Unwanted Link Layer Traffic in Large IEEE 802.11 Wireless Networks

Ramya Raghavendra[†], Elizabeth M. Belding[†], Konstantina Papagiannaki[‡],
Kevin C. Almeroth[†]
[†]Department of Computer Science, University of California, Santa Barbara
[‡]Intel Research, Pittsburgh
{ramya, ebelding, almeroth}@cs.ucsb.edu, dina.papagiannaki@intel.com

Abstract—Wireless networks have evolved into an important technology for connecting users to the Internet. As the utility of wireless technology grows, wireless networks are being deployed in more widely varying conditions. The monitoring of wireless networks continues to reveal key implementation deficiencies that need to be corrected in order to improve protocol operation and end-to-end network performance. In wireless networks, where the medium is shared, unwanted traffic can pose significant overhead and lead to suboptimal network performance. Much of the previous analyses of unwanted traffic in wireless networks have focused on malicious traffic. However, another major contributor of unwanted traffic is incorrect link layer behavior. Using data we collected from the 67^{th} Internet Engineering Task Force (IETF) meeting held in November 2006, we show that a significant portion of link layer traffic stems from mechanisms that initiate, maintain, and change client-AP associations. We further show that under conditions of high medium utilization and packet loss rate, handoffs are initiated incorrectly. We analyze the traffic to understand when handoffs occur and whether the handoffs were beneficial or should have been avoided.

1 Introduction

IEEE 802.11-based WLANs have experienced rapid growth in recent years as the chief means of providing Internet connectivity to users. Large WLAN deployments are popular in locations such as conferences, university campuses, hotels, and airports. A 2006 survey[1] shows a significant increase in mobile application deployment in North American enterprises (as compared to 2004) wherein 63% already use in-house WLANs and 58% plan to increase their WLAN investment. These networks are characterized by a large number of access points (APs) that are densely deployed to support network usage by many simultaneous users. Dense AP deployment helps ensure that the overall user demand is met and network coverage is provided, especially if users are mobile.

One of the limitations of IEEE 802.11 WLANs is the limited number of orthogonal channels, three in the case of 802.11b/g and 12 in the case of 802.11a. Because there is a limited number of orthogonal channels, it is commonly the case that a large WLAN deployment has several APs within range of each other, and often multiple of these APs are configured to transmit on the same channel. Large WLAN deployments are hence likely to suffer from high interference and high loads. This is particularly true when WLANs need to support flash crowds, which are defined as sudden surges in the number of users attempting to connect to and access the WLAN [1]. Increased interference and load gives rise to several problems such as intermittent connectivity, low throughput and high losses, resulting in an unreliable network and sometimes a complete breakdown.

To investigate the prevalence of the aforementioned problems in WLANs, we collected traces from the 67^{th} Internet Engineering Task Force (IETF) meeting held in San Diego in November 2006. The network consisted of over 100 APs on both 802.11a and 802.11g networks, and was used by more than 1200 users over a span of five days. We collected both the 802.11a and 802.11g traces for four of the five days, resulting in, to the best of our knowledge, the most comprehensive trace of a large conference WLAN to date. Our analysis of the traces shows that the network suffered from high interference and loss rates. There was significant overhead on the clients and APs to transmit a single frame of useful data. Repeated association and reassociation attempts, due to lost connections, aggravated the problem. The result was that clients could only maintain a short association period with an AP, leading to both suboptimal network performance and deterioration in application performance.

Unwanted traffic has been used to refer cumulatively to those traffic components originated directly or indirectly either by malicious or "non productive " activities [2]. Much of the previous analysis of unwanted traffic in wireless networks has focused on malicious traffic. However, another major contributor of unwanted traffic are the applications that aggressively attempt to maintain connectivity and high quality client service, leading to undesirable traffic on the link layer. Through the analysis of the IETF traces, our goal is to understand the causes for high overhead and short association times of clients. We observe that a significant portion of the network overhead stems from mechanisms that initiate, maintain, and change client-AP connectivity. We show that much of this traffic is unnecessary, and actually compounds the problem of maintaining client-to-AP

1. http://www.forrester.com/Research/Document/Excerpt/-0,7211 ,40720,00.html

associations. We identify two such mechanisms that contribute to unwanted traffic on the wireless network:

- Keepalive traffic that is used to maintain client-AP associations in the absence of data traffic.
- The probing mechanism used by clients to frequently collect neighbor information.

Unwanted traffic is detrimental to the performance of large wireless networks such as that deployed during the IETF as it leads to missed transmission opportunities and inefficient medium utilization. As a result, clients erroneously conclude that they have lost their connections to their APs, and hence initiate handoffs. As congestion increases, the rate of handoffs increases, even in the absence of mobility. We show that a majority of these handoffs are unnecessary and at times negatively impact throughput, wherein the clients' throughput suffered immediately following a handoff.

Analysis of such unwanted traffic is very important to understand and improve the performance of congested networks. We believe that the problems identified in this trace are not unique to the IETF network. These problems can occur in any wireless network, particularly large networks that are deployed to support many simultaneous users. Recent studies have identified key implementation deficiencies in frame retransmissions, frame sizes and rate adaptation in congested networks [3], [4], [5].

Our study continues to identify key deficiencies in the 802.11 protocol and its implementations in adapting to conditions of high usage and congestion. These insights will be useful in designing systems and protocols that are more adaptive to network conditions. We believe that through protocol improvement and better implementations, the ability of large scale networks to handle high loads can be significantly enhanced.

In an earlier work, we analyzed the handoff behavior of clients in congested environments [6]. We showed that clients perform handoffs at a high rate in a congested network, some of which lead to a throughput deterioration. In this paper, we make the following contributions:

- We show that client overhead increases with the increase in network density.
- We analyze the two types of overhead mechanisms that are prominent in a congested network - probes and keepalive packets.
- We show that the handoff rates increase with an increase in network utilization, even in the absence of mobility.
- We perform handoff analysis for different card vendors, and show that the behavior of cards across vendors is relatively consistent.

The remainder of this paper is organized as follows. Section 2 presents the related work and motivates the study of unwanted traffic and client associations. An overview of 802.11 frame types and the handoff process is described in Section 3. In Section 4, we provide details on the IETF network, monitoring methodology and the network usage characteristics. Section 5 discusses our findings on the unwanted traffic in the IETF network. We report our analysis on handoff behavior observed in section 6. Finally, we conclude our work in Section 7.

2 Related Work

Over the last few years, several studies have examined wireless network traces to understand the usage and performance of these networks. Starting with studies that focus on analysis of wireless network usage in campuses [7], [8], metropolitan-area networks [9] and mobility models [10], research has progressed to analyze the performance of these networks, including application workloads and session durations [11], [12]. These studies were based on the analysis of wired distribution network traffic and polled SNMP management data. As a result, these studies focus on *how* networks were used and how they perform but do not provide insights into *why* the networks operated or applications performed in a particular way.

To address this gap, recent studies have analyzed traces captured from the wireless side of the network using monitors. Yeo *et al.* were one of the first to capture link layer information and analyze the performance of a campus network [13]. This work identifies the challenges of wireless monitoring and explores the feasibility of merging traces from multiple sniffers using beacon frames.

Jardosh *et al.* monitored the IETF conference and analyzed link layer traces to understand congestion in wireless networks [3], [5]. Their work identifies some key deficiencies of the 802.11 protocol in congested environments with respect to rate adaptation, frame sizes and the RTS-CTS mechanism.

Rodrig *et al.* collected link layer traces from the SIGCOMM conference and analyzed the causes for high retransmission rates in the network [4]. Their work identifies that both contention and wireless transmission errors are the cause for retransmissions, and retransmissions due to contention affect rate adaptation in an incorrect way. Congestion-aware rate adaptation policies that take medium utilization into account have shown up to a 300% increase in throughput in congested environments, compared to other well known adaptation schemes [14].

While much of the previous work focused on the effect of congestion on retransmissions and rate adaptation, none of the studies focused on unwanted traffic on the wireless side. Prior studies in unwanted traffic have focused on malicious traffic in WLANs [2], [15]. However, unwanted traffic can result from protocol operations, and an inability of the protocol implementations to adapt to the environment. We show that such traffic can result in significant overhead and performance deterioration in the network.

