Global Path-Cache Technique for Fast Handoffs in WLANs

Weetit Wanalertlak and Ben Lee
School of Electrical Engineering and Computer Science
Oregon State University
Corvallis, OR 97331, USA
{wanalewe, benl}@eecs.orst.edu

***Abstract*—This paper proposes a technique called Global Path-Cache (GPC) that provides fast handoffs in WLANs. GPC maintains a history of mobile stations' mobility patterns in a network to assist in the prediction of the next point-of-attachment. GPC properly captures the dynamic behavior of the network and mobile stations, and provides accurate next AP predictions. Our simulation study shows that GPC virtually eliminates the need to scan for APs during handoffs and results in much better overall handoff delay compared to existing methods.**

Keywords-WLANs, mobility, handoffs, scanning, mobile stations, access points.

I. INTRODUCTION

IEEE 802.11 wireless local area networks (WLANs) have become pervasive in our society. WLANs offer high data transfer rate that allows portable devices such as laptops, PDAs, and smart phones to not only connect to the Internet but also transfer real-time multimedia data, such as streaming audio and video. Moreover, smart phones in the near future will be able to automatically turn off its connection to a GSM or CDMA network and register to a lower cost Voice over IP (VoIP) service over a WLAN [1]. In fact, as more and more WLAN access points (APs) are installed in public and commercial areas, they will soon cover majority downtown areas in many cities. One of the greatest benefits of WLAN is mobility, which allows a user to continually talk on a VoIP application or watch a video stream while walking between city blocks or riding a downtown bus. However, mobility incurs large handoff delays when mobile stations (MSs) switch connections from one AP to another. In a crowded network, such as office or university campus environments, APs are installed close together. This causes frequent handoffs that make the problem more severe. Long handoff delay is undesirable and yet a recent study found that the handoff delay in WLANs can be much as 1.1~1.9 seconds [2]. This becomes a major concern for mobile multimedia applications, such as VoIP, where the end-to-end delay is recommended to be not more than 50 ms [3].

A number of techniques have been proposed to reduce handoff delay [4, 5, 6, 7]. These techniques focus on optimizing the probing process, since the probing delay represents more than 90% of overall handoff delay. This is achieved by using extra hardware, either in the form of additional radios [6] or an overlay sensor network [7], to detect APs, limiting the number of channels to be probed by predefining the topological placement of APs [4], and predicting the next point-of-attachment based on signal strength [5]. Unfortunately, these techniques do not consider the dynamic nature of WLANs and mobility patterns of MSs.

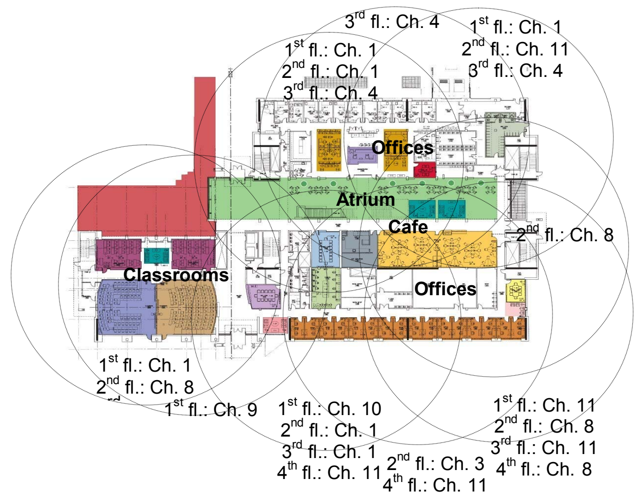

Figure 1. WLAN coverage for the KEC building.

In order to illustrate the problems with existing methods, Fig. 1 shows the coverage area for the first floor of the new four-story, 153,000-ft^2 Kelley Engineering Center (KEC), which is the home of the School of Electrical Engineering and Computer Science at Oregon State University. The figure shows that the coverage area characteristics of a typical WLAN deployment such as ours can be rather ad hoc. First, APs are installed in relative close proximity to users, i.e., offices and classrooms. Thus, the topological placement of APs does not follow an ideal hexagonal cell layout. Second, some cells are highly overlapped to provide high bandwidth for MSs in high traffic areas, and adjacent cells do not necessarily use non-overlapped channels 1, 6, and 11. In addition, the signals transmitted from APs are not limited to just a single floor but extend omni-directionally beyond the ceilings, floors and walls. Therefore, a MS on the 1st floor can discover signals from APs on the 2nd, 3rd, and 4th floors. Fourth, the operating environment of WLANs changes frequently and drastically due to multipath effects, user mobility, and electromagnetic interference. Therefore, the quality of signals from APs cannot be guaranteed over time. Most importantly, mobility patterns of MSs are not random but rather dictated by the structure of a building.

