Wireless Networks 11, 115–133, 2005
© 2005 Springer Science + Business Media, Inc. Manufactured in The Netherlands.

Analysis of a Campus-Wide Wireless Network *

DAVID KOTZ **
Department of Computer Science, Dartmouth College, Hanover, NH 03755, USA

KOBBY ESSIEN
Department of Bioengineering, University of Pennsylvania, 120 Hayden Hall, 3320 Smith Walk, Philadelphia, PA 19104, USA

Abstract. Understanding usage patterns in wireless local-area networks (WLANs) is critical for those who develop, deploy, and manage WLAN technology, as well as those who develop systems and application software for wireless networks. This paper presents results from the largest and most comprehensive trace of network activity in a large, production wireless LAN. For eleven weeks we traced the activity of nearly two thousand users drawn from a general campus population, using a campus-wide network of 476 access points spread over 161 buildings at Dartmouth College. Our study expands on those done by Tang and Baker, with a significantly larger and broader population.

We found that residential traffic dominated all other traffic, particularly in residences populated by newer students; students are increasingly choosing a wireless laptop as their primary computer. Although web protocols were the single largest component of traffic volume, network backup and file sharing contributed an unexpectedly large amount to the traffic. Although there was some roaming within a network session, we were surprised by the number of situations in which cards roamed excessively, unable to settle on one access point. Cross-subnet roams were an especial problem, because they broke IP connections, indicating the need for solutions that avoid or accommodate such roams.

Keywords: wireless network, Wi-Fi, 802.11, WLAN, workload characterization

1. Introduction

Wireless local-area networks (WLANs) are increasingly common, particularly on university and corporate campuses. For example, a contemporary survey of 392 academic institutions [5] found that nearly all plan to install a wireless network, about half already have a limited deployment, and a few (7%) have a "comprehensive" deployment. Although technology such as IEEE 802.11b is broadly deployed and usage is increasing dramatically, little is known about how these networks are used. A clear understanding of usage patterns in real WLANs is critical information for those who develop, deploy, and manage WLAN technology, and those who develop systems and application software for wireless networks.

This paper presents results from a large and comprehensive trace of network activity in a large, production wireless LAN. Dartmouth College has 11 Mbps 802.11b coverage for nearly every building on campus, including all administrative, academic, and residential buildings, and most athletic facilities. We collected extensive trace information from the entire network throughout the Fall term of 2001. (As of this writing in April 2003, the network includes over 550 access points.)

Our work significantly expands upon the WaveLAN study by Tang and Baker [17], which traced 74 computer-science users in one building for 12 weeks. Our study traces nearly two thousand users drawn from a general campus population, across 161 buildings for one academic term (11 weeks). It also expands upon the Metricom study by Tang and Baker [16,18], which traced a metropolitan-area network for seven weeks. Although that trace covers a wide geographical area and almost 25,000 users, our trace includes detailed information about the amount and nature of the network traffic. Others have used SNMP to trace networks of four [2], 109 [9], and 117 [3] access points, but never on the scale of our study, or with the detailed movement data provided by our syslogs.

The size, population diversity, and detail of our data collection offers extensive insight into wireless network usage. Although every environment is different, our study has characteristics common to both residential and enterprise deployments.

We next describe the environment of our study, the campus of Dartmouth College, and then detail our tracing methodology in section 3. In section 4 we present and discuss the most interesting characteristics of the data. Section 5 compares our results with those of earlier studies, and section 6 draws conclusions.

2. The test environment

The Dartmouth College campus is compact, with over 161 buildings on 200 acres, including administrative, academic, residential, and athletic buildings. Every building is wired to the campus backbone network. Every office, dorm room, and lecture hall, and in some places every seat in a lecture hall, has wired Ethernet. In 2001 Dartmouth installed 476 access points from Cisco Systems, each an Aironet model 350, to provide 11 Mbps coverage to nearly the entire campus. Each access point (AP) has a range of about 130–350 feet indoors, so there are several APs in all but the smallest buildings. Although there was no specific effort to cover outdoor spaces,

* This paper is revised and extended from our Mobicom'02 paper.
** Corresponding author.
E-mail: dfk@cs.dartmouth.edu

the campus is compact and the interior APs tend to cover most outdoor spaces.

All APs share the same network name (SSID), allowing wireless clients to roam seamlessly from one AP to another. On the other hand, a building's APs are connected through a switch or hub to the building's existing subnet. The 161 covered buildings span 81 subnets, so in many cases a wireless client roaming from one building to another will be forced to obtain a new IP address. (Dartmouth chose not to construct a separate campus-wide subnet for the wireless network, unlike the Wireless Andrew project [4].)

Dartmouth College has about 5,500 students and 1,215 full-time professors. During Fall 2001 approximately 3,330 undergraduate students lived on campus. Each is required to own a computer. Each year, approximately 1000 undergraduate students enter Dartmouth College, and most purchase a computer through the campus computer store. Of those purchases, laptops have become increasingly dominant in recent years: 27% in 1999, 45% in 2000, 70% in 2001, and 88% in 2002. Assuming that that students obtaining computers elsewhere choose laptops in the same fraction, and that in 1998 (for which no data is available) about 15% purchased laptops, about 40% of undergraduates owned laptops at the time of our study in Fall 2001. All laptops purchased since summer 2001 had built-in wireless support, and over 1000 802.11b cards were sold over the 2000–2001 year to other users. The Tuck school of business requires all 480 students to own laptops. In addition, most engineering-school graduate students, own laptops. We estimate that about half of those laptops were wireless-enabled in Fall 2001.

3. Trace collection

We began collecting data in April 2001, when the first access points were installed. After preliminary study of the data in May 2001 [15], we began full-scale data collection when students returned to campus in September 2001. In this paper we focus on the data collected during the eleven-week Fall 2001 term, Tuesday September 25 through Monday December 10, inclusive. Although we have data for about a week prior and about a month after, there was significantly less usage during vacation periods and so we limit our analysis to the active period.

At the beginning of the trace period there were 465 access points (APs). Eleven more APs were installed in the first month to bring the total to 476 by October 21. As we discuss below, it appears that some of the "installed" APs were not completely or correctly configured during the tracing period, however, which resulted in fewer APs represented in our data.

We used three techniques to collect data about wireless-network usage: syslog events, SNMP polling, and tcpdump sniffers.

3.1. Syslog

We configured the access points to transmit a syslog message every time a client card authenticated, associated, reassociated, disassociated, or deauthenticated with the access point (see definitions below). The syslog messages arrived via UDP at a server in our lab, which recorded all 3,533,352 of them for later analysis.

Most APs contributed to the syslog trace as soon as they were configured and installed. Of the 476 APs, only 430 were represented in our trace. Although some appear never to have been used, many were misconfigured and did not send syslog messages. Furthermore, we have incomplete data for a few dates when the campus experienced a power failure, or when a central syslog daemon apparently hung up. Finally, since syslog uses UDP it is possible that some messages were lost or misordered. As a result of these spatial and temporal holes in the trace, some of our statistics will undercount actual activity.

Our syslog-recording server added a timestamp to each message as it arrives. Each message contained the AP name, the MAC address of the card, and the type of message (figure 1):

Authenticated. Before a card may use the network, it must authenticate. We ignore this message.

Associated. After authentication, a card chooses one of the in-range access points and associates with that AP; all traffic to and from the card goes through that AP.

Reassociated. The card monitors periodic beacons from the APs and (based on signal strength or other factors) may choose to reassociate with another AP. This feature supports roaming. Unfortunately, some cards apparently never use the Reassociate protocol, and always use Associate [6];[1] see figure 2.

Roamed. When a card reassociates with a new AP, the new AP broadcasts that fact on the Ethernet; upon receipt, the old AP emits a syslog "Roamed" message. We ignore this message; because it depends on an inter-AP protocol below the IP layer, it only occurs when a card roams to another AP within the same subnet.

Disassociated. When the card no longer needs the network, it disassociates with its current AP. We found, however, that the syslog contained almost no such messages.

Deauthenticated. While it is possible for the card to request deauthentication, this almost never happened in our log. Normally, the associated AP deauthenticates the card after 30 minutes of inactivity. In our log it is common to see several deauthentication messages for a widely roaming card, one message from each subnet visited in the session; we ignore all but the message from the most recent AP.

Our network does not use MAC-layer authentication in the APs, or IP-layer authentication in the DHCP server. Any card may associate with any access point, and obtain a dynamic IP address. We thus do not know the identity of users, and the IP address given to a user varies from time to time and building to building. We make the approximating assumption to

[1] Cards from all common vendors exhibit this behavior: notably, 11% of Lucent cards, 19% of Cisco cards, and 21% of Apple cards were never seen to Reassociate.

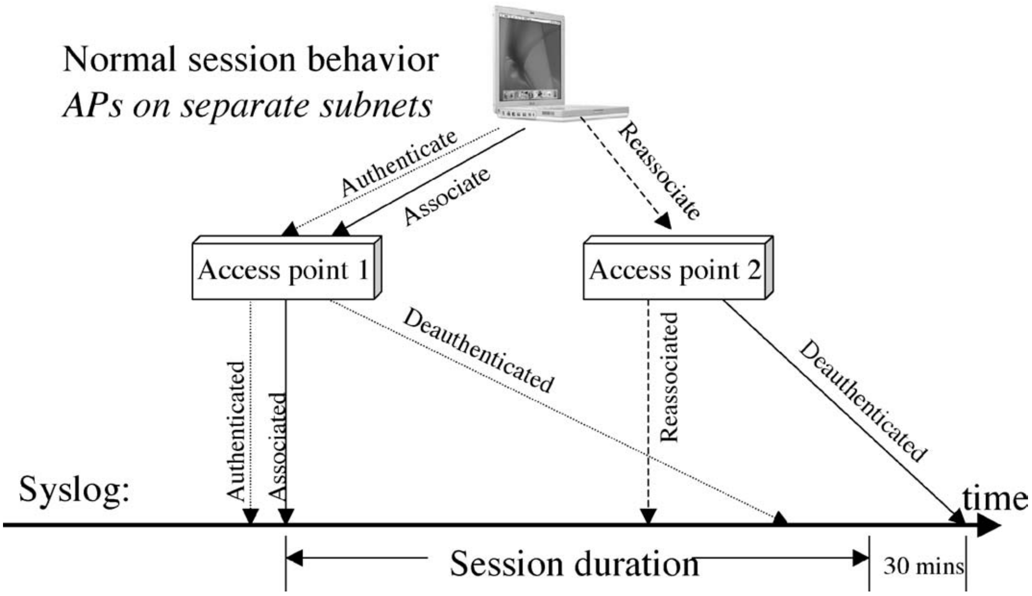

Figure 1. A normal session begins with an Authentication message and an Association message, followed by optional Reassociation messages, and ends with a Deauthentication message.

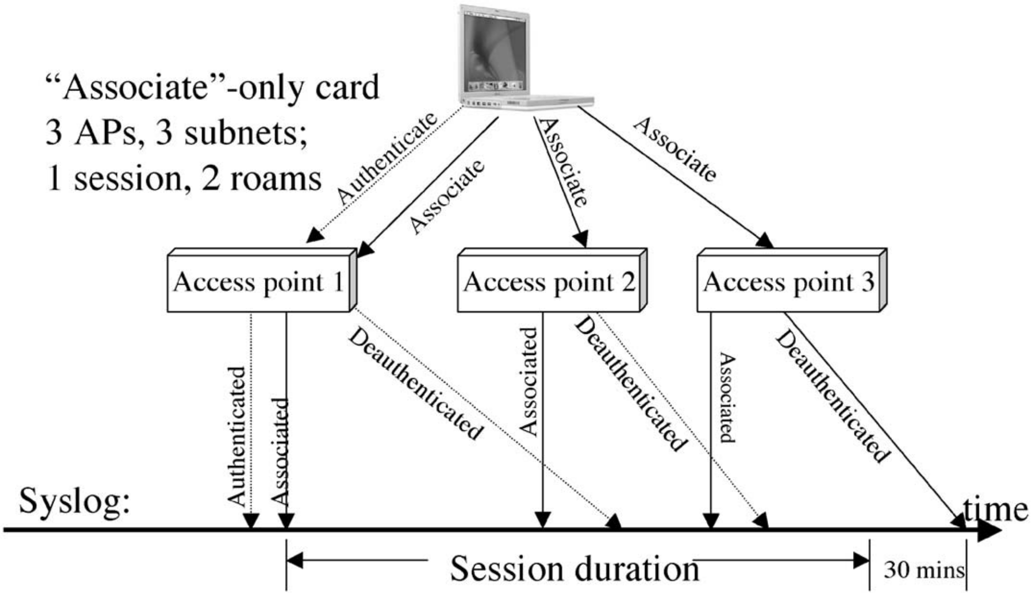

Figure 2. Some cards never use the Reassociate protocol, so most Associate messages are treated as if they were Reassociate.

equate cards with users, although some users may have multiple cards, or some cards may be shared by multiple users. Throughout this paper we use the term "card" for precision, although with the intention that cards approximate users.