A number of studies have evaluated the performance of 802.11 handoff mechanisms. Mishra *et al.* performed an empirical analysis of handoffs using cards from several vendors and identified that the probe mechanism is the main cause of handoff latency [16], and that this latency is significant enough to reduce application performance. Several improvements have been suggested to perform faster handoffs [17], [18], [19]. Recent studies have also shown that the current AP selection and triggering mechanisms are sub-optimal. Mhatre *et al.* showed that the use of long term iaveraged signal strength instead of instantaneous signal strength measurements results in better handoff decisions [20]. Potential bandwidth available after the handoff [21] and the quality of the AP's connection to the Internet [22] have

	Management Frame Subtype	Description
AUTH	Authentication Frame	Used by clients and APs for exchanging credentials.
DEAUTH	Deauthentication Frame	AP sends to a client when it wishes to terminate secure communication.
ARQ	Association Request	Client sends to AP when it wishes to connect to the AP.
ARP	Association Response	AP responds to the client's request with acceptance or rejection.
RRQ	Reassociation Request	Client sends to a new AP when the connection with the old AP has been lost.
DASS	Disassociation Frame	Client or AP use this frame to terminate an association.
BCN	Beacon Frame	AP sends periodically to announce its presence.
PRQ	Probe Request	Client broadcasts to obtain information on neighboring APs.
PRP	Probe Response	AP sends information in response to a probe request.

TABLE 1
Overview of IEEE 802.11 management frame types.

been suggested as better AP selection mechanisms than signal strength.

The above handoff studies are conducted on experimental testbeds in controlled conditions, and do not analyze protocol behavior in real settings. We believe that understanding how handoff mechanisms operate in a real network is essential to improve the existing algorithms. In our work, we show that current handoff mechanisms do not differentiate losses based on congestion. State of the art techniques such as beacon loss cannot be used to initiate handoffs in a congested network where the loss rate is high. We believe that the insights gained from this work will help in the design and implementation of better handoff techniques for large WLANs.

3 IEEE 802.11 Frame Types

Before we analyze our collected traces for unwanted traffic and handoffs, we begin with a brief overview of the various 802.11 frames and the role of each of the frame types in the client-AP association process. We limit the scope of this description to the aspects essential for understanding the protocol operations discussed in the paper.

The IEEE 802.11 standard defines three frame types: 1) Management; 2) Control; and 3) Data. Management frames enable the stations to establish and maintain connections. Control frames assist in the delivery of data frames. Data frames carry the application data and header information. Each frame type is comprised of several subtypes, each of which is used for a specific purpose in the protocol operation. Since we focus primarily on the analysis of management traffic in this paper, we limit the scope of this section to management frame subtypes. Table 1 summarizes the management frame subtypes and their role in the client-AP association process.

Handoff procedure: A client that wishes to join a network begins by authenticating itself to the AP. On successful authentication, the client sends an association request along with its radio capability information, such as supported data rates. The AP allocates resources for the client and sends its own information such as association ID and supported rates. Once a client is authenticated and associated, it can communicate with other clients through the AP as well as other systems on the distribution side of the AP.

When a client moves and loses connectivity to the AP, it starts gathering information on the APs present within its vicinity by broadcasting probe messages. The client receives responses from potentially multiple APs, and based on some implementation-dependent policy, it sends a reassociation request to one of the APs. The AP responds with either a success or a failure. On a successful response, the client is associated with the new AP. This process is called a *Layer 2* (L2) handoff. In some cases, such as enterprise networks, the pre-handoff AP exchanges client-specific context information with this new AP.

A L2 handoff consists of four phases i) triggering; ii) discovery; iii) AP selection; and iv) commitment [17], [20]. In the trigger phase, a handoff is initiated when a wireless client identifies the need to associate with another AP. When a trigger is generated, the client collects information about the APs in the vicinity, called the "discovery" phase. In the "selection" phase, the client identifies one AP that meets the particular vendor-specific performance criterion, usually signal strength. In this case, clients associate with the AP with the highest value of the Received Signal Strength Indicator (RSSI). Finally, in the "commitment" phase, the client disassociates with the current AP and reassociates with the new AP.

4 Data Collection Methodology

In this section, we first describe the IETF wireless network architecture. We then explain our monitoring framework, and finally, some of the challenges of this framework.

4.1 The 67^{th} IETF Network Configuration and Data Collection Framework

The IETF network consisted of 55 Cisco and D-Link Access Points (APs), spread across the East and West Towers of the hotel. The conference rooms were in the West Tower, which had 38 APs. Each AP was equipped with dual radios, with one radio tuned to operate on the 5 GHz spectrum (802.11a network) and the other on 2.4 GHz spectrum (802.11b/g network). Thus, there were 76 APs in total in the West Tower where we installed our monitoring setup. We focused our monitoring efforts on a subset of these APs to capture the client behavior during the daily sessions. To enable spatial reuse, the APs on the 802.11g network were configured on three orthogonal channels, 1, 6, and 11, and the APs on the 802.11a network were configured on four orthogonal channels, 36, 40, 44, and 48.

Figure 1 shows the AP and sniffer locations in the rooms at the conference venue. The APs did not support load balancing,

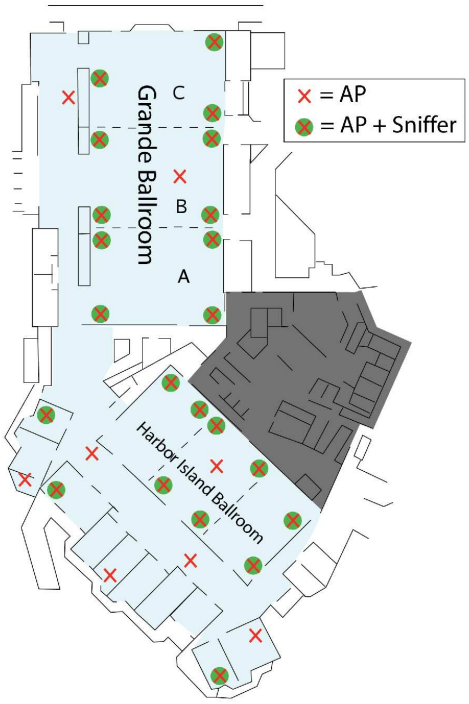


Fig. 1. IETF floor plan with AP and sniffer locations. Only the APs located in the conference rooms are depicted.

transmission power control or dynamic channel assignment. We used the *vicinity sniffing* technique to collect data from the MAC layer [13], [5]. This is a technique in which a set of wireless devices, known as *sniffers*, are deployed to passively monitor the packets in the wireless medium. A total of 12 sniffers were deployed in the conference rooms at various locations, which were chosen based on the number of users in the rooms. These locations used during the week are indicated in Figure 1. The sniffers were placed directly underneath the AP to maximise the likelihood of capture of all the packets received by and sent from the APs.

The sniffers were IBM R32 and T40 ThinkPad laptops with linux 2.6 kernel. Each sniffer was equipped with an Atheros chipset 802.11a/b/g PCMCIA card. The radios were configured in "monitor mode" to capture all packets. In this mode, we are able to capture all MAC layer frames including the control and management frames. In addition, the prism header information, which contains send rate, received signal strength, and noise level was also recorded for each packet. We captured the first 250 bytes of the packet to record header information only. Packets were captured using the *tethereal* utility.

Meetings were held in two separate sessions, the day and the late evening sessions, the latter of which is also called the *plenary*. We monitored the network during both the day and plenary sessions using different sniffer configurationsi described as follows.

Day session: The day sessions were held between 09:00 hrs and 17:30 hrs from November 6-10, 2006. Each day session was divided into six to eight parallel tracks, each of which was held in one of the conference rooms. During the day sessions, we collected usage statistics in the beginning of the day and ranked the APs based on the number of users associated with each. We configured the sniffers to monitor the top 12 ranked APs for the entire day. Some of the sniffers were thus configured on the 802.11g network, and the rest on 802.11a network, depending on the usage.

Plenary session: The evening sessions were held on November 9^{th} (Plenary I) and 10^{th} (Plenary II) between 17:00 hrs and 19:30 hrs (on these days, the day sessions ended at 16:00 hrs). During the plenary sessions, the partitions between the *Grande Ballroom* A and B were removed, so that the entire room could used. Sniffers were placed in this room at the locations shown in Figure 1 underneath the eight APs. During plenary I, we configured eight sniffers on the 802.11a network (one underneath each AP), and four on the 802.11g network, placed below the four APs located at the entrance of the Grande Ballroom. During plenary II, we similarly placed the sniffers; however, in this case the eight sniffers monitored the APs on the 802.11g network and four monitored the 802.11a network.

