Practical Schemes for Smooth MAC Layer Handoff in 802.11 Wireless Networks

Yong Liao, Lixin Gao
University of Massachusetts
Department of Electrical and Computer Engineering
Amherst, Massachusetts 01003, USA
{yliao, lgao}@ecs.umass.edu

May 01, 2005

Abstract

The limited service range of the access points demands mobile wireless stations to handoff frequently between different cells in the IEEE 802.11 infrastructure networks. However, the handoff scheme used in the current 802.11 infrastructure networks is far from graceful. In this paper, we propose a smooth MAC layer handoff scheme and a greedy smooth MAC layer handoff scheme. Our handoff schemes scan channels in a smooth manner so that the handoff can have less impact on upper layer applications. In order to limit the frequency of channel scanning, an adaptive mechanism is used to dynamically adjust the threshold triggering the channel scanning operation. We have implemented our handoff schemes using commodity 802.11 devices and extensive experiments have been conducted to evaluate the performance. The experimental results demonstrate that our schemes reduce packet delay and loss during handoff. Our handoff schemes are implemented in the client side only and do not require changes to access points. Therefore, our schemes can be deployed in the existing 802.11 wireless networks.

1 Introduction

Over the past few years, numerous 802.11-based wireless networks have been deployed in campuses, hotels, airports and companies. Besides providing convenient access to the Internet, 802.11 wireless networks have sufficient bandwidth to support various promising applications such as VoIP. However, the small service range of the access points (AP) makes mobile wireless stations frequently handoff between different cells. Previous studies have shown that the handoff latency can be several hundred milliseconds, which is too long for real-time applications such as VoIP or video streaming. Therefore, an efficient handoff scheme is critical to ensure that multimedia applications can be deployed in 802.11 wireless networks.

Existing measurements have shown that the channel scanning phase is the most time consuming phase in handoff. It contributes as much as 90% to the entire handoff latency[6]. During handoff, all incoming and outgoing frames are dropped. Previous works propose to scan less channels to reduce the handoff latency [2] [3] [5]. In one approach, the wireless station figures out which channels may have working APs and scans only those channels. The other approach is the AP reports its neighboring APs to the wireless station. We choose to implement the handoff on the client side and not to make any change to the AP side. So our schemes can be deployed in the existing 802.11 wireless networks.

From the point of view of applications, we can generalize the goal of improving handoff as minimizing the handoff's impact on applications. Most active scan handoff schemes focus on making handoff *fast* by reducing the number of scanned channels as well as the duration of scanning a channel. We observe that besides making the handoff fast, there is another way to improve the handoff process. We can break the long duration discovery phase into several pieces. By this means, there is not a *long* transmission interuption, so the handoff will be smooth and graceful and the upper layer applications may even not notice the handoff occurs.

We propose a handoff scheme, *smooth handoff,* in which the scan channel phase is splitted into multiple subphases. The wireless station can use the interval between two consecutive subphases to send and receive data frames. Obviously, this can reduce packet delay and jitter during the channel scanning phase, which is important for time critical applications as VoIP. If the wireless station's available queue buffer size is small, our scheme can efficiently reduce packet loss as well. We further extend the smooth handoff into a greedy smooth handoff scheme. The later scheme not only scans channel smoothly but also reduces the number of channels being scanned.

We have implemented our handoff schemes using commodity 802.11b devices. Extensive experiments and measurements have been conducted to evaluate their performance. In order to assess the impact on applications, we had experiments to test our handoff schemes in the context of real-time voice traffic transmission and TCP file transmission. The results demonstrate that our handoff schemes are far more imperceptible to applications than the existing handoff scheme used in 802.11 networks.

The rest of this paper is organized as follows. In section 2 we introduce the background of IEEE 802.11b/g standard and previous works on MAC layer handoff. Section 3 presents the details of our handoff schemes. Section 4 describes the implementation issues. In Section 5 we present the experimental evaluation. Section 6 concludes this paper.

2 Background and Related Work

2.1 Background of IEEE 802.11b/g Wireless LAN Standard

There are three main variants of 802.11 standard, designated by the letters b, g and a. In this paper, we present our handoff schemes in the context of 802.11b/g since the majority of existing WLANs use 802.11b/g. The 802.11b/g networks operate in the 2.4GHz unregulated spectrum, which is further divided into 11 partly overlapped channels. Among the 11 channels, channel 1, channel 6, and channel 11 are the maximum number of channels that are not overlapping in spectrum. The 802.11b/g wireless networks are usually organized in the *infrastructure* mode, in which the system is subdivided into cells. Each cell has a central controller called the *access point* (AP). The AP acts as the bridge between the wired and the wireless world.

There is a mechanism defined in the 802.11b/g standard for measuring the RF energy. The numeric value of measured signal quality is an integer with a range of $0 \sim 255$ (1-byte value), which is called the *Receive Signal Strength Indicator* (RSSI). The standard also provides a layer-2 handoff process that allows the station to associate with a new AP, if the current AP's RSSI drops below a threshold. The 802.11b/g standard provides a set of functions for the MAC layer handoff, such as the active/passive scan, the authentication process, and the reassociation process. The layer-2 handoff usually lasts several hundred milliseconds. During handoff, the wireless card cannot send or receive data frames, because the card does not work on the same channel of its current AP.

In active scan, the wireless station broadcasts a *Probe Request* frame on one channel after contending to the medium and starts a *Probe Timer*. If no activity is detected in the wireless media when the *Probe Timer* reaches *MinChannelTime*, the station believes that no AP is working in that channel and it should scan another channel. If the station detects that the channel is not idle, it will wait for *Probe Response* frames from working APs until the *Probe Timer* reaches *MaxChannelTime*. An empirical measurement shows that *MinChannelTime* is about $20ms$, and *MaxChannelTime* is about $40ms$ [6].

Passive scan can be used to discover nearby APs as well. The 802.11b/g APs can periodically broadcast *beacon* frames to announce their presence. To discover working APs on one channel, the wireless station should switch to that channel and waits for the beacon frames from APs. The default beacon generating interval is $100ms$. It takes $100ms \times 11 \approx 1$ second to discover all APs woking in 11 channels. Since passive scan always has longer latency than active scan, most of the current wireless cards use active scan to probe available APs [7].

The handoff procedure can be divided into three phases, *discovery phase*, *authentication phase*, and *reassociation phase*. Corresponding to the three phases, the entire handoff latency consists of three components: *probe delay*, *authentication delay*, and *reassociation delay*. Among all operations in handoff process, the discovery phase contributes as much as 90% to the overall handoff latency [6]. The authentication and reassociation usually take only a few milliseconds to finish [1]. Therefore, the most efficient way to improve the handoff is to change the discovery phase.

2.2 Related Work

Previous studies on reducing the handoff latency fall into two categories: the active scan based methods and the passive scan based methods.

2.2.1 Active Scan Handoff

We use *full scan handoff* to denote the original active handoff scheme of the wireless card, which scans all channels *consecutively* in the discovery phase. Most improvements to the active scan handoff strive to scan less channels. This is called as *selective scan handoff*. Obviously, if the wireless station knows which channels have working APs, it probes only those channels. The 802.11k standard introduces *Neighbor Report*, which makes selective scan a reality. But we could not expect the existing networks will switch to 802.11k in the near future.

The authors of [2] propose a MAC layer fast handoff. They use selective scan and record the scan results in the “AP cache” for future use. When a wireless station moves to a location visited before, it knows which channels have APs by checking the AP cache. Only those channels and channel 1, 6, and 11 will be scanned. But in the case of cache miss or incorrect cached information, the handoff latency is the same as that of the full scan handoff.