3.2. SNMP

We used the Simple Network Management Protocol (SNMP) to periodically poll the APs; 451 of the 476 APs responded to our polls. We chose to poll every 5 minutes to obtain information reasonably frequently, within the limits of the computation and bandwidth available on our two polling workstations. Our trace period includes 193,111,734 of these SNMP records. Unfortunately, we have incomplete data for the following dates: October 7, 9, and 12 (maintenance of our server), November 19 (unknown causes), and December 5 (a campus-wide power failure). We chose to entirely exclude those dates from our analysis, because most of our SNMP-based plots examine traffic per day, a number that would be polluted by "short" days.

Each poll returned the MAC addresses of recently associated client stations, and the current value of two counters, one for inbound bytes and one for outbound bytes. The AP does not reset the counters when polled, so we compute the difference between the values retrieved by one poll and the values retrieved by the next poll. The counters are 32-bit unsigned integers, and our computation properly handles counter roll-over. We ignore the result, however, in two instances: (a) when the time between successful polls is more than 12 minutes (twice the polling interval plus a little slack); (b) when the resulting number of bytes is more than the wireless interface could have sent or received in the time since the last poll. In the former case, the AP was unreachable for more than one poll, and we were unsure how many times the counter may have rolled during those missed polls. In the latter case, the AP (and its counters) were likely reset due to maintenance or a power failure.

Although each SNMP record contains a list of cards associated with the AP, we chose to use the syslog data for tracking cards because the syslog data provides the exact series of events for each card, whereas the SNMP polling data was less precise.

3.3. Sniffers

The syslog and SNMP traces allowed us to compute basic statistics about traffic, users, and mobility. To get a better picture of what the users were doing with the network, we used tcpdump to capture all of the packet headers on a selection of the APs around campus. Because of the volume of data, and privacy concerns, we recorded only packet headers. Because of the number and geographic distribution of APs, the structure of our network (many subnets, and switched Ethernet), and the volume of traffic, it was not possible to capture all of the wireless traffic. In each of four locations we attached a computer and the building's APs to a common hub, and attached the hub's uplink to a switch port on the campus network. With this "sniffer" in promiscuous mode, we used tcpdump to record the header of every packet passing by; in our later analysis, we focus only on the wireless packets.

Our sniffers often recorded packets unrelated to the wireless access points on their hub, so we needed a way to study only the wireless packets. To decide whether a given packet was wireless, we examined the source and destination MAC addresses in its Ethernet frame, and compared them with a list of MAC addresses appearing in our syslog trace. Unfortunately, tcpdump[2] recorded invalid MAC addresses for about 78% of all frames.[3] For frames containing IP packets, we examined the source and destination IP address; if the IP address was associated with a valid, wireless MAC address in a recent IP packet, then we assumed this packet used the same MAC, and treated it as a wireless packet. We fixed about a third of bad MACs this way. For frames not containing an IP packet, we were unable to repair the MAC address and thus assumed the frame was not wireless. As a result, our statistics undercount non-IP wireless frames.

We chose four representative locations:

Sudikoff: the Department of Computer Science (6 APs). There were three holes amounting to 21 hours.

Brown: a dormitory with many first-year students (2 APs). There were 15 holes amounting to 213 hours.

Berry: the main campus library. Due to the size of the building and the switched nature of its network, we were only able to sniff 5 of the 13 APs. There were seven holes amounting to 139 hours.

Collis/Thayer: two buildings, the student center and dining hall, containing five cafes, several lounge areas, several meeting rooms, and some offices (total 9 APs). There were eight holes amounting to 337 hours.

Many of the holes were caused by power outages, in which case the sniffer lost power, but so did the the access point and nearby networking hardware. Thus there was no traffic to sniff during the power failure. Since, after power was restored, the sniffer no doubt took more time to boot than the access point and network hardware, we probably missed a small amount of data. Thus our statistics will slightly undercount the traffic. The Collis sniffer, unfortunately, was more seriously affected by one power failure and required several days to repair.

Since we did not use the sniffer data for any daily analyses, we did not discard any data.

3.4. Definitions

One goal of this study is to understand user behavior. We imagine user "sessions" in which a user (card) joins the network, uses the network, possibly roams to other APs, and leaves the network. We need precise definitions:

Card: a wireless network interface card, identified by MAC address.

Active Card: a card involved in a session (see below), during the hour, during the day, or at the place, in question.

Mobile Card: an active card that visits more than one building during the hour, during the day, or at the place, in question. We aim to understand physical mobility, so we focus on buildings rather than access points.

Roamer Card: an active card that roams (see below) during the hour, during the day, or at the place, in question. We aim to understand network mobility within a session.

Session: A session starts when a card associates with an access point. Exception 1: any Associate messages that arrive less than *SessionThreshold* after the preceding Associate or Reassociate message are treated as if they were a Reassociate message rather than starting a new session (figure 3). Thus they indicate a roam. Exception 2: for any card that never used Reassociate during our trace, we assumed that card is of the variety that uses Associate (within a session) to mean Reassociate, so we counted as roams any Associate arriving within an existing session.

A session ends in one of three ways:

1. If a Deassociate or Deauthenticate message is received from the last access point used by the card (other such messages are ignored), the session is clearly over. If the reason is "Inactivity", and this message arrived more than 30 minutes after the session start time, we compute the session end time to be 30 minutes prior to this

[2] Apparently, due to a bug in tcpdump.

[3] 0:0:0:0:0:0, 0:0:0:0:0:1, 1:0:0:0:0:0, or 1:0:1:0:1:0.

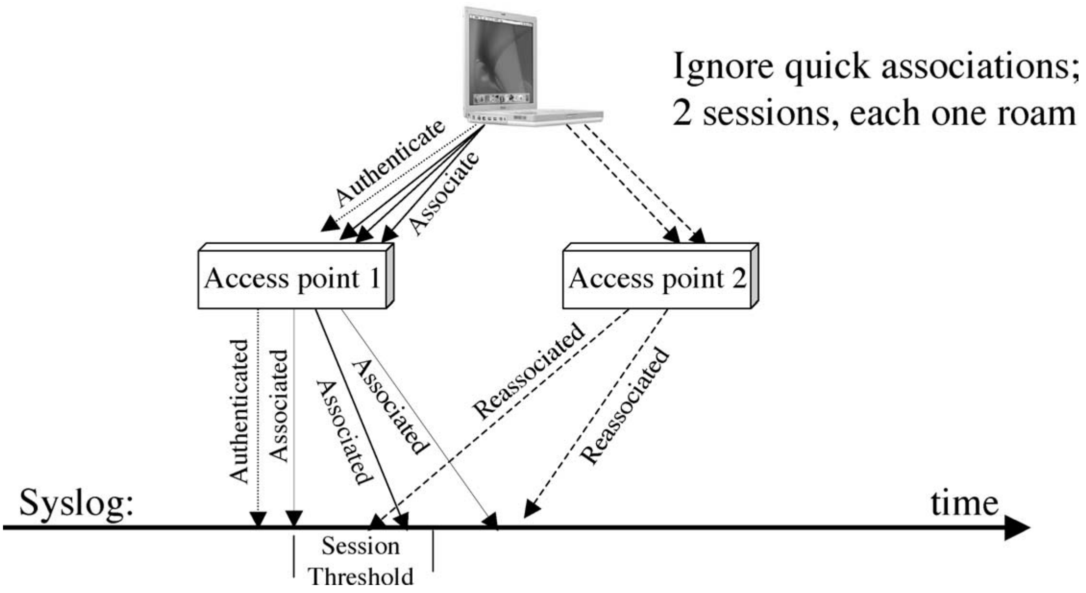

Figure 3. A card that, for some reason, Associates soon after a recent Association or Reassociation. If less than 30 seconds have passed, the Association is treated as a roam rather than a new session.

message's time. Otherwise, the session end time is this message's time.

2. As mentioned above, we treat some Associate messages arriving during an existing session as marking a new session. The time of this Associate message defines the end time of the current session and the start time of the new session. This rule was necessary because it appeared that many sessions did not end with a Disassociate or Deauthenticate message, either because the AP did not send the message or we did not receive it.
3. The end of the trace is reached. When this occurs, all ongoing sessions end at the last AP being used by the card and the session is assumed to end at the time of the last log in the entire trace.

Roam: a card switches access points within a session, identified by a Reassociate message to a new AP, or by an Associate message that is treated as a roam (as described above).

Extra-subnet roam: a roam to an AP in another subnet.

Intra-subnet roam: a roam to an AP in the same subnet.

Stationary session: a session containing no roams.

Roaming session: a session containing roams.

Extra-subnet session: a session containing an extra-subnet roam.

Intra-subnet session: a roaming session containing no extra-subnet roams.

Inbound: traffic sent by the access point to the card.

Outbound: traffic sent by the card to the access point.

These card-centric definitions of "in" and "out" are the reverse of those used in our earliest drafts [10,15], but match those used by Tang and Baker [17].

If a sniffer saw a frame with a wireless source *and* destination (rare), we counted it as "inbound", rather than double-counting it as inbound and outbound. In the SNMP analysis, we believe the AP counted such traffic twice.

A note about the *SessionThreshold* mentioned above. On occasion, a card would Associate rather than Reassociate, apparently because the state machine on the card was out of sync with that on the AP [6]. Figure 3 illustrates the situation. It is difficult to identify precisely which of these Associate messages should define a new "session", and which really represent a roam within the current session. We set *SessionThreshold* to 30 seconds, under the assumption that anything shorter is certainly not a new "session" in the eyes of the user.

A note about the timestamps in the syslog. Although the messages may be delayed or reordered as they pass through the campus network to our server, the delays are small relative to our timestamp granularity (one second) and any reordering that affects causality should be rare. To affect our analysis, the reordering must affect two messages about the same card, the messages must be causally related, and the messages must be significant to our definition of session and roam. There are a few cases of concern. (1) A Reassociate message arrives before the start of its session; our analysis script might be confused about the current AP for the session and possibly miss the session-ending Deauthenticate message. In such a situation, though, it is likely that the card will reassociate several times (in our experience) so the loss of the first in a sequence of quick roams has no effect on our analysis script. (2) A Reassociate message arrives after the end of its session, in which case we will miss counting this as a roam. (3) A Deauthenticate message arrives before its session-starting Associate message; but most Deauthenticate messages are generated by the 30-minute timer, and would thus be sent much later than the Associate message. (4) An Associate message arrives before the Deauthenticate or Disassociate message of the prior session. This situation may arise when an AP Deauthenticate or Disassociates due to an error, and then the card quickly re-Authenticates and re-Associates. If the messages arrive in order we count two sessions. If the messages arrive out of order we count either one session with an extra roam (if the Associate message closely follows an earlier roam) and we miss the rest of the real second session, or we count two sessions as before (if the Associate message does not come too soon after an earlier roam, we will have treated it as the start of a new session anyway).

It is difficult, if not impossible, to quantify the effect of the possible reordering, but we believe that these occurrences are rare in the context of our large trace.

Availability. All of the data used in this study, and more, is available at `http://www.cs.dartmouth.edu/~campus/`

4. Results

We collected an enormous amount of data, and can present only a subset of the interesting characteristics in this paper. First, the basics. In the 77-day trace period we saw 1706 distinct cards.[4] Of the 476 installed access points, we monitored 430 by syslog, 451 by SNMP, and 22 by tcpdump. The access points were distributed among 161 buildings, which we divide into five categories: 82 Residence, 32 Academic, 6 Library, 19 Social, and 22 Administrative. The residential buildings are mostly undergraduate dormitories and fraternities, but also include some Dartmouth-owned housing for faculty and staff, and a residential facility for the business school. All business-school students have laptops and (as the data shows) many are busy wireless users. The social buildings include dining facilities, the arts center, and athletic facilities (including a lodge at the ski area and a boathouse on the river).

In the rest of this section we present a series of questions about the network's usage, and our analysis based on the data. For each figure or table, we identify the data source as [syslog], [SNMP], or [tcpdump].

4.1. Traffic

Perhaps the most fundamental questions about a new network involve how much it is used, and when:

- How much traffic does the network handle?
- How much traffic per card?
- How does traffic vary across hours, days, weekdays?
- How much traffic remains on campus?

Over the course of our study period we measured 3.3 terabytes of total traffic, although more than half that traffic was caused by only 5% of cards.

The daily traffic also varied considerably. Figure 4 is a time series, and figure 5 shows the distribution of traffic across days in our trace. On the busiest day the network moved over 240 GB, whereas the median daily traffic was 53 MB. There is a clear dip around the Thanksgiving holiday. There was always less outbound traffic than inbound traffic, but the proportion of outbound data varied daily between 18 and 89%.

