Selective Channel Scanning for Fast Handoff in Wireless LAN using Neighbor Graph

Sang-Hee Park, Hye-Soo Kim, Chun-Su Park, Jae-Won Kim, and Sung-Jea Ko

Department of Electronics Engineering, Korea University,
Anam-Dong Sungbuk-Ku, Seoul, Korea
Tel: +82-2-3290-3672
{jerry, hyesoo, cspark, jw9557, sjko}@dali.korea.ac.kr

Abstract. Handoff at the link layer 2 (L2) consists of three phases: scanning, authentication, and reassociation. Among the three phases, scanning is dominant in terms of time delay. Thus, in this paper, we propose an improved scanning mechanism to minimize the disconnected time while the wireless station (STA) changes the associated access points (APs). According to IEEE 802.11 standard, the STA has to scan all channels in the scanning phase. In this paper, based on the neighbor graph (NG), we introduce a selective channel scanning method for fast handoff in which the STA scans only channels selected by the NG. Experimental results show that the proposed method reduces the scanning delay drastically.

1 Introduction

In recent years, wireless local area network (WLAN) with wide bandwidth and low cost has emerged as a competitive technology to adapt the user with strong desire for mobile computing. The main issue of mobile computing is handoff management. Especially, for real-time multimedia service such as VoIP, the long handoff delay has to be reduced. Many techniques [1]-[4] have been proposed by developing new network protocols or designing new algorithms. Their approaches are categorized as network layer (L3), L2, and physical layer (PHY).

One of the previous works based on L3 uses the reactive context transfer mechanism [1],[2]. This mechanism is designed solely for access routers (ARs) and is reactive rather than pro-active. On the other hand, Nakhjiri [5] proposed a general purpose context transfer mechanism, called SEAMOBY, without detailing transfer triggers. In SEAMOBY, a generic framework for either reactive or pro-active context transfer is provided, though the framework does not define a method to implement either reactive or pro-active context transfer.

To reduce the L2 handoff delay in WLAN using inter access point protocol (IAPP) [7], an algorithm on the context transfer mechanism utilizing the NG [6] was suggested in [7]. But originally, IAPP was only reactive in nature and creates an additional delay in handoff. Thus, this algorithm can not shorten the original L2 handoff delay.

One approach on PHY is the method using two transceivers. In this method, an STA has two wireless network interface cards (WNICs), one for keeping connection to current AP and the other for scanning channels to search alternative APs [8].

In this paper, we propose a selective channel scanning mechanism using the NG to solve the L2 handoff delay. In the proposed mechanism, the STA scans only channels selected by the NG without scanning all the channels. And on receiving a *ProbeResponse* message, the STA scans the next channels without waiting for the pre-defined time. Therefore, the delay incurred during the scanning phase can be reduced.

This paper is organized as follows. Section 2 describes the operations in IEEE 802.11 [9]. In Section 3, the proposed algorithm is presented. Finally, Section 4 shows the results experimented on our test platform and presents brief conclusion comments.

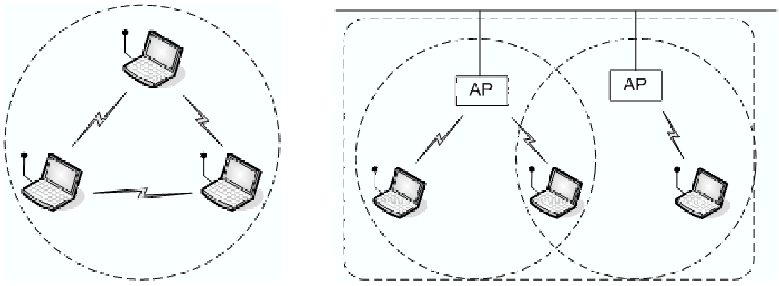


Fig. 1. Types of WLAN (a) ad hoc mode(b) infrastructure mode.

2 IEEE 802.11

As shown in Fig. 1, IEEE 802.11 MAC specification allows for two modes of operation: ad hoc and infrastructure modes. In the ad hoc mode, two or more STAs recognize each other through beacons and establish a peer-to-peer relationship. In this configuration, STAs communicate with each other without the use of an AP or other infrastructure. The ad hoc mode connects STAs when there is no AP near the STAs, when the AP rejects an association due to failed authentication, or when the STAs are explicitly configured to use the ad hoc mode. In the infrastructure mode, STAs not only communicate with each other through the AP but also use the AP to access the resource of the wired network which can be an intranet or internet depending on the placement of the AP. The basic building block of IEEE 802.11 network is a basic service set (BSS) consisting of a group of STAs that communicate with each other. A set of two or more APs

connected to the same wired network is known as an extended service set (ESS) which is identified by its service set identifier (SSID).