In [3], each AP, instead of the wireless stations, record the neighboring APs’ information in the “neighbor graph” data structure. Then the AP can inform wireless stations about which channels have neighboring APs. The wireless stations need to scan only those channels. Besides, wireless stations need not wait until *Max Channel Time* if all neighboring APs’ response frames have already arrived. To construct and maintain the neighbor graph, all APs have to upgrade their firmwares, which makes this scheme not easy to deploy because of the large number of existing 802.11b/g networks.

In [5] the authors propose a fast scan handoff scheme. Instead of broadcasting the probe request frame to all APs, wireless station might be interested in some specific APs. The probe request frame

[1] We assume that all APs use *open system* authentication, which is the default setting for most vendors’ APs. The authentication delay might be the major part of the handoff latency if more sophisicated authentication process is used.

is sent to a specific AP who will be the sole responder. The designated AP sends probe response frame after SIFS deferral. By this means, a wireless station waits for probe response for only a few microseconds (a SIFS duration). Because this scheme needs to change both the wireless stations and the APs, it cannot be deployed in the existing 802.11 networks.

2.2.2 Passive Scan Handoff

The second category of previously proposed handoff scheme strives to improve the performance of passive scan. The authors of [4] propose a MAC layer handoff called SyncScan. It assumes all APs working on the same channel are synchronized and broadcast beacon frames almost at the same time. Wireless station switches to the particular channel at the time when beacon frames are broadcasted. The actual time used to listen on each channel can be very short, because the wireless station exactly knows when the APs will announce themselves. But it may be difficult to synchronize the beacon frame broadcast of all APs in large scale wireless networks.

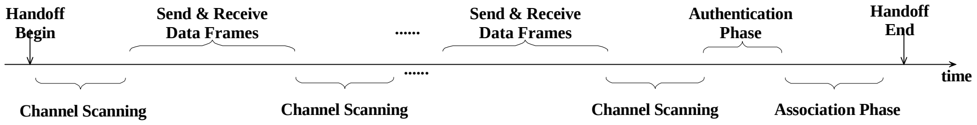


Figure 1: Operation of smooth MAC layer handoff

3 Smooth MAC Layer Handoff Schemes

Different from previous works, we mainly focus on scanning channels in a different manner, instead of scanning less channels.

3.1 Smooth Handoff

In existing active scan handoff schemes, the channels are scanned one after another without interuption. There is an several hundred milliseconds interruption during which the wireless station cannot send and receive any data frame.

In our smooth handoff scheme, the 11 channels are divided into *groups*. Instead of consecutively scanning all channels, after one group of channels are scanned, the wireless station takes a break from the channel scanning operation and switches back to normal data transmission mode. After working in the normal mode for some time, the wireless station switches to the channel scanning mode again and discovers APs working on another group of channels. After all channels are scanned, the wireless station knows all APs working on 11 channels. Then it can choose one AP with good signal quality to associate with. Figure 1 shows the operation of our smooth handoff scheme.

The total time it takes for scanning all channels is so long that the handoff is far from imperceptible to users. In our scheme, the scanning operation is divided into multiple subphases and there is sufficient time left for transmitting data frames between two subphases. Therefore, packet delay jitter will be smaller. Although the total time needed to scan all channels in a smooth handoff is the same as that of the full scan handoff, our scheme can still reduce packet loss during handoff in some cases. We will explain this issue in details at section 3.3.

Having multiple channel scanning subphases means a longer discovery phase before actually switching to another AP. So the discovery phase in our smooth handoff scheme should start earlier than that of the standard 802.11b/g handoff. In the 802.11b/g infrastructure network, there are usually overlapping areas between neighboring cells. Ideally, the discovery phase starts when wireless station enters an overlapping area and it should finish before the wireless station leaves the overlapping area. We limit the scenario of deploying our smooth handoff scheme in the wireless networks where there is enough overlapping area between two neighboring cells, e.g. the wireless networks deployed in the indoor environment. The wireless station has sufficient time to scan channels if it moves in a modest speed. For example, the ordinary walking speed of human being is about $1.5m/s$. If the discovery phase takes $2s$ to finish, the client moves about $3m$ ahead. Usually, the overlapping area between two 802.11b/g APs should be far more than $3m$. We will describe the AP deployment requirements in section 3.4.

In the standard handoff of the current 802.11b/g networks, the wireless station does not have to discover another AP before losing connection with its current AP, i.e. the RSSI of the current AP drops to a very low threshold. In our smooth handoff scheme, the wireless station needs to start the discovery phase earlier than the standard handoff scheme. In order to do so, we use a higher threshold to trigger the discovery phase, to make sure the wireless station has enough time to scan channels gracefully.

Using a high threshold (denoted by $Thres$) to trigger the discovery phase may lead to more scan operations when the wireless station moves to a place where all available APs' RSSI is lower than $Thres$, such as near the network periphery. To limit the frequency of scanning channels, we adopt an adaptive algorithm to dynamically change the threshold triggering the discovery phase.

Initially the wireless station uses a high threshold. It starts to probe for nearby APs when the current AP's RSSI is lower than the threshold $Thres$. After scanning all channels, it turns out that all available APs' RSSI is below the high threshold. Then the threshold is adjusted to a lower level (the threshold should be decreased by α). When the wireless station successfully finds an AP with good signal quality to associate with, the threshold should be increased by β, with the expectation to find a better signal quality AP. The threshold should be bounded by a maximum and a minimum value, denoted by $Thres_{max}$ and $Thres_{min}$. This is to prevent the wireless station from scanning channel too aggressively or not doing handoff even the current AP's signal quality is poor. When the threshold drops to $Thres_{min}$, the signal quality of current AP is too poor to make successful transmission. The wireless station needs to find another AP as soon as possible, so a full scan should be scheduled. Figure 2 shows the pseudocode of the smooth handoff algorithm.

3.2 Greedy Smooth Handoff

To shorten the discovery phase in handoff, we can scan channels both smoothly and selectively. That is, we adopt selective scan in our smooth handoff scheme. We call this handoff algorithm *Greedy Smooth Handoff*. In the discovery phase of the greedy smooth handoff, after the wireless station scans a group of channels and finds a suitable AP, it connects to that AP without scanning the remaining channels. If no suitable AP is discovered, the wireless station continues scanning the next group of channels. The threshold should also be adjusted as described in the previous section.

How to arrange channels into groups is the challenge here. Ideally, the channels that are likely to have working APs should be put into the first group to be scanned. Some apriori information is needed to properly arrange channels into groups. For example, the neighboring APs of an AP working on channel 1 may usually work on channel 6 or channel 11, considering that there are only 3 noninterference channels in 802.11b/g. Even if the wireless station randomly arranges the channels into groups, on average, only 6 channels are needed to be scanned before discovering a suitable AP.