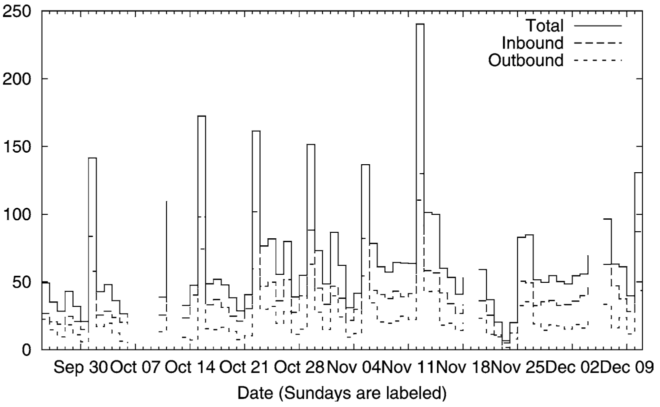

Figure 4. [SNMP] Daily traffic (GB). A date's bar appears to the right of its ticmark. Gaps in the plot represent holes in our data. Note that there is typically more inbound than outbound traffic.

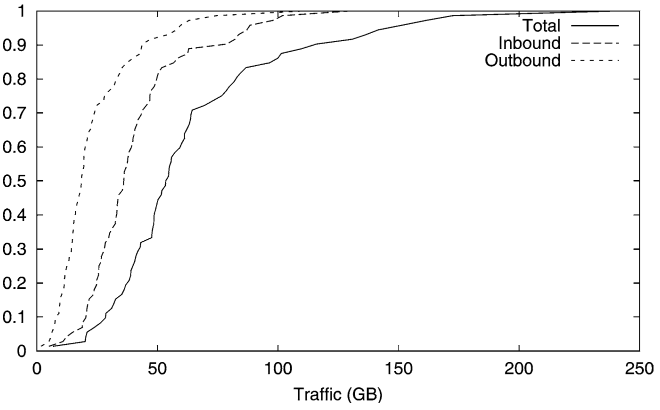

Figure 5. [SNMP] Daily traffic (GB), distribution across days.

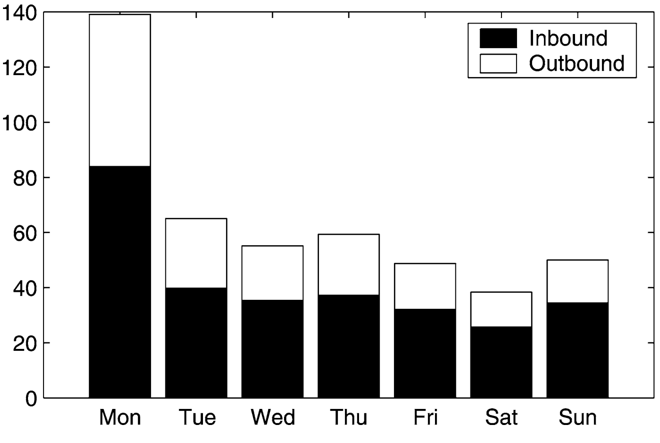

Figure 6. [SNMP] Average daily traffic (GB), by weekday.

These figures show a reasonably strong weekly pattern with some surprising peaks on Mondays. In figure 6 we see the weekly patterns more clearly. Friday and Saturday are

[4] Update: we have seen 7,566 distinct cards visit the network in the two years between its inception in April 2001 and mid-March 2003. Clearly usage is growing dramatically.

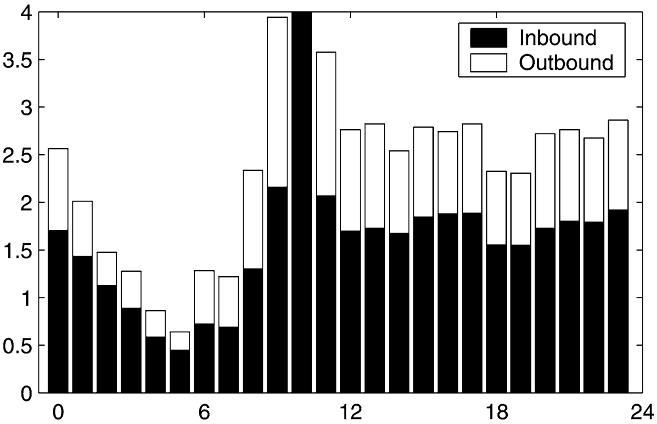

Figure 7. [SNMP] Average hourly traffic (GB), by hour (truncated). The 10 am peak reaches 11.4 GB (60% inbound).

the quietest days, as students relax, but Sunday picks up as students begin their homework. Monday's average is skewed by activity in one building, Whittemore, which includes study rooms and residences for students of the Tuck School of Business. If Whittemore is removed from figure 6, Monday's traffic matches Tuesday's (not shown).

Figure 7 displays the variation of traffic over the hours of the day. The bar for 10 AM is skewed above 11 GB by the traffic in Whittemore; without Whittemore, the 10 AM bar drops slightly below that of the 11 AM bar. When we examined the Whittemore traffic, we saw a dramatic burst of activity, both inbound and outbound, every Monday during the 10 AM hour, often accounting for nearly 100% of campus-wide traffic during that hour. We do not have data to determine the application causing the traffic, but we learned that the Tuck business school held a regular class in Whittemore in that hour, and apparently the students were often busy collaborating on group projects.

The traffic is steady throughout the afternoon and evening with a dip around dinner, tailing off through the night when students finally go to sleep, and rising again as employees return to work. Because our environment is a mixture of residential and academic uses, this plot shows a mixture of the workday bell curve and the residential evening bell curve.

In the discussion so far we present the quantity of traffic. In the following sections we examine the distribution of that traffic across campus, across users, and across protocols. But how much of this traffic is between two on-campus hosts, and how much involves off-campus hosts? This information is not available from the access points, which operate below the IP layer. Figure 8 uses the data from 22 access points captured by our tcpdump sniffer to provide an approximate answer to this question. Nearly all captured frames contained IP packets (see section 4.6), so we ignore the non-IP traffic. The figure shows that about 64% of wireless IP traffic, whether measured in bytes or packets, went to or came from a "far" (outside the dartmouth.edu domain) host, while 36% went to or came from a "near" host. Inbound traffic was the majority of far traffic (52% of bytes and 62% of packets), but the minority of near traffic (48% of bytes and 45% of packets).[5] Certainly most users are more likely to upload files to local file servers, or attachments to local mail servers, than to off-campus servers.

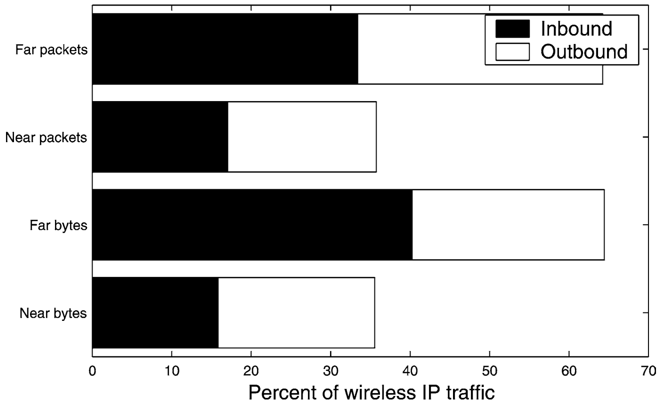

Figure 8. [tcpdump] Proportion of near and far traffic. "Near" traffic is to or from dartmouth.edu, all else is "Far".

Table 1
[syslog] Common card vendors.

Number	Vendor
624	Lucent/Agere
536	Apple Computer
489	Cisco/Aironet
57	Other (15 brands)
1706	Total

4.2. Users and user mobility

We did not (and could not) track *users*, but since for the most part each card is associated with one user, and most users have just one card, we examined cards as if they represent users. We ask:

- How many cards are there? From which vendors?
- How many days is each card active?
- How many APs does a card visit?
- How many buildings does a card visit?

There were 1706 unique MAC addresses seen in our syslog trace, most from a few common vendors (table 1). Dartmouth's campus computing store resells exclusively Apple and Dell computers, and as of 2001 all laptops sold to first-year students have wireless cards built-in: Agere (then part of Lucent) cards in the Dell laptops, and Apple Airport cards in the Apple laptops. The store also sells Cisco (Aironet) wireless PC cards, an option for those with older laptops.

Users varied in the number of days that they used their cards, from only once to every day in the 77-day trace (figure 9). Many users are students, living on campus, and it is not surprising to see some with wireless laptops on their

[5] A negligible portion of the traffic, less than one half of one percent, was both inbound and outbound, that is, between two wireless hosts monitored by the same sniffer; this graph counts that traffic only as inbound.

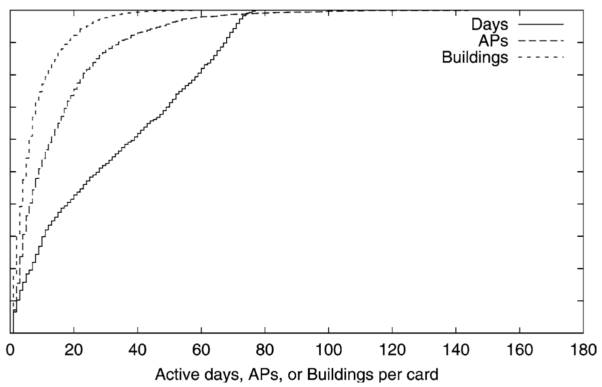

Figure 9. [syslog] Activity per card, distribution across cards. Maximums: 77 days, 64 buildings, and 161 APs. Medians: 28 days, 5 buildings, and 9 APs.

dorm-room desk, always on-line. Interestingly, the distribution is roughly uniform between one and 77 days, with a median of 28 days.

The graph also shows that few cards move around much, with a median of five buildings and nine APs, and no card visiting even half of the entire network. Indeed, nearly 18% of the cards spent all their time in one building. Clearly, most users limit their activity to a few key sites in their daily routine. We expect to see this pattern change as more small devices, such as PDAs with built-in Wi-Fi, ease mobility.

4.3. Card activity

Now that we have seen the network from the card's perspective, we examine the cards from the network's perspective:

- How many cards are active?
- When are cards active?
- How long are sessions?
- How many sessions are started each day?
- How are sessions distributed among buildings?
- How many sessions are roaming? extra-subnet?
- How often do cards roam per session?

Although there were 1706 cards seen in our traces, not all were active every day. Figure 10 shows the number of cards active in each day of our trace period. Clearly visible are the Thanksgiving holiday, weekly cycles, and a tail-off at the end of the term. Also visible is a slow trend toward more active cards per day, as more users obtain wireless capability and choose to use it more often. Here we define "active" to mean any card that is associated with an access point, regardless of whether the user is actually using the computer or network. The plot also shows "mobile" cards, which visited more than one building on that day, and "roamer" cards, which visited more than one AP during any session that day.

In another view, figure 11 shows the distribution of the number of active, roamer, and mobile cards in any given day. Almost half of our card population was active on a typical day, and over a third of those were mobile.

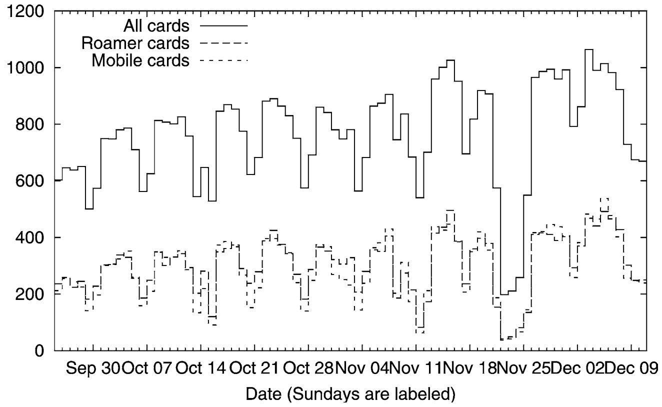

Figure 10. [syslog] Number of active, mobile, and roamer cards per day. A date's data appears to the right of its tick-mark.

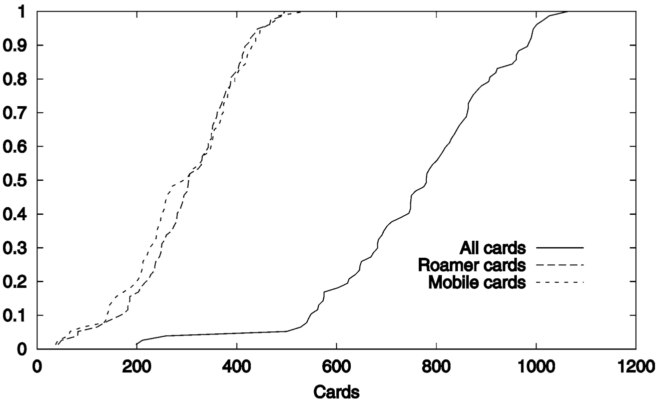

Figure 11. [syslog] Number of active, mobile, or roamer cards per day; distribution across days. Medians are 780 (all), 304 (roamer), and 301 (mobile).