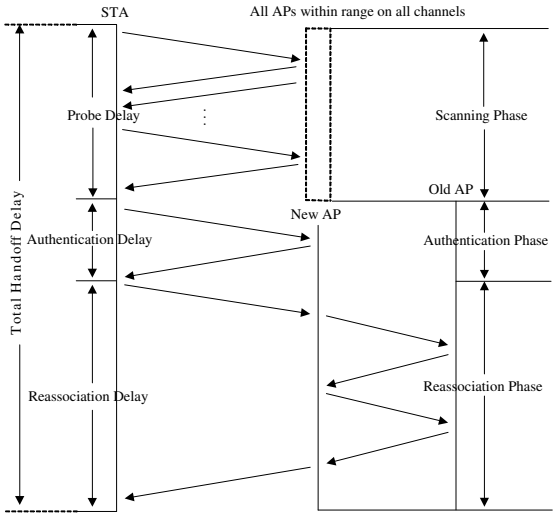


Fig. 2. IEEE 802.11 handoff procedure with IAPP.

The STA continuously monitors the signal strength and link quality from the associated AP. If the signal strength is too low, the STA scans all the channels to find a neighboring AP that produces a stronger signal. By switching to another AP, the STA can distribute the traffic load and increase the performance of other STAs. During handoff, PHY connectivity is released and state information is transferred from one AP to another AP. Handoff can be caused by one of the two types of transition including BSS and ESS transitions. In the BSS transition, an STA moves within an extended service area (ESA) that constructs an ESS. In this case, an infrastructure does not need to be aware of the location of the STA. On the other hand, in the ESS transition, the STA moves from one ESS to another ESS. Except for allowing the STA to associate with an AP in the second ESS, IEEE 802.11 does not support this type of transition. In this paper, we focus on BSS transition.

2.1 Handoff Procedure

The complete handoff procedure can be divided into three distinct logical phases: scanning, authentication, and reassociation. During the first phase, an STA scans for APs by either sending *ProbeRequest* messages (Active Scanning) or listening for *Beacon* messages (Passive Scanning). After scanning all the channels, the STA selects an AP using the received signal strength indication (RSSI), link quality, and etc. The STA exchanges IEEE 802.11 authentication messages with

the selected AP. Finally, if the AP authenticates the STA, an association moves from an old AP to a new AP as following steps:

(1) An STA issues a *ReassociationRequest* message to a new AP. The new AP must communicate with the old AP to confirm that a previous association existed;
(2) The new AP processes the *ReassociationRequest*;
(3) The new AP contacts the old AP to finish the reassociation procedure with IAPP;
(4) The old AP sends any buffered frames for the STA to the new AP;
(5) The new AP begins processing frames for the STA.

The delay incurred during these three phases is referred to as the L2 handoff delay, that consists of probe delay, authentication delay, and reassociation delay. Figure 2 shows the three phases, delays, and messages exchanged in each phase.

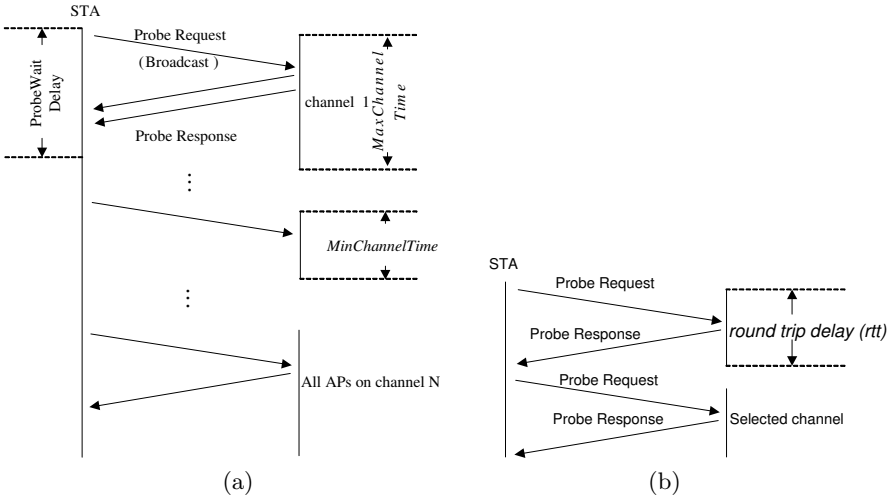


Fig. 3. Active Scanning (a) full channel scanning (b) selective channel scanning.