For example, users will travel along hallways but not across walls. Therefore, handoff patterns tend to repeat as MSs travel from one coverage area to another.

This paper proposes a new solution called the *Global Path-Cache* (*GPC*), which uses past history of mobility patterns to assist in the predictions of next point-of-attachments. The proposed GPC handoff technique has two major advantages over existing methods. First, GPC not only maintains the handoff history of the current MS but also the handoff history of other MSs. This eliminates the need to 'learn' new handoff sequences. Second, GPC uses past *N* handoffs, which keeps track of MS's direction of movement relative to the topological placement of APs, to determine the next point-of-attachment. Moreover, GPC predicts the next AP to reassociate based on the frequency of occurrences rather than signal strength. Therefore, when GPC fails to properly predict the next AP, it is recalibrated for future predictions. The proposed technique is an adaptive algorithm, which is independent of the topological placement of APs and the number of channel frequencies used. It minimizes the need to perform scanning for available APs and results in faster handoffs to meet the requirements of mobile multimedia applications. Our simulation study shows that GPC results in superior average handoff delay compared with Selective Scan with Caching [5] and Neighbor Graph [4].

The paper is organized as follows. Section II presents the related work. Section III discusses the proposed GPC technique. Section IV compares and evaluates the performance of the proposed method and compares it with the existing solutions. Finally, Section V concludes the paper and discusses possible future work.

II. Related Work

There has been a lot of work done to reduce the handoff delay in WLANs. This section focuses on optimizing the probing or scanning process, which is the most time consuming part of a handoff [2, 8].

MultiScan uses multiple WLAN network interfaces to opportunistically scan and pre-associate with alternative APs to avoid disconnections [6]. The basic idea is to have the first WLAN interface communicate with the current AP while the second WLAN interface scans for new APs. This scan information is then used to connect to the new AP before the connection is lost from the current AP. *Selective Active Scanning* uses an overlay senor network to obtain information on the neighboring APs [7]. This technique requires sensor nodes, which operate at a different frequency than the WLAN, to be uniformly placed over the network coverage areas. Although both techniques can provide fast handoffs, they require extra hardware, implemented either on the client side or as a separate control plane, which may be impractical and/or power inefficient.

Another technique, called *SyncScan* [9], requires APs to send staggered periodic beacons, which allows MSs to scan for additional APs while it is still connected to the current AP. Although the overall handoff delay can be reduced, there is a hidden cost since a MS has to periodically suspend its communication to listen for other APs. Moreover, the proposed method is an orthogonal approach to SyncScan and thus they can be deployed together to reduce the cost of performing a full scan.

The works closest to ours are *Neighbor Graph* [4, 10] and *Selective Scan with Caching* [5]. The Neighbor Graph technique reduces the number of channels to scan by defining a directed graph that represents the topological placement of APs and the mobility patterns of MSs. Moreover, edges between APs, which represent handoffs, are added or deleted to reflect the changing conditions. Even though the Neighbor Graph technique significantly reduces the average number of channels probed, it does not provide next point-of-attachment predictions and thus all edges (i.e., adjacent channels) emanating from a node needs to be scanned.