Challenges: While the vicinity sniffing technique facilitates capture of data, control, and management frames on the wireless side of the network, there are multiple challenges, as indicated in previous work [5], [23]. One of the critical challenges of this technique is unrecorded frames, and how to reconstruct missing frames using data from multiple sniffers. This problem has been addressed in the work by Mahajan *et al.* [23]. A challenge in our setup was to determine how reliably the sniffer detected packets that the AP received. If the receive sensitivity of the radio in the AP was higher than the receive sensitivity of the sniffer's radio, the sniffer would not record every packet that the AP successfully received.

For accuracy of analysis, we need to determine the reliability of the sniffer in capturing all the packets on the wireless medium. To this end, we compute the *sniffing fidelity*, defined as the ratio of frames received by the AP to frames undetected by the sniffer. We use the strict frame sequencing defined by the 802.11 protocol to compute the sniffer reliability. That is, for every response logged by the sniffer, there should have been a corresponding request from the client. The sniffer was placed directly below the AP and so we can safely assume that it should have received the vast majority responses from the AP.

For every message received from the AP, we need to make sure that a corresponding message was received from the client. To do this, we first need to determine which of the request-response frames can be used for the computation of sniffing fidelity. We cannot use the RTS-CTS pair since the 802.11g mechanism uses CTS-to-self packets to silence 02.11b neighbors. Probe request-response messages also cannot be used since probe requests are broadcast and all APs that receive a request transmit a reply. Fortunately, the association (ARQ, ARP) and reassociation (RRQ, RRP) request and response frames arrive atomically and can be used in our computation. Similarly, the DATA-ACK arrival atomicity can be leveraged. The drawback of this technique is that it does not account for missing frames when both the request and response are unrecorded. However, since the sniffers were placed directly beneath the APs, the

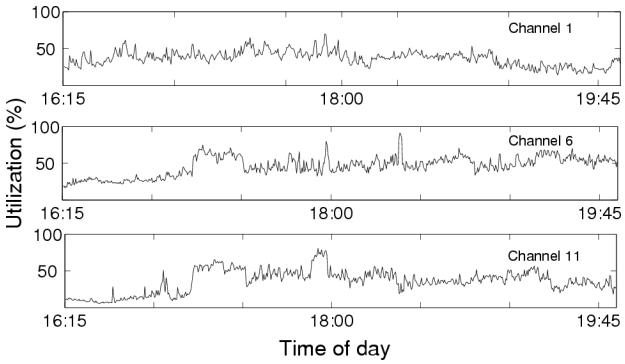


Fig. 2. Airtime utilization of Channels 1, 6 and 11 over one second intervals.

probability of missing a response is low and we obtain a close estimate.

Sniffing fidelity is computed as the ratio of the number of response frames to the number of request frames recorded by the sniffer, given by:

$$\text{Sniffing fidelity} = \frac{N_{ARQ} + N_{RRQ} + N_{ACK}}{N_{ARP} + N_{RRP} + N_{DATA}} \quad (1)$$

The average sniffing fidelity of the eight sniffers during the plenary session varied between 0.9 and 0.96. This implies sniffers captured at least 90% of the frames and as many as 96%. While we believe that the results obtained from the analysis of the trace will not be significantly altered if the missing frames were also present in the analysis, better techniques are required to determine accurate sniffer locations during the trace collection for maximum fidelity. Investigation of such techniques is an area left for future work.

4.2 Data Set Analysis

Over 140 gigabytes of uncompressed wireless network traces were collected during the week. With a goal of analyzing network behavior under conditions of high load and network activity, we focus on the 802.11g network during the plenary II session. There were three times as many users on the 802.11g network as there were on the 802.11a network, and hence, the effects of high network usage were more pronounced.

Previous studies have collected data at a single vantage point and analyzed the client's performance in terms of throughout, rate adaptation and retransmissions. While some initial efforts[2] exist to analyze handoff behavior in wireless networks, to the best of our knowledge this is the first attempt to capture wireless data from the entire network's perspective and perform handoff analysis for a network of this scale.

We perform preliminary analysis on the captured data to compute the network usage statistics. We characterize the network scale in terms of utilization, number of APs and clients. We then characterize the performance in terms of loss rates.

Network utilization: Figure 2 shows the airtime utilization on all three channels of the 802.11g network. We compute airtime utilization at each second as the sum of the time spent transmitting all data, management, and control frames recorded by the sniffer, and the total number of delay components, such as the Distributed Inter-Frame Spacing (DIFS) and Short Inter-Frame Spacing (SIFS) intervals. The APs supported both long and short preambles. We have used the transmission time for the long preamble, namely 192 μs, in our computation. Each point on the graph is an average over a 20 second interval. We observe that the utilization level increases at around 17:00 hrs, when the plenary begins. The network continues to be heavily utilized throughout the plenary.

Number of access points: Since the network at the IETF was densely deployed, we expect multiple APs on each channel to be within the range of each of the sniffers. In Figure 3(a), the x-axis shows the number of APs within the sniffers' range, and the y-axis shows the percentage of time the sniffers detected that number of APs. The figure shows that the sniffers had between one and eleven APs in range. We observe that about 90% of the time, between four and eight APs on the same channel were within range of the sniffers. The beacon frames that are periodically broadcast by the APs are used to compute the number of APs in range for that interval. The cumulative percentage is computed for all the sniffers over the entire length of the plenary.

Number of users: Figure 3(b) shows the instantaneous number of clients detected per second, summed over all channels, throughout the plenary session. The vertical lines mark the duration of the plenary. A client who transmitted at least one data frame during a one second interval is said to be present in that interval. For visual clarity, the average value for 20 seconds is represented in the figure. A maximum of 300 users were detected to be simultaneously present on the network, and the number of users continues to be high, over 150, throughout the session.

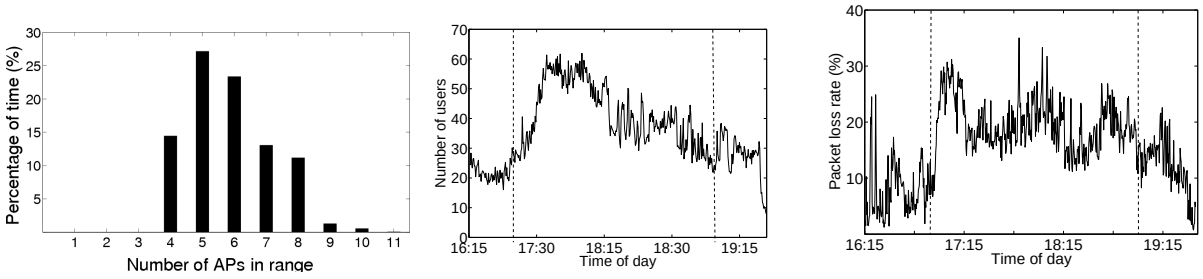
Loss: We compute loss rate as the number of MAC layer frames marked as retransmissions. Figure 3(c) shows the instantaneous packet loss rate in the network for every one second interval during the plenary. The vertical lines mark the duration of the plenary session. Again, the points in the figure are an average of 20 seconds. The packet loss rate is computed as the ratio of the number of retransmitted frames to the total number of data frames logged by the sniffer. Retransmitted frames are data frames with the *retry* bit set. We observe that the loss rate increases to over 20% as the plenary session begins, and continues to be high throughout the session, with a maximum loss of about 35%. Clearly, the network suffered high loss rates throughout the session.

The graphs indicate that, during the plenary session, the network was heavily utilized and had a dense deployment of APs. At the same time, the network suffered from a high loss rate. In comparison with previous studies of large scale wireless networks, our traces are extensive in that they cover the entire set of APs throughout the session. The high utilization and loss rates motivate us to evaluate the wireless protocol behavior to investigate the cause for these high loss rates.

5 Traffic Analysis

The 802.11 DCF protocol uses Carrier Sense Medium Access with Collision Avoidance (CSMA/CA) to manage and reduce

2. http://www1.cs.columbia.edu/ãndreaf/new/ietf.html

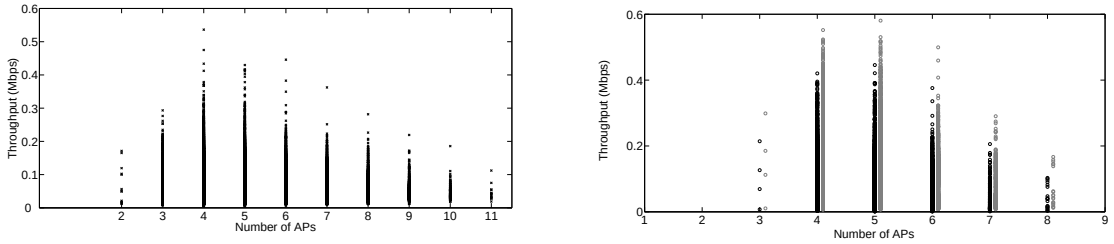


(a) Number of APs within range of the sniffers.