2.2 Passive and Active Scanning Modes

The STA operates in either a passive scanning mode or an active scanning mode depending on the current value of the *ScanMode* parameter of the MLME-SCAN.request primitive. To become a member of a particular ESS using the passive scanning mode, the STA scans for *Beacon* messages containing SSID indicating that the *Beacon* message comes from an infrastructure BSS or independent basic service set (IBSS). On the other hand, the active scanning mode

attempts to find the network rather than listening for the network to announce itself. STAs use active scanning mode with the following procedure:

(1) Move to the channel and wait for either an indication of an incoming frame or for the *ProbeTimer* to expire. If an incoming frame is detected, the channel is in use and can be probed;
(2) Gain access to medium using the basic distributed coordination function (DCF) access procedure and send a *ProbeRequest* message;
(3) Wait for the *MinChannelTime* to elapse.
  a. If the medium has never been busy, there is no network. Move to the next channel.
  b. If the medium was busy during *MinChannelTime* interval, wait until *MaxChannelTime* and process any *ProbeResponse* messages.

When all the channels in the *ChannelList* are scanned, the MLME issues an MLME-SCAN.confirm primitive with the *BSSDescriptionSet* containing all information gathered during the active scanning. Figure 3 (a) shows messages, *MaxChannelTime*, and *MinChannelTime* for the active scanning.

During the active scanning, the bound of scanning delay can be calculated as

$$N \times T_b \leq t \leq N \times T_t, \tag{1}$$

where N is the total number of channels which can used in a country, T_b is *MinChannelTime*, T_t is *MaxChannelTime*, and t is the total measured scanning delay.

Mishra [10] showed that the scanning delay is dominant among the three delays. Thus, to solve the problem of the L2 handoff delay, the scanning delay has to be reduced. Therefore, our paper focuses on the reduction of the delay time of the active scanning.

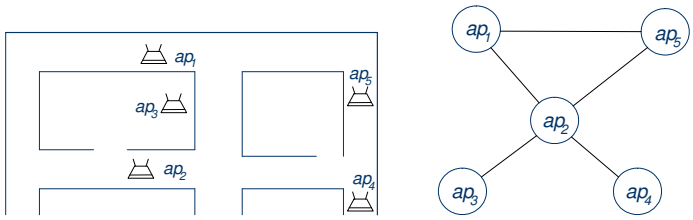
3 Proposed Scanning Method

Before introducing our proposed method, we briefly review the NG. The NG is an undirected graph where each edge represents a mobility path between APs. Therefore, for a given edge, the neighbors of the edge become the set of potential next APs. Figure 4 shows the concept of the NG with five APs.

The undirected graph representing the NG is defined as

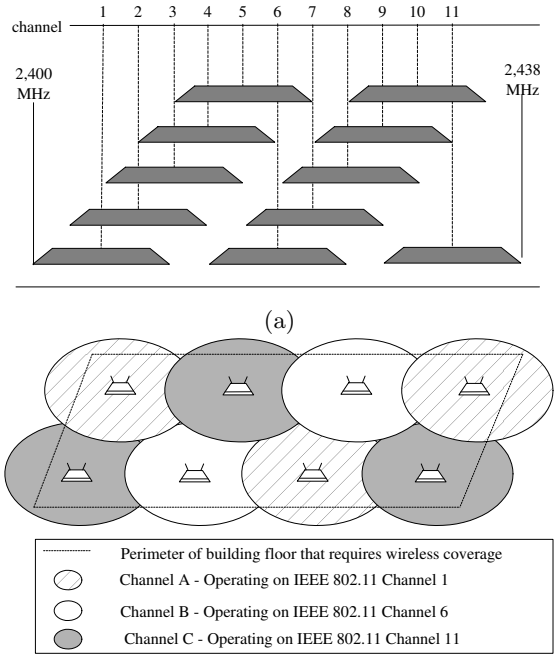
$$\begin{aligned} &G = (V, E), \\ &V = \{ap_1, ap_2, \ldots, ap_i\}, \\ &e = (ap_i, ap_j), \\ &N(ap_i) = \{ap_{ik} : ap_{ik} \in V, (ap_{i,} ap_{ik}) \in E\}, \end{aligned} \tag{2}$$

where G is the data structure of NG, V is a set containing all APs, E is a set consisting of edges, e's, and N is the neighbor APs near an AP.



