Fast Intra-Network and Cross-Layer Handover (FINCH) for WiMAX and Mobile Internet

Jui-Hung Yeh, *Member*, *IEEE*, Jyh-Cheng Chen, *Senior Member*, *IEEE*, and
Prathima Agrawal, *Fellow*, *IEEE*

Abstract—To support fast and efficient handovers in mobile WiMAX, we propose *Fast Intra-Network and Cross-layer Handover (FINCH)* for *intradomain (intra-CSN)* mobility management. FINCH is a complementary protocol to Mobile IP (MIP), which deals with *interdomain (inter-CSN)* mobility management in mobile WiMAX. FINCH can reduce not only the handover latency but also the end-to-end latency for MIP. Paging extension for FINCH is also proposed to enhance the energy efficiency. The proposed FINCH is especially suitable for real-time services in frequent handover environment, which is important for future mobile WiMAX networks. In addition, FINCH is a generic protocol for other IEEE 802-series standards. This is especially beneficial for the integration of heterogeneous networks, for instance, the integration of WiMAX and WiFi networks. Both mathematical analysis and simulation are developed to analyze and compare the performance of FINCH with other protocols. The results show that FINCH can support fast and efficient link layer and intradomain handovers. The numerical results can also be used to select proper network configurations.

Index Terms—Intradomain mobility management, cross-layer design, WiMAX, mobile Internet.

◆

1 INTRODUCTION

The IEEE 802.16 standard [1] is a promising standard for next-generation broadband wireless access networks. It provides *last mile* solution and supports high-speed multimedia services. The IEEE 802.16e amendment [2] enhances IEEE 802.16 with mobility support for users moving at vehicular speeds. Like other IEEE 802-series standards, 802.16 standardizes physical (PHY) layer and Media Access Control (MAC) layer only. To build a complete system, higher layers are still necessary. One of the major objectives of WiMAX Forum [3], thus, is to promote conformance and interoperability of the IEEE 802.16 standards. The network reference model proposed by WiMAX Forum is depicted in Fig. 1 [4]. The *Access Service Network (ASN)* provides radio access to WiMAX subscribers. It consists of one or more ASN Gateways (ASN GWs) and Base Stations (BSs). ASNs are connected by *Connectivity Service Network (CSN)*, which provides Internet Protocol (IP) connectivity services. To support IP mobility, Mobile IP (MIP, IETF RFC 3344) is adopted by WiMAX Forum [5]. The Home Agent (HA) of a Mobile Station (MS) is located in the CSN of the MS's *Home Network Service Provider (H-NSP)*. ASN GW supports the Foreign Agent (FA) functionality. For intra-ASN mobility, there is no need to update MS's care-of-address (CoA). For inter-ASN mobility, on the other hand, the MS needs to update its CoA and register the new CoA (NCoA) with its HA.

MIP is a simple and effective way to deal with mobility management in the network layer. However, it also has some deficiencies, which include frequent location update, long handover (HO) delay, and long end-to-end latency. The registration of CoA with HA will cause many control messages if an MS moves frequently between subnets. When an MS is far away from its HA, the registration will result in long delay. Thus, it causes long HO delay. The registration sent from an MS to its HA also results in extra delay. When an MS is far away from its HA, the redirection of user packets by HA will cause long end-to-end latency as well. Although route optimization has been proposed, it is designed as an optimization and can result in latency, which is highly variable. Moreover, all of the problems related to a roving MS outlined above are exaggerated for *real-time* services that require very fast HO. Although MIP is suitable for *interdomain mobility* (also referred to as *macromobility*[1]), it is generally realized that MIP may not be suitable for *intradomain mobility* (also referred to as *micromobility*), especially for real-time services.

Mobile WiMAX has been designed to support users moving at vehicular speeds. In addition, real-time services such as voice and multimedia applications are expected to be important services in future mobile WiMAX networks. Because MIP is adopted, mobile WiMAX is likely to inherit the MIP deficiencies discussed above. Although IPv6 might be more efficient than IPv4, it, however, is not widely deployed. Because we aim to provide an immediate

- *J.-H. Yeh is with Realtek-NCTU Joint Research Center, National Chiao Tung University, Hsinchu 30010, Taiwan, R.O.C. E-mail: jhyeh@wire.cs.nthu.edu.tw.*
- *J.-C. Chen is with the Department of Computer Science and the Institute of Communications Engineering, National Tsing Hua University, Hsinchu 300, Taiwan, R.O.C. E-mail: jcchen@cs.nthu.edu.tw.*
- *P. Agrawal is with the Department of Electrical and Computer Engineering and Wireless Engineering Research and Education Center, Auburn University, Auburn, AL 36849. E-mail: pagrawal@eng.auburn.edu.*

Manuscript received 8 Aug. 2007; revised 2 July 2008; accepted 29 Sept. 2008; published online 15 Oct. 2008.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number TMC-2007-08-0236.
Digital Object Identifier no. 10.1109/TMC.2008.149.