Selective Scan with Caching minimizes the need to probe during a handoff by predicting next point-of-attachment based on signal strength. A MS joining the network for the first time performs a full scan. Then, the corresponding bits in the channel mask are set for all the probe responses received from APs, as well as bits for channels 1, 6, and 11 with the premise that these channels are more likely be used by APs. As MS connects to the AP with the strongest signal, the corresponding bit in the channel mask is reset based on the assumption that the likelihood of adjacent APs having the same channel is very small. In addition, the addresses of two APs representing the second and third strongest signals are stored in the *AP-cache* using the current AP's address as the key. These two APs represent the best and second best candidates for subsequent handoffs. During the next handoff, MS will attempt to reassociate with these two APs in order. If MS fails to reassociate with both APs or an entry is not found in the AP-cache, a selective scan is performed based on the channel mask to choose two APs with the strongest signals and stores them in the AP-cache. If no APs are discovered with the current channel mask, bits in the channel mask are inverted and another scan is performed. If the partial scan fails to discover APs, a full scan is performed. However, in order to use the information from the last scanning period for the current handoff, the direction of MS movement relative to the cell layout must be identical to the one in the last handoff. This is often not the case and thus the AP-cache will frequently fail to provide correct Next-AP predictions.

III. The Proposed GPC Technique

The basic idea behind GPC is to track past mobility patterns and then use this information to predict future handoffs. This virtually eliminates the need to scan channels when MSs move through the same APs' coverage area. In order to illustrate the motivation behind GPC, Fig. 2 shows an example of a coverage area that contains three APs. As MS moves from AP_y to AP_x and back to AP_y, it is unclear which AP it will attach to next since there are two possible candidates (i.e., AP_x or AP_z). Therefore, the history of handoff sequences is maintained and used to predict behavior of future handoffs.

In order to keep track of a MS's handoff sequence, a *local history* is maintained using an *N*-entry *Handoff-Sequence Window* (HSW) containing information of the current AP as well as *N*-1 past APs (i.e., the MAC address and the channel number). Fig. 2 illustrates HSW for *N*=3. A MS joining the net-

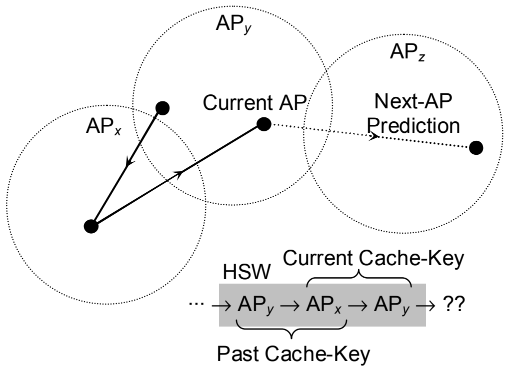

Figure 2. Local history using HSW for N=3.

work for the first time has no local history and thus its HSW contains null entries. When MS reassociates with a cell, the information of the current AP is queued in HSW. During each subsequent handoff, MS sends to the server a *path-cache request* containing HSW as part of an authentication request.

When the server receives path-cache requests from MSs, a *global history* of all the MSs in the network is maintained in the *Path-Cache*, where each entry contains a *Cache-Key* represented by *Current-AP* and *N*-2 *Past-APs*, *Next-AP*, and a *Counter* indicating the number of hits on this entry. An example content of the Path-Cache for Fig. 2 is shown in Table I.

The following operations are performed when the server receives a path-cache request from a MS:

- *Path-Cache update* - the *past cache-key* represented by the handoff sequence $\{AP_0, AP_1, \ldots, AP_{N-2}\}$ in HSW is used to search in the Path-Cache for a matching Cache-Key. If a match is found, a check is made to see if AP_{N-1} also matches the Next-AP entry. If it matches, the counter for that entry is incremented by one. If a match is not found, it means the HSW is new. Therefore, the new handoff sequence is stored in the Path-Cache and its counter is initialized to one.
- *Next-AP prediction* - the *current cache-key* represented by the handoff sequence $\{AP_1, AP_2, \ldots, AP_{N-1}\}$ in HSW is used to search in the Path-Cache for a matching Cache-Key. If a null HSW is received, it indicates the MS is joining the network for the first time. Therefore, a special HSW $\{null_1, null_2, \ldots, AP_{tuned\text{-}in}\}$, where $AP_{tuned\text{-}in}$ represents the current AP the MS is tuned into, is used to search in the Path-Cache. If a match or multiple matches are found, a *path-cache response* containing a list of Next-AP predictions sorted in descending order of their counter values is sent to MS as part of an authentication response. Otherwise, a null Next-AP prediction is sent back to notify of Path-Cache miss.

Note that the size of *N* depends on the complexity of the network topology and the building structure. If the coverage area is small and yet there are many APs, a longer handoff history will be preferred. However, our study shows that in general N = 2 or 3 is sufficient to provide a good Next-AP prediction. In addition, all the Path-Cache entry counters are periodically decremented to prevent saturation.