Smooth Handoff Algorithm

```
while(true) {
    sample the $RSSI$ of current AP;
    if($RSSI < Thres$ ) {
        if($Thres > Thres_{min}$) {
            for(each group $i$) {
                scan channels in group $i$;
                sleep for some time;
            }
        } else {
            scan all channels consecutively;
        }
        choose AP with the best $RSSI$;
        if($RSSI_{new} > Thres_{min}$ &&
                ($RSSI_{new} - RSSI > \Delta$)) {
            associate with the chosen AP;
            $Thres = \min(Thres_{max}, Thres + \beta)$;
        } else {
            $Thres = \max(Thres_{min}, Thres - \alpha)$;
        }
    }
}
```

Figure 2: Smooth handoff algorithm

3.3 Benefits of Smooth Handoff Schemes

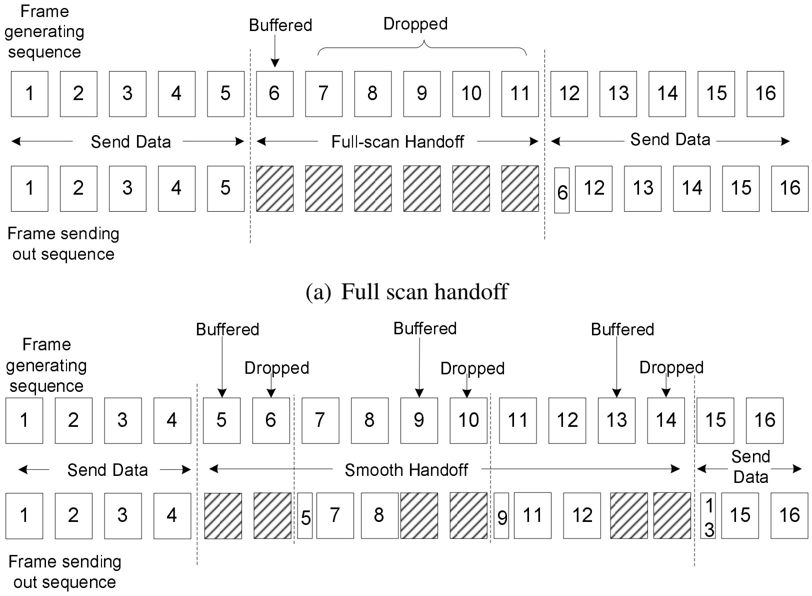
An obvious benefit of our smooth handoff schemes is that the packet delay and jitter during handoff will be smaller than that of the standard handoff, since the wireless station can still intermittently transmit packet during the discovery phase.

When a station sends out frames, the application layer writes the data into a buffer of the OS kernel and then the data are written to a buffer on the wireless card. The buffer in the OS kernel and the onboard buffer of the card can be modeled as one queue. Before the discovery phase starts, because the current AP's signal quality is too poor to have successful transmission or the transmission rate drops to a low level, the queue is almost full of backlog packets. Therefore, the available queue buffer is usually very small at the beginning of the discovery phase. In the standard handoff, the wireless card cannot send and receive any data frame for several hundred milliseconds. Packets generated by applications can overflow the queue, and therefore packet loss occurs.

Generally, packet loss during handoff depends on the available queue buffer size and packet generating rate of the applications. We consider only the client side here. If the available queue buffer size is large enough to cache all packets generated during handoff, there is no packet loss in handoff. In this case, our smooth handoffs can reduce delay jitter only. If the available queue buffer size is small, which is quite likely to be true at the beginning of the discovery phase, our smooth handoff has smaller packet loss than the full scan handoff does. Suppose P packets are generated during a full scan handoff and the available queue buffer size is C. The discovery phase is divided into N subphases in the smooth handoff. We use L and L' to denote the number of lost packets during full scan handoff and smooth handoff. When the queue buffer is too small to cache all the packets generated in each subphase, $N \times C$ packets are buffered and sent out later in the smooth handoff, while the full scan handoff can buffer only C packets. The smooth handoff has $(N-1)C$ less lost packets than the full scan handoff does. If the queue buffer can cache all packets in each subphase, there is no packet loss in smooth handoff, while $P - C$ packets are lost in full scan handoff. If the queue buffer can cache all packets during the full scan handoff, there is no packet loss in smooth handoff and full scan handoff. The improvement of the smooth handoff in terms of packet loss is shown in (1). Since the greedy smooth handoff scans less channels than the smooth handoff, it can be expected that the packet loss can be further reduced.

$$L - L' = \begin{cases} (N-1)C & (C \leq \frac{P}{N}) \\ P - C & (\frac{P}{N} < C \leq P) \\ 0 & (C > P) \end{cases} \tag{1}$$

A side-effect of more packets being buffered in the smooth handoff is that more packets will be delayed during handoff. We use D' and D to denote the number of delayed packets in smooth handoff and full scan handoff. It can be derived that $D' - D$ has the same expression as (1). Although more packets are delayed in the smooth handoff, the latency is much shorter than that of the full scan handoff. In general, the latency is proportional to the duration of a full scan (denoted by T seconds) in full scan handoff. Our smooth handoff can reduce the latency to $\frac{T}{N}$, since the wireless station has a chance to transmit packets every $\frac{T}{N}$ seconds during handoff.



(a) Full scan handoff

(b) Smooth handoff

Figure 3: How queue buffer affects packet loss and delay in handoff

Figure 3 shows an example of what happens during handoff. A box in Figure 3 represents a frame. For simplicity, we assume the queue buffer can cache one frame. When the queue is full, new arriving frames are droppped. We also assume frames are generated in a constant rate, i.e. one frame is generated every "time unit". We consider the discovery phase only since it is the most time consuming operation in handoff.

Figure 3(a) shows an example of full scan handoff. The discovery phase starts after the fifth frame is generated. For the 6 frames generated during handoff (frame $6 \sim 11$), only frame 6 is buffered and sent out when handoff completes. The handoff loses 5 frames and delays one frame by 5 time units. In the smooth handoff shown in Figure 3(b), the entire discovery phase is divided into 3 subphases and one subphase uses 2 time units. After frame 4 is sent out, the wireless station starts to scan a group of channels, which causes frame 5 to be buffered and frame 6 to be dropped. Then the wireless station stops scanning channels for 2 time units and the backlog frame 5 is sent out, followed by 2 newly generated frames. We should note that frame 7 and frame 8 have almost no delay, because frame 5 is immediately sent out when the wireless station stops scanning channel. The transmission delay can be ignored when the frame is small. The entire smooth handoff loses 3 frame and delays 3 frames by one time unit.

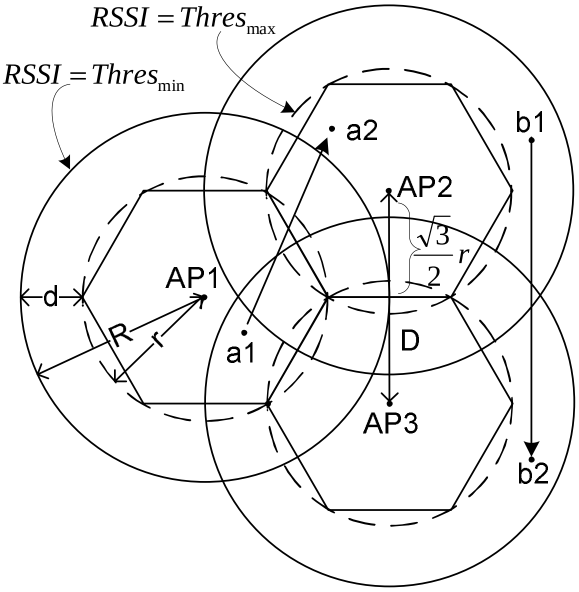


Figure 4: Overlapping areas between neighboring cells

3.4 Station Movement and AP Deployment Constraints

Each hexagon in Figure 4 represents the cell of one AP. The solid circle represents where AP's RSSI is $Thres_{min}$. The wireless station will disconnect with its current AP if moving out of the solid circle. $Thres_{min}$ is determined by the sensitivity of the radio module on wireless card. Once the $Thres_{min}$ is fixed, the solid circle's radius, R, is determined by the transmission power of AP and the propagation of radio signals. We can consider R as a constant once the network is deployed. The dashed circle's radius is r, representing where AP's RSSI is $Thres_{max}$. The relationship between AP's $RSSI$ and the distance to the AP L is: $RSSI = \frac{C}{L^k}$, where k is the factor contributing to the attenuation of signal propagation and C is related with the transmission power of the AP. We have equation (2) for r:

$$\begin{cases} Thres_{min} = \frac{C}{R^k} \\ Thres_{max} = \frac{C}{r^k} \end{cases} \Rightarrow r = R\sqrt[k]{\frac{Thres_{min}}{Thres_{max}}} \tag{2}$$

$$\begin{aligned} T_{max} \geq \; & GroupNum \times BreakTime + \\ & ChannelNum \times MaxChannelTime \end{aligned} \tag{3}$$