(b) Total number of clients on all channels, averaged over 20 second intervals.

(c) Packet loss rate.

Fig. 3. Usage statistics of the IETF wireless network during Plenary II.



(a) Per-client throughput.

(b) Aggregate data and control throughput.

Fig. 4. Per-client and aggregate throughput.

contention. According to the algorithm, a node that wants to transmit a frame is required to perform carrier sensing to check whether the medium is busy. If the medium is not busy, the node transmits the packet. If the channel is busy, then the node backs off for a specific interval known as the *backoff interval*. For every slot time that the channel is not busy, the BO is decremented. The node transmits the packet when the backoff timer reaches zero. If this transmission results in a collision, maximum length of backoff interval doubles.

This algorithm requires that a node must contend for every packet that it needs to transmit. In a network where there are a large number of nodes, several nodes will be within each other's carrier sense range. In a highly utilized network, these nodes will have a large number of frames that they need to send. When nodes within carrier sense range repeatedly contend for the medium, the nodes spend a significant amount of time in backoff, instead of packet transmission. Consequently, the medium is not utilized efficiently even though the contention is high.

In a large network such as the IETF, we expect the aforementioned problem to be prevalent. To avoid unnecessary backoff and utilize the medium efficiently, it becomes critical that we avoid transmitting any unwanted frames on the wireless medium. Our goal is to analyze the traffic to identify *unwanted traffic* on the wireless side, recognize the protocol components that generate this traffic and suggest ways to mitigate the unwanted traffic. In doing this, we reduce the overhead of unnecessary medium contentions and backoff. The transmission opportunities available to nodes to transmit useful data frames is thus significantly increased. In the context of our work,

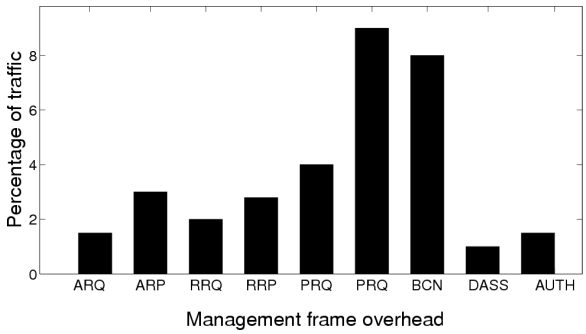
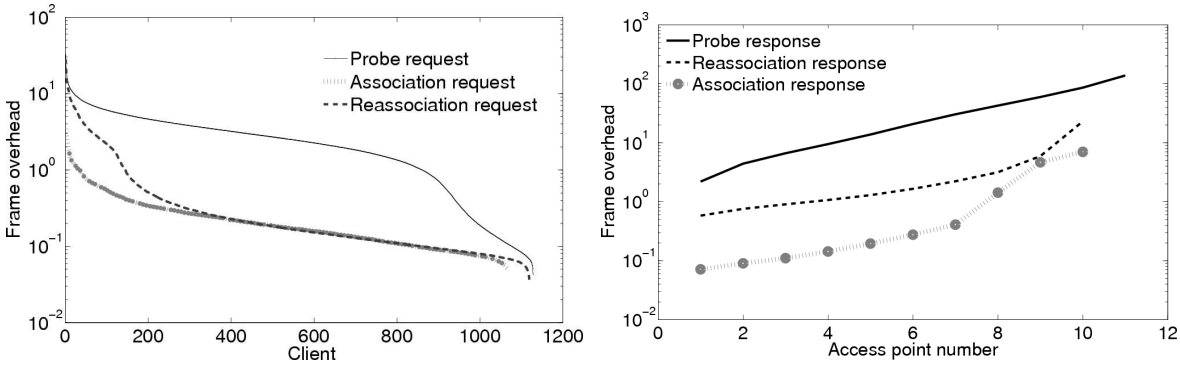


Fig. 5. Breakdown of management traffic as a percentage of total traffic.

unwanted traffic is defined as traffic that is unnecessarily sent on the medium, due to a deficiency in the protocol or its implementation. In the remainder of this section, we analyze the amount of overhead and identify protocol components that generate unnecessary traffic.

We begin with an analysis of user and network throughput. Per-user throughput and aggregate network throughput are computed based on the instantaneous number of users recorded in our data sets. To compute these metrics for a particular one-second interval, we consider all users who contributed at least one data frame during that interval. Figure 4(a) shows the per-user throughput versus the number of APs within range during the same one second interval. The initial increase in throughput as the number of APs increased can be understood as the time when clients obtained benefits of multiple APs in the vicinity, in terms of selecting the AP with the best signal strength. As



(a) Frame overhead per client.

(b) Frame overhead per AP.

Fig. 6. Frame overhead.

the number of APs in the vicinity increases beyond four, the throughout begins to deteriorate. This decrease is due to the increased interference that results from dense AP deployment. The throughput interestingly peaks when there are four APs in the vicinity, which can be understood as follows. When there are three APs in range of each other, they will be assigned to orthogonal frequencies. Contention arises as more APs are within range, resulting in more than one AP operating on the same channel.

Figure 4(b) shows the aggregate network control and data throughput computed for every one second interval, plotted against the number of APs within range during the same one second interval. Again, when the number of APs is greater than four, we observe that the aggregate throughput decreased. However, the rate at which the data throughput decreased is much greater than that of the control throughput. Here, the term control throughput is used to refer to throughput of all non-data frames, i.e., control frames and management frames. The decrease in aggregate throughput is the consequence of increased interference and contention. The higher percentage of control traffic is a result of an increase in the number of overhead frames caused by the presence of multiple APs in the vicinity.

To understand the cause of this overhead, we first need to categorize the traffic based on the different frame types and subtypes. An overview of the management frame types is given in Section 3. In this paper, we focus on the unwanted management and data traffic subtypes, and not the control traffic. The control frame subtypes found in the traces are Request-To-Send (RTS), Clear-To-Send (CTS), and Acknowledgment (ACK). ACKs are necessary to acknowledge the successful reception of data packets. Only about 15% of the users used the RTS-CTS mechanism, and the percentage contribution of these frames to the overhead was not significant. Hence, we focus our analysis on the management and data frames.

A high percentage of the total frames, nearly 40%, were management frames. This high percentage of management traffic has also been reported in previous studies [4]. Figure 5 shows the percentages of each management frame subtype as recorded by the sniffers, averaged over all three channels. The x-axis in the graph represents each of the management frame subtypes explained in Table 1 and the y-axis shows the percentage of frames of each subtype.

To understand the effect of this overhead on the clients and APs, we calculate a metric called *frame overhead*. Frame overhead is computed as the ratio of number of management frames to the number of data frames transmitted in every one second interval. For a client, the overhead consists of probe, association and reassociation requests. For an AP, the overhead frames are the corresponding response frames. This metric is useful as it gives a sense of how many overhead frames a station transmits before obtaining the opportunity to transmit a data frame. Each time a node transmits an overhead frame, it implies a missed transmission opportunity for a data frame.

The frame overhead for each client is shown in Figure 6(a) and for each AP is shown in Figure 6(b). Each value on the x-axis represents a single station (client or AP). The y-axis shows frame overhead for each of the three frame types. The clients and APs are arranged in ascending order of frame overhead for the purpose of clarity. As we can see, the frame overhead for a majority of the clients is greater than one. This implies that a majority of stations must transmit multiple overhead frames before transmitting a single data frame.

In the following sections, we investigate the causes of such high overhead. We show that this overhead is an artifact of a network that is dense and heavily utilized, and much of this overhead is unwanted and degrades the network performance. We then explore ways in which this overhead can be reduced and study the gains of reducing the unwanted traffic. In our traces, we identified two major contributors to unwanted traffic frames. First is a data frame subtype, the null data frame which we discuss in Section 5.1. In Section 5.2, we analyze probes, which are another type of unwanted traffic. We investigate the causes for the high volume of probe traffic and the effects of dense AP deployments on the amount of probe overhead.