---

[1] In mobile WiMAX, macromobility refers to *inter-ASN mobility* and micromobility refers to *intra-ASN mobility*, which are different with the definitions here. To make it clear, we use *interdomain mobility* and *intradomain mobility* for the rest of this paper. The scope of each mobility is indicated in Fig. 1.

1536-1233/09/$25.00 © 2009 IEEE Published by the IEEE CS, CASS, ComSoc, IES, & SPS

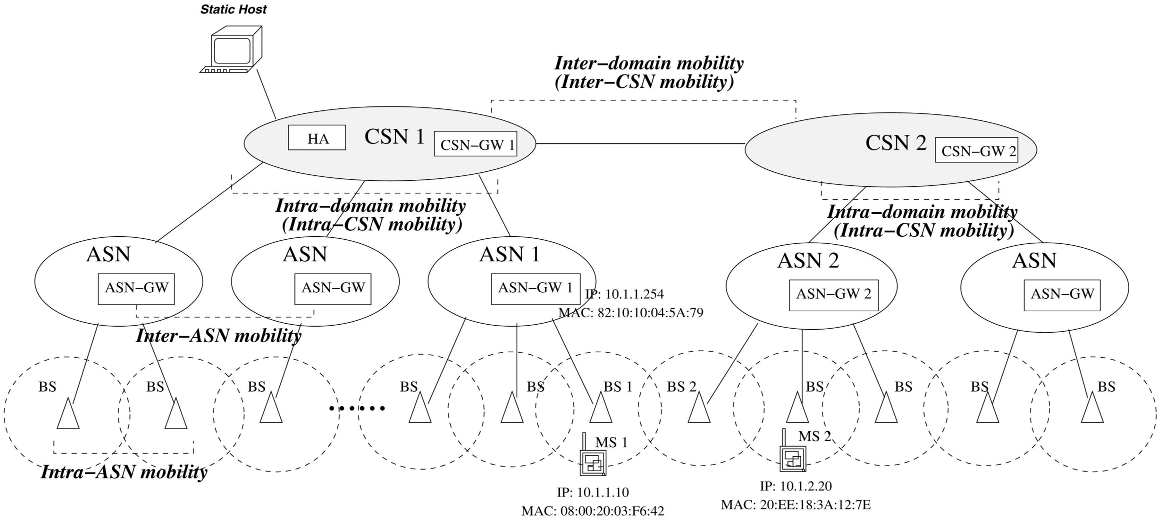


Fig. 1. Generic mobile WiMAX network architecture.

solution for current deployment of mobile WiMAX networks, this paper only focuses on the IPv4 over *Ethernet-like link model* [6]. In IPv4 over Ethernet, the Address Resolution Protocol (ARP) can incur significant delay for both packet delivery and HO. We propose to use MIP in mobile WiMAX for interdomain mobility (inter-CSN mobility) only. We propose a new protocol, *Fast Intra-Network and Cross-layer Handover (FINCH)*, for intradomain mobility (intra-CSN mobility), which can achieve fast HO, especially for real-time services. FINCH limits frequent HOs within CSN. It cooperates with MIP, which serves as the interdomain mobility management protocol. FINCH intends to localize location update to reduce the HO latency in MIP. It also reduces end-to-end latency because packets are delivered in a shorter path than that in MIP. In addition, FINCH is a cross-layer protocol, which also considers link layer. Thus, FINCH can further improve the performance. The proposed FINCH is especially suitable for real-time services in frequent HO environment. In addition, paging extension is designed to conserve the energy of MS and reduce the signaling overhead for location update. Although FINCH considers both IP and link layers, we limit the link layer to those which are common in all IEEE 802-series standards only. Thus, FINCH is also a generic protocol for other IEEE 802-series standards. This is especially beneficial for the integration of heterogeneous networks. For example, FINCH can be used in WiFi and WiMAX integrated networks.

The rest of this paper is organized as follows: In Section 2, we summarize the related work and point out the contributions of this paper. Section 3 delineates the design principles. The proposed FINCH is presented in Section 4. The analytical models and numerical results are presented in Sections 5 and 6, respectively. Section 7 concludes this paper.