Let T_{max} denote the maximum duration of discovery phase. T_{max} depends on the number of channel scanning subphases, the duration of data transmission after each subphase, the number of channels, and the time needed to scan one channel, as shown in (3). Users may configure those parameters according to their requirements when deploying the networks. Intuitively, if $Thres_{max}$ is set to a higher value, the discovery phase will start earlier (r is smaller). Then V_{max} can be higher if T_{max} is a fixed value.

$$V_{max} < \frac{d}{T_{max}} = \frac{R-r}{T_{max}} \tag{4}$$

From (2) and (4) we can derive the limitation of wireless station's moving speed as(5).

$$V_{max} < \frac{R}{T_{max}} \left(1 - \sqrt[k]{\frac{Thres_{min}}{Thres_{max}}} \right) \tag{5}$$

Another constraint is that there should be an AP with better signal quality when the station finishes scanning channels. From Figure 4, we can see that in order to ensure the wireless station enters another AP's dashed circle when it moves out of its current AP's dashed circle, the three neighboring dashed circles should have at least one overlapping point. Let D denote the distance between neighboring APs, we can derive that $D \leq \sqrt{3}r$. After replacing r with (2), we have the following constraint on the distance between two neighboring APs:

$$D \leq \sqrt{3}R\sqrt[k]{\frac{Thres_{min}}{Thres_{max}}} \tag{6}$$

There are two kinds of movements that cause handoff. The first one is moving in the middle of the network, like from point a_1 to a_2 in Figure 4. In this case, the wireless station can always have enough time to smoothly scan channels and find another AP with good signal, if the moving speed and neighboring APs' distance are limited by (5) and (6) respectively. The second kind of movement is at the border of the network, where all nearby APs have poor signal quality, e.g., moving from point b_1 to b_2 in Figure 4. In this situation, the threshold $Thres$ should be adjusted to a low value (lower than $Thres_{max}$). Otherwise the wireless station will keep scanning channels because all nearby APs have RSSI lower than $Thres_{max}$.

4 Implementation

We have implemented full scan handoff, smooth handoff and greedy smooth handoff by software. The full scan handoff is used to emulate the handoff controlled by the wireless card firmware. The station is equipped with a 802.11b PCMCIA card using the Intersil Prism2 chipsets. The HostAP driver [10] is used to drive the wireless card.

The HostAP driver has a *manual scan and roam* mode, in which both channel scanning and AP selection are left to users. A 16 bits *Channel Mask* controls which channels should be scanned during handoff. We had some modifications to the HostAP driver so user can make an ioctl() system call to set the Channel Mask.

A user space daemon, *handoffd*, implements the function of handoff. It monitors the current AP's signal quality and starts the discovery phase when the current AP's RSSI is low. To avoid the "ping-pong" effect, the wireless station only attempts to associate with the new AP when condition

Table 1: Default values in implementation

Parameters	RSSI Value
$Thres_{max}$	15
$Thres_{min}$	5
α	5
β	2
Δ	2

(7) holds. This is similar to the scheme specified in [13]. After scanning channels, the *handoffd* daemon chooses one AP to associate with.

$$\begin{cases} RSSI_{newAP} \geq Thres \\ RSSI_{newAP} - RSSI_{oldAP} > \Delta \end{cases} \tag{7}$$

Table 1 shows the default value of parameters in the implementation. $Thres_{min} = 5$ and $\Delta = 2$ are the values used by the firmware of our wireless card. $Thres_{min}$ is the minimum RSSI value that the wireless card can transmit data frames. $Thres_{max} = 15$ is an empirical value derived from experiment. We will explain the reason for choosing this value in section 5.5.1. The α parameter is set to 5, therefore the threshold will be adjusted to the same value as the firmware-based handoff after two unsuccessful channel-scanning. We set β to be equal to Δ in our implementation, because the RSSI of the new AP is at least Δ higher than previous AP's RSSI after one successful handoff.

5 Experimental Evaluation

We use a testbed network and an existing network deployed in a three-floor building in our experiments. The later one is used to test our handoff schemes in a large network with various signal coverage conditions. We have tested the scenario that the wireless station moves to places where all nearby APs' signal quality is poor and the threshold adjusting function is activated.

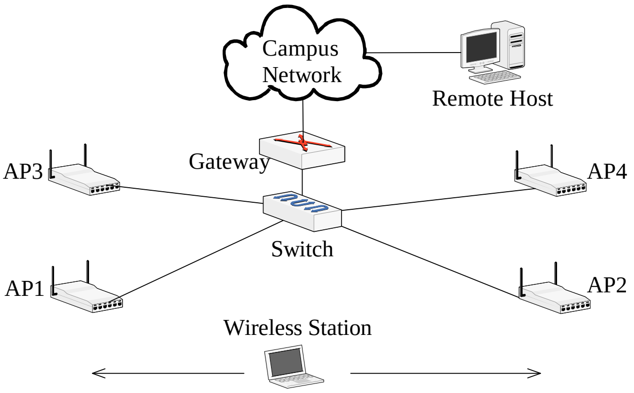


Figure 5: The testbed network

5.1 Testbed Setup

Figure 5 shows the testbed network. The two APs on the left side work on channel 6. The AP_2 and AP_4 on the right side work on channel 11. The left-side APs are far away from the right-side ones. The wireless station needs to do handoff when moving between them. Figure 6 shows the RSSI of AP_1 and AP_2 when the wireless station moves from left to the right. This network has the main feature of the cellular network infrastructure, i.e. the wireless station can always make a choice between two candidate APs after the discovery phase. *Open authentication* is used for layer-2 authentication. The *power saving* mode is disabled during the experiments. So the AP will not cache frames destined to the wireless station.