5.1 Keepalive Traffic

We analyze the effect of packets transmitted by client cards to maintain connectivity with the access point. We call these packets *keepalive traffic*. In our traces, we observed a large number of null data frames transmitted by the clients to the

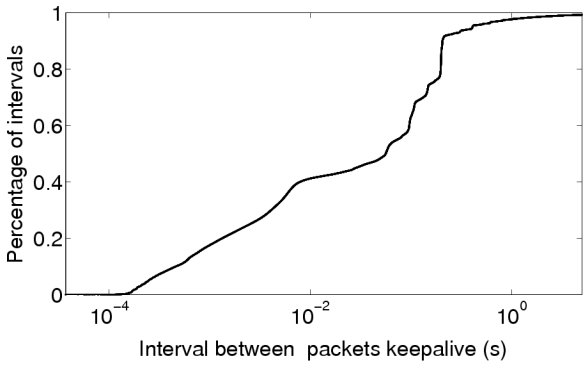


Fig. 7. CDF of interarrival times of the keepalive packets.

Bytes	Airtime	Frames
8%	9%	12%

TABLE 2
Volume of keepalive traffic as a percentage of the entire trace.

access point that were then ACKed by the AP. A null frame is a data frame subtype with zero bytes of data. Further analysis of the traces and open source client implementations (such as Intel) showed that this was part of the AP book-keeping mechanism.

APs maintain an entry for each client in order to store the client's connectivity information. This overhead increases as the number of clients grows. In an effort to minimize this book-keeping overhead, the APs maintain state information only for those clients that are actively sending data packets, and disassociate those clients which have not sent any. The amount of time an AP waits before disassociating an inactive client is implementation dependent. In the absence of data packets, a client transmits null packets, which are essentially keepalive packets, to avoid disconnection by the AP.

Figure 7 is a cumulative distribution of the frequency at which clients transmitted keepalive packets. For each client, the interval between two successive null data packets is calculated and the cumulative distribution of all the inter-packet intervals is plotted. The plot shows the CDF of packet intervals for all the clients for the duration of the plenary. Nearly 50% of the keepalive packets were sent within an interval of 100 ms and 90% of the packets are sent within 1 second. This high rate of transmission results in significant overhead. The impact of this mechanism on the traffic is summarized in Table 2. We examine three different metrics: number of bytes, airtime and number of frames. Each value in the table is expressed as a fraction of the entire trace and averaged over all the channels. As we can see from the table, the keepalive packets pose a considerable overhead, and intelligent techniques that reduce this overhead in a network that is heavily utilized are needed.

5.2 Probe Traffic

A client broadcasts probe requests when it needs to obtain information on which APs are in range. Any AP that receives this request sends a probe response containing information necessary for association, such as capability information and

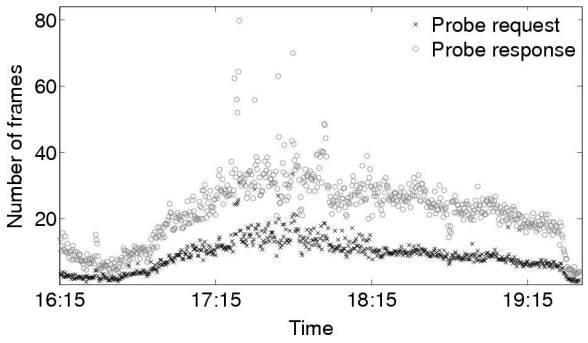


Fig. 8. Comparison of probe traffic with utilization.

supported data rates. Probe requests are sent when a client disconnects or roams from the AP with which it is associated. A client also probes the medium periodically to check which APs are in the vicinity, and whether it is still associated with the AP with the strongest signal.

This aggressive probing is beneficial when clients are mobile. When a client moves and loses connectivity with an AP, the process of scanning and performing a handoff to another AP can take hundreds of milliseconds. This delay is large enough to deteriorate application performance, especially delay sensitive applications such as voice. Instead of being reactive to packet loss, clients are proactive in probing the medium and collecting neighbor information.

While aggressive probing of the medium facilitates faster handoffs for mobile clients, this behavior in a static, congested network imposes unnecessary overhead and leads to inefficient medium utilization. Figure 8 shows the number of probe requests and responses logged by the sniffers per second, averaged over a period of 20 seconds. On average, there were 22 probe requests every second, and at times, as many as 80 probes per second. To understand the high occurrence of probe frames, we look at how frequently the clients probed the medium.

From Figure 8, we also observe that there are a large number of probe responses every second, even more than the number of probe requests. Our reasoning is as follows: since probe requests are broadcast packets, all APs within range of the client hear the request and send unicast responses to the client. In the IETF network, there were multiple APs deployed on each channel, resulting in multiple responses per request. As we saw in Figure 3(a), there were at least four access points detected 85% of the time. This implies that each probe request is likely to elicit at least four responses 85% of the time. This is significant extra overhead in a network that is already highly utilized.

Figure 9 shows the cumulative distribution of the frequency with which clients probed the medium. The x-axis represents the intervals between successive probe requests by any client and the y-axis represents the cumulative percentage of probes at each of the intervals. The graph plots the cumulative distribution of the inter-probe intervals of all the probe requests for each client in the plenary that sent at least one data frame. Nearly 60% of all probe requests occur in intervals smaller than 30 ms, and close to 80% in intervals smaller than 2 seconds. This indicates that a majority of the clients probe the medium frequently, contributing to the overhead. The clients whose probe intervals are very high,

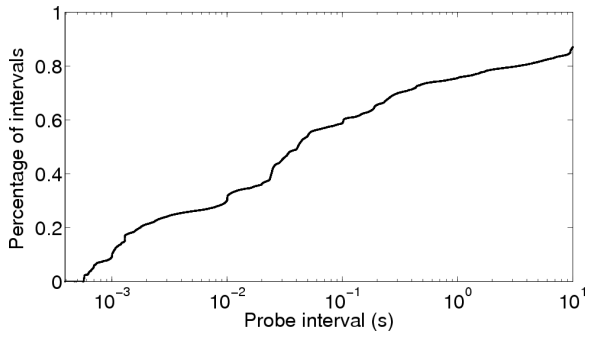


Fig. 9. CDF of the interval between successive probe requests.

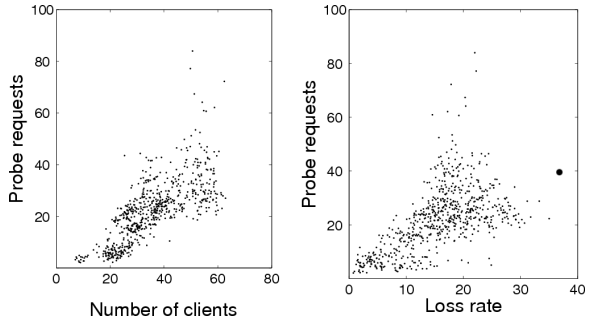


Fig. 10. (a) Scatter plot showing the relationship between the number of clients and probe requests. The correlation coefficient is 0.73. (b) Scatter plot showing the relationship between loss rate and probe requests. The correlation coefficient is 0.65.

on the order of 100 seconds, are the clients for whom we did not observe active sessions in the traces. This may indicate the radios were in sleep mode and he laptop was not in use.

To understand the factors that affect probe traffic, we study the correlation between the number of probe requests and the number of users and loss rate. As the number of clients increases, we expect a proportional increase in the number of probe requests. Also, when the congestion in the medium increases, the number of retransmissions increases. We expect this increase to result in the loss of some of the probe request and response frames. When response frames are lost due to collision, the client will retransmit a probe request, thereby aggravating congestion. Hence we expect a correlation between the loss rate and the number of probe requests. To verify these claims, we plot the number of probe requests per second against the number of clients and loss rate in F igures 10(a) and 10(b), respectively. Figure 10(a) indicates that, as the number of users increases, the number of probe requests increases. In Figure 10(b), the number of probe requests generally increases with an increase in loss rate.

5.3 Discussion

Management frames, together with keepalive traffic, comprise nearly half the total frames transmitted during our collection period. This extreme overhead is detrimental to network performance. With nodes frequently contending for the medium and then backing off, the medium is utilized inefficiently. With a large number of transmitted packets, the probability of packet collision and retransmission is significant. The high frequency of medium probing by the nodes is wasteful. We show in Section 6 that users were predominantly static in the plenary session and did not need to aggressively search for new APs. The same argument holds for keepalive messages. While this mechanism reduces storage overhead on the AP by aggressively removing the disconnected clients, it results in high traffic overhead. This behavior is particularly undesirable in static networks where there is a high probability that a user will reconnect to the same AP. In such a case, having the AP keep a client record active may be more beneficial than aggressively removing it.