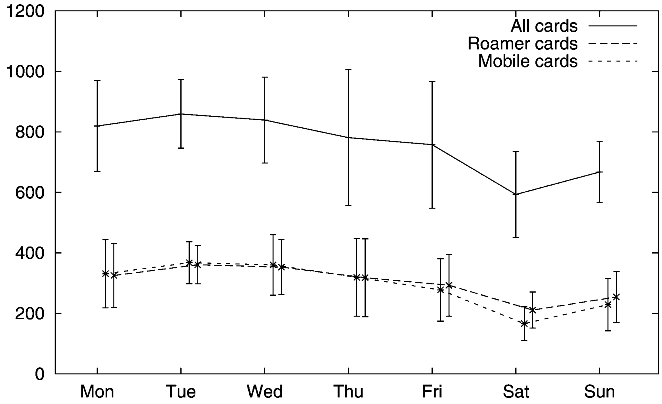

Figure 12. [syslog] Number of active, mobile, or roamer cards per weekday. The curve shows the mean, while the bars show standard deviation. The three curves are slightly offset so the bars are distinguishable.

The visible weekly cycle of figure 10 is reinforced in figure 12, which we believe reflects a typical student pattern of activity, hustling to complete their work early in the week, relaxing on Friday and Saturday, and picking up again on Sunday.

Figure 13 shows diurnal patterns. As in the hourly traffic graph, this pattern matches a mixture of workplace and

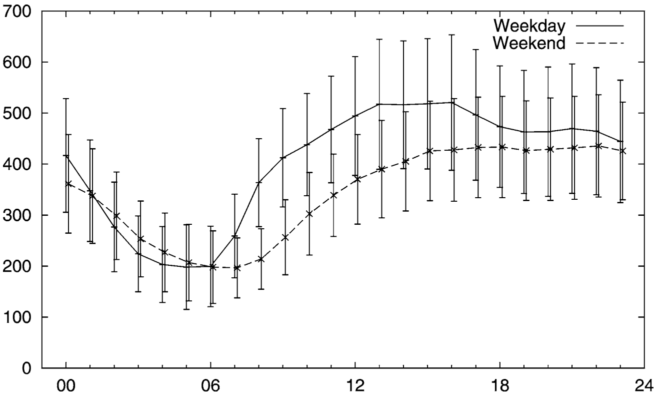

Figure 13. [syslog] Number of active cards per hour. The number of active cards for each hour of the day, separately for weekdays and weekends. The curve shows the mean, while the bars show standard deviation. The two curves are slightly offset so the bars are distinguishable.

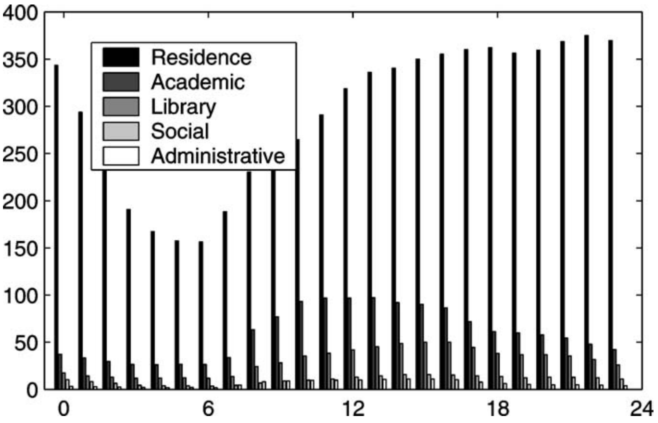

Figure 14. [syslog] Mean active cards per hour, by category. A card visiting multiple building categories within an hour was counted once for each category it visits.

residential patterns. The bulk of the activity was during the afternoon, with substantial activity during the evening and a slow decline in activity through the wee hours of the morning. Curiously, although on weekdays there were fewer active cards in the evening than in the afternoon, the traffic remains relatively flat (figure 7); perhaps evening users are busier.

With most office workers away on weekends, the weekend mid-day activity is lower, but due to the residential population the evening and overnight hours remain about the same on weekends and weekdays. We reach similar conclusions about roamer and mobile cards, not shown.

Figure 14 demonstrates the different patterns, and relative activity, of different categories of buildings on campus. Residential activity dominates. Residences and social spaces tend to be used more in the evening hours, whereas academic and administrative buildings are active during the work day, and libraries are somewhat in-between. Figure 15 shows far fewer mobile cards, particularly during the overnight hours when people move less. Figure 16 shows the roamer cards.

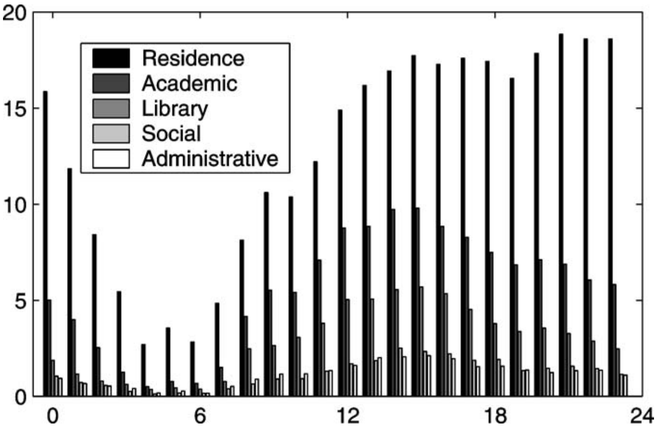

Figure 15. [syslog] Mean mobile cards per hour, by category. A card visiting multiple building categories within an hour was counted once for each category it visits.

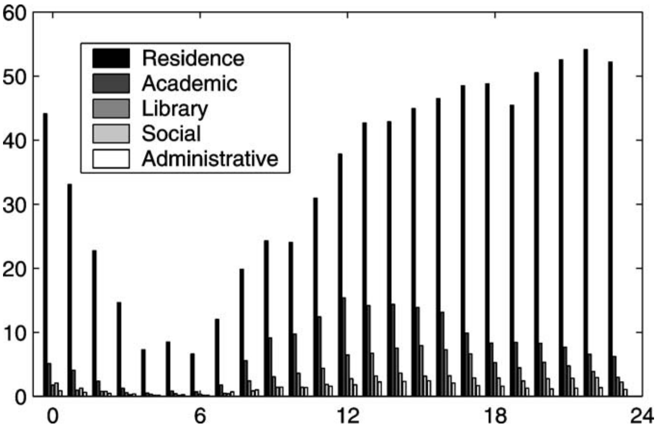

Figure 16. [syslog] Mean roamer cards per hour, by category. A card visiting multiple building categories within an hour was counted once for each category it visits.

Sessions

We are interested in when, and for how long, users choose to use the wireless network. In the preceding section we define a "session", intuitively, to be the period of activity with the network, although it is difficult to accurately detect the beginning and end of all sessions given the syslog data. We believe that our results are a reasonable approximation of the notion of a user session.

Our data (figure 17) shows that most sessions are short. The median session length was 16.6 minutes, and 71% of sessions finish in less than one hour. Given that students move frequently from class to class to dining to dorm, and like to check email in between, these numbers are reasonable.

On the other hand, there were a few sessions that were very long (69 days in one case). These extremely long sessions are likely artifacts of holes in the syslog data, in which we lost the session-ending message. There are many short sessions: 27% of sessions last less than a minute. Despite our 30-second SessionThreshold, our session-begin definition was apparently too liberal. Nonetheless, this data begs the question about why the cards associate so quickly and frequently. Examination of sample sessions show many instances in which a card

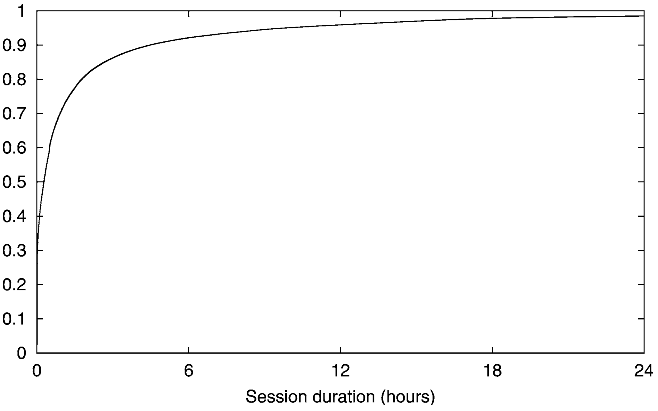

Figure 17. [syslog] CDF of session duration (truncated to 1 day). The longest session measured 69 days, although that is probably an error due to holes in our data. The median is 16.6 minutes.

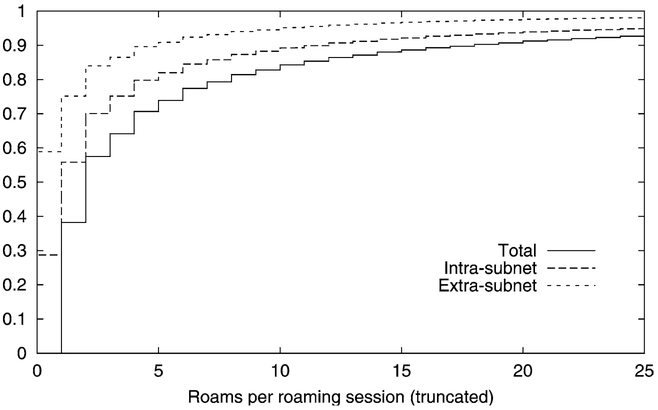

Figure 18. [syslog] Roams per roaming session, distribution across sessions. This graph is truncated. The maximum is 19,902 roams. About 18% of sessions were roaming.

Associates with an AP despite (from our reading) being associated already, an indication that the state machine in the card and in the AP are out of sync [6]. Although further study is necessary, it appears that there is substantial room for improvement in the card firmware and possibly in the 802.11 protocols.

Although most (82%) sessions are non-roaming, roaming sessions do include one or more roams. Figure 18 shows the distribution of the number of roams during roaming sessions. Most roaming sessions were short and roamed infrequently (the median is two roams). Some sessions roamed extremely frequently: one session roamed nearly 20,000 times! Nearly 60% of roaming sessions roamed only within one subnet. Unfortunately that means that over 40% roamed across a subnet boundary, which breaks connections and forces the user to obtain a new IP address.

So, why do cards reassociate so frequently? The cards aggressively search for a strong signal, and in an environment with many APs and overlapping cells, cards will roam frequently [6]. (In some cases, where the APs were from multiple subnets, it is doubtful the user had much luck using the network!) We did see a weak relationship between the card vendor and the amount of roaming; it is impossible to firmly determine the role of the card because the cards were used at different times and in different places. In any case, our three major vendors (Cisco, Lucent, and Apple) rarely had excessive roaming behavior.

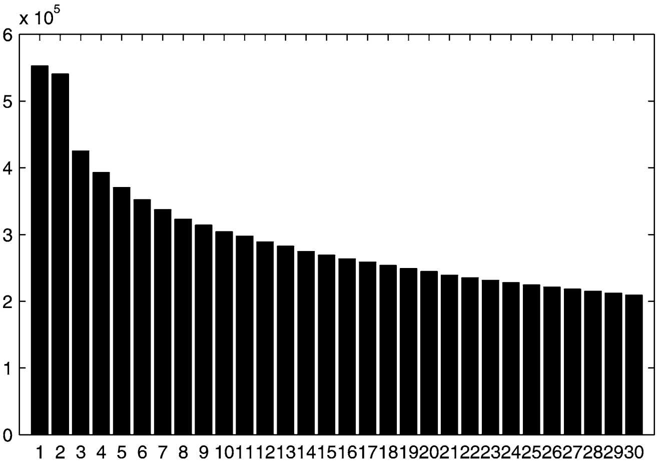

Figure 19. [syslog] Number of roams, by Roam Threshold. Over the entire trace period. Notice the *y* scale.

Either card firmware needs to be less aggressive, or our environment needs to reduce cell overlap, to reduce the roaming, reduce the resulting load on the network, and give better service to the user. Furthermore, since it is expensive to deploy a single campus-wide subnet for the wireless network [7,8], Mobile IP [13] or similar services are required to support seamless roaming.

We experimented with a *RoamThreshold* parameter, which ignores any roam if it occurs less than *RoamThreshold* seconds after the session start or a previous roam, and the results are in figure 19. Clearly, this parameter filters out many roams. It is not clear, however, what *RoamThreshold* would be appropriate for general use in our analysis. If we set the threshold too high, we may mask roams caused by real user movement. If we set it too low, our data represents the "jumpy" nature of the real cards. For the purposes of this paper, we chose threshold 0, because within the range 0–30 seconds, the choice has little effect on the graphs presented in this paper. Perhaps most significant is the measure of roams per session (figure 18). This truncated plot does not change, but the maximum value drops to 1574 for threshold = 30 rather than 19,902 for threshold = 0. In short, the threshold removes the short-term jumpiness but does not affect the conclusions drawn from figure 18.

This aggressive roaming is the likely reason that the number of "mobile" cards is similar to the number of "roaming" cards. There are many locations where a card may associate with APs in multiple buildings, despite being physically stationary. Thus the "mobile" card count is an overestimate of large-scale mobility.

