Assessing the Completeness of Wireless-side Tracing Mechanisms

Aniket Mahanti Martin Arlitt Carey Williamson
Department of Computer Science
University of Calgary
Calgary, AB, Canada T2N 1N4
{amahanti, arlitt, carey}@cpsc.ucalgary.ca

Abstract

Analyzing traces of wireless network activity has many pragmatic purposes, from capacity planning to network design. Unfortunately, capturing complete traces of wireless traffic is difficult, and using incomplete traces can degrade the quality of the aforementioned analyses. In this paper we examine three different methods for estimating the completeness of wireless traces. We find that a method that examines MAC-layer sequence numbers provides the most accurate results. We also examine the effect of the placement of wireless sensors on the completeness of wireless-side traces. We determine that locating sensors such that the signal strengths between clients and access points is over 40% results in low miss rates at the sensor, and few CRC errors.

1 Introduction

The use of Wireless Fidelity (WiFi) technology has become ubiquitous. WiFi allows a person with a laptop or handheld computer such as a Personal Digital Assistant (PDA) to connect to the Internet without using any cabling. These wire-free networks are called Wireless Local Area Networks (WLANs).

The global usage of WiFi has increased significantly over the past 6 years. Recent estimates indicate that there are currently 165 million WiFi users worldwide [4]. Nowadays, it is easy to find WLANs almost anywhere: airports, coffee shops, university campuses, enterprises, and homes. In most cases, WLANs are set up as "hotspots" covering a small area allowing customers easy access to the Internet.

The surge in the popularity of WLANs motivates the study of how such networks are used. A commonly used methodology in such studies is to analyze empirical traces of wireless traffic. There are two different techniques for collecting wireless traffic traces. The first method, *wired-side measurement*, attaches Ethernet sensors to routers that transfer wireless traffic, and collects supplementary information using SNMP polling, syslog, and authentication logs. The second technique, *wireless-side measurement*, requires sensor devices to be deployed throughout the WLAN to capture frames directly from the wireless medium.

In the past, researchers have studied user behaviour, user sessions, roaming, network load, and traffic characteristics from WLANs on campuses [7, 8, 14, 15, 16], in enterprises [2], and at public hotspots [1, 3, 12] . All of these studies utilized wired-side measurements in their analyses. More recent studies have used wireless-side traces [9, 13].

Wireless-side measurements are desirable in that they enable similar analyses to wired-side measurements, as well as numerous other analyses that are not possible from the wired-side. For example, network designers can use Radio Frequency (RF) signal analysis for site and capacity planning. An understanding of signal strength/quality, physical errors, and retransmissions can help network designers in these planning exercises. Network workload analysis can help in improving quality of service (QoS) for users, which is especially important for IP-based multimedia services. Software designers can use the data to create robust, wireless-friendly multimedia applications. The traces are also useful for addressing common WLAN issues such as multi-path reflections, hidden node problems, RF denial of service attacks, contention, or congestion. Finally, researchers can use these traces to improve the operational performance of the 802.11 MAC protocol.

Deployment of a wireless-side measurement infrastructure is a non-trivial process. The number of sensors to be placed in the WLAN and their vantage points are important considerations. The problem is further complicated when dealing with a geographically-distributed WLAN. An understanding of the measurement losses incurred is of cardinal significance. To decide upon the effectiveness of a deployment, a sound loss estimation technique is required. Sensor placement is another important issue in order to maximize the completeness of the collected trace while minimizing the number of sensors required (this saves both the cost of purchasing and deploying additional sensors).

In this paper we present a detailed assessment of the

completeness of wireless-side measurements. We propose three different methods for assessing the completeness of such traces: the *beacon method*, the *ACK method*, and the *sequence number method*. The beacon method is the simplest but least accurate. The ACK method is more accurate than the beacon method, but tends to underestimate the number of missed frames, and in some situations fails completely. The sequence number method is the most accurate of the tested methods, but is more complex to implement due to the idiosyncracies of wireless network devices. We also examine the effect of the placement of sensors on the completeness of wireless-side traces. We find that locating sensors such that the signal strength between clients and access points is at least 40% results in negligible miss rates and CRC error rates at the sensor.

The remainder of the paper is organized as follows. Section 2 introduces the concept of wireless-side measurement. Section 3 describes the measurement methodology we used. Section 4 compares the three different methods proposed for assessing the completeness of wireless-side traces. Section 5 explores the implications of sensor placement on wireless-side trace quality. Section 6 proposes a robust sensor layout for our wireless network. Section 7 discusses related work. Section 8 summarizes our work and lists future directions.

2 Wireless-side Measurement

There are two approaches to wireless-side measurement of traffic to and from a wireless Access Point (AP). One approach is called AP-centric, and the other is called AP-triangulation.

In the first scenario, a single sensor is placed close to an AP. Such a deployment allows the sensor to have a perspective of the WLAN that is consistent with the viewpoint of the AP. Typically in such a setup the sensor is unable to see wireless stations that are beyond the association range of the AP. AP-centric monitoring facilitates the analysis of client-related problems for the monitored AP.

The second approach involves surrounding a monitored AP with three sensors such that they form a triangle, with the AP at its centre. The triangle should be sized such that the signal strength of the AP as perceived by the sensors has diminished by half [19]. Here, the RF environment of the client stations is being monitored in contrast to that of the AP alone. This approach monitors client behaviour as they associate or disassociate with the AP, correspondingly entering or leaving the service set. This approach is usually suitable when dealing with security issues such as rogue APs and unauthorized users.

3 Measurement Methodology

The most common measurement design employed in the networking literature is RF Monitoring (RFMON) [10, 13, 21]. This configuration places a wireless network interface card (NIC) into monitor mode, allowing the NIC to passively observe all nearby wireless traffic. NICs placed in RFMON mode can only sniff frames on a single channel. Furthermore, not all NIC chipsets and drivers support RFMON mode [11]. For those that do support RFMON mode, some chipsets may not function properly [11]. Some chipsets and/or operating systems may just support RFMON promiscuous mode, where only wireless data frames are captured. Also, not all drivers are supported on all operating systems (e.g., Windows, Mac OS). In almost all cases, those employing the RFMON design have used notebook computers with a wireless NIC, with a protocol analyzer (e.g., Ethereal, tcpdump) running to capture frames. This means that the placement points and operating range (if not using an external antenna) of the sensor will be constrained.