6 Handoff Analysis

In Section 5, we studied the breakdown of network traffic and quantitatively analyzed the amount of management and keepalive traffic. Both these frame types are necessary to maintain the client-AP associations. Even when clients are not moving, neighbor discovery is performed frequently to check whether an AP with a higher signal strength is available, thus attempting to improve performance. When a client wishes to associate with a different AP, a handoff process is initiated. An overview of the handoff procedure was provided in Section 3. Handoff trigger is the first stage of handoff wherein a client identifies the need to look for another AP. The implementation of this mechanism is left to vendors, however, it is usually a reaction to one or more of the following: 1) consecutive missed beacons[3]; 2) unacknowledged packets [17]; and 3) beacon frame loss or quality degradation [20]. As a result of frequent probing and implementations that use packet loss information to trigger handoffs, we expect a high rate of handoffs in a congested network. In this section, we analyze the duration and frequency of these associations and the handoff behavior of clients.

Channel 1	Channel 6	Channel 11
614	586	627

TABLE 3
Number of handoffs during the plenary session.

6.1 Trace Analysis

To explore the handoff behavior observed in our traces, we investigate the number and frequency of handoffs and the nature of handoffs between different channels. Most importantly, we investigate whether the handoff resulted in a performance improvement for clients.

The number of handoffs on each channel observed during the plenary is summarized in Table 3. We observe a total of nearly 1800 handoffs during the three hours of the plenary, which is unexpected since we visually observed client mobility to be minimal during the session. To better understand the client handoff behavior and validate our anecdotal observation of low client mobility, we compute the length and frequency of

3. http://ipw2200.sourceforge.net

client-AP associations. We define two metrics for this computation: *prevalence* and *persistence*. Prevalence and persistence of Internet routes were previously studied by Paxson [24]. We define these terms in the context of client-AP associations, and compute these metric values for the IETF traces.

6.2 Prevalence

Adapting the notion of prevalence as defined by Paxson [24], we define prevalence of clients as follows: Given that we observed a client c associated with an AP A, what is the probability of observing c associated with A in the future? Prevalence has specific implications on client mobility. If a client is predominantly static, the prevalence of a client-AP association pair is high, we call this AP the *dominant* AP. On the other hand, evenly distributed prevalence values indicate that there was no single dominant AP, and that the client was mobile. In a well functioning network characterized by clients with low mobility, we expect the majority of the client-AP associations to have high prevalence values indicating that clients did not bounce back and forth between APs.

We compute prevalence values at a high granularity of one second and a coarse granularity of one minute. We divide the trace into n intervals. Let n_s be the total number of one second intervals in the trace. At each one second interval, we check whether a client has sent at least one data packet to the AP. If it has, then it is still connected to the AP, else it has either roamed or become inactive. We consider the client to have reconnected to the AP when we see a data packet from that client again. Let k_s be the total number of one second intervals in which the client was active. The prevalence of the client on the AP is given by

$$\pi_s = k_s/n_s \tag{2}$$

The prevalence values at one second granularity are shown in Figure 11. The prevalence values at this granularity are evenly distributed, which indicates that at a high granularity, not all clients were highly prevalent on the dominant AP. About 40% of the clients had only a 50% chance of being associated with its dominant AP.

Prevalence at granularity of one minute is calculated similarly. If n_m is the total number of one minute intervals in the traces, and k_m is the number of intervals in which a client was active, the prevalence is given by

$$\pi_m = k_m/n_m \tag{3}$$

From Figure 11, we see that the majority of clients are more prevalent on the dominant AP on a one minute granularity. Only about 30% of clients had a prevalence of 80% or less on the dominant AP. The remaining 70% of the clients were prevalent on the dominant AP over 80% of the time. These results indicate that clients frequently associated with the same AP, implying that mobility in the network was low. Even though multiple APs on the same channel were within the range of a client, we observe that a client tends to be prevalent on one AP, the dominant AP. As described in Section 3, most clients use signal strength to select an AP for association. Consequently, the dominant AP is most likely the AP closest to the client. The

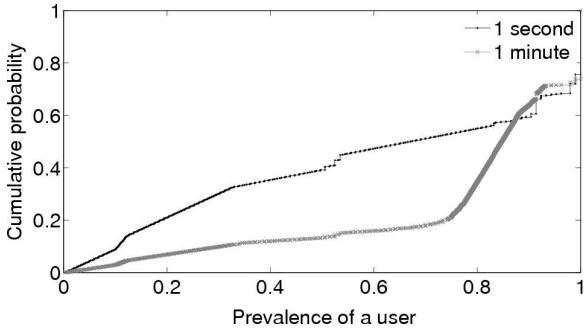


Fig. 11. Client prevalence on an AP, given as the cumulative distribution of the probability of a client being associated with an AP.

lower prevalence at a higher granularity of time implies one of the two things: i) clients were sending data frames infrequently; or ii) clients were bouncing back and forth between APs within short intervals. Given the rate at which the keepalive packets were transmitted, as shown in Figure 7, and the per second client throughput, as shown in Figure 4(a), we believe that frequent switching of clients between APs contributed significantly to the lower prevalence rates over one second intervals.

6.3 Persistence

We define the persistence of a client as follows: Given that a client is associated with a particular AP, how long before the client is likely to have changed its association to another AP? Thus, persistence is the length of time a client remains associated with an AP. A low persistence value indicates that the clients did not remain connected to an AP for a long time. In a well-functioning network characterized by clients with low mobility, we expect clients to have high persistence values. That is, clients stay connected to an AP for long periods while they are static, and only infrequently change APs during movement.

We calculate the persistence of clients connected to their dominant AP. The dominant AP for a client is the AP on which the client has high prevalence. An association length is calculated as the time elapsed between the first and last data frame observed from the client, including null data frames. The persistence is computed for a one second time interval; if no data frame has been observed for up to one second, we assume the session has ended. The one second interval for this computation is based on the observed rate at which null packets are transmitted by the clients to keep their sessions alive. From figure 7, we learned that about 98% of the time, a null packet is sent within one second. Furthermore, if we observe a data frame from a client at second s_1 and do not observe a frame in the subsequent second s_2, we make a "best guess" that the disassociation occurred halfway between these two time intervals.

Figure 12 shows the cumulative distribution of persistence values for the users present during the plenary session. The figure captures values for all client-AP pairs observed in the traces. The x-axis represents the length of associations in minutes and the y-axis represents the cumulative percentage of associations. About 40% of the associations were under two

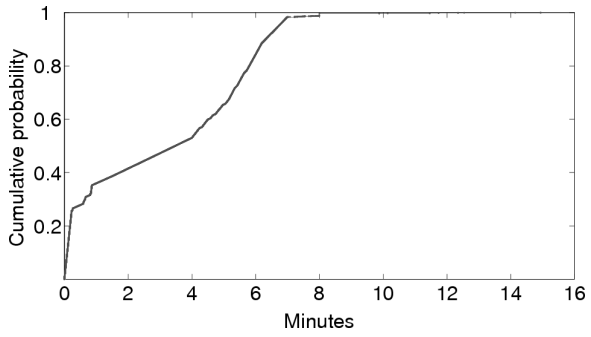


Fig. 12. Client persistence on an AP, given as the cumulative distribution of client-AP association duration.

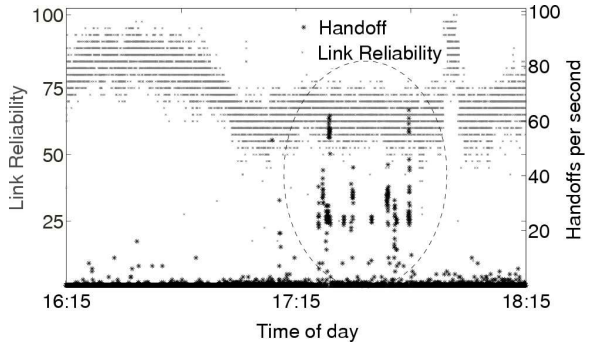


Fig. 13. Comparison of utilization and number of handoffs across all channels.

minutes and 90% of associations were under seven minutes. This indicates that clients remained connected to APs for fairly short periods of time.

In a network with dense AP deployment and a large number of users connected to the network simultaneously, the number of handoffs is high in spite of low mobility. The reason for this behavior lies in the handoff mechanisms. Handoff triggering mechanisms rely on packet loss information to detect when a client has moved away from its AP. This loss can consist of either consecutive beacon framesi losses or unacknowledged data packets. In our traces, we found that the number of beacons received by a client, called *link reliability*, influences the number of handoffs, as shown in Figure 13. Link reliability is computed as the average percentage of beacons received by the sniffer from each AP within range. Sniffers are physically close to the APs and have a higher probability of beacon reception than the clients. Hence, this graph provides an upper bound on the number of beacons that a client is likely to have received. The graph is a time series plot of the percentage of beacons the sniffer received from all APs in one second, and the corresponding number of handoffs that occurred. The beacons were sent at 100ms intervals, implying that the sniffer should receive 10 such beacons per second from each AP in its range. The graph shows a sharp increase in the number of handoffs when the beacon reception rate decreased.