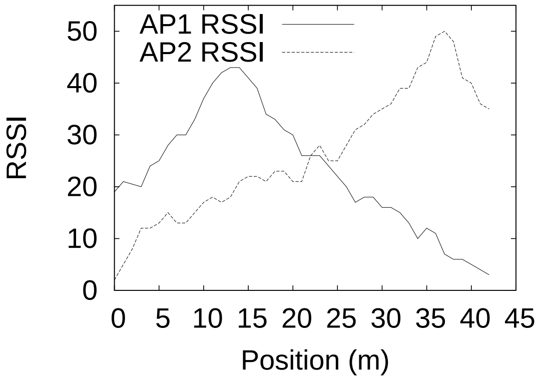


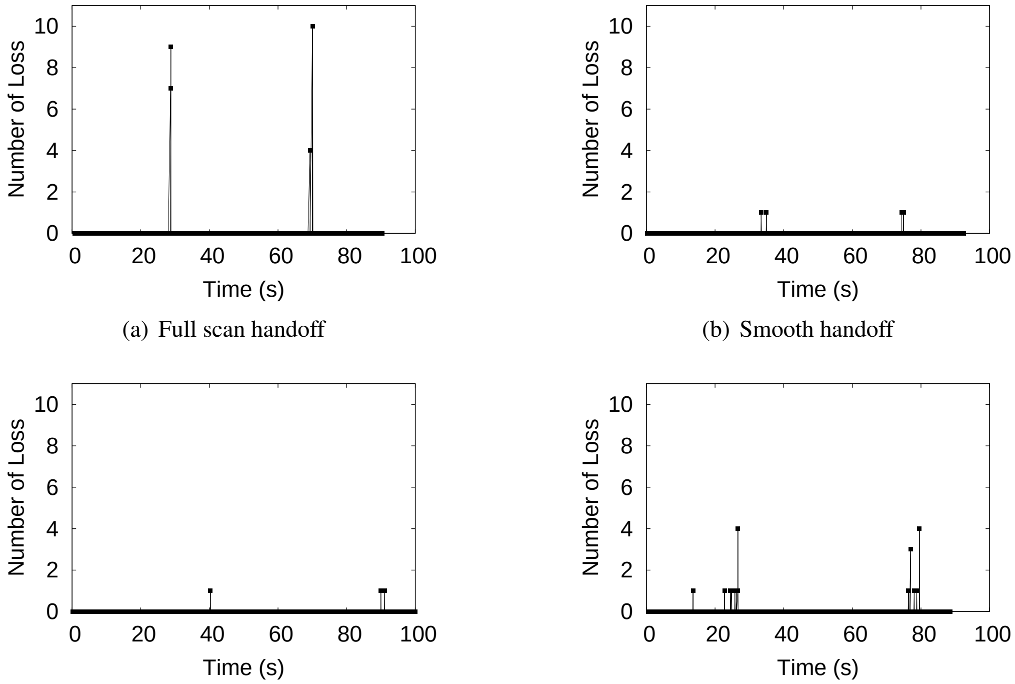
Figure 6: The RSSI disribution

All APs and the gateway machine are connected to a high speed ethernet switch. The gateway connects the testbed network with the campus network. A laptop equipped with a LinkSys WPC 11 PCMCIA card is the mobile station. The driver is our modified version of HostAP 0.2.5, compiled for Linux kernel 2.4.26.

We configure the group size to be 1 in our handoff schemes. In the following experiments, we do not deliberately choose the order in which the channels are scanned. Instead, our handoff schemes scan channels in the order of channel 1, channel 2, channel 3, and so on. This is the worst case scenario for the greedy smooth handoff scheme, since we do not use any apriori information to order the channels being scanned. Between two scan subphases, there is a $50ms$ interval left for the wireless station to send and receive data frames. Considering $MaxChannelTime$ is about $40ms$ and the switching channel overhead is a few milliseconds, it takes about $50ms$ to scan one channel. If we set group size to 1, the duration of the discovery phase in our smooth handoff is about $11 \times (50ms + 50ms) \approx 1.1s$.

5.2 Delay & Loss Measurement

The purpose of this experiment is to evaluate the performance of different handoff schemes in terms of packet loss and delay. The wireless station sends out an ICMP echo request message to the gateway machine every 50 milliseconds. When an echo reply message from the gateway arrives at the wireless station, the arrival time and the message sequence number are recorded. The packet delay is half of the round trip time. The gap between the sequence numbers of two successively arriving echo reply messages indicates the number of lost packets. We have conducted the experiments of full scan handoff, smooth handoff, and greedy smooth handoff. There are two handoffs in each experiment and we use the same threshold to trigger the discovery phase. We will first discuss one typical experiment instance for each handoff scheme. Then the average exprimental results are presented in Table 2.



(a) Full scan handoff

(b) Smooth handoff

(c) Greedy smooth handoff

(d) Firmware-based handoff

Figure 7: Packet loss in different handoff schemes (channel group size is 1 in smooth handoff and greed smooth handoff schemes)

The experimental results of the automatic handoff by the wireless card firmware are also presented, although it is unfair to compare the software-based handoff schemes with the firmware-based handoff, since the software implementation always has more overheads.

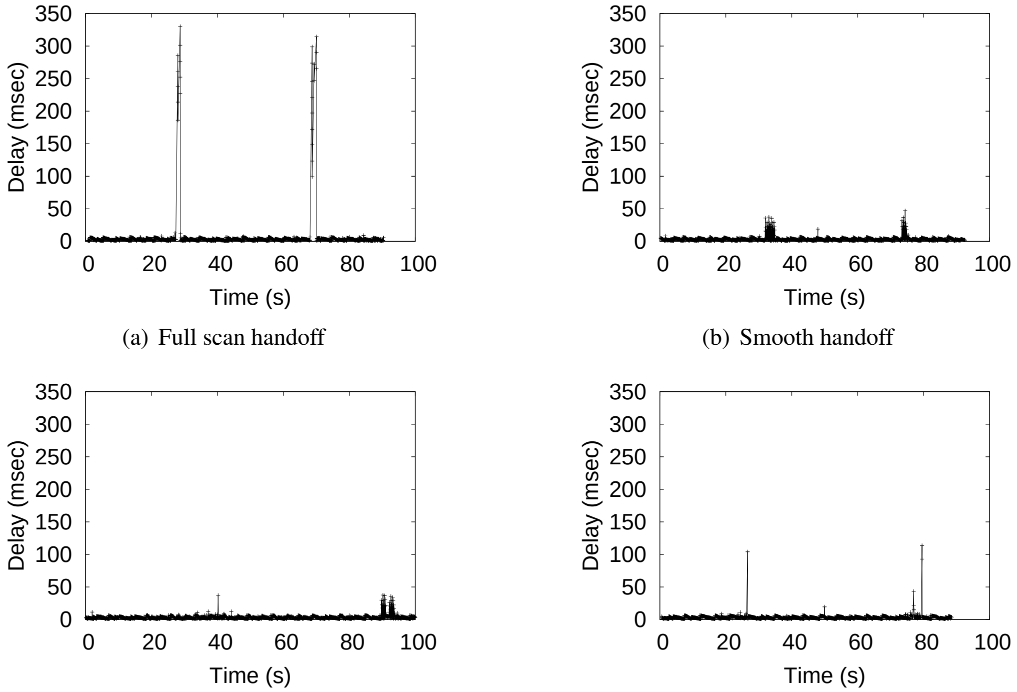
5.2.1 Full Scan Handoff

In this experiment, the *handoffd* daemon samples the current AP's RSSI. If the RSSI is lower than threshold $Thres$, the wireless station uses a consecutive full scan to discover all nearby working APs.

Figure 7(a) and Figure 8(a) plot the packet loss and delay. We can see that the loss and delay demonstrate a burst characteristic during handoff. This is because the wireless card scans all channels without interuption, it cannot send or receive any data frame for a long time. During handoff, the maximum delay can be $350ms$ and there are 26 delayed packets during the two handoffs. Forty packets are lost and all of them are consecutive losses. The maximum consecutive loss is 10 packets. When there is no handoff, most packets have delay less than $5ms$ and no packet loss occurs.

5.2.2 Smooth Handoff

In our smooth handoff, a high threshold is used to proactively start the discovery phase. The thresholds in full scan handoff and smooth handoff are set to the same value ($Thres_{max} = 15$), in order to compare them in the same scenario. We scan one channel in each subphase to fully demonstrate the feature of our smooth handoff.



(a) Full scan handoff

(b) Smooth handoff

(c) Greedy smooth handoff

(d) Firmware-based handoff

Figure 8: Packet delay in different handoff schemes (channel group size is 1 in smooth handoff and greed smooth handoff schemes)

Figure 7(b) and 8(b) plot the results in this experiment. Both packet loss and delay are less than those in the full scan handoff experiment. There are only 4 packets lost and the maximum latency is less than $40ms$. Thirty-five packets are delayed during two handoffs. Our smooth handoff has more delayed packets than the full scan handoff, which is consistent with our analysis in Section 3.3.