To overcome the above-mentioned shortcomings, we use a specialized trace capture program called Airopeek NX [17]. Airopeek is a real-time 802.11 a/b/g WLAN analyzer used by network designers and administrators for performing site surveys, security audits, application-layer protocol identification, and troubleshooting. Airopeek works in conjunction with a network adapter (e.g., wireless NIC) to sniff frames from the air. For our purposes we used an off-the-shelf WLAN adapter called 802.11 Remote Distributed Sensor [17].

3.1 Airopeek

Airopeek can capture the link-, network-, transport-, and application-layer headers of a frame. Airopeek records additional information such as a timestamp, the signal strength, channel number, data rate, and size of each frame. This additional information is stored as a separate header that precedes the MAC header. Airopeek offers the following useful features, which are not (all) available with other wired-side and RFMON-based wireless-side measurement approaches:

Multiple WLAN adapters: Airopeek allows multiple simultaneous capture sessions, each using a different adapter. This means that a single workstation with multiple network interfaces can be used to run multiple trace capture sessions, thus reducing the hardware required for the task.

Channel hopping: For WLANs that operate on multiple channels, Airopeek supports channel scanning on the network adapter. Airopeek can be used for setting the channels that need to be scanned, the order in which they are scanned, and the duration to gather data on each channel.

Frame slicing: Because 802.11 WLAN frames have a variable header length, a constant capture size will not always record the same (desired) information for every frame. The frame slicing feature provides variable length capture, to record the selected information.

Filters: Airopeek offers a wide selection of built-in filters that can be used to capture frames that satisfy certain

criteria. It also allows new filters to be created. By using frame slicing and filters, the size of the resulting trace capture files can be reduced significantly.

3.2 802.11 Remote Distributed Sensor

The 802.11 Remote Distributed Sensor (henceforth referred to as a *sensor*) is an Ethernet-connected WLAN adapter that acts like a "listen-only" AP. The sensor is a compact wall mountable device that can be powered using AC or Power over Ethernet (PoE). The sensor operates on the 5.0 GHz and 2.4 GHz frequencies and can capture all 802.11a/b/g WLAN frames at a remote location. The sensor plugs into an Ethernet LAN and sends copies of captured WLAN frames back to Airopeek running on any network-accessible computer. The captured frames are encapsulated in UDP packets.

The firmware of the sensor supports channel hopping. The hopping order for the device is random with the following properties [18]: (a) Two-thirds of the time, the channel is chosen from the set of all selected channels; (b) One third of the time, the channel is chosen from the set of channels that have been identified by beacons. In both cases, no channels from the set are repeated until all channels (in the set) have been scanned.

The wireless traces were collected on a dedicated Dell OptiPlex GX270 2.8 GHz PC with 3 GB memory and 80 GB disk. The PC was time synchronized using the Network Time Protocol (NTP). Trace files were automatically transferred to a file server.

4 Qualitative Assessment of Methodology

The efficacy of the sensor in capturing frames depends on many factors, such as operating range, network congestion, and hardware limitations. Accurate determination of frames that are "missed" (not captured) by the sensor is a non-trivial task. Information regarding the frame miss rates of the sensor is necessary to determine the completeness of wireless traces and the accuracy of any subsequent analysis.

In most cases, we have to rely on the existing data set to infer the number of missed frames. This is because one may not be able to collect the wireless traces from the network routers. Also, statistics acquired using SNMP polling of APs are unreliable [6]. Thus, we have to come up with estimation techniques that provide a true measure of the missed frame counts. We implement and compare the following three methods for estimating the number of frames missed:

Beacon frames: Most APs transmit beacon frames every 100 milliseconds. Counting beacon frames provides a simple estimate of the continuity of WLAN coverage. We refer to this as the *beacon method*. While beacon frame sizes vary (usually 60-100 bytes), the beacon miss rate can be used to estimate the link quality between the AP and the sensor. If the sensor fails to capture a short beacon frame, the probability of capturing a large data frame is low. Increasing beacon miss rates may also indicate increased traffic intensity in the network. A sensor may be overwhelmed during such times, leading to more frame misses. Although this method may underestimate the actual number of missed data and management frames, it is relatively simple to calculate, and can quickly indicate if there is a serious problem with the completeness of a trace.

802.11 ACK frames: All data frames and certain management frames (except broadcasts) sent by a wireless station or AP are acknowledged (at the data-link layer) by the receiver after a short inter-frame space (SIFS). During this SIFS, no other wireless device in the operating range is allowed to send a frame. Acknowledgement (ACK) frames have the address of the sender in the MAC header. Ideally, such a situation would be represented in a trace as a data frame followed by an ACK frame. If an ACK is present and the corresponding data frame is missing, then it means that the sensor was unable to capture that data frame. By counting such ACK frames one can estimate the number of missed data and management frames. We refer to this as the *ACK method*.

MAC sequence numbers: All data and management frames (except retransmissions) sent by an 802.11 wireless device can be distinguished by a sequence number in the MAC header. Every time a wireless station or an AP sends out a data or management frame, the sequence number counter is incremented by 1. Sequence numbers can have any value in the $mod\ 4095$ set. Once the maximum value is reached, the counter wraps. By counting the gaps in the sequence numbers of frames captured by a sensor, the number of missed data and management frames can be estimated. We refer to this as the *sequence number method*.