TABLE I. CONTENT OF THE PATH-CACHE FOR FIG. 2.

Cache-Key		Next-AP	Counter
Past-AP	Current-AP		
AP_x	AP_y	AP_z	6
AP_x	AP_y	AP_x	2
AP_y	AP_x	AP_y	1
AP_y	AP_z	AP_y	7
AP_z	AP_y	AP_x	3
AP_z	AP_y	AP_z	8

The algorithm for the GPC technique is described below based on the assumption that the Next-AP predictions for the current handoff have been determined from the previous handoff:

1. MS directly tunes into the AP provided by the Next-AP prediction. If Next-AP prediction is null, MS performs a full-scan and tunes into the AP with the strongest signal.
2. MS sends authentication request containing path-cache request to the server to obtain Next-AP predictions for the next handoff.
3. If authentication is successful, the server performs Path-Cache Update and Next-AP Prediction based on the received HSW. Otherwise, choose the next element in the Next-AP prediction list and go to Step 1.
4. MS and AP exchange reassociation request/response. If no reassociation response is received, choose the next element in the Next-AP prediction list and go to Step 1.
5. Information of the new AP is queued in HSW.

If a path-cache request hits on the Path-Cache and its 1st Next-AP prediction is successful, GPC will reduce overall handoff delay down to only the time required for MS to perform a channel switch plus authentication and reassociation. With each additional Next-AP misprediction, the overall handoff delay increases incrementally by the channel switching time plus authentication timeout period. For example, if the 1st Next-AP prediction fails but the 2nd Next-AP prediction is successful, MS first tunes into the first predicted Next-AP and waits until the authentication times out, then tunes into the second predicted Next-AP.

In case of a Next-AP misprediction, or authentication failure, MS will revert back to the conventional handoff, which requires a full scan. A Path-Cache miss will occur if a handoff sequence is encountered for first time. Afterwards, the Path-Cache will record the new sequence and use it for future handoffs. Therefore, as long as the Path-Cache is current, all MSs can benefit from this information to provide fast handoffs. Finally, note that path-cache request/response is piggy-backed on authentication request/response for the current AP. Therefore, no extra messages are needed.

IV. PERFORMANCE EVALUATION

This section presents the performance evaluation of the proposed GPC technique and its comparison against the *Selec-*

TABLE II. DELAY PARAMETERS USED IN THE SIMULATION.

Parameters	Set 1	Set 2
Channel Switching Time (t_{switch})	11.4 ms	11.4 ms
MinChannelTime (t_{min})	20 ms	1 ms
MaxChannelTime (t_{max})	200 ms	10 ms
Authentication delay (t_{auth})	6 ms	6 ms
Reassocaitioan delay (t_{assoc})	4 ms	4 ms

tive Scan with Caching (*SSwC*) [5] and *Neighbor Graph* (*NG*) [4, 10] techniques.

A. Measurements of Delay Parameters

The overall handoff delay analysis for each technique is based on two sets of parameters shown in Table II: *Channel Switching Time* (t_{switch}) is the time required to switch from one channel to another; *MinChannelTime* (t_{min}) is the minimum amount of time MS has to wait on an empty channel; *MaxChannelTime* (t_{max}) is the maximum amount of time MS has to wait to collect all the probe responses, which is used when a response is received within MinChannelTime; *Authentication delay/timeout* (t_{auth}) is the time required to perform authentication based on MAC addresses; and *Reassociation delay* (t_{assoc}) is the time requires to perform reassocation.

The Parameter Set 1 represents the current off-the-shelf devices, and was obtained using an experimental setup that consisted of two laptops with PCMCIA 802.11a/b/g NICs based on Atheros AR 5002X chipsets [11] (running Linux 2.6 with Madwifi driver [12] on Laptop #1 as a traffic generator and FreeBSD 6.1 on Laptop #2 as a traffic observer), a Sun SPARC Server with Ethernet LAN NIC (running SunOS 5.1), and an HP ProCurve Wireless Access Point 420. The NICs on the AP and on both laptops are operating on Ch. 1. Measurements were obtained by having the first laptop transmit a stream of 16-byte UDP packets to the server. Then, the NIC on the first laptop is forced to switch to Ch. 2, which has no APs, and immediately switch back to Ch. 1. During this time, tcpdump running on the second laptop sniffs the traffic. Parameter Set 1 was obtained from average values of 2400 measurements over a period of a day to reduce variations due to network traffic.