Table 2: Average results of different handoff experiments

Scheme	Total Lost Packet	Max Burst Packet Loss	Max Delay
Full Scan	50.3	15.2	384.4ms
Smooth	6.2	2.8	48.1ms
Greedy	4.9	1.5	33.8ms
Firmware	24.6	6.2	102.2ms

5.2.3 Greedy Smooth Handoff

Figure 7(c) and 8(c) show the results of one experiment instance. As the greedy smooth handoff scans less channels than the smooth handoff, there are 30 delayed packets, which is less than that in smooth handoff experiment. The number of lost packets here is 3 and the maximum delay observed in this experiment is about $40ms$. There are less delayed packets during the handoff at $40s$ than the handoff at $90s$ in Figure 8(c). That is because one access point in this experiment works on channel

6 and the other one works on channel 11. Only 6 channels are scanned during the first handoff. For the second handoff, all 11 channels are scanned before the station discovers the AP working on channel 11.

5.2.4 Firmware-based Handoff

Firmware-based handoff is the original handoff controlled by the wireless card firmware. It can be deemed as the hardware implementation of the full scan handoff algorithm. Figure 7(d) and 8(d) plot the results of one experiment. The packet delay can be more than $100ms$ when handoff occurs. There are 21 packets lost and 9 packets are delayed during the two handoffs. We can also see that there are 6 individual losses before a consecutive 4 packets loss in the first handoff. The 6 individual lost packets represent that the current AP's signal quality has degraded, but the wireless station still connects to that AP. The 4 consecutive lost packets represent the period when the wireless station is doing handoff.

5.2.5 Comparison of Handoff Schemes

The above experiments have been repeated for 10 times and Table 2 shows the average results. We can see from Table 2 that the smooth handoff and greedy smooth handoff can significantly reduce the accumulative packet loss and the consecutive packet loss. The full scan handoff is used to emulate the firmware-based handoff. They both probe channels continuously, so packet delay and loss demonstrate the same pattern. The firmware-based scheme has better performance than the full scan handoff because hardware solution has less overhead than software solution. Therefore, it can be expected that if our handoff schemes are implemented by hardware, the packet loss and delay can be further reduced.

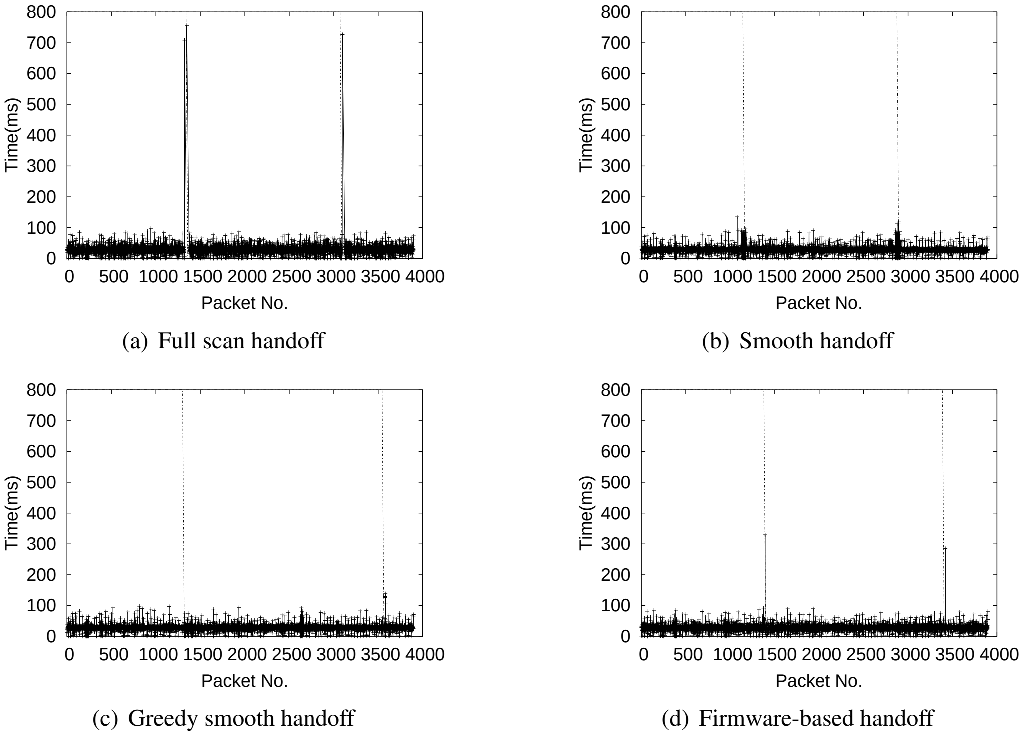
Table 3: Packet losses during handoff in voice traffice experiment

Handoff	Full Scan	Smooth	Greedy	Firmware
losses	30	3	2	9

5.3 VoIP Traffic Experiment

This experiment is to evaluate performance of our handoff schemes for audio stream traffic. We use sniffer tools to capture all the UDP packets from the the speaker to the listener in a 100 seconds VoIP session generated by the NetMeeting software. We replay the captured packets in this experiment when using four handoff schemes alternately. A wired machine connected to the campus network is the receiver, as shown in Figure 5. We consider the situation that the wireless station is a speaker and the wired station is a listener. The wired station records the interarrival time of all UDP packets. The packet interarrival time in each experiment is plotted in Figure 9. The vertical dashed lines in those figures indicate when handoffs occur. The average number of packet losses is shown in Table 3.

The maximum packet interarrival time in the full scan handoff experiment is about $700ms$. That is to say the handoff causes about one second silence period in the VoIP session. While Figure 9(b) and Figure 9(c) show that the smooth handoff and the greedy handoff do not cause much additional fluctuation to the packet interarrival time. Only 2 or 3 packets are lost during handoff, as we can see from Table 3. Figure 9(d) plots packet interarrival time in the firmware-based handoff. We can see that it shows the same pattern as the full scan handoff experiment.



(a) Full scan handoff

(b) Smooth handoff

(c) Greedy smooth handoff

(d) Firmware-based handoff

Figure 9: Packet interarrival time in VoIP experiment for different handoff schemes (channel group size is 1 in smooth handoff and greed smooth handoff)

5.4 TCP Transmission Experiment

We use the same network configuration as shown in Figure 5. The remote station is a FTP server connected on campus network. The wireless station downloads or uploads a $35M$ binary file from the FTP server. In each experiment there are two handoffs.

Figure 10 shows the normalized number of packets received by the wireless station as a function of time in the FTP download experiment. We can see that both the full scan handoff and the firmware-based handoff have considerable impact on the goodput. For the full scan handoff experiment, almost no packet arrives at the wireless station from $18s$ to $20s$, since during that time the wireless station is doing handoff. The corresponding period in the firmware-based handoff is from $38s$ to $40s$. For the firmware-based handoff, the period from $25s$ to $38s$ represents the AP "cling" effort, i.e. the wireless station does not try to associate with another AP even the current AP's signal quality is poor and no packet can be transmitted. As we can see from Figure 10, the curves for smooth handoff and greedy smooth handoff experiments are more closer to linear functions. The handoffs cause the FTP download speed dropping for only a short time. Figure 11 plots the normalized number of packets sent out by the wireless station in the FTP upload experiments. Both figures show that our handoff schemes have less impact on the TCP-based FTP application.