4.1 Test Environment

We collected a test trace from the computer science department (CPSC) WLAN in the Information and Communications Technology (ICT) building between 10 pm Saturday April 29, 2006 and 10 pm Friday May 5, 2006. The CPSC WLAN is a non-encrypted single-channel 802.11 b/g network. It is restricted for use by the CPSC students, staff, and faculty members. The CPSC WLAN spans the 5^{th}, 6^{th}, and 7^{th} floors of the ICT building. The 5^{th} floor has 1 AP, and the 6^{th} and 7^{th} floors each have three APs. We placed a single sensor on the 7^{th} floor of the ICT building, in the southwest corner. The objective of our test was to use empirical data to measure the number of frames missed, the operating range, and the effect of environmental factors.

4.2 Metrics

We gauge the performance of the sensor using a metric called miss ratio. We measure two different miss ratios, namely the *beacon miss ratio* and the *frame miss ratio*.

Miss ratio is calculated using the following formula:

$$Miss\ ratio = \left(1 - \frac{Captured}{Total}\right) \times 100\%$$

$Captured$ is the number of captured beacons or frames in a time interval t (expressed in hours). When calculating beacon miss ratio, $Total = t \times 36,000$; since one beacon is sent every 100 milliseconds and one hour has 3,600 seconds, there are $\frac{3,600 \times 1,000}{100} = 36,000$ beacons transmitted per hour.

The frame miss ratio refers to the number of data and management frames missed by the sensor. When calculating the frame miss ratio, $Total = Captured + Estimated$. $Estimated$ is the number of frames missed by the sensor, as estimated by either the sequence number or ACK method.

4.3 Beacon Miss Ratio

We use an AP-centric view of the 7 APs in our WLAN. All data and management frames observed in the trace are sent either to the AP from clients (To-AP) or to the clients from the AP (From-AP). The beacon miss rate for three cases are presented: 'Best AP', 'Median AP', and 'Worst AP'. We label the AP for which the sensor captured the highest overall percentage of becaons the 'Best AP'. The median AP case divides the APs in the WLAN into two categories: the APs with lower beacon miss rates than the median AP, and those with higher beacon miss rates than the median AP. The AP for which the sensor recorded the highest overall beacon miss ratio is labeled the worst AP.

For context, we first provide an indication of the traffic volume observed on the WLAN. Figure 1 shows the number of frames captured by the sensor for the best AP, median AP, and worst AP. Although the frame rates for the APs are different, we observe several commonalities. First, there is a persistent background traffic load of 10 frames/second, due to beacon frames broadcast by the AP at regular intervals. Second, we observe the effect of diurnal and weekly usage patterns. The humps in the graph represent the work hours of weekdays, while the troughs represent nights. There is minimal traffic on April 30 because it was a Sunday.

Next, we consider the beacon miss ratio. The overall beacon miss ratios for all APs in the WLAN was relatively low, with the average hourly beacon miss ratio varying between about 2% for the best AP and about 9% for the worst AP.

The first row (a) of Figure 2 shows the variability of beacon miss ratio over the trace duration. The graphs show the percentage of beacons missed in each one-hour long interval. The results highlight the influence of traffic intensity, time of day, and network contention in the frame capture process. The extremely low beacon miss ratio for the best AP in Figure 2(a) is due to the close proximity of the sensor with the AP.

The best AP is located on the 6^{th} floor, directly below the room where the sensor was placed. In comparison to the best AP, the median AP has significantly higher beacon miss ratios. The median AP was located 30 metres away on the same floor as the sensor. In an indoor campus environment, where each floor has multiple rooms separated by walls, the chance of signal attenuation increases as the distance from the AP increases. Metal and reflective material cause signal attenuation. Distortion is also seen when RF waves are reflected off obstructions and the direct signal combines with the scattered signals.

The rightmost graph shows the beacon miss ratios for the worst AP. This AP was located in the northeast corner of the floor (the opposite corner from the sensor). The beacon miss ratio here shows high variability.

The beacon miss ratio allows us to understand the "wall penetration" of the monitored APs. It allows us to find how many physical barriers (walls) that frames (beacons) sent from the AP can traverse. This method indirectly provides us with an understanding of the type of wall construction in the building. For example, penetration would be lower in case of a concrete construction as opposed to drywall construction.

4.4 Frame Miss Ratios

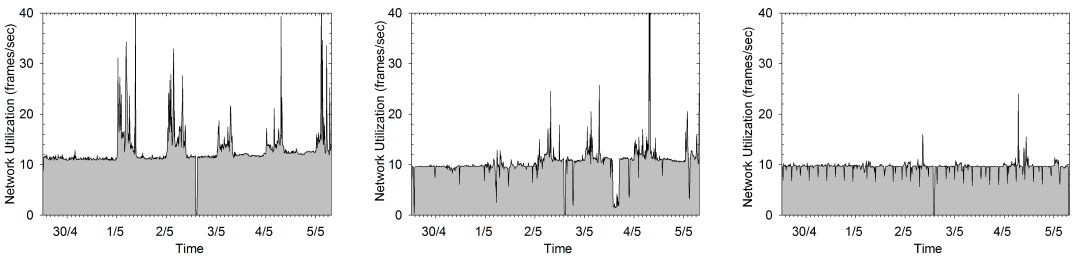
The beacon miss ratio provides the base case for quantifying the completeness of traces recorded by the sensor. The second row (b) of Figure 2 shows the frame miss ratio variations of the three selected APs using the sequence number method, while the third row (c) shows the frame miss ratio variations using the ACK method. The two methods provide differing views of the frame capture capability of the sensor. Before we delve into the reason for this and decide which method is more accurate, let us look at some of the similarities.[1]

The miss ratio observed depends on the directionality of the traffic. Both figures show higher miss ratios in the To-AP direction than in the From-AP direction. This phenomenon can be explained as follows. The wireless clients associated with an AP tend to be spatially distributed. The wireless devices also have a low-gain antenna and a limited reception range. In contrast, the APs in our test have high-gain external antennae. This means that a frame sent at a low signal level could be received by the AP. However, the sensor in the AP's vicinity may not be able to do so because it lacks a high-gain external antenna.