(a) (b)

Fig. 4. Concept of neighbor graph (a) placement of APs (b) corresponding neighbor graph.



(a)

(b)

Fig. 5. Selecting channel frequencies for APs (a) channel overlap for 802.11b APs (b) example of channel allocation.

The NG is configured for each AP manually or is automatically generated by an individual AP over time. The NG can be automatically generated by the following algorithm with the management messages of IEEE 802.11.

(1) If an STA sends *Reassociate Request* to AP_i with *old-ap* = AP_j, then create new neighbors (i, j) (i.e. an entry in AP_i, for j and vice versa);
(2) Learn costs only one '*high latency handoff*' per edge in the graph;
(3) Enable mobility of APs which can be extended to wireless networks with an ad hoc backbone infrastructure.

In general, the STA scans all channels from the first channel to the last channel, because it is not aware of the deployment of APs near the STA. However, as shown in Fig. 5 (a), the STA can use only three channels at the same site (in case of United States, channel 1, 6, 11) because of the interference between adjacent APs. Thus, all channels are not occupied by neighbor APs to permit efficient operation although multiple APs are operating at the same site. If channels of APs existing near the STA are known, the STA does not need to scan all channels as shown in Fig. 5 (b). In this section, based on the NG, we introduce a selective channel scanning method for fast handoff in which an STA scans only channels selected by the NG. The NG proposed in [6] uses the topological information on APs. Our proposed algorithm, however, requires information on channels of APs as well as topological information. Thus, we modify the data structure of NG defined in (2) as follows:

$$\begin{aligned} &G' = (V', E), \\ &V' = \{v_i : v_i = (ap_i, channel), v_i \in V\}, \\ &e = (ap_i, ap_j), \\ &N(ap_i) = \{ap_{ik} : ap_{ik} \in V', (ap_i, ap_{ik}) \in E\}, \end{aligned} \tag{3}$$

where G' is the modified NG, and V' is the set which consists of APs and their channels.

Assume in Fig. 4 (b) that an STA is associated to ap_2. The STA scans only 4 channels of its neighbors (ap_1, ap_3, ap_4, ap_5) instead of scanning all channels. Figure 3 (b) shows that the STA scans potential APs selected by the NG.

In order to scan a channel, the STA must wait for *MaxChannelTime* after transmitting a *ProbeRequest* message whose destination is all APs as shown in Fig. 3 (a). Because the STA does not have information on how many APs would respond to the *ProbeRequest* message. However, if the STA knows the number of APs occupying the channel, the STA can transmit another *ProbeRequest* message to scan the next channel without waiting for *MaxChannelTime* when it receives the number of predicted *ProbeResponse* messages. Our mechanism can be summarized as follows:

(1) Retrieve the NG information (neighbor APs and their channels) according to the current AP;
(2) Select one channel;

(3) Transmit a *ProbeRequeset* message to the selected channel and start *ProbeTimer*;

(4) Wait for *ProbeResponse* messages not until *ProbeTimer* reaches *MaxChannelTime*, but until the number of the received *ProbeResponse* messages becomes the same as the number of potential APs in the channel.

a. If the channels to be scanned remain, process (1).

b. Otherwise, start the authentication phase.

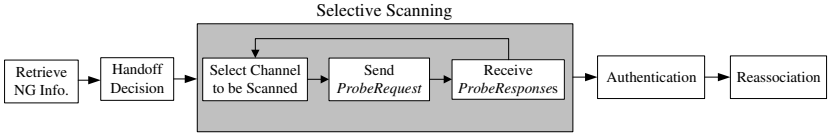


Fig. 6. Block diagram of the proposed method.

Figure 6 shows the block diagram of the proposed method described above. With the proposed scanning algorithm, the scanning delay can be expressed as

$$\begin{aligned} t &= N' \times (t_{proc} + rtt) + (N - N')/2 \times (t_{proc} + rtt) + \alpha \\ &= (N + N')/2 \times (t_{proc} + rtt) + \alpha, \end{aligned} \tag{4}$$

where N is the number of the potential APs, N' is the number of channels selected by the NG, t_{proc} is the processing time of MAC and PHY headers, rtt is the round trip time, and α is the message processing time. While *MinChannelTime* and *MaxChannelTime* in (1) is tens of milliseconds, t_{proc} is a few milliseconds.