The Parameter Set 2 represents possible future NICs with reduced handoff delays based optimized t_{min} and t_{max} values from [8]. This study determined that the value of t_{min} that leads to minimized handoff delay is given by $t_{min} \geq DIFS + (aCWmin \times aSlotTime)$ [8], where *DIFS* is Distributed Inter-Frame Space, *aCWmin* is the number of slots in the minimum contention window, and *aSlotTime* is the length of a slot. In the IEEE 802.11g standard [13], the values for *DIFS*, *aCWmin*, and *aSlotTime* are 28 µs, 15 µs, and 9 µs, respectively, which results in $t_{min} \geq 163$ µs. However, t_{min} is defined in terms of Time Units (TU), where 1 TU = 1024 µs. Therefore, the smallest possible value of t_{min} is 1024 µs. Moreover, t_{max} is estimated as the transmission delay required when 10 MSs try to access the same AP. In their simulation, the bit rate of the channel is set to 2 Mbps, which is the maximum possible rate for management frames. The same bit rate for control frame also applies to IEEE 802.11g [12, 13]. Therefore, the estimated t_{max} is 10 ms.

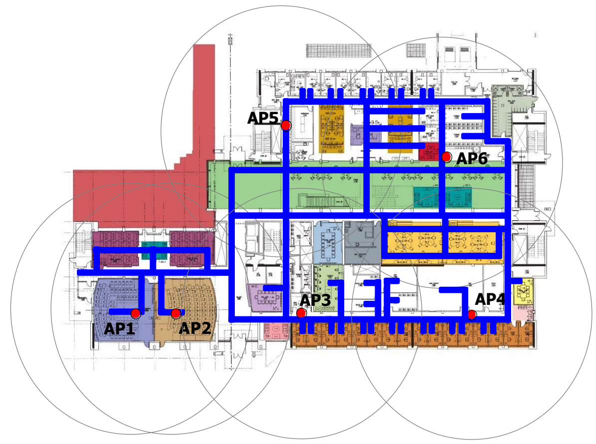

Figure 3. Topology and possible paths for the simulated network.

B. Simulation Environment

The topology used in the simulation study is shown in Fig. 3, which is the floor plan of the 1st floor of the KEC building. The red circles indicate AP locations and the blue lines show the possible paths of MSs. The simulator consists of two main modules: the *path generator* and the *handoff detector*. The path generator randomly selects a location on the topology then uses the path-finder algorithm [14] to generate a path for MS. On the other hand, the handoff detector monitors a MS's movement and performs a handoff when the distance between the MS and the associated AP reaches the maximum radius of the coverage area, which is based on log-distance path loss model [15]. This process is repeated for every one meter of movement. The handoff detector records the number of channel switches, the number of times MS has to wait for t_{max}, t_{min}, t_{auth}, and t_{assoc}. The AP coverage area was estimated by monitoring the connection while slowly moving a test MS further away from the AP in different directions. Our measurements show that the average radius of a coverage area is approximately 31 meters.

C. Sumlation Results

In order to provide a fair comparison, SSwC was extended to have an unlimited number of AP cache entries and Next-AP predictions per entry rather than only 10 AP cache entries and two Next-AP predictions per entry (i.e., best AP and second Best AP) used in the original SSwC.

We first analyze the overall Next-AP prediction accuracies of GPC and SSwC. The overall accuracy is defined as the number of correct predictions divided by the total number of handoffs. The NG technique is not included in this comparison since it does not provide a Next-AP prediction mechanism. Fig. 4 compares the overall accuracy of GPC and SSwC as function of history, which is represented as the number of handoffs. As can be seen, when the number of handoffs is low (10^2~10^3), GPC lacks sufficient history and thus the overall accuracy is below 100%, and decreases as N increases. This is because a larger N leads to a larger number of possible handoff sequences, and thus longer history is required to record all possible handoff sequences in GPC. However, beyond 10^4 handoffs, the overall accuracy for GPC becomes 100% because all