For the median AP, the To-AP frame miss ratios are even higher than those for the best AP. The above explanation applies in this case as well. If the sensor is situated farther away from the AP than the associated wireless clients, then the chances of not capturing a frame sent by the client increases.

For the worst AP, we observe an interesting phenomenon. Based on the sequence number method (Figure 2(b)),

[1] We did not estimate the number of missed frames to an AP in intervals when the number of captured frames was less than 20 (typically off-peak hours). Due to the constant transmission of beacons by the AP, we always had more than 20 captured frames in the From-AP direction.



(a) Best AP (b) Median AP (c) Worst AP

Figure 1. Amount of traffic transferred over 10 minute intervals

the From-AP miss ratio is 90% or higher for most of the trace. In this case the sensor is unable to capture any data or management frame from the AP because the AP is evidently out of the sensor's operating range. In contrast, the sensor is able to capture more frames in the To-AP direction due to the proximity of the sensor to a set of clients that are associated with the AP. Figure 2(c) reveals that the ACK method significantly underestimates the number of frames, as in this scenario the ACK frames were often missed when the corresponding data frame was missed, which causes this method to fail.

The miss ratio observed also depends on the traffic volume. The occasional spikes in Figure 2(b) and (c) can be explained using Figure 1. For example, the spikes in Figure 2(c) show a clear correlation to the traffic volume, which increases almost four-fold during work hours. These results indicate that during times of increased network activity a sensor may have higher miss ratios.

Table 1 summarizes the number of frames captured and missed (estimated) by the sensor. Table 1 shows that the From-AP captured frame counts are significantly higher than To-AP frame counts. This is due to the beacon frames (10 per second) emitted by the APs. The best AP has an overall miss ratio of 4% using the sequence number method and 2% using the ACK method. We see that the two methods show similar results for the best AP scenarios, while differing significantly on the median and worst AP cases.

4.5 Sequence Number and ACK Methods

In this section, we discuss the differences in the accuracy of missed frames estimate for the two methods. Consider a simple example where a WLAN consists of an AP and 2 clients. We assume that a total of 20 frames were transmitted or received by the wireless stations during the period of observation. We also assume that the sensor was able to capture 13 frames. This scenario is illustrated in Figure 3, with data frames in the left column, and the corresponding ACK frame in the right column. Note that frames with a grey background represent frames captured by the sensor. We further assume that the sensor has captured all previously transmitted frames.[2]

We now look at the sequence number approach and describe how it functions. Note that each station maintains its own sequence number counter. We serially look at all of the captured frames. Starting at frame number 3 (the first captured frame), we observe that the sequence number is 500. Because we assumed all previous frames were captured we do not register a miss. We do the same when we look at frame number 7. Similarly, processing frames 9 and 11 identifies no missed frames. When we look at frame 15, however, we register a miss. This is because there is a jump in the sequence numbers for the AP. Similarly, when we look at frame 19, we observe that sequence number of S1 is 102, while the last observed sequence number for S1 was 100. Thus, we estimate that two frames were missed. Summing all the misses, we find that there were three missed frames, which is consistent with the actual number of missed frames. This approach is similar to the method used by `ping` to measure round trip times; ping estimates losses using sequence numbers placed in the payload.

For the example in Figure 3, the ACK method does not identify any of the missed data frames.[3] This occurs because (in the given example) all of the captured ACK frames correspond to data frames that were also captured. Thus, in this example the ACK method underestimates the number of missed frames. In particular, if no ACK frames corresponding to missed data frames are captured, then this method fails to identify any missed frames.

As a result, the sequence number method tends to provide a more accurate approach for estimating the number of missed frames. However, there are some complications in applying this approach. For example, the sequence number counters are not reset when a client changes APs. Thus, long gaps may be introduced in the sequence numbers when a client switches from one AP to another and then back to the previous AP again. In this work we ignored long

[2]The implementation of the methods addresses issues that arise due to trace end effects.

[3]The ACK method could identify the missed ACK frame (frame 4), but we are primarily concerned with identifying missing data frames.