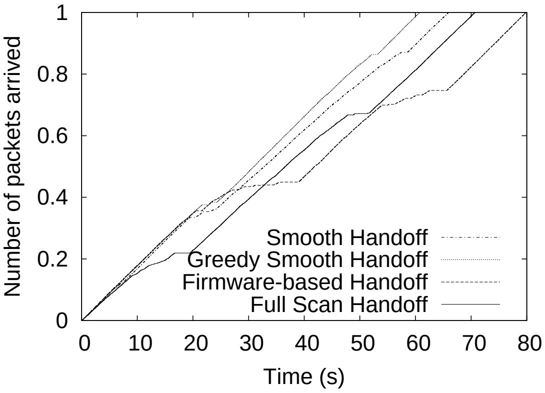


Figure 10: Download experiment

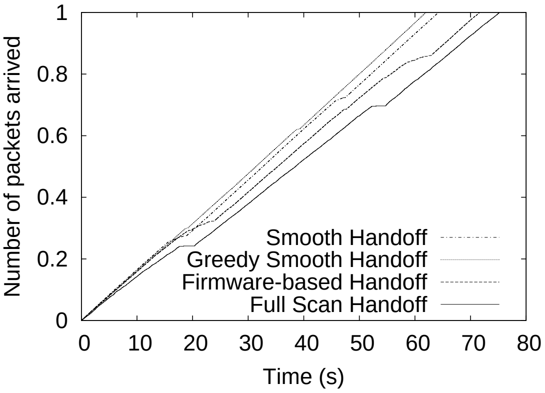


Figure 11: Upload experiment

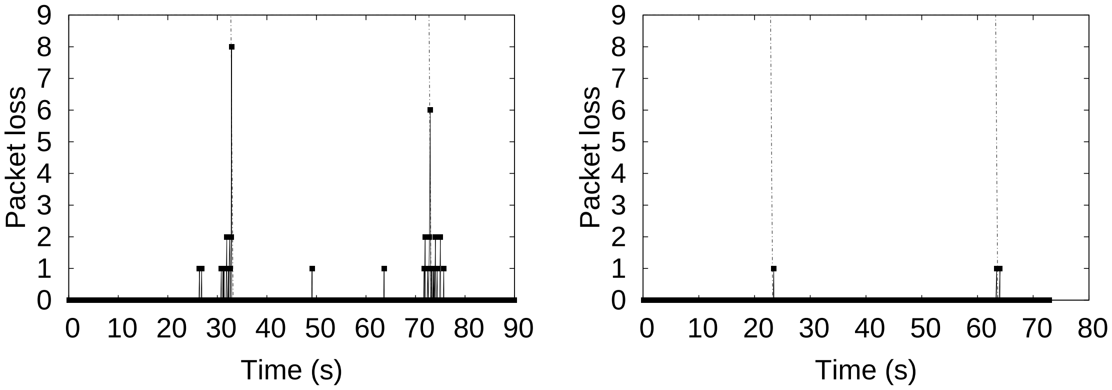
5.5 Effect of Algorithm Parameters

5.5.1 Thresholds Triggering Discovery Phase

To proactively start the discovery phase in our handoff schemes, a high RSSI threshold is used to trigger the discovery phase. This experiment is to demonstrate how the $Thres_{max}$ parameter affects the performance of our handoff schemes.

The wireless station generates ICMP echo request messages to the gateway machine and records the reflected echo reply messages. The packet generating rate is one packet every $50ms$. We set $Thres_{max}$ to four different RSSI value, 5, 10, 15, and 20.

Figure 12(a) plots the packet losses when using $RSSI = 5$ as the $Thres_{max}$. We can see that more packet are lost when using a low threshold, because even the wireless station has the chance to transmit data frames between two channel scanning operations, the signal quality of its current AP is too poor to make successful transmissions. As shown in Figure 12(b), there can be only one or two losses during each handoff when using $RSSI = 15$ as the threshold. Our experiment shows that using threshold higher than 15 does not result in significant less packet losses and delay.



(a) $Thres = 5$

(b) $Thres = 15$

Figure 12: Packet loss in different threshold experiments

We have conducted this experiment for 10 times. Table 4 shows the average number of packet losses in handoff when setting $Thres_{max}$ to different values. The threshold can greatly affect the performance of our handoff schemes. In order to have the full advantage of our handoff schemes, we should use a high threshold to make sure that the discovery phase is proactively started.

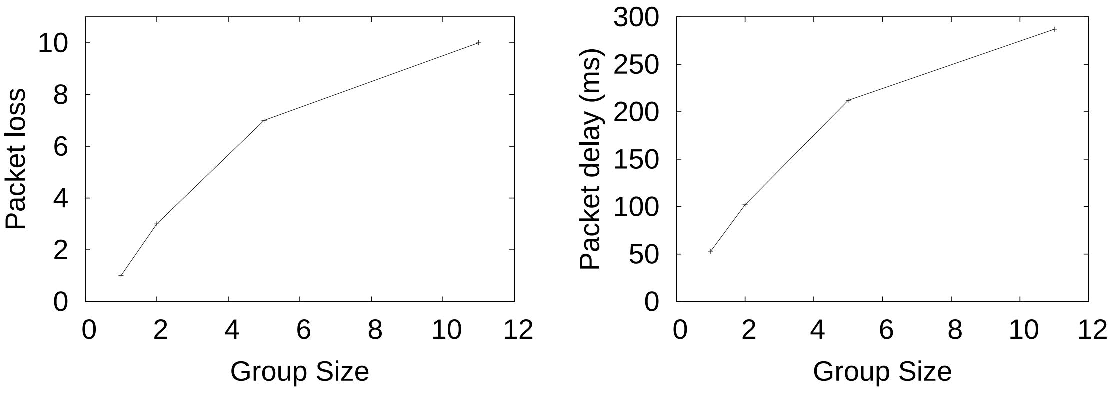
Table 4: Average packet losses in different thresholds experiments

Handoff Scheme	$Thres$ = 5	$Thres$ = 10	$Thres$ =15	$Thres$ = 20
Smooth	20.5	4.8	3.5	3.1
Greedy	16.3	3.2	2.3	1.8

5.5.2 Channel Group Size

As discussed in Section 3.3, both packet delay and loss depends the number of channel scanning subphases, which is determined by the channel group size. This experiment is to demonstrate the impact of group size to the performance of our handoff schemes.

The wireless station still sends out ICMP echo request messages to the gateway machine and records the echo reply messages. We measure the packet loss and delay when the group size is set to 1, 2, 5, and 11. The results are plotted in Figure 13(a) and Figure 13(b). We can see that the larger the group size, the more packets will be lost and the longer the packet delay will be, which is consistent to what we have discussed before.



(a) loss

(b) delay

Figure 13: Packet loss and delay in different group size experiments

5.6 Accommodating Mobility in Large Area

Besides the testbed network, we have also conducted experiments in a large wireless network, where some areas are not well covered by APs. When the wireless station moves to those areas, the adaptive mechanism adjusts the threshold to a low value to prevent the station from keeping scanning channels. The wireless network used in this experiment is deployed in a three-floor building. There are 12 APs in this network, four on each floor of the building. The four APs in the same floor are installed in the ceiling of a sixty-meter corridor, with two on the east side and two on the west side. Figure 14 shows the network topology and the moving trace of the wireless station.