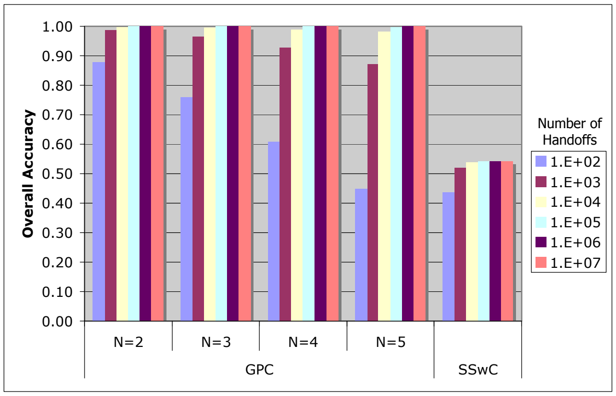

Figure 4. Overall Accuracy as function of history.

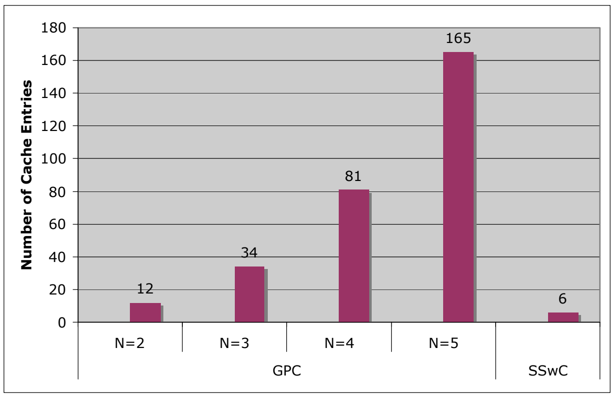

Figure 5. Number of required cache entries.

the possible handoff sequences have been recorded in GPC. Thus, all path-cache requests will be provided with correct Next-AP predictions. The overall accuracy of SSwC also increases as function of number of handoffs, but saturates at ~54%. The reasons for this will be explained the following paragraphs.

In order to properly compare the performance, all the subsequent results were obtained based on the assumption that (1) GPC maintains a complete history of handoff patterns, (2) AP-cache of SSwC contains entries for all the APs in the network, and (3) NG is preconfigured. This is done by first running the simulations for 10^4 handoffs to fill up the respective caches and performing NG construction, and then gathering statistics for up to 10^7 handoffs.

Fig. 5 shows the maximum number cache entries needed for GPC and SSwC. Again, NG is not included in this comparison. The AP-cache used in SSwC requires only six entries, which is the number available APs in the 1st floor of the KEC building. In contrast, GPC keeps track of MSs' more complex moving paths as *N* increases but requires more entries. Note that the number of entries cannot be directly compared because multiple GPC entries provide multiple Next-AP predictions, where as each entry in AP-cache provides multiple Next-AP predictions. Therefore, a more accurate metric is the average number of Next-AP predictions returned per handoff shown in Fig. 6. As can be seen, GPC provides higher average number Next-AP predictions per handoff than SSwC.

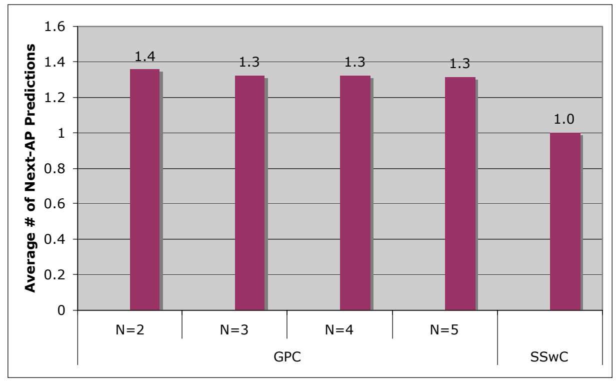

Figure 6. Average Number of Next-AP Predictions.