Using link reliability as a handoff trigger is incorrect and problematic in a congested environment. At high utilization levels, the beacon reception rate decreases for two reasons.

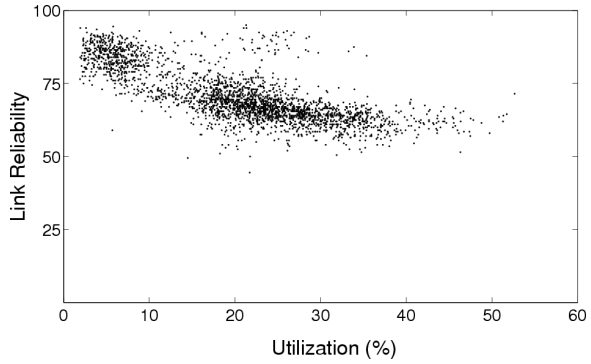


Fig. 14. Scatter plot of beacon reception rate vs. utilization. The correlation coefficient is -0.65.

First, the packet loss rate increases, as illustrated in Figure 3(c), resulting in missed beacon packets. Second, certain AP implementations are known to not queue beacon packets, and will broadcast beacons at the specified beacon interval only if the send queue is empty[4]. Figure 14 illustrates this effect. When the medium is utilized over 50%, the sniffer received beacons only slightly more than 50% of the time.

	Channel 1	Channel 6	Channel 11
Channel 1	33%	7%	2%
Channel 6	2%	24%	6%
Channel 11	4%	3 %	19%

TABLE 4
Percentage of handoffs between different channels for each channel pair. The row value indicates the channel before handoff. The column value indicates the channel after handoff.

The use of packet loss information as a handoff trigger has adverse effects in a congested network. Missed beacons initiate a client to commence roaming, wherein a client actively probes the medium and waits for responses from APs. This behavior not only results in unwanted probe traffic in the wireless medium, but also results in unwanted handoffs. We analyzed the nature of handoffs between channels and the results are summarized in Table 4.

As indicated by table 4, 76% of the handoffs occur between APs on the same channel (found by summing along the diagonal). About 58% of the total handoffs were to the same AP from which the client disconnected. This behavior can be explained as follows: a handoff is triggered due to packet loss, as we have seen earlier. On a trigger, the client scans the medium and obtains information on all the available APs. Currently implemented AP selection mechanisms typically select the AP from which the client receives the strongest signal, without any knowledge of the load on the AP or on the channel. For clients that are predominantly stationary, the AP with the strongest signal strength will be, with a very high probability, the AP from which the client disconnected.

Reassociation with the same AP is wasteful; not only does

4. http://hostap.epitest.fi

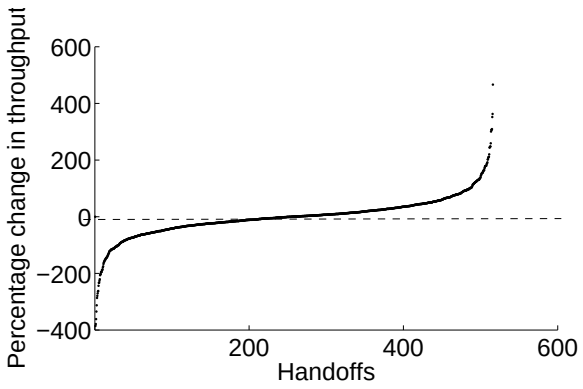


Fig. 15. Percentage change in throughout after handoff over a period of 30s. The x-axis represents each handoff event ordered by throughput improvement.

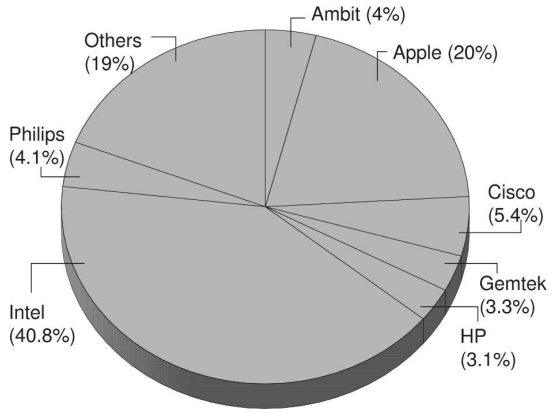


Fig. 16. Distribution of cards per vendor.

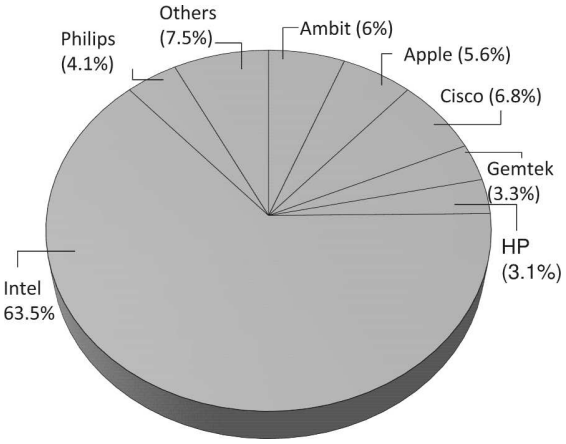


Fig. 17. Percentage of handoffs per vendor.

it result in MAC overhead, but it also causes application performance deterioration. Handoffs to APs on the same channel can be beneficial only if the new AP is less loaded than the AP to which the client was previously connected. However, connecting to APs with lower signal strength is likely to result in lowered data rates. Further, if the network around the client is congested, switching to a different AP on the same channel is not beneficial since the client continues to see a similar level of congestion.

Switching to an AP on a different channel can be beneficial if the new channel is less congested and can offer better throughput to the clients. Since the three channels were utilized uniformly during the plenary and loss rates were comparable, as shown in Figure 3, we do not expect users to have obtained significant gains from handoffs.

To determine whether the handoffs were beneficial, we compute the percentage change in throughput immediately before and after a handoff for each handoff between two different APs. To calculate the percentage throughput improvement of the client, we consider the throughput obtained by the client 30 seconds before and after the handoff and plot the difference. These values are plotted in Figure 15, where the handoffs events are ordered in the ascending order of the throughput improvement. The x-axis represents individual handoff events and the y-axis represents the percentage improvement in throughput as a result of the handoff. The graph indicates that about 50% of the handoffs had a negative impact on the throughput. While 50% handoffs resulted in an increase in throughput, 20% of these handoffs resulted in less than a 10% increase in throughput. These results indicate that a significant portion of the handoffs were not beneficial, and may even have been detrimental. A reduction in unbeneficial handoffs will reduce the amount of management traffic, leading to greater transmission opportunities for nodes with data packets and an increase in efficient medium utilization.

6.4 Vendor Handoff Analysis

Much of the handoff behavior discussed in the earlier sections depends on the way the handoff mechanism is implemented by the wireless card vendor. The IEEE 802.11 specification does not specify the exact implementation of handoff mechanisms, leaving it to the vendor to implement efficient algorithms for handoff triggers and AP selection. In this section, we investigate the behavior of different cards to analyze whether the handoff behavior observed is common among the different vendors or simply a manifestation of bugs in a single vendor's implementation.

Over 600 unique cards were present in the plenary session. The breakdown of the cards based on the vendors is shown in Figure 16. Figure 17 shows a breakdown of the percentage of handoffs per vendor. Cards from different vendors exhibited similar handoff behavior, with the exception of Apple cards. Apple cards experienced a low percentage of handoffs during the entire plenary. Figure 18 shows the prevalence of clients on the dominant AP, grouped by vendor. The figure shows that different cards are relatively consistent in reassociation with the same AP regardless of vendor; nearly 40% of the cards reconnect to the same AP within five minutes.