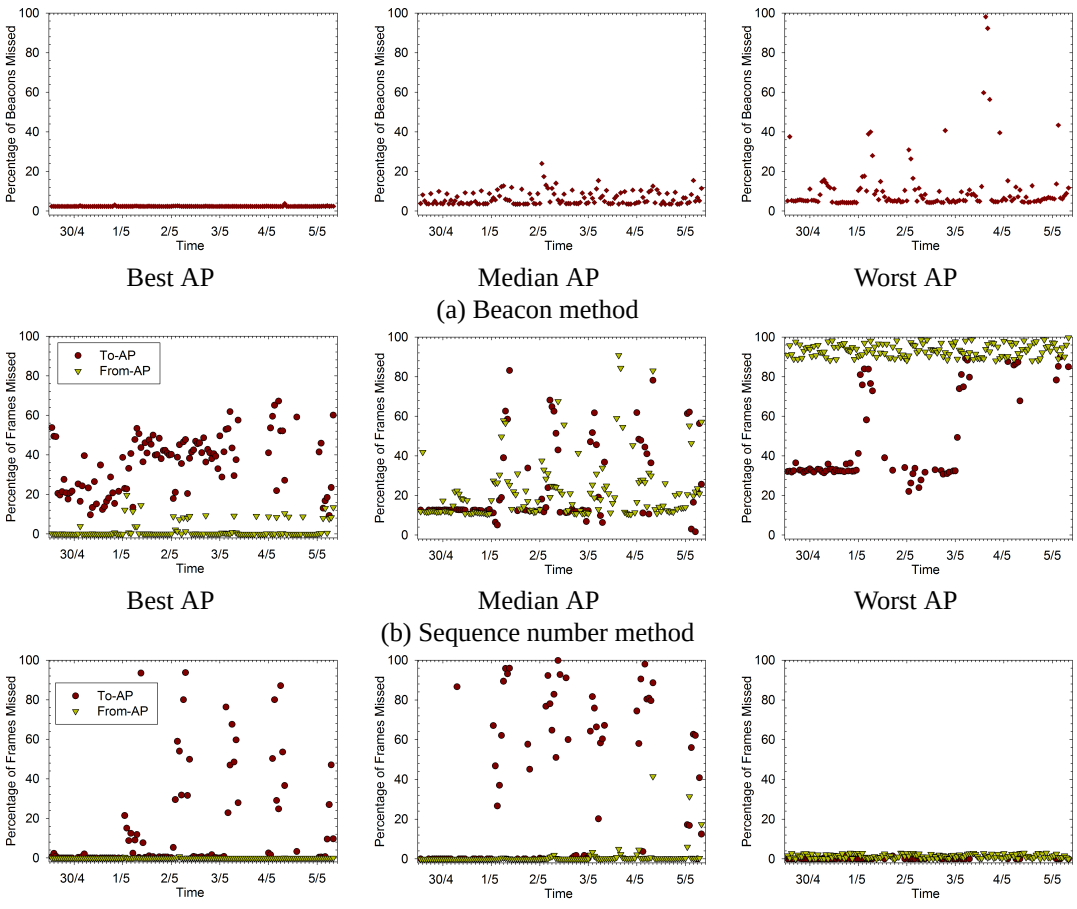


Figure 2. Frame miss ratios calculated using beacon, sequence number, and ACK methods

Frame	Type	From	To	Seq#	Frame	Type	From	To	Seq#
1	DATA	S1	AP	100	2	ACK	AP	S1	
3	DATA	S2	AP	500	4	ACK	AP	S2	
5	DATA	S1	AP	101	6	ACK	AP	S1	
7	DATA	AP	S1	1000	8	ACK	S1	AP	
9	DATA	AP	S1	1001	10	ACK	S1	AP	
11	DATA	S2	AP	501	12	ACK	AP	S2	
13	DATA	AP	S2	1002	14	ACK	S2	AP	
15	DATA	AP	S1	1003	16	ACK	S1	AP	
17	DATA	AP	S1	1004	18	ACK	S1	AP	
19	DATA	S1	AP	102	20	ACK	AP	S1	

Figure 3. Example showing how sequence number and ACK methods estimate missed frames

gaps that resulted due to these conditions (we elaborate on a similar issue below). A second issue to address is frame retransmissions. In our trace we observed a high number of retransmissions in the To-AP direction. This is due to some clients not receiving ACKs from the AP, which typically happens when the original frame never reached the AP (or was found to be corrupt and thus dropped by the AP). A client NIC will try to retransmit the same frame at most seven times, after which the old frame is discarded. Retransmissions carry the same sequence number, while a new frame is sent with an incremented sequence number.

A third complication arises because of vendor-specific implementation differences. In our traces we observed that D-Link APs used a separate sequence counter per associated station, rather than a single (global) sequence number counter. We also noticed jumps in the sequence numbers of some Intel NICs. Addressing these types of issues may require the use of heuristics, which complicates the sequence number method. For example, for several of the Intel NICs the sequence numbers of two consecutive frames sent from the NICs were not sequential. In one case the consecutive frames from the NIC had sequence numbers 1001, 1004, 108, 109, and 110. For the initial two frames we determined that two frames were missed by the sensor (sequence numbers 1002 and 1003). Between the third and fourth captured

Table 1. Captured and missed frames using sequence numbers and ACK methods

Frame	Best AP			Median AP			Worst AP		
	Captured	Missed (Estimated)		Captured	Missed (Estimated)		Captured	Missed (Estimated)	
		Seq Num	ACK		Seq Num	ACK		Seq Num	ACK
To-AP	202,900	92,822 (31%)	117,278 (37%)	92,300	89,071 (50%)	45,885 (34%)	417,204	411,327 (70%)	11,752 (2%)
From-AP	6,062,357	144,042 (2%)	1450 (1%)	4,977,740	1,948,304 (29%)	103,755 (2%)	743,165	942,040 (90%)	31,401 (3%)
Total	6,265,257	236,864 (4%)	118,728 (2%)	5,070,040	2,037,375 (29%)	149,610 (1%)	1,160,369	1,503,577 (58%)	43,152 (1%)

frames the sequence number jumps from 1004 to 108. In this case we do not register a missed frame. Here we used a threshold (i.e., a maximum difference of 50 between two consecutive sequence numbers) to determine if there were missed frames or a jump in sequence numbers. Obviously as the idiosyncrasies of a wider range of devices are identified, the heuristics may need to be updated. Similarly, a time-based threshold may also be required to address the roaming of a client among a set of APs.

We occasionally observed out of order frames in our trace, where frames with higher sequence numbers arrived before frames with lower sequence numbers. We believe that this is an artifact of the trace infrastructure and not the result of missed frames. This observation indicates the need for a more sophisticated approach than examining a trace on a frame-by-frame basis. For example, a buffer of the next N frames in the trace could be kept, and the frames re-ordered by sequence number before checking for missed frames.

5 Determining Sensor Placement

Before we can decide upon the vantage points for the sensors, it is essential that we measure their operating range. Here we are interested in determining at what distance the capture capability of the sensor reduces to zero. To achieve this, we conducted an experiment with a wireless notebook running a UDP ping client. A server is installed on a PC that is on the wired-side of the network. The wireless client sends out UDP packets to the server at a fixed rate. Upon receipt of the packet, the server returns it back to the client. We placed a sensor at a fixed location (7^{th} floor) to capture the packets exchanged between the client and the server. By varying the position of the client with respect to the sensor we can quantify the operating range of the sensor.

We conducted several trials of our experiment at different points of interests on the 7^{th} and 6^{th} floors of the ICT building. We refer to these points of interest as loci. We devise three metrics for this purpose, namely signal strength, miss probability, and CRC error probability. Signal strength (expressed as a percentage) is used to measure the RF energy level of a signal as experienced by the sensor. The average miss ratio for n trials is called miss probability. CRC error probability is the probability that a frame captured by the sensor is corrupt due to signal deterioration. We use these metrics to measure the quality of the link between sensor and AP (From-AP) and sensor and client (To-AP).