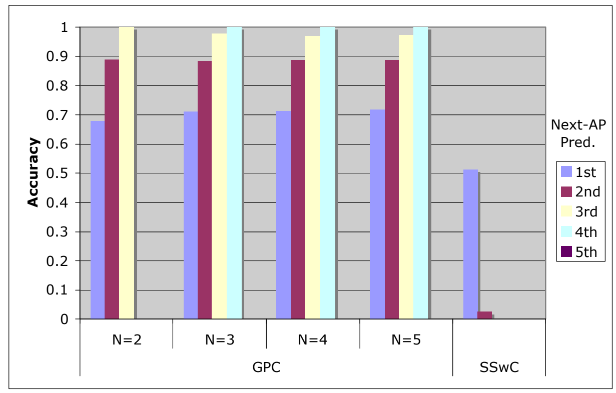

Figure 7. Accuracies of Next-AP Predictions.

Fig. 7 compares the accuracies of Next-AP predictions. The set of returned predictions is prioritized based on their hit counter values for GPC and signal strengths for SScW. The significance of these priorities is that each misprediction adds to the overall handoff delay. For GPC, the accuracy for the 1st Next-AP prediction starts at 68% and increases slightly as function of *N*. 1st Next-AP predictions that fail are satisfied by 2nd Next-AP predictions with accuracy of 89%. 3rd and 4th predictions only become effective with a longer handoff history and provide accuracies of 97%-100% and 100%, respectively. In contrast, SSwC provides significantly lower 1st and 2nd prediction accuracies of 51% and 2.6%, respectively. Note that SSwC provides at most only two predictions, while GPC offers up to four predictions. The reason for this can be explained from the characteristic of overlapped cells shown in Fig. 3. Our simulations show that 89% of the overlapped regions traveled by MSs are cover by two cells, and only 11% have three cells. Thus, SSwC will have at most two Next-AP predictions. In contrast, the maximum number of Next-AP predictions depends on the number of adjacent cells, which is four.

The GPC's superior prediction accuracy comes from not only the history handoff sequences but also the set of returned predictions is prioritizes based on how often these paths are encountered. In contrast, SScW relies only on signal strength, which depends on when and where full or selective scans were performed and the relative directions of MSs. Moreover, the AP-cache only caches all the unique APs in the network. Therefore, if the actual paths taken by MSs are different from

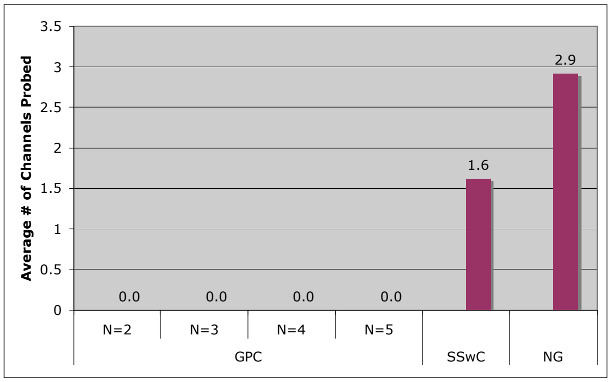

Figure 8. Average Number of Channels Probed.

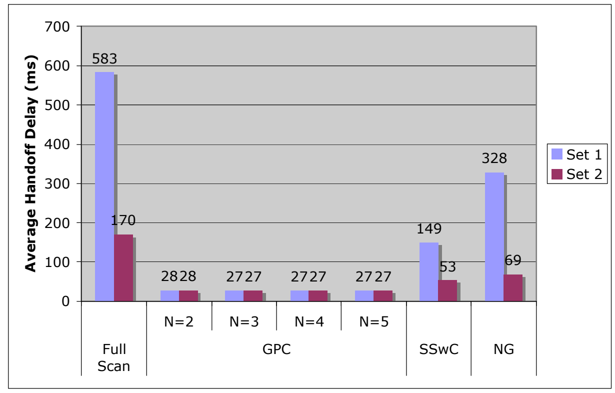

Figure 9. Average Handoff Delay.

predictions recorded in the AP-cache, they lead to higher mispredictions that add to the overall handoff delay.

These mispedictions are reflected in the average number of channels probed per handoff shown in Fig. 8, which also includes the result for NG. The SSwC scheme probes on average 1.6 channels. This is because Next-AP prediction provided by SSwC has very low accuracy (see Fig. 7) that cause 47.7% of the handoffs to mispredict and have to rely on selective scanning, which involves selecting the best AP from channels 1, 6, 11, and channels heard from either previous full scan or selective scan. The average number of channels probed for NG is higher at 2.9, and depends on the number of neighbor nodes encountered at each point-of-attachment. For GPC, the number channels probed per handoff is zero because once the GPC has a complete history it is guaranteed to provide accurate Next-AP predictions.