Figure 19 shows the client persistence on the dominant AP. Consistent with earlier results, we see that up to 25% of user sessions were under a second and nearly half of these were under one minute. This behavior is consistent across the different card vendors, with the exception of Apple. This result shows that, across vendors, there is a need for better handoff triggering and AP selection mechanisms. Since the Apple drivers are not open

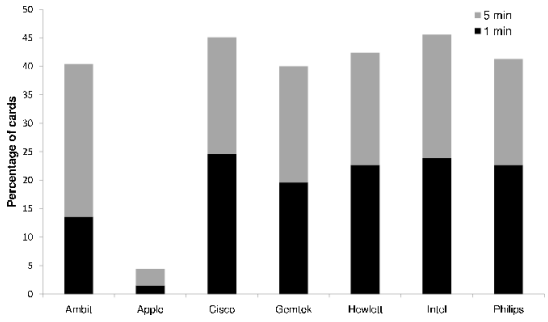


Fig. 18. Client prevalence. Black indicates the percentage of clients that reconnect to the same AP within one minute. Grey indicates percentage of clients that reconnect within five minutes.

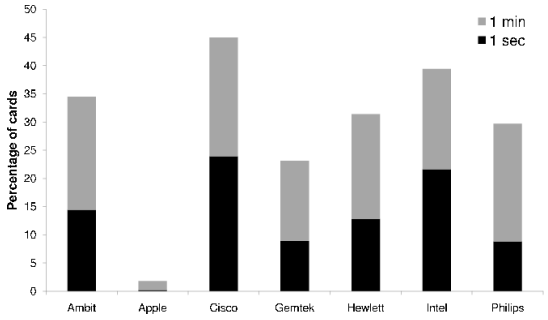


Fig. 19. Session length. Black indicates percentage of clients whose session lengths were under 1s and grey indicates the percentage of clients whose session lengths were under 1m.

source, we are unable to investigate why Apple cards perform better than the other vendors.

7 Conclusion

The ease of deployment and low cost of infrastructure have led to the rapid deployment of WLANs to provide network access to users. Analysis of real world deployments are critical to identify deficiencies in the 802.11 protocol and its implementations. To this effect, we collected data from the 67^{th} Internet Engineering Task Force (IETF) meeting held in November 2006 in San Diego CA. Through the analysis of this data, we have identified the causes for high overhead in the transmission of data frames. In particular, we have analyzed the unwanted link layer traffic that stems from mechanisms that initiate, maintain, and change client-to-AP associations. We further show that clients have short association times with the APs. This result is a consequence of the current mechanisms that trigger a handoff under conditions of high medium utilization and packet loss rate, even in the absence of client mobility. We analyze the traffic to understand when handoffs occur and whether the handoffs were beneficial or should have been avoided.

Observations made in this paper suggest that there is a need to design algorithms that are adaptive to network conditions and usage. In particular, the frequency of keepalive messages should be lowered when the clients are stationary and when the network congestion levels are high. Similarly, the periodic probe requests must be reduced in stationary, congested networks. In a network that is used heavily, a client may be able to leverage probe responses transmitted to its neighbors. Thus, a mechanism in which a node passively monitors the probe responses in its vicinity, and sends a request only when it does not detect any neighbor responses, will help in reducing the probe traffic.

Finally, handoff mechanisms should be adaptive to congestion losses. Use of packet loss information to trigger handoffs creates in a high rate of handoffs, even in the absence of mobility. In the IETF network, a significant fraction of these handoffs were to the same AP, and thus unnecessary. Further, many of the handoffs that occurred to other APs impacted the clients negatively. Schemes that use signal strength trends to detect disconnection, and schemes that incorporate network information such as load or loss rates, are needed.

References

[1] A. P. Jardosh, K. Mittal, K. N. Ramachandran, E. M. Belding, and K. C. Almeroth, "IQU: practical queue-based user association management for wlans," in *Proceedings of MobiCom*, Sept. 2006, pp. 158–169.

[2] F. Ricciato, P. Svoboda, E. Hasenleithner, and W. Fleischer, "Unwanted traffic in 3G networks," *SIGCOMM Computer Communication Review*, vol. 36, no. 2, 2006.

[3] A. P. Jardosh, K. N. Ramachandran, K. C. Almeroth, and E. M. Belding-Royer, "Understanding Link-Layer Behavior in Highly Congested IEEE 802.11b Wireless Networks," in *Proceedings of EWIND*, Philadelphia, PA, Aug. 2005, pp. 11–16.

[4] M. Rodrig, C. Reis, R. Mahajan, D. Wetherall, and J. Zahorjan, "Measurement-based characterization of 802.11 in a hotspot setting," in *Proceedings of EWIND*, Aug. 2005, pp. 5–10.

[5] A. P. Jardosh, K. N. Ramachandran, K. C. Almeroth, and E. M. Belding-Royer, "Understanding congestion in ieee 802.11b wireless networks," in *Proceedings of IMC*, Berkeley, CA, Oct. 2005.

[6] R. Raghavendra, E. M. Belding, K. Papagiannaki, and K. C. Almeroth, "Understanding Handoffs in Large IEEE 802.11 Wireless Networks," in *Proceedings of IMC*, San Diego, CA, Oct. 2007.

[7] D. Tang and M. Baker, "Analysis of a local-area wireless network," in *Proceedings of the ACM MobiCom*, Boston, MA, Aug. 2000, pp. 1–10.

[8] D. Kotz and K. Essien, "Analysis of a campus-wide wireless network," in *Proceedings of the ACM MobiCom*, Atlanta, GA, Sept. 2002, pp. 107–118.

[9] D. Tang and M. Baker, "Analysis of a metropolitan-area wireless network," *Wireless Networking*, vol. 8, no. 2/3, pp. 107–120, 2002.

[10] M. Balazinska and P. Castro, "Characterizing Mobility and Network Usage in a Corporate Wireless Local-Area Network," in *Proceedings of MobiSys*, San Francisco, CA, May 2003, pp. 303–316.

[11] A. Balachandran, G. M. Voelker, P. Bahl, and P. V. Rangan, "Characterizing user behavior and network performance in a public wireless LAN," in *Proceedings of the ACM SIGMETRICS Conference*, Marina Del Rey, CA, June 2002, pp. 195–205.

[12] T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campus-wide wireless network," in *Proceedings of the ACM MobiCom*, Philadelphia, PA, Sept. 2004, pp. 187–201.

[13] J. Yeo, M. Youssef, and A. Agrawala, "A framework for wireless LAN monitoring and its applications," in *Proceedings of WiSe*, Philadelphia, PA, Oct. 2004, pp. 70–79.

[14] P. A. K. Acharya, A. Sharma, E. M. Belding, K. C. Almeroth, and K. Papagiannaki, "Congestion-Aware Rate Adaptation in Wireless Networks: A Measurement-Driven Approach," in *Proceedings of SECON*, San Fransisco, CA, Oct. 2008.

[15] R. Beyah, S. Kangude, G. Yu, B. Strickland, and J. Copeland, "Rogue access point detection using temporal traffic characteristics," in *Proceedings of IEEE Globecom*, Nov. 2004.

[16] A. Mishra, M. Shin, and W. A. Arbaugh, "An empirical analysis of the IEEE 802.11 MAC layer handoff process," *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 2, pp. 93–102, Apr. 2003.

[17] H. Velayos and G. Karlsson, "Techniques to reduce IEEE 802.11b handoff time," in *Proceedings of ICC*, Paris, France, June 2004.

[18] H.-S. Kim, S.-H. Park, C.-S. Park, J.-W. Kim, and S.-J. Ko, "Selective Channel Scanning for Fast Handoff in Wireless LAN using Neighbor Graph," in *Proceedings of ITC-CSCC*, Sendai/Matsusima, July 2004, pp. 303–316.

[19] I. Ramani and S. Savage, "SyncScan: Practical Fast Handoff for 802.11 Infrastructure Network," in *Proceedings of IEEE Infocom*, Miami, FL, Mar. 2005.

[20] V. Mhatre and K. Papagiannaki, "Using smart triggers for improved user performance in 802.11 wireless networks," in *Proceedings of MobiSys*, June 2006, pp. 246–259.

[21] S. Vasudevan, K. Papagiannaki, C. Diot, J. Kurose, and D. Towsley, "Improved access point selection," in *Proceedings of IMC*, Oct. 2005.

[22] A. J. Nicholson, Y. Chawathe, M. Y. Chen, B. D. Noble, and D. Wetherall, "Improved access point selection," in *Proceedings of MobiSys*, June 2006, pp. 233–245.

[23] R. Mahajan, M. Rodrig, D. Wetherall, and J. Zahorjan, "Analyzing the mac-level behavior of wireless networks in the wild," in *Proceedings of SIGCOMM*, Sept. 2006, pp. 75–86.

[24] V. Paxson, "End-to-end routing behavior in the internet," in *Proceedings of SIGCOMM*, 1996, pp. 25–38.