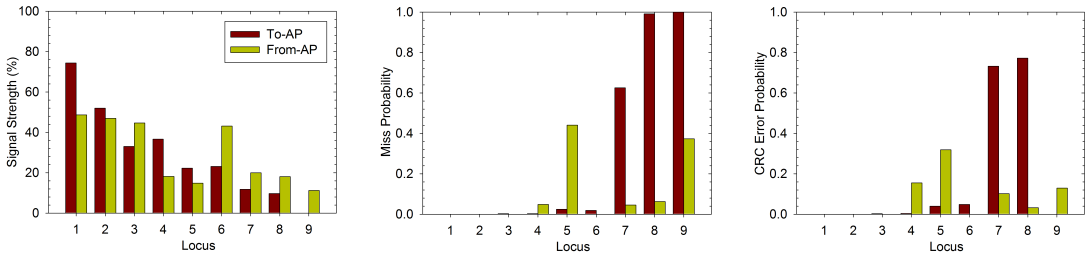
We configured the wireless notebook to send UDP packets with 512 bytes of payload at a rate of 150 packets/second. Traffic analysis of the WLAN showed us that the frame size distribution is bimodal. We thus chose a packet size of 512 bytes to represent the average size of a frame transmitted on the WLAN. The chosen packet rate is used to emulate a WLAN with high traffic intensity.

We conducted five trials per locus and during each trial at least 5,000 packets were sent in each direction. We identified loci on the horizontal plane (i.e., 7^{th} floor) as well as on the vertical plane (i.e., 6^{th} floor) of coverage of the sensor. The modus operandi for choosing the loci is as follows. Assuming the sensor to be the centre of an imaginary circle, we carried our experiments at arbitrary distances in four directions - north, south, east, and west. We used the same approach in the vertical plane. Figure 4 shows the relationship between signal strength as perceived by the sensor with its miss probability and CRC error probability. We only present results for 9 selected loci. Loci 1-5, 9 are on the horizontal plane and loci 6-8 are on the vertical plane.

Loci 1-3 represent best case scenarios for the sensor. Here the client (on the 7^{th} floor) is closest to the sensor. The sensor also happens to be in close proximity with the associated AP (on the 6^{th} floor). In all these three cases the signal strength is about 40% or higher (see Figure 4(a)). As the client moves away from the sensor, the signal strength of the To-AP frames comparatively decreases and errors increase. In Figures 4(b) and (c) we see that the miss probability and CRC error probability are near zero. When the client and AP are close to the sensor, the signals are strong and there is no measurement loss.

Locus 4 shows the scenario where the client (on the 7^{th} floor) is relatively far from the sensor and associated with an AP on the 6^{th} floor. With a larger coverage area on the horizontal plane the perceived signal strength of the client is higher (37%) than that of the AP (18%). This results in 4% of the frames sent by the AP being missed and an increase in number of corrupt frames (15%) that are received. In case of locus 5, although being on the same horizontal plane, the AP is farther away from the sensor than the client. This leads to a higher miss probability (44%) and CRC error probability (40%) for the From-AP traffic.

Locus 6 shows the case where both the client and the AP are situated on the vertical plane. The signal strength of the AP (43%) is twice that of the client (23%). This is due



(a) Signal Strength (b) Miss Probability (c) CRC Error Probability

Figure 4. Relation between signal strength, miss probability, and CRC error probability

to two reasons: the AP has better hardware (antenna) and has a better link quality than the client. Loci 7 and 8 show the narrowing coverage radius of the sensor in the vertical plane. In both cases the client and AP are distant from the sensor. Client traffic has high miss probabilities ($> 70\%$) and CRC error probabilities ($> 73\%$).

Locus 9 is the most interesting case. Here, both client and AP are on the horizontal plane. The AP is slightly closer in distance to the sensor than the client, but as the results indicate both are out of the sensor's operating range. We observe that 100% of the client traffic and 40% of the AP traffic is missed by the sensor. Additionally, 12% of the captured AP traffic is corrupt. Because the sensor captured not a single client frame, the CRC error probability is 0.

Figure 5 and Figure 6 show time-series representations of measured values for two selected locations (Loci 5 and 6). In both figures, there are four graphs, showing: (a) variation of signal strength, (b) the number of frames captured, (c) the number of captured frames with failed checksums, and (d) the number of MAC layer retransmissions. In Figure 5(a) we observe that the signal strength of the AP is lower than that of the client. Comparing Figure 5(a) and (b) we see that whenever the AP signal strength dips below 15%, we see a decrease in the number of AP frames captured. Subsequently, we notice an increase in the rate of CRC error frames. Figure 5(d) shows the number of retransmitted frames over time. This measure tells about the link between the client and AP. We found that MAC-layer retransmissions are common during our experiments. This is mostly due to the receiver not sending an ACK to the sender or the received ACK being corrupted. Both events depend on the wireless environment.

In Figure 6 we see a different picture. Here, perceived signal strength from the client and the AP is higher than 20%. In Figure 6(b) we find the lines showing frames captured in each direction coinciding, indicating a good link between the sensor and the AP/client. This leads to very low measurement loss. Because the AP has a signal strength greater than 40%, we do not see any CRC error frames in the From-AP direction, however, there are some observed in the To-AP direction. Figure 6(d) is consistent with Figure 5(d), where we observe a steady rate of MAC-layer retransmissions.