Fig. 9 shows the average handoff delays for all three techniques based on the two parameter sets defined in Table II, and includes the result for full scan as a reference. These results show that GPC results in lowest average handoff delay due to better Next-AP prediction accuracy. Overall, GPC incurs average handoff delay of 27~28 ms for both parameter sets and is significantly lower than SSwC and NG. Finally, the suggested size for *N* is 2 or 3 for a network such as ours because the average handoff delay is relatively constant regardless of *N* and yet they require only a minimal number of entries in GPC.

V. Conclusion

This paper described the GPC technique to minimize the time required to probe for APs. GPC is different from the other existing methods because it uses global history handoffs to determine directions of moving MSs. Therefore, it captures the mobility patterns of MSs much like NG and at the same time provides a much more accurate Next-AP predictions than SSwC. Our simulation study shows that GPC eliminates the need to perform scanning and thus results in much lower overall handoff delay compared to the existing techniques.

For future work, we plan to investigate several issues. First, we plan study the effectiveness of GPC in larger WLANs and metropolitan area networks. Second, we also plan to study the effectiveness of GPC in high traffic areas where MSs can be disconnected because of large number of packet losses due to contention. Third, we plan to further improve the Next-AP prediction accuracy by considering mobility patterns of users as function time-of-day and idiosyncrasies. Finally we would like to investigate how GPC can be utilized to speed up vertical handoffs.

References

[1] M. Lewis. CiceroPhone™ Smartphone VoIP client. Jan. 20, 2006. http://www.arcchart.com/blueprint/show.asp?id=383.

[2] A. Mishra, M. Shin, and W. A. Arbaugh, "An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process," *ACM Computer Communications Review*, vol. 33, pp. 93-102, April 2003.

[3] International Telecommunication Union. General Characteristics of International Telephone Connections and International Telephone Circuits. ITU-TG.114, 1988.

[4] M. Shin, A. Mishra, and W. A. Arbaugh, "Improving the latency of 802.11 hand-offs using Neighbor Graphs," *ACM MobiSys Conference*, June 2004.

[5] S. Shin *et al.*, "Reducing MAC Layer Handoff Latency in IEEE 802.11 Wireless LANs," *ACM International Workshop on Mobility Management and Wireless Access (MobiWac)*, Sept. 26 - Oct. 1, 2004.

[6] V. Brik, A. Mishra, and S. Banerjee, "Eliminating handoff latencies in 802.11 WLANs using Multiple Radios: Applications, Experience, and Evaluation," *International Measurement Conference* (IMC), Oct. 19-21, 2005.

[7] S. Waharte, K. Ritzenthaler, and R. Boutaba, "Selective Active Scanning for Fast Handoff in WLAN Using Sensor Networks," *IFIP/IEEE International Conference on Mobile and Wireless Communication Networks* (MWCN 2004), Oct. 25-27, 2004.

[8] H. Velayos and G. Karlsson, "Techniques to Reduce IEEE 802.11b MAC Layer Handover Time," *Proc. IEEE International Conference on Communications*, June 2004.

[9] I. Ramani and S. Savage, "SyncScan: Practical Fast Handoff for 802.11 Infrastructure Networks," *IEEE International Conference on Computer Communication* (INFOCOM), March 13-17, 2005.

[10] A. Mishra, M. Shin, and W. A. Arbaugh, "Context Caching using Neighbor Graphs for Fast Handoffs in a Wireless Network," *IEEE International Conference on Computer Communication* (INFOCOM), March 7-11, 2004.

[11] Atheros AR5002X 802.11a/b/g Universal WLAN Solution. http://www.atheros.com/pt/AR5002XBulletin.htm

[12] Madwifi_0.9.2. URL: http://www.madwifi.org

[13] IEEE. *IEEE Standard 802.11. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, 1999 Edition, Reaffirmed 2003.

[14] A. J. Patel, Path Finder Algorithm. http://theory.stanford.edu/~amitp/GameProgramming

[15] T. S. Rappaport, *Wireless Communications: Principles and Practice, Second Edition*, Prentice Hall, 2002.