Figure 20 is another view of daily network activity, in which we count the number of sessions started in each day, and here present the count as a distribution across days. The large number of sessions seen here is consistent with the shortness of sessions noted earlier. Although most session

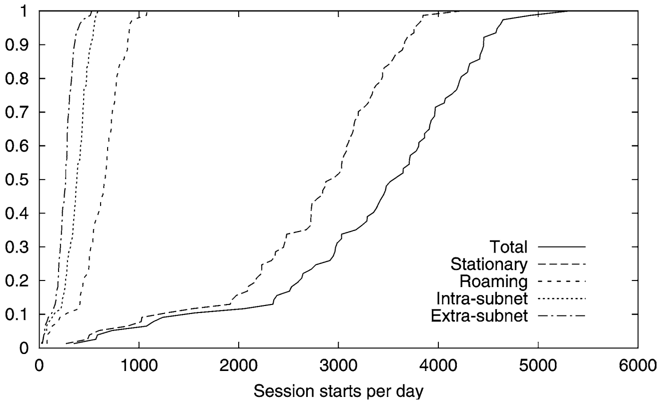

Figure 20. [syslog] Number of session starts per day, distribution over days. Median is 3582 sessions, or 664 roaming sessions.

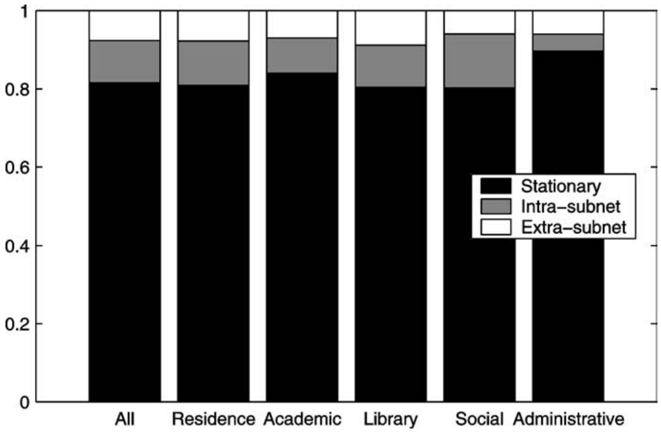

Figure 21. [syslog] Number of session starts (normalized), by category.

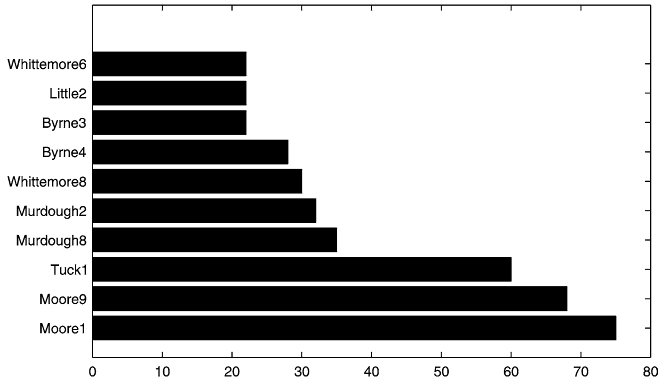

Figure 22. [syslog] Maximum cards per hour, for the busiest APs. Ranked by their busiest hour, in terms of active cards.

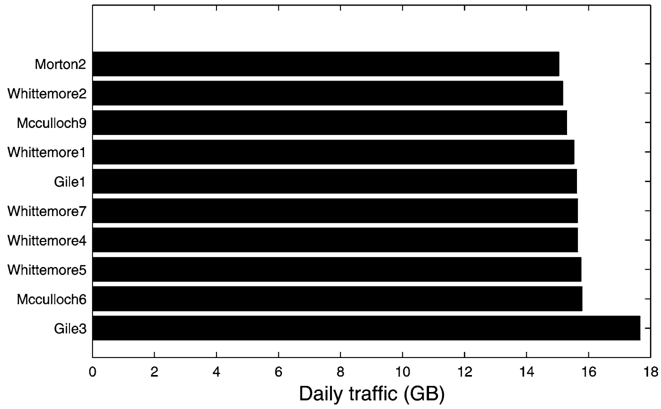

Figure 23. [SNMP] Maximum daily traffic (GB), for the busiets APs. Ranked by their busiets day.

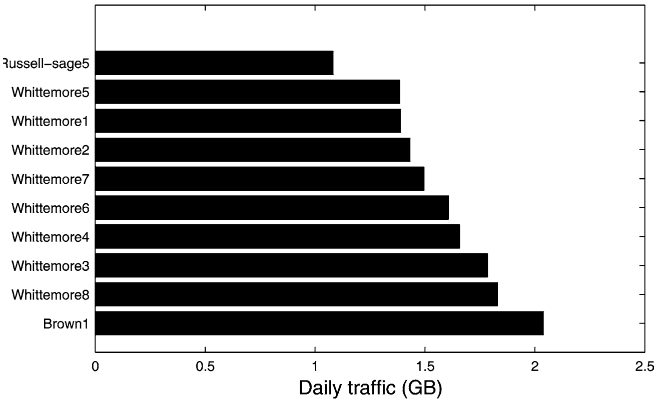

Figure 24. [SNMP] Average daily traffic (GB), for the busiets APs. Ranked by daily traffic.

starts are in the dominant category (residence), we found (figure 21) that sessions started in academic or administrative buildings tend to be more stationary, and that those in libraries tend to have slightly more extra-subnet roams. The latter may have more to do with the configuration of the libraries and subnets than any real physical mobility.

4.4. AP activity

We now examine network activity in terms of the APs:

- How many APs are there?
- When are APs active?
- How does activity vary across APs, and which are most active?
- How does traffic vary across APs, and which have most traffic?

There were 476 APs installed by the end of the study. The data in this section are based on the 430 APs in the syslog trace and the 451 responding to our SNMP polls.

A detailed identification of the busiest APs is perhaps only of internal interest at Dartmouth College, and in any case we examine the related question about the busiest buildings in the next subsection. The APs with the most active cards in their busiest hour were those located near large lecture halls; in its busiest hour, the busiest AP had 71 active cards. The traffic was elsewhere, however: the APs with the largest maximum and average daily traffic were from residences.

Figures 22–24 show the ten “busiest” access points, for three different definitions of busy. Figure 22: during our study, one professor provided a wireless PDA to each of 70 students in his class, which met in a lecture hall near Moore1. Figure 23: Gile is a dorm, and apparently one or more of its

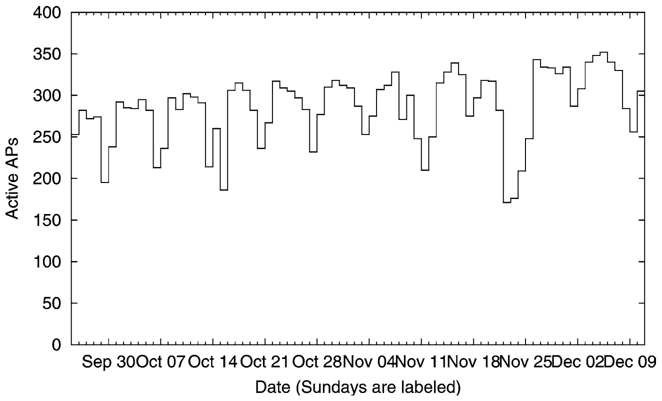

Figure 25. [syslog] Number of active APs per day. A date's data appears just to the right of its tick-mark.

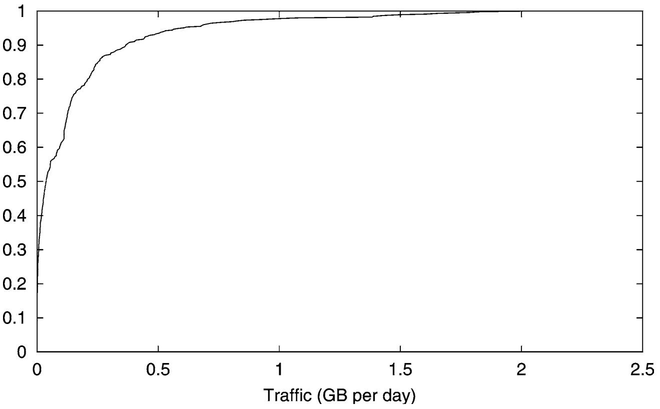

Figure 26. [SNMP] Average daily traffic (GB), distribution across APs. Median is 39 MB, maximum is 2.0 GB.

users occasionally caused a lot of traffic. Figure 24: Brown is a dorm with many first-year students (recall that in 2001 70% of first-year students own wireless laptops), and Whittemore is the residential facility in the business school (where students are required to own laptops, and are assigned a groupware assignment nearly every day).

Figure 25 shows the variation in the number of APs active each day. Clearly visible are the weekly cycle, the Thanksgiving holiday, and a general trend to use more APs, as the number of cards increased and as people used the network more. Each day saw between 171 and 352 access points in use, with a median of 292. Otherwise, we found (not shown) that the temporal patterns of active APs follows a pattern similar to the number of active cards shown in figures 13 and 14.

Over the life of the trace, the APs varied widely in the amount of traffic they handled (figure 26), with the median AP handling an average of only 39 MB per day, while the busiest AP handled an average of over 2 GB per day.

4.5. Building activity

An examination of buildings allows us to classify the most active locations on campus.

- How many buildings are there?
- When are buildings active?

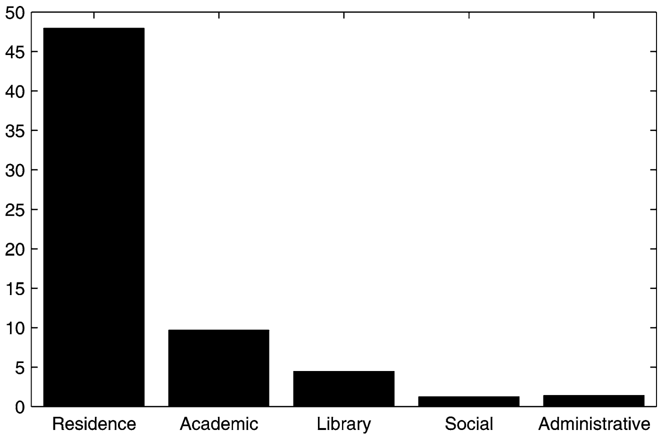

Figure 27. [SNMP] Average daily traffic (GB), by category.

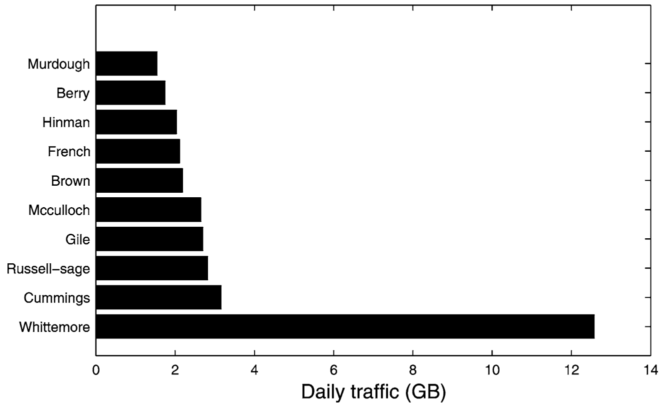

Figure 28. [SNMP] Average daily traffic (GB), for the busiest buildings. Ranked by daily traffic.

- How does activity vary across buildings, and which are most active?
- How does traffic vary across buildings, and which have most traffic?
- How does activity vary across building categories?
- How does traffic vary across building categories?

There were 161 buildings with installed APs, ranging widely from huge central libraries to tiny houses, and even a shed at the tennis courts. Although figure 27 shows that the bulk of the traffic was seen in the residential buildings (averaging 48 GB per day), when normalized by building size (number of APs, not shown) we see somewhat more balanced traffic. Residential users spend more hours in residences than most people spend in other buildings, accounting for some of this difference.

The building with the largest average daily traffic (figure 28) was Whittemore. Recall that Whittemore's traffic is skewed by the Monday 10 AM peak mentioned earlier; on the other hand, about a third of Tuck School students do have a wireless laptop. Cummings is the engineering school and Murdough is the library between Cummings and Whittemore. Berry is the main library, and the other buildings are dormitories with large populations of first-year undergraduates.

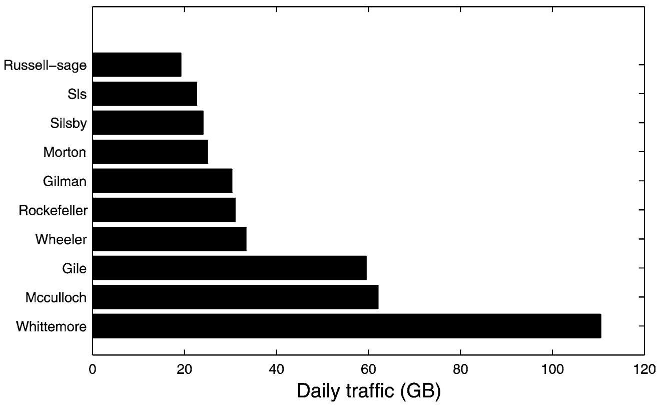

Figure 29. [SNMP] Maximum daily traffic (GB), for the busiest buildings. Ranked by their busiest day.