Table 2. Frames captured and missed from the WLAN

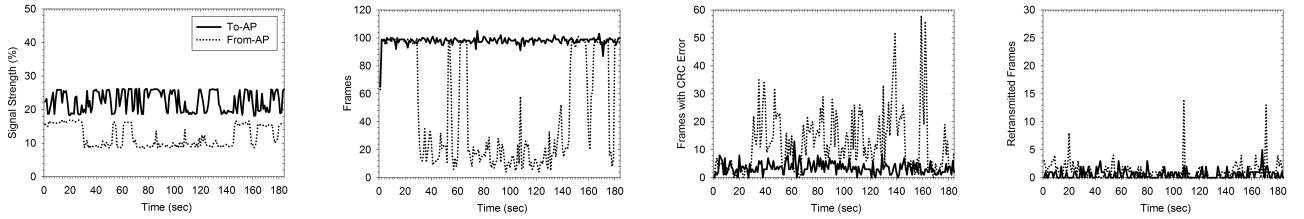
Floor	AP	Captured	Missed (Estimated)
7^{th}	1	875,648	4,666 (1%)
	2	828,760	28,815 (3%)
6^{th}	3	961,705	9,460 (1%)
	4	928,273	5,609 (1%)
	5	887,247	67,504 (7%)
5^{th}	6	1,473,947	68,280 (4%)
WLAN		5,955,580	184,328 (3%)

These results indicate that for the sensor to capture almost all of the traffic (both client and AP), the perceived signal strength must be greater than 40% in each direction. While much of the traffic can still be captured at 20% signal strength, the tradeoff is more captured frames with failed CRC checksums. The radius of coverage of the sensor on the vertical plane is significantly lower than that on the horizontal plane.

6 Sensor Layout

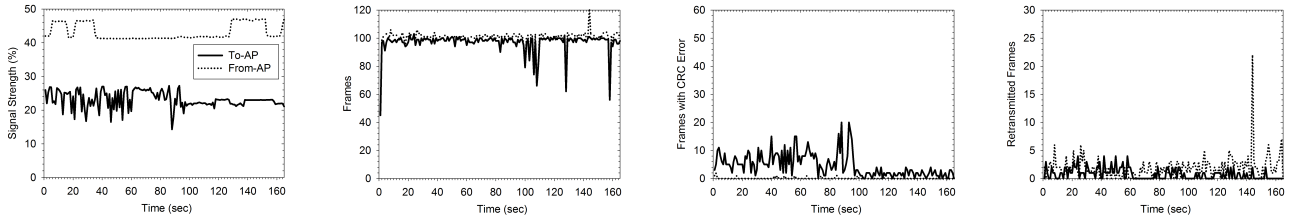
We utilize the results obtained from the tests described in Section 5 to present a sensor layout scheme with low measurement loss. We deployed 4 sensors in our WLAN: two on the 7^{th} floor, one on the 6^{th} floor, and one on the 5^{th} floor. Figure 7 illustrates the exact location of the sensors and APs. The coverage area of the sensors are represented using circles with broken lines, much like contour lines on a geographical map. Note that there are two types of circles in the figure. The circle with the larger area represents the operating range of the sensor on the same floor (i.e., horizontal plane). For example, the sensor situated on the 5^{th} floor covers all the rooms on that floor. The smaller (oval-shaped) circle represents the operating range on floors above or below the floor on which the sensor is placed. For instance, the smaller circle on the 7^{th} floor shows the range for any WLAN traffic captured by the sensor on the 6^{th} floor. The perimeter of the circle marks the distance at which the sensor's perceived signal strength diminishes to 40%.

To understand the effectiveness of this specific layout,



(a) Signal Strength (b) Frames Sent and Received (c) CRC Error Frames (d) Wireless Retransmissions

Figure 5. Details for Locus 5



(a) Signal Strength (b) Frames Sent and Received (c) CRC Error Frames (d) Wireless Retransmissions

Figure 6. Details for Locus 6

we collected a 24-hour trace during a workday from all the deployed sensors. The sensors collectively captured approximately 6 million frames. Table 2 shows the number of frames captured and missed frames for the trace. We present results for 6 APs only as one of the APs on the 7^{th} floor was not functional during trace capture. We employed the sequence number method for estimating missed frame counts. We observe that our layout provides a holistic view of the WLAN, with only 3% of overall traffic remaining uncaptured. We intend to implement this scheme on a much larger scale and use the resultant traces for WLAN analysis.

7 Related Work

Yeo et al. [20, 21] studied the difficulties associated with wireless-side measurement of WLANs. The authors set up a controlled WLAN environment (restricted to a single floor) to evaluate a technique for accurately capturing frames from the wireless medium. Specifically, they found that using multiple sensors can reduce the number of uncaptured frames. Their results suggest that one sensor should be placed near the target AP and the remainder of the sensors should be positioned close to the predicted locations of clients. Our work complements and extends these prior works. We provide a way to understand the completeness of a trace and the operating range of sensors. Once the completeness exceeds a certain threshold (which may vary depending on the intended use of the trace), then adding additional sensors becomes unnecessary (and more costly). We also present a robust scheme for sensor layout across multiple floors to capture traffic from a production WLAN.

More recently, Jardosh *et al.* [10] used three laptop sensors to study link-layer behavior in a congested WLAN. Rodrig *et al.* [13] took wireless measurements using five PC sensors from the SIGCOMM 2004 conference WLAN to study the operational behavior of the 802.11 MAC protocol. Both studies used the RFMON measurement design and used the ACK method to determine measurement losses. Cheng *et al.* [5] developed a system called Jigsaw that provides large scale synchronization of wireless traces from distributed sensors. They determined their trace accuracy by capturing artificial traffic workloads on the wireless-side and comparing to traces captured on the wired-side. Our work is orthogonal to these. They focused on MAC-layer analysis and trace merging, while our work presents a methodology for efficient collection of traces from the wireless-side and measurement loss estimation.

8 Conclusions

In this paper we examined three different methods (beacon, ACK, and sequence number) for estimating the completeness of wireless traces. The methods differ in the features they examine, their simplicity, and their accuracy. We found the sequence number method to be the most accurate, although its implementation is complicated by the idiosyncracies of different wireless devices.

We also examined the placement of sensors within WLAN environments, with the goal of improving the completeness of the collected traces, while minimizing the number of sensors needed. We found that placing sensors in locations where the signal strength of client-AP communications is at least 40% results in relatively complete traces with a few sensors.

