	282	449	367	179	38	0	0	0	0	0	0	0
D	256	379	437	415	424	106	102	0	0	0	0	0	0
E	0	197	94	329	415	354	95	74	0	0	0	0	0
F	0	0	6	86	228	394	331	86	0	0	0	0	0
G	0	0	0	3	44	273	518	149	53	0	0	0	0
H	0	0	1	0	118	283	446	545	523	411	38	0	0
I	0	0	1	0	0	0	78	208	535	354	214	1	0
J	0	0	1	0	0	0	34	140	502	533	458	57	1
K	0	0	1	0	0	0	0	62	482	503	582	572	307
L	0	0	1	0	0	0	0	0	8	331	551	571	500
M	0	0	0	0	0	0	0	0	0	10	508	484	579

Table 1. Number of Beacons received from each channel.

5. Conclusions and Future Work

The primary contribution of this paper is an empirical analysis of how active and passive scanning affect concurrent data transmission, Additionally, we propose a novel scanning technique for 802.11-based positioning systems called Monitor Sniffing. We found out that with an active scanning interval of 2 seconds the resulting network conditions can be considered stable enough and suitable for common data transmission. The same is true for a passive scanning if a scan interval of 7 seconds is applied. For smaller scan intervals, the network conditions cannot be considered stable.

Our novel scanning technique allows 802.11-based positioning systems to stay on a certain channel while concurrently receiving Beacons from access points assigned to adjacent channels. We achieved this by overhearing the wireless interface running in monitor mode. This allows undisturbed data transmissions and high rate signal strength measurements to support precise position estimates.

In our ongoing work, we are trying to conceive an algorithm that automatically switches between active and monitor sniffing scans to maintain undisturbed data transmission and to deliver a minimum number of signal strength measurements from different access points.

References

[1] W. Arbaugh, M. Shin, and A. Mishra. An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. *ACM SIGCOMM Computer Communications Review*, 33(2):93–102, April 2003.

[2] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *Proc. IEEE InfoCom*, 2000.

[3] B. Bing. Measured Performance of the IEEE 802.11 Wireless LAN. In *Proc. IEEE LCN*, 1999.

[4] B. Brumitt, B. Meyers, J. Krumm, A. Kern, and S. Shafer. EasyLiving: Technologies for Intelligent Environments. In *Proc. HUC*, 2000.

[5] Y.-C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm. Accuracy Characterization for Metropolitan-scale Wi-Fi Localization. In *Proc. ACM MobiSys*, 2005.

[6] K. Cheverst, N. Davies, K. Mitchell, and A. Friday. Experiences of Developing and Deploying a Context Aware Tourist Guide: The GUIDE Project. In *Proc. ACM MobiCom*, 2000.

[7] D. Duchamp and N. F. Reynolds. Measured Performance of a Wireless LAN. In *Proc. IEEE LCN*, 1992.

[8] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda. Performance Anomaly of 802.11b. In *Proc. IEEE InfoCom*, 2003.

[9] IEEE Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Website: `http://standards.ieee.org/getieee802/`, 1999.

[10] E. Kaplan and C. Hegarty, editors. *Understanding GPS: Principles and Applications*. Artech House Incorporated, second edition, 2005.

[11] T. King. Passive Scanning not supported? ipw2100 developer mailinglist: `http://sourceforge.net/mailarchive/forum.php?thread%5Fid=27023011&forum%5Fid=38938`, July 2006.

[12] T. King, T. Haenselmann, S. Kopf, and W. Effelsberg. Overhearing the Wireless Interface for 802.11-based Positioning Systems. Technical Report TR-2006-018, University of Mannheim, 2006.

[13] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg. COMPASS: A Probabilistic Indoor Positioning System Based on 802.11 and Digital Compasses. In *Proc. ACM WiNTECH*, 2006.

[14] A. LaMarca, Y. Chawathe, S. Consolvo, J. Hightower, I. Smith, J. Scott, T. Sohn, J. Howard, J. Hughes, F. Potter, J. Tabert, P. Powledge, G. Borriello, and B. Schilit. Place Lab: Device Positioning Using Radio Beacons in the Wild. In *Proc. IEEE PerCom*, 2005.

[15] V. Mhatre and K. Papagiannaki. Using Smart Triggers for Improved User Performance in 802.11 Wireless Networks. In *Proc. ACM MobiSys*, 2006.

[16] A. J. Nicholson, Y. Chawathe, M. Y. Chen, B. D. Noble, and D. Wetherall. Improved Access Point Selection. In *Proc. ACM MobiSys*, 2006.

[17] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket Location-Support System. In *Proc. ACM MobiCom*, 2000.

[18] I. Ramani and S. Savage. SyncScan: practical fast handoff for 802.11 infrastructure networks. In *Proc. IEEE InfoCom*, 2005.

[19] H. Velayos and G. Karlsson. Techniques to Reduce the IEEE 802.11b Handoff Time. In *Proc. IEEE ICC*, 2004.

[20] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location System. *ACM Transactions on Information Systems*, 10(1):91–102, January 1992.

[21] G. Xylomenos and G. C. Polyzos. TCP and UDP Performance over a Wireless LAN. In *Proc. IEEE InfoCom*, 1999.

[22] M. Youssef. *Horus: A WLAN-Based Indoor Location Determination System*. PhD thesis, University of Maryland at College Park, 2004.