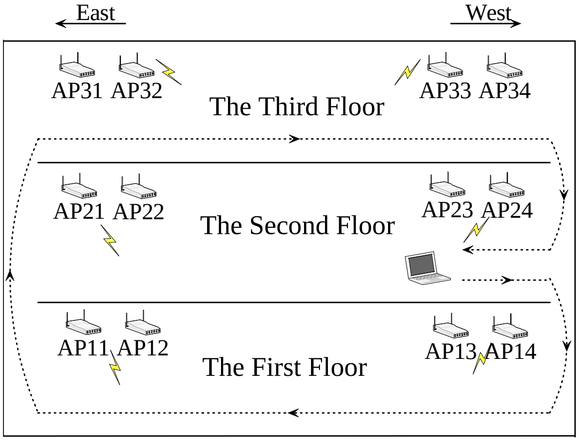


Figure 14: Network topology and the moving trace

Figure 15 shows the packet loss when wireless station uses the original firmware-based handoff. In this experiment, packet loss burst happens 7 times, which indicates 7 handoffs. Figure 16(a) plots the packet loss in the smooth handoff experiment when the threshold adjustment is disabled by setting $\alpha = \beta = 0$. From this figure we see that the first handoff, the fourth one, and the sixth one have more packet losses. That is because wireless station has moved to the stairwells in the east and west sides of the building, where are not well covered by the wireless network. In those areas, since all nearby APs have RSSI value less than the high threshold, the wireless station had multiple channel scanning operations before finding an AP with good signal quality. Those scan operations can cause additional packet losses. Figure 16(b) shows the packet loss in a smooth handoff experiment with the threshold adjustment. The threshold adjusting mechanism changes the threshold to a low value when all nearby APs have poor signal quality. Therefore, even in those areas which are not so well covered by APs, our smooth handoff scheme does not cause more packet losses comparing with the handoff occurring in the well covered areas.

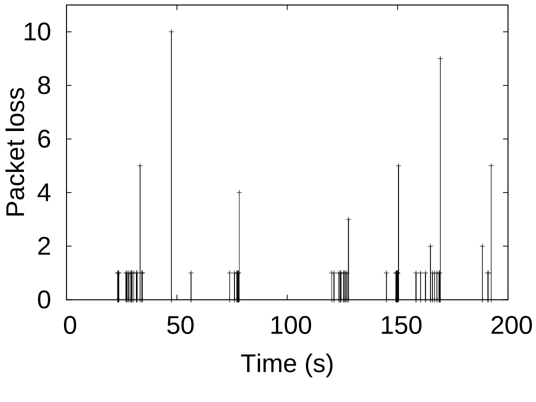
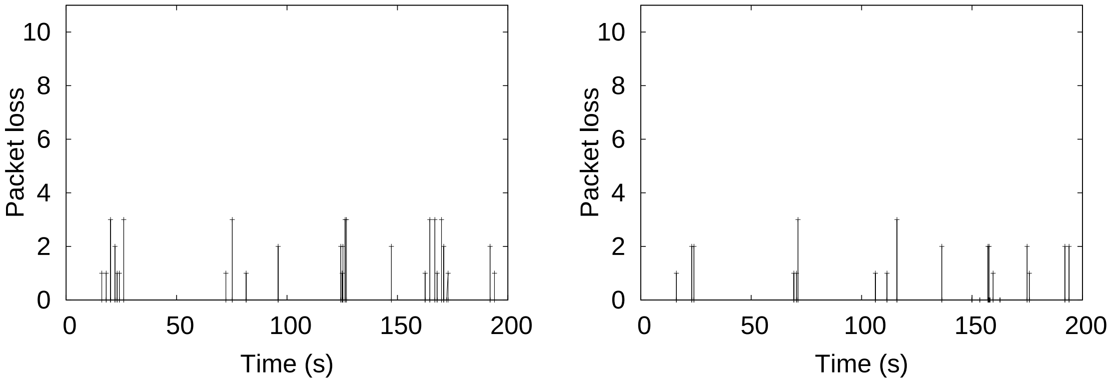


Figure 15: Packet loss in firmware handoff



(a) without threshold adjustment

(b) with threshold adjustment

Figure 16: Comparison of packet losses when adjusting the threshold or not

6 Conclusions

In this paper we propose a smooth MAC layer handoff scheme and a greedy smooth MAC layer handoff scheme. Our handoff schemes split the time for discovering APs into multiple subphases. So the station can use the interval between subphases to send and receive data frames. To prevent the station from keeping scanning channels when it moves within the areas which are not well covered by APs, an adaptive algorithm is adopted to dynamically adjust the threshold triggering the handoff. Our handoff schemes are staightforward and easy to deploy. They do not require any change to the AP side. We have implemented our handoff schemes using commodity 802.11 devices. Experimental results show that our handoff schemes have considerable improvements in terms of packet delay and loss. They can effectively make the handoff imperceptible to applications.

References

[1] IEEE Computer Society LAN MAN standards Committee, "IEEE Standard for Information Technology: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," 1999.

[2] S. Shin, A. Forte, A. Rawat, and H. Schulzrinne, "Reducing MAC Layer Handoff Latency in IEEE 802.11 Wireless LANs," in *Proceedings of the second international workshop on Mobility management and wireless access protocols,* Philadelphia, PA, USA, 2004.

[3] M. Shin, A. Mishra, W. A. Arbaugh, "Improving the Latency of 802.11 Hand-offs using Neighbor Graphs," in *Processings of the ACM MobiSys Conference,* Boston, MA, USA, June 2004.

[4] I. Ramani, and S. Savage, "SyncScan: Practical Fast Handoff for 802.11 Infrastructure Networks," in *Proceedings of the IEEE Infocom Conference 2005,* Miami, FL, March 2005.

[5] Moo Ryong Jeong, Fujio Watanabe, Toshiro Kawahara, "Fast Active Scan for Measurement and Handoff,", DoCoMo USA Labs, Contribution to IEEE 802, May 2003.

[6] A. Mishra, M. Shin, and W. Arbaugh, "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Proces," *ACM Computer Communications Review,* vol. 33, no. 2, Apr. 2003.

[7] H. Velayos and G. Karlsson, "Techniques to Reduce IEEE 802.11b MAC Layer Handover Time," KunglTekniska Hogskolen, Stockholm, Sweden, Tech. Rep. TRITA-IMIT-LCN R 03:02, ISSN 1651-7717, ISRN KTH/IMIT/LCN/R-03/02.SE, April 2003.

[8] International Telecommunication Union. General Characteristics of International Telephone Connections and International Telphone Circuits. ITU-TG.114, 1998.

[9] Jean Tourrilhes, "Wireless LAN resources for Linux," `http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Wireless.html`

[10] Host AP driver for Intersil Prism2/2.5/3 and WPA Supplicant. `http://hostap.epitest.fi/`

[11] "Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation", IEEE Std 802.11F-2003, June 2003.

[12] "Medium Access Control (MAC) Security Enhancements," IEEE Std 802.11i-2004, July 2004.

[13] Lucent Technologies Inc., "Roaming with WaveLAN/IEEE 802.11," Technical Report WaveLAN Technical Bulletin 003/A, November 1998.

[14] R. Shirodkar, J. Kabara, and P. Krishnamurthy, "A QoS-based Indoor Wireless Data Network Design for VoIP Applications," in *proceedings of IEEE International Vehicular Technology Conference*, Atlantic City, NJ, Fall 2001.

[15] N. Jordan, R. Fleck, C. Ploninger, "Fast Handover Support in Wireless LAN based Networks," In *Proceeding of the Fifth IFIP-TC6 International Conference on Mobile and Wireless Communications Networks (MWCN'03)* p.: 49-52, ISBN:981-238-686-6, Singapore, October 2003.

[16] A. Weyland, G. Stattenberger, and T. Braun, User-Controlled Handover in Wirless LANs, *IEEE Workshop on Applications and Services in Wireless Networks 2002 (ASWN 2002)*, Paris, France, July 3-5, 2002.