As part of future work we intend to examine additional methods for evaluating the completeness of wireless traces. For example, hybrid approaches of two or more of the tested methods could provide more accurate estimates of the num-

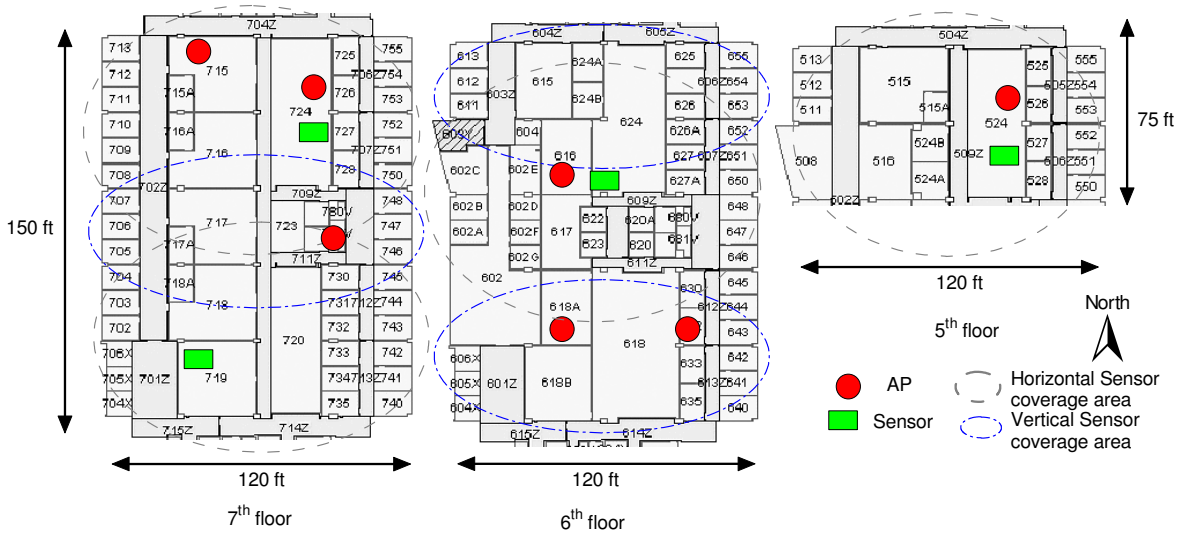


Figure 7. Layout of sensors and APs in the WLAN

ber of frames missed by a sensor. We also plan more controlled tests to better quantify the accuracy of each method.

9 Acknowledgements

Financial support for this research was provided by Informatics Circle of Research Excellence (*i*CORE) in the Province of Alberta, as well as by Canada's Natural Sciences and Engineering Research Council (NSERC). The authors thank the CPSC technical support staff for deploying the sensors.

References

[1] A. Balachandran, G. Voelker, P. Bahl, and V. Rangan. Characterizing User Behavior and Network Performance in a Public Wireless LAN. In *SIGMETRICS*, 2002.

[2] M. Balazinska and P. Castro. Characterizing Mobility and Network Usage in a Corporate Wireless Local-area Network. In *MobiSys*, 2003.

[3] D. Blinn, T. Henderson, and D. Kotz. Analysis of a Wi-fi Hotspot Network. In *WiTMeMo*, 2005.

[4] BusinessWeek. Wi-Fi Gaining Traction, October 2005. `http://www.businessweek.com/technology/tech_stats/wifi051003.htm`.

[5] Y. Cheng, J. Bellardo, P. Benko, A. Snoeren, G. Voelker, and S. Savage. Jigsaw: Solving the Puzzle of Enterprise 802.11 Analysis. In *SIGCOMM*, 2006.

[6] T. Henderson and D. Kotz. Problems with the Dartmouth Wireless SNMP Data Collection. Tech. Report, Dartmouth College, 2003. `http://www.cs.dartmouth.edu/~dfk/papers/henderson:problems.pdf`.

[7] T. Henderson, D. Kotz, and I. Abyzov. The Changing Usage of a Mature Campus-wide Wireless Network. In *MOBICOM*, 2004.

[8] R. Hutchins and E. Zegura. Measurements from a Campus Wireless Network. In *ICC*, 2002.

[9] A. Jardosh, K. Ramachandran, K. Almeroth, and E. Belding-Royer. Understanding Congestion in IEEE 802.11b Wireless Networks. In *IMC*, 2005.

[10] A. Jardosh, K. Ramachandran, K. Almeroth, and E. Belding-Royer. Understanding Link-layer Behavior in Highly Congested IEEE 802.11b Wireless Networks. In *SIGCOMM E-WIND*, 2005.

[11] Kismet Wireless FAQ. `http://www.kismetwireless.net/documentation.shtml`.

[12] T. Ojala, T. Hakanen, T. Mäkinen, and V. Rivinoja. Usage Analysis of a Large Public Wireless LAN. In *WirelessCom*, 2005.

[13] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, and J. Zahorjan. Measurement-based Characterization of 802.11 in a Hotspot Setting. In *SIGCOMM E-WIND*, 2005.

[14] D. Schwab and R. Bunt. Characterising the Use of a Campus Wireless Network. In *INFOCOM*, 2004.

[15] K. Simler, S. Czerwinski, and A. Joseph. Analysis of Wide Area User Mobility Patterns. In *WMCSA*, 2004.

[16] D. Tang and M. Baker. Analysis of a Local-area Wireless Network. In *MOBICOM*, 2000.

[17] WildPackets. `http://www.wildpackets.com/`.

[18] WildPackets. *Airopeek NX User Manual*. 2003.

[19] WildPackets. Remote Analysis of a Wireless LAN Environment. White Paper, 2003. `http://www.wildpackets.com/elements/whitepapers/WirelessLAN.pdf`.

[20] J. Yeo, M. Youssef, and A. Agrawala. A Framework for Wireless LAN Monitoring and its Applications. In *WiSe*, 2004.

[21] J. Yeo, M. Youssef, T. Henderson, and A. Agrawala. An Accurate Technique for Measuring the Wireless Side of Wireless Networks. In *WiTMeMo*, 2005.