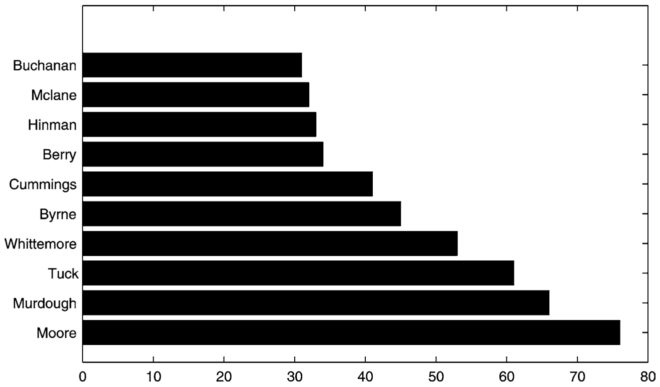

Figure 30. [syslog] Maximum cards per hour, for the busiest buildings. Ranked by their busiest hour (in number of active cards).

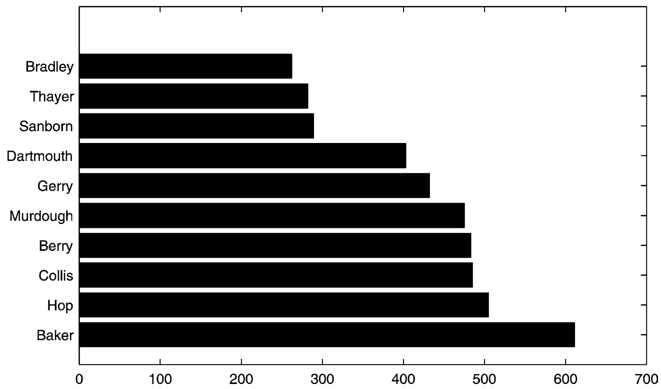

Figure 31. [syslog] Number of active cards per building, for the ten most popular buildings. Ranked by the number of unique cards visiting that building, over the whole trace.

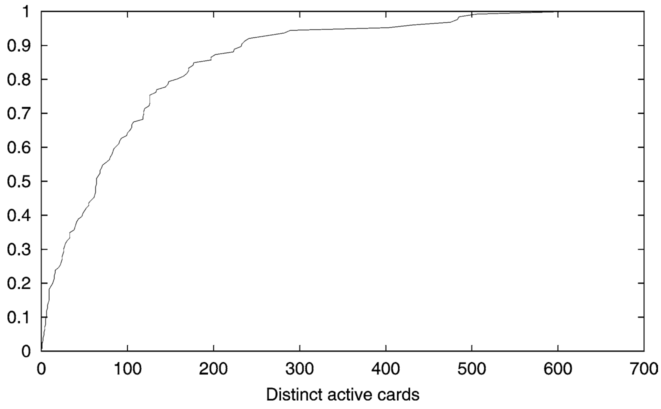

Figure 32. [syslog] Number of active cards per building, distribution over buildings.

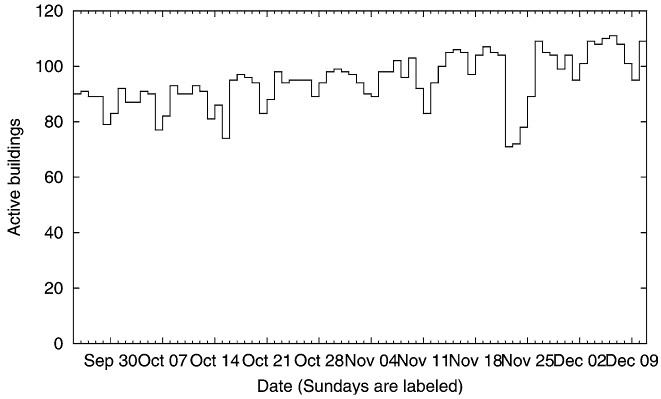

Figure 33. [syslog] Number of active cards per day. A date's data appears just to the right of its tick-mark.

Examining the busiest day for each building (figure 29), we first notice that maximum traffic is nearly a hundred times larger than the averages in figure 28. Also, other than the Whittemore, Gile, and Russell–Sage dorms, this chart has a different set of buildings: some academic buildings (Gilman, Rockefeller, and Silsby), an administrative building (Sls), and dorms. Average behavior is not a good predictor of bursty behavior.

In figure 30, the buildings with the busiest hour, in terms of the number of active cards, are mostly buildings with large lecture halls (Moore, Murdough, Tuck, Byrne, and Cummings), the main campus library (Berry), and some residences (Whittemore, Hinman, McLane, and Buchanan). Clearly network designers need to plan carefully for such large concentrations.

Finally, in figure 31, we see the buildings with the largest number of cards visiting over the entire trace. These are all large buildings where you expect a diverse population: libraries (Baker, Berry, Murdough, and Sanborn), social and dining spaces (Hop, Collis, and Thayer), an academic building with large lecture halls (Dartmouth) frequented by students in introductory courses, and the campus computer store and repair shop (between Gerry and Bradley), where wireless cards are often first installed and tested. Figure 32 shows, though, that these buildings were unusual. Half of all buildings saw fewer than 65 users over the life of the trace, less than Moore saw in a single hour.

The number of active buildings followed a pattern similar to the number of active APs and number of active cards, although the variation was dampened somewhat as we consolidate the activity into buildings. Figure 33 demonstrates the effect. Interestingly, of the 82 residences only about half are active in any given hour.

4.6. Protocols

Although the sniffer data covers only four buildings and 22 APs, it covers a variety of populations (library, dormitory, student center, and academic computer science). Above, we examine questions about where, when, and how much people use the wireless network; now, we ask about *how* they used the network:

- Which protocols are the most commonly used?
- Which protocols consume the most traffic?
- For each protocol, how many bytes (or packets) flow each way?

We captured 1.2 billion frames, of which we identified about 357 million (28%) as explicitly wireless. Of all frames, about 2.6% were sent to the broadcast MAC, and thus would be transmitted to all wireless clients, but for our purposes these frames were not "wireless" unless the source was a wireless client. Of the wireless frames, 99.7% contained IP packets, evenly split between outbound and inbound. Due to a glitch in the trace data (section 3.3), we were not always able to identify wireless non-IP packets, but the 1.2 million we could identify were all ARP (66%), Appletalk (31%), or IPX (3%).

We sniffed nearly 228 GB of wireless IP data, counting only IP data bytes (not headers). The dormitory accounted for 135 GB and the rest was roughly evenly distributed. Although we saw a tiny amount of ICMP, IGMP, PIM, RSVP, and NARP, more than 99% of the IP traffic was UDP (2.5% of bytes, 5.0% of packets) or TCP (97.5% of bytes, 94.3% of packets).

More than half (956) of all wireless cards were caught at least once by our sniffers. Although Brown dormitory saw 142 cards, and Sudikoff (Computer Science) saw 134 cards, the Collis student center saw 476 and Berry Library 729, as they are larger, public spaces with a diverse population.

We were able to identify many application-layer protocols in the TCP and UDP packets we sniffed, by recognizing "well-known" port numbers. We used the official IANA list[6] associating 3801 protocol names with TCP and UDP port numbers (or in many cases, both) and added a list of 116 Dartmouth-specific protocol assignments (which overrode 50 of the IANA definitions with local meanings). We also added another 19 assignments from another useful port list[7] that identified several common protocols that were not identified by IANA or Dartmouth. We examined each packet individually. If the packet was a TCP SYN packet, we associated the packet with the destination (server) port; if the packet was a TCP SYN/ACK packet, we associated the packet with the source (server) port; for other packets we examined both source and destination ports. If neither were well-known, we associated that packet with the "unknown" protocol. If either port was well-known, we associated that packet with that protocol. If both ports were well-known, we associated that packet with the protocol corresponding to the lower port number: in many such cases, a randomly assigned client's port happened to match one of the well-known port numbers, but in most such cases the server is using a low-numbered port (such as 80 for http) and the traffic will be associated with the server's port.

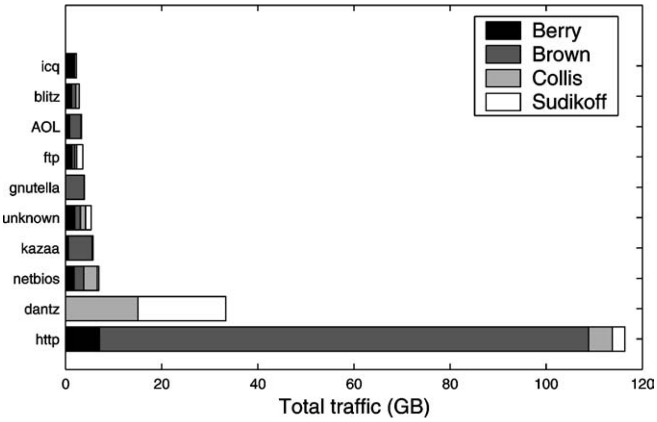

Figure 34. [tcpdump] Total traffic (GB), by TCP or UDP protocol. [Fall].

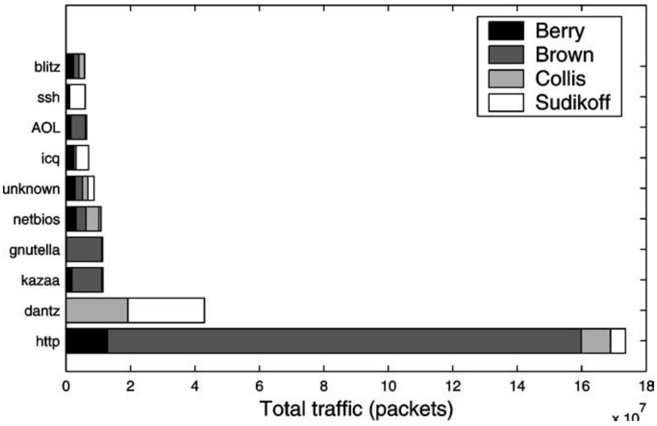

Figure 35. [tcpdump] Total traffic (packets), by TCP or UDP protocol. [Fall].

This technique is an approximation, of course, since it is possible that some applications use a "well-known" port for other purposes, but it provides a good overall estimate.

After stripping their headers, we measured 218.7 GB of TCP and UDP data. Nine protocols, and the pool of unknown protocols, account for 85.4% (186.8 GB), as shown in figure 34. Nearly the same protocols account for most of the packets as well, presumably because most of these protocols use packets whose size is a full MTU (maximum transfer unit); see figure 35. The symmetry of the traffic of figure 34 is explored in figure 36. Although most were asymmetric by bytes, they were all nearly symmetric by packet count (not shown), presumably because even one-way file transfer typically requires one acknowledgement packet for each data packet. We now look at each in detail.

http (116 GB, 53%), including both http and https, and some other common http ports (such as 8000). Clearly, web browsing is a significant fraction of any network traf-

[6] www.iana.org/assignments/port-numbers

[7] www.seifried.org/security/ports/

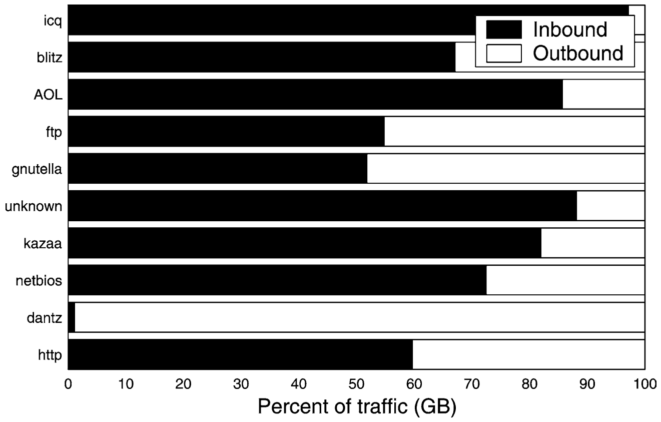

Figure 36. [tcpdump] Total traffic (GB), by TCP or UDP protocol, normalized. [Fall].

fic today. It is not dominant everywhere, however: in Collis and Sudikoff, there was less http traffic than "dantz". Although most http traffic is inbound, there is substantial outbound traffic in Brown; most likely this traffic represents file-sharing programs operating over http, or the use of web-based email clients. Although it cannot be determined from figures 34–36, it turns out that over 96% of the http bytes and packets involve off-campus hosts. Thus it does not seem, for example, that the outbound traffic is destined for an on-campus homework server.

dantz (33 GB, 15%), a protocol for the Retrospect backup product from Dantz corporation, in use here for office Macintosh computers. Collis and Sudikoff have several such offices, and the "dantz" protocol dominates the traffic seen by those sniffers. The traffic is mostly outbound, of course, as wireless clients are backed-up to a wired server. While it was an unexpected frontrunner, a few periodic backups accounted for the volume.

netbios (6.9 GB, 3.2%), a set of Windows protocols (dgm, ns, ssn) that support Windows print and file sharing, including Samba.

kazaa (5.7 GB, 2.6%) and gnutella (3.9 GB, 1.8%), two popular Internet peer-to-peer file-sharing applications. Seen mostly in the dorm and the library.

unknown (5.3 GB, 2.4%): For these packets, neither the source nor the destination port number was on our port lists. Other traffic matched obscure ports on the list, but we doubt they actually use the associated protocols, so the "unknown" category should actually be larger. We speculate that these connections may be related to file-sharing or gaming applications in which a coordinator arranges peer-to-peer connections through arbitrary ports.

ftp (3.6 GB, 1.6%), including all variants of the common file-transfer protocol, including ftp, ftp-data, ftplog, bftp, tftp, ftps, and sftp. Curiously, there was nearly an even split between outbound and inbound data, although in each sniffer (not shown) it is more skewed toward either outbound (Collis and Berry) or inbound (Brown and Sudikoff). This protocol does not appear on the packet-counting figure 35, where ssh appears instead. Clearly, ftp transfers large amounts of data using relatively few large packets, whereas ssh (often used for interactive connections) uses many tiny packets.