4 Experimental Results

Figure 7 shows our experimental platform consisting of an STA, APs, router, and correspondent node (CN). To exchange the NG information, socket interface is used, and Mobile IPv6 is applied to maintain L3 connectivity while experimenting. The device driver of a common WNIC was modified such that the STA operates as an AP. And emulated APs perform the foreign agents on our experimental platform. To simulate the operation of the proposed mechanism, we developed three programs: *NG Server*, *NG Client*, and *Monitor*. The *NG Server* manages the data structure of NG on the experimental platform and processes the request of the *NG Client* which updates the NG information of the STA after the STA moves to the another AP. The *Monitor* collects the information on wireless channel, decides if the STA has to handoff, then starts the handoff. In Fig. 7, *NG Client* and *Monitor* are deployed at the STA and *NG Server* is deployed at the CN, respectively. We defined the scanning delay as the duration

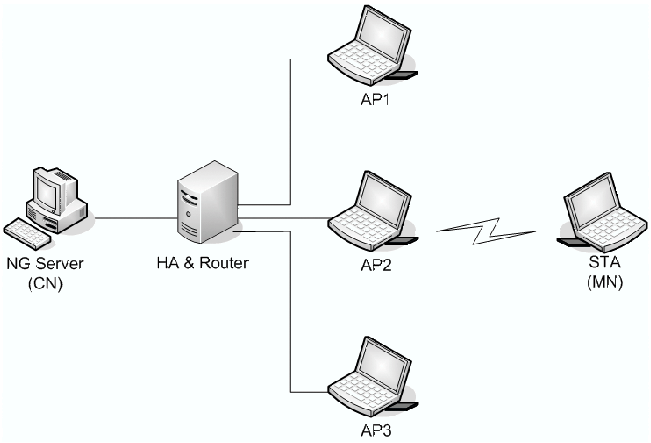


Fig. 7. Experimental platform.

Table 1. Average probe delay of each method.

Neighbors	Method	delay [ms]
1	Full channel active scanning	322
	Selective scanning	55
	Selective scanning without *MaxChannelTime*	12
2	Full channel active scanning	322
	Selective scanning	109
	Selective scanning without *MaxChannelTime*	21
3	Full channel active scanning	322
	Selective scanning	145
	Selective scanning without *MaxChannelTime*	30
4	Full channel active scanning	322
	Selective scanning	190
	Selective scanning without *MaxChannelTime*	39

between the first *ProbeRequest* message and the last *ProbeResponse* message. To capture these two kinds of messages, we used the sniffer program, AiroPeek.

To evaluate the proposed algorithm, we experimented with three mechanisms: full channel active scanning mechanism; selective scanning; and selective scanning without *MaxChannelTime* for the number of neighbors. As shown in Table 1, the scanning delay by the selective scanning is shorter than the full channel active scanning. Note that the selective scanning without *MaxChannelTime* produces a smallest delay time among the three mechanisms.

5 Conclusions

we have introduced the selective channel scanning method for fast handoff in which an STA scans only channels selected by the NG. The experimental re-

sults showed that the proposed method can reduce the overall handoff delay drastically. Therefore, we expect that the proposed algorithm can be an useful alternative to the existing full channel active scanning due to its reduced delay time.

References

1. Koodli, R., Perkins, C.: Fast Handovers and Context Relocation in Mobile Networks. ACM SIGCOMM Computer Communication Review. Vol. 31 (2001)
2. Koodli, R.: Fast Handovers for Mobile IPv6. Internet Draft : draft-ietf-mipshop-fast-mipv6-01.txt (2004)
3. Cornall, T., Pentland, B., Pang, K.: Improved Handover Performance in wireless Mobile IPv6. ICCS. Vol. 2 (2003) 857–861
4. Park, S. H., Choi, Y. H.: Fast Inter-AP Handoff Using Predictive Authentication Scheme in a Public Wireless LAN. IEEE Networks ICN. (2002)
5. Nakhjiri, M., Perkins, C., Koodli, R.: Context Transfer Protocol. Internet Draft : draft-ietf-seamoby0ctp-09.txt. (2003)
6. Mishra, A., Shin, M. H., Albaugh, W.: Context Caching using Neighbor Graphs for Fast Handoff in a Wireless. Computer Science Technical Report CS-TR-4477. (2003)
7. IEEE: Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.1 Operation. IEEE Standard 802.11. (2003)
8. Ohta, M.: Smooth Handover over IEEE 802.11 Wireless LAN. Internet Draft : draft-ohta-smooth-handover-wlan-00.txt. (2002)
9. IEEE: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Standard 802.11. (1999)
10. Mishra, A., Shin, M. H., Albaugh, W.: An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. ACM SIGCOMM Computer Communication Review. Vol. 3 (2003) 93–102