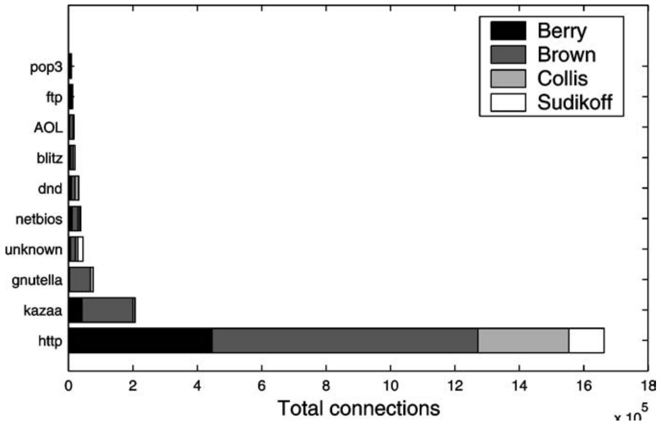

Figure 37. [tcpdump] Total connections, by TCP protocol. [Fall].

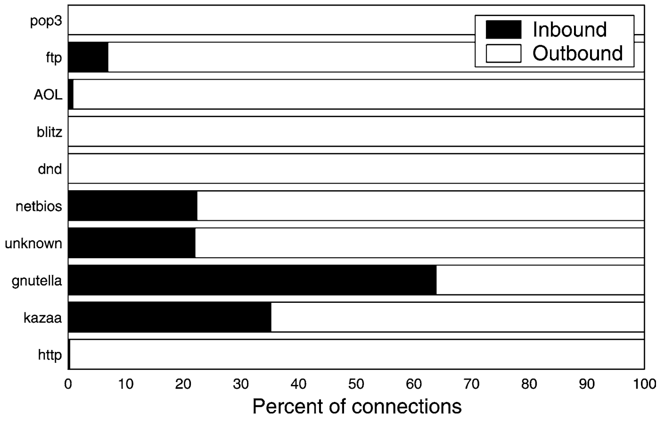

Figure 38. [tcpdump] Total connections, by TCP protocol, normalized. [Fall].

AOL (3.4 GB, 1.5%): Instant messaging is gaining in popularity.

blitzmail (2.9 GB, 1.3%): BlitzMail is a locally developed email client, with a custom protocol, in ubiquitous use outside Sudikoff. The high volume is no doubt the result of large enclosures.

icq (2.3 GB, 1.0%): ICQ is a popular Internet instant-messaging application. ICQ is far more skewed toward inbound traffic than AOL, perhaps because ICQ is used more for group chat than for one-to-one communication. There are also several Trojan-horse programs that use related ICQ ports, but that traffic was negligible.

Most of the above protocols are commonly used for file transfer, which accounts for their dominance in this ranking based on volume. Nearly the same protocols dominate when ranked by the number of TCP connections, as shown in figure 37. Two new protocols appear here: dnd is a custom Dartmouth name-service protocol, and pop3 is a standard email access protocol. Figure 38 confirms that most wireless hosts are clients of these standard services, although kazaa and gnutella peer-to-peer protocols see a substantial fraction of inbound connections.

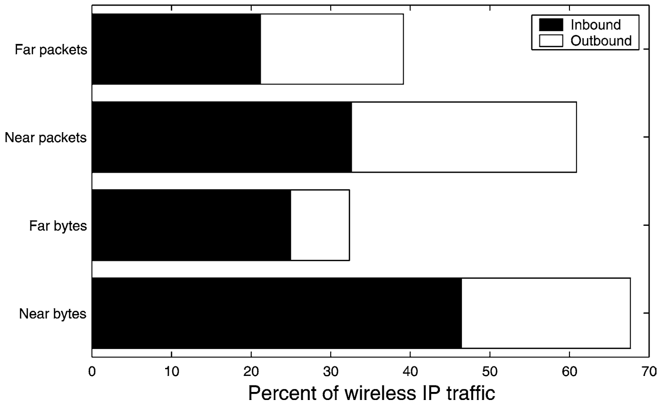

Figure 39. [tcpdump] Proportion of near and far traffic. “Near” traffic is to or from `dartmouth.edu`, all else is “Far”. [Spring; compare with figure 8].

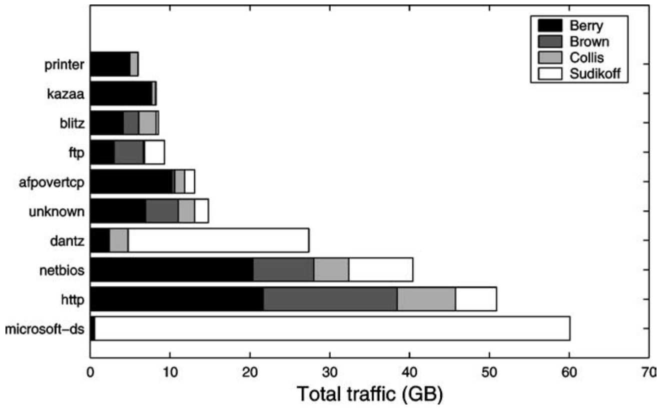

Figure 40. [tcpdump] Total traffic (GB), by TCP or UDP protocol. [Spring; compare with figure 34].

Springtime

Six months after our Fall 2001 study, we collected data from March 24, 2002 through June 9, 2002, inclusive. This data presents an interesting contrast to the fall data, indicating a need for caution in interpreting the fall data too closely. In any case, remember that both sniffer data sets involve only four buildings and 22 access points.

First, figure 39 stands in stark contrast to figure 8: in the fall, most traffic involved off-campus hosts (“far”), but in the spring, the pattern was completely reversed. About 65% of the wireless traffic relates to on-campus hosts, and most of the traffic is inbound in either case. Looking inside the numbers, the far traffic increased slightly from fall to spring, but the near traffic increased by a factor of 4–5.

A better understanding of the change in traffic appears in figure 40; compare this to figure 34. The dormitory (Brown) had dramatically less http traffic (17 GB rather than 102 GB). We suspect that much of the fall http traffic was peer-to-peer file sharing over http; in the spring, Kazaa and Gnutella also declined in the rankings. The spring http was skewed strongly toward inbound traffic (figure 41), and yet the connections were outbound (figure 41), so the spring data seems unlikely to reflect peer-to-peer sharing over http.

The most stunning feature of the spring data in figure 40 is microsoft-ds which consumed a whopping 60 GB of bandwidth, largely in the computer-science building Sudikoff. Fortunately, figure 43 makes it clear that the attack was originating externally. Its packet count (figure 42) was lower than that of http, indicating that its packets are relatively large. This part (445) is used for Microsoft Windows file system traffic, accounting for the large packets and large amount of traffic. It is also, however, the vector for attacks on Windows, so it is possible that some or all of the traffic is related to inbound attacks.

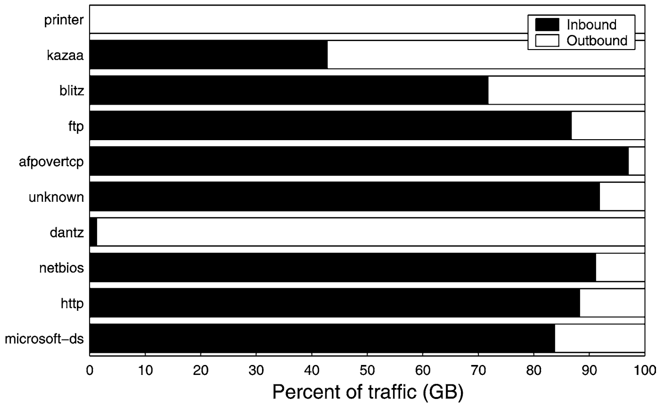

Figure 41. [tcpdump] Total traffic (GB), by TCP of UDP protocol, normalized. [Spring; compare with figure 36].

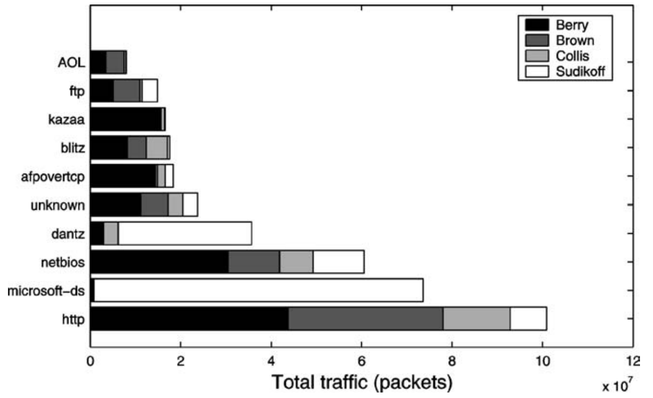

Figure 42. [tcpdump] Total traffic (packets), by TCP of UDP protocol, normalized. [Spring; compare to figure 35].

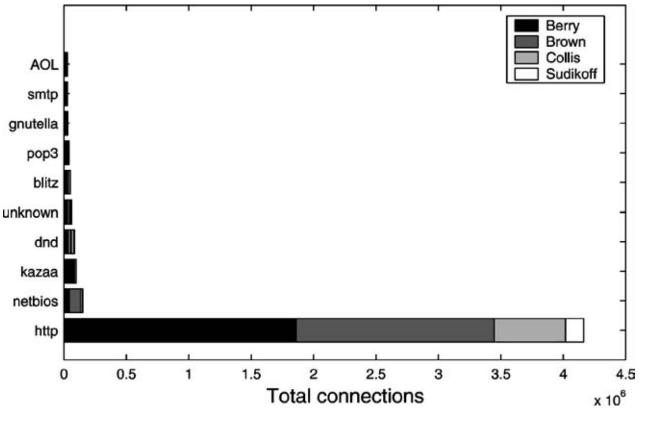

Figure 43. [tcpdump] Total connections, by TCP protocol. [Spring; compare to figure 37].

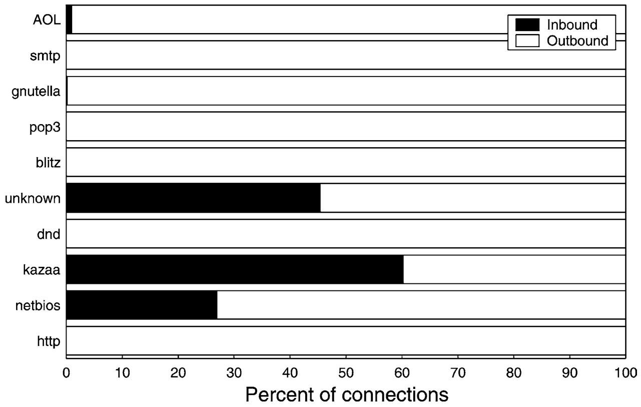

Figure 44. [tcpdump] Total connections, by TCP protocol, normalized. [Spring; compare to figure 38].

The protocol "afpovertcp" is Apple Filing Protocol over TCP, allowing Macintosh computers to share file systems without using Appletalk. Uncommon in the dorm, it seems most commonly used by staff and visitors in the library, and (figure 42) mostly for downloading.

The "printer" protocol is port 515, the lpd protocol for remote printing. Dartmouth introduced a new network printing service that spring, with printers located in Berry and Collis, so it is not surprising to see significant usage in those buildings.

Summary

While the details of our protocol distribution may be specific to Dartmouth, we expect that others in academic environments will see approximately the same set of activities dominating: web, email, backup, messaging, file transfer, and file sharing.

5. Related work

At the time of this writing (April 2003), our study is the largest and most comprehensive characterization of wireless LAN users. In three earlier studies, Tang and Baker characterized wireless-network usage. In 1998 they used tcpdump in a limited study of eight laptops over eight days [12], focusing on the number of times the laptops switched between wired and wireless, and on the latency encountered by packets. They note that users did tend to behave differently on the wireless network than on the wired network, due to extremely high latencies. In 1999 they characterized the users of the Metricom Ricochet network, a wireless metropolitan-area network service [16,18]. This study is notable for its size (24,773 clients and 14,053 access points) and duration (about seven weeks). Given the nature of the data available, their analysis focuses on network activity and client mobility. Finally, in 2000 they use tcpdump and SNMP records to characterize the activity of 74 Wi-Fi users in the Stanford Computer Science Department, over a 12-week period [17]. While this study is similar to our own, our population is much larger and more diverse, and our roaming patterns are more complex than a single subnet in a single building. We have syslog data that allows more precise measurements of roaming, but we do not have authentication data that allow us to associate MAC addresses with users. Although we do not have sniffer data for the entire population, we do have it for four diverse buildings. Their top five protocols (http, netbios, ftp, unknown, and ssh + telnet) represent the CS workload; ours (http, dantz, unknown, netbios, and kazaa) include CS as well as a more diverse workload.

During our study, Balachandran et al. [2] traced 195 Wi-Fi users at one location during the active hours of the 2.5 day SIGCOMM conference. Their results are necessarily limited by the homogeneous activity, as all attendees follow the conference schedule. They found that most sessions were short, less than 10 minutes; that longer sessions tended to be idle most of the time; that the cards were evenly distributed across the four APs, but the offered load was highly unbalanced across users and hence across APs. The peak bandwidth requirements were well within the capability of four access points. Of their TCP traffic, http accounted for 46% of all bytes (for us, 53%) and 18% ssh (for us, 0.57%), reflecting their computer-scientist audience.

In another recent study, Hutchins and Zegura study usage patterns on the campus of Georgia Institute of Technology [9]. During winter and spring of 2001 they captured the wireless activity of the 802.11b network on part of their campus, including 109 wireless access points spread across 18 buildings. During their two-month study they recorded the activity of 444 wireless hosts using a sniffer, SNMP polling, and records from their authentication server. The authentication data allows them to precisely determine session boundaries, rather than using the syslog records as we do. Their results show a strong diurnal pattern in the number of sessions and the number of TCP flows. Both trail off significantly in the early evening, unlike our related data for active cards and traffic, probably because their network does not cover residential areas of campus.

They used SNMP to poll each access point every fifteen minutes. They use this data to detect user mobility, as in the Balachandran study. They did see a significant amount of short inter-move times, and seem to have seen some cards oscillate between access points. The shortest inter-move times were lost in their 15-minute polling interval, so their distribution is skewed toward longer inter-move times; 40% or more of all inter-move times were longer than one day. On the other hand, a third of all users never moved.

They used a single sniffer to monitor TCP and UDP flows through all access points. They count flows and measure flow duration, rather than the number of gigabytes. They present flow lengths and counts for FTP, SSH, Telnet, SMTP, DNS, HTTP, POP3, IDENT, NNTP, NETBEUI, IMAP, and SSL. Although it is clear in their data that HTTP and SSL dominate the others, it is not clear whether these 12 protocols were in fact the most popular protocols.

Finally, another recent study by Balazinska and Castro [3] used SNMP for four weeks to trace the activity of 1366 users of 117 access points spread over three large buildings of a corporate research campus. They compare their data to ours and the other prior studies and comment on the similarities. They also compute two new metrics, *prevalence* of an access-point in user traces, and user *persistence* at locations.

The Wireless Andrew project at Carnegie-Mellon University created the first large WaveLAN installation, and their papers discuss the design and deployment of that network [4, 7,8]. Although they hint of plans for a usage study [4], there are as yet no published results.

Kunz et al. studied customers using WAP web browsers on their cell phones [11]. For seven months they used tcpdump to capture packets at the WAP gateway. Unfortunately, they were unable to identify unique users or phones, but the number of IP addresses assigned in any given day increased to about 400 by the end of the trace. The PCS network reassigned an IP address whenever the browser was idle for 90 seconds, so the session lengths were quite short (average 3.38 minutes). Otherwise, the usage followed the expected weekly and daily patterns.

A more recent study [1] characterizes the behavior of cellular wireless users of a specific commercial service, an information browsing and notification service. Due to the nature of the service, the nature of the mobile clients (mostly cellular phones), and the nature of the data available (URLs), their study focuses on different questions than ours. They focus on content; we focus on the network: protocols, mobility, and traffic.

Another recent study [14] traced all HTTP and peer-to-peer (P2P) traffic at the border routers on the campus of the University of Washington, for nine days in spring 2002. Although they have no data related to wireless networks, their results provide significant insight into the nature of web and P2P traffic, notably Kazaa and Gnutella. They found that outbound P2P traffic dominated, by an order of magnitude; in contrast, our results for Kazaa and Gnutella show a dominance of inbound traffic (figure 36). It seems likely that most wireless users do not leave a Kazaa or Gnutella peer running for much longer than the time needed to obtain the desired content, and thus the peer is a net consumer.

6. Conclusions

We conducted a large trace-based study of wireless LAN users in an effort to understand patterns of activity in the network. Although our population was large and diverse, it is important to interpret our results within its context. Our residential university campus population may not reflect activity on a corporate campus, a public space, or other venues.

The activity and traffic varied widely from hour to hour, day to day, and week to week. While we do see clear daily and weekly patterns, they reflect a mixture of a residential campus and an academic workplace, including more overnight usage than might be common in enterprise WLANs. We found that many wireless cards are extremely aggressive when associating with access points, leading to a large number of short "sessions" and a high degree of roaming within sessions. About 17% of sessions involved roaming, and of these "mobile sessions" about 40% involved roaming to a different subnet. From anecdotal evidence, these extra-subnet roams often occur when when the user is stationary, leading to failures of IP traffic.

Network designers should note the high variance in the activity of buildings, access points, and cards, over both time and space. We need new solutions to prevent cards from roaming too frequently, without sacrificing coverage. We need network-layer [13] and application-layer solutions to support multi-subnet roaming. Finally, note that the traffic is not definitively dominated by outbound or inbound traffic. The ratio varied significantly from day to day, building to building, and protocol to protocol. This conclusion argues against any design with asymmetric bandwidth.

In the early stages of the wireless project, the staff at Dartmouth College debated whether it would be important to provide wireless coverage in the dormitories, which were already wired with at least one port per resident. Our data shows that the bulk of wireless activity occurs in the residences. Furthermore, for wireless network connectivity to be useful to a mobile user, it needs to be pervasive, allowing the user to grab their laptop on the way out the door, confident that there will be network access wherever they may go. Nonetheless, we saw that most users visited few APs and buildings over the life of the trace, and most users were stationary within a session.

Future work

Our study, and nearly all of the studies before it, characterized only the wireless network. It would be useful (but extremely difficult, on switched networks) to collect simultaneous information about usage on the wired and wireless networks, to determine the characteristics unique to the wireless environment.

We would like to study the geographic patterns of mobility. Presumably most users have regular habits as they move from dorm to class to dining hall.

We were unable to distinguish users or types of users (students, faculty, staff). It may be possible to infer the type of users from their behavior (for example, students are seen frequently in dorms), or to use clustering techniques [17]. We were also unable to distinguish the mobile host hardware (PDA, laptop, or desktop) or operating system, but for those seen in a tcpdump trace we may be able to learn something from the protocols they use.

Availability

All of the data used in this study, and more, is available at `http://www.cs.dartmouth.edu/~campus/`

Acknowledgements

The authors graciously acknowledge the contribution of Pablo Stern; for his Senior Thesis he constructed many of the scripts for collecting and analyzing the SNMP and sniffer data, and used them in an early analysis of the network as it came online in Spring 2001 [15]. Abhishek Nair assisted in collecting the Spring 2002 data set. More recently, Clara Lee helped to program some of the analysis software, and she and Libo Song helped to anonymize the data for release.

We are also grateful for the assistance of the staff of Dartmouth Computing Services, particularly Steve Campbell, Punch Taylor, Jim Baker, and Charles Clark, and of Computer Science, particularly Wayne Cripps, Arne Grimstrup, Ron Peterson, and Tim Tregubov. We are indebted to the open-source developers of ucd-snmp, tcpdump, CoralReef, gawk, perl, gnuplot, and other great tools, and we appreciate the feedback from the anonymous reviewers.

Finally, we thank Cisco Systems for their funding, equipment, and technical assistance. This study was funded by the Cisco Systems University Research Program and is a project of Dartmouth's Center for Mobile Computing.

References

[1] A. Adya, P. Bahl and L. Qiu, Characterizing alert and browse services for mobile clients, in: *Proceedings of the 2002 USENIX Technical Conference*, USENIX Association (2002) pp. 343–356.

[2] 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 2002 ACM SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, Marina Del Rey, June 2002 (ACM Press, 2002) pp. 195–205.

[3] M. Balazinska and P. Castro, Characterizing mobility and network usage in a corporate wireless local-area network, in: *Proceedings of the 2003 International Conference on Mobile Systems, Applications, and Services (MobiSys)* (USENIX Association, May 2003).

[4] B.J. Bennington and C.R. Bartel, Wireless Andrew: Experience building a high speed, campus-wide wireless data network, in: *Proceedings of the 3rd Annual International Conference on Mobile Computing and Networking* (ACM Press, September 1997) pp. 55–65.

[5] R. Boggs and P. Arabasz, The move to wireless networking in higher education, Technical Report 7, Research Bulletin of the EDUCAUSE Center for Applied Research (April 2002).

[6] J. Christy, Cisco Systems engineer, Personal communication, March 4 (2002).

[7] A. Hills, Wireless Andrew, IEEE Spectrum 36(6) (1999) 49–53.

[8] A. Hills and D.B. Johnson, Seamless access to multiple wireless data networks: A wireless data network infrastructure at Carnegie Mellon University, IEEE Personal Communications 3(1) (1996) 56–63.

[9] R. Hutchins and E.W. Zegura, Measurements from a campus wireless network, in: *Proceedings of the IEEE International Conference on Communications (ICC)*, Vol. 5 (IEEE Computer Society Press, New York, April 2002) pp. 3161–3167.

[10] D. Kotz and K. Essien, Characterizing usage of a campus-wide wireless network, Technical Report TR2002-423, Department of Computer Science, Dartmouth College (March 2002).

[11] T. Kunz, T. Barry, X. Zhou, J.P. Black and H.M. Mahoney, WAP traffic: Description and comparison to WWW traffic, in: *Proceedings of the Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2000)* (ACM Press, Boston, MA, August 2000) pp. 11–19.

[12] K. Lai, M. Roussopoulos, D. Tang, X. Zhao and M. Baker, Experiences with a mobile testbed, in: *Worldwide Computing and Its Applications*, Lecture Notes in Computer Science, Vol. 1368 (Springer, 1998) pp. 222–237.

[13] C.E. Perkins, Mobile networking in the Internet, Mobile Networks and Applications 3(4) (1999) 319–334.

[14] S. Saroiu, K.P. Gummadi, R.J. Dunn, S.D. Gribble and H.M. Levy, An analysis of Internet content delivery systems, in: *Proceedings of the 2002 Symposium on Operating Systems Design and Implementation* (USENIX Association, Boston, MA, December 2002) pp. 315–328.

[15] P. Stern, Measuring early usage of Dartmouth's wireless network, Senior Honors Thesis, Technical Report TR2001-393, Department of Computer Science, Dartmouth College (June 2001).

[16] D. Tang and M. Baker, Analysis of a metropolitan-area wireless network, in: *Proceedings of the 5th Annual International Conference on Mobile Computing and Networking* (ACM Press, August 1999) pp. 13–23.

[17] D. Tang and M. Baker, Analysis of a local-area wireless network, in: *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking* (ACM Press, Boston, MA, August 2000) pp. 1–10.

[18] D. Tang and M. Baker, Analysis of a metropolitan-area wireless network, Wireless Networks 8(2–3) (2002) 107–120.

David Kotz is a Professor of Computer Science at Dartmouth College in Hanover, NH. He received the M.S. and Ph.D. degrees in computer science from Duke University in 1989 and 1991, respectively. He received the A.B. degree in computer science and physics from Dartmouth College, Hanover, NH, in 1986. He rejoined Dartmouth College in 1991 and was promoted to Professor in 2003. His research interests include wireless networks, context-aware mobile computing, and intrusion detection. He is a member of the ACM, IEEE Computer Society, and USENIX associations, and of Computer Professionals for Social Responsibility.
E-mail: dfk@cs.dartmouth.edu
WWW: http://www.cs.dartmouth.edu/∼dfk/

Kobby Essien is doctoral student in the Department of Bioengineering at the University of Pennsylvania. At the time of this research he was an undergraduate computer-science major at Dartmouth College; he received the A.B. degree in computer science from Dartmouth College in 2002. His research interests involve networking and the application of computer science techniques to biomedical problems.
E-mail: Kobby.Essien@alum.dartmouth.org