2 RELATED WORK

Mobile IPv4 (MIPv4) is defined by the IETF as the mobility management protocol in IPv4 networks. Unlike MIPv4 that deals with the mobility in IP layer, Session Initiation Protocol (SIP, IETF RFC 3261) is adopted for application-layer mobility [7], [8]. In SIP-based mobility management, an SIP Redirect Server keeps tracking an MS's location. However, different from MIP, the SIP server only engages in setting up the communication link between the users. After that, user traffic can be delivered directly between the users.

Recently, the IETF specifies a Host Identity Protocol (HIP, IETF RFC 5201) that supports secure mobility management (IETF RFC 5206) over IP networks. A new protocol layer, HIP, between the network and transport layers is introduced. HIP decouples the roles of *locator* and *identifier* of an IP address. In HIP, IP address is only used for packet forwarding. Host Identifier (HI) is a public key that is used to represent the host identity. A pair of IPsec Security Associations (SAs) is then bound to the HIs, instead of the IP addresses, of the end peers. Therefore, communications between the peers can be protected by Encapsulation Security Payload (ESP, IETF RFC 5202) SAs over any IP address. The connections will not break when the underlying IP address change. Rendezvous Server (RVS) maps the HIs and the IP addresses for MSs. When an MS gets a new IP address in a foreign network, the MS updates its record in the RVS. The Corresponding Host (CH) can reach the MS by querying the RVS. When an MS changes its IP address while a session is still going on, it notifies both the RVS and the CH to update the new IP address of the MS. Therefore, the connection can be continued when the MS roams to a foreign network.

The protocols discussed above generally deal with the interdomain mobility management. There are many protocols designed to improve the performance of intradomain HOs in MIP. *MIPv4 Regional Registration* [also referred to as *Hierarchical MIP* (HMIP)] [9] employs two or more hierarchical level of FAs to reduce HO delay. At the top level of the hierarchy, there are Gateway FAs (GFAs). Beneath the GFA, there may be one or more Regional FAs (RFAs). An MS registers the GFA address as its CoA to the HA. The MS also maintains a local CoA that is used for receiving packets in the visited domain. Furthermore, RFAs handle the *Regional Registration Request* from MS to the GFA and maintain the *visitor lists* with the location information of the MS. Consequently, downlink packets can be successfully tunneled from the GFA through RFAs to the local CoA of the MS.

Cellular IP [10] was proposed to import the mobility management of cellular systems into an IP paradigm. Passive connectivity, paging, and fast HO are implemented in Cellular IP access networks. The routing in Cellular IP identifies MS with its home address and directly routes packets without tunneling or address conversion. *Uplink* packets originated from MS are sent to the gateway in a *hop-by-hop* manner. At the same time, each node on the path will cache the source direction of the MS. Thus, the *downlink* packets addressed to the MS can be forwarded back to the MS with the routing caches. Cellular IP intends to minimize the usage of explicit signaling messages. When there is no data to send, MS must send a special IP packet toward the gateway to indicate its current location.

In *Handoff-Aware Wireless Access Internet Infrastructure (HAWAII)* [11], a network is divided into several small domains. MIP is still used for *interdomain mobility*. In contrast, the HAWAII protocol handles the *intradomain mobility* management. Network nodes in HAWAII maintain mobile-specific routing entries on the legacy routing tables. Unlike Cellular IP, MS creates, updates, and modifies the location information with explicit signaling messages. HAWAII supports *forwarding* and *nonforwarding* path setup strategies. The forwarding scheme buffers packets forwarded to the old access point and redirects them to the new access point. On the other hand, the nonforwarding scheme drops the packets sent to the old access point.

The characteristics of various intradomain mobility management protocols are quantified by simulation in [12]. The results show that HMIP may experience more packet loss when MS performs HO. This is because the FAs keep tunneling data to the original path until the registration message from the MS reaches the GFA. The results also show that in nontree topology, the MS using HAWAII may choose *suboptimal routes* after handing off.

Many other intradomain mobility management protocols have also been proposed. Generally speaking, they can be categorized as *tunnel-based* and *host-specific-routing-based* protocols. The tunnel-based protocols usually employ hierarchical mobility architecture or require a *mobility gateway* to tunnel packets to and from MSs. Examples of the tunnel-based intradomain mobility management protocols include HMIP [9], Intradomain Mobility Management Protocol (IDMP) [13], Multicast-based mobility [14], and Dynamic HMIP (DHMIP) [15]. The host-specific-routing-based protocols, on the other hand, adopt new routing schemes to support intradomain mobility. That is, standard IP routing is not used for intradomain mobility management. Examples of this category include Cellular IP [10], HAWAII [11], and Mobility-aware MPLS [16]. The proposed FINCH also belongs to this category.

In addition to intradomain mobility management, *fast HO* is another technique to reduce HO latency and packet loss rate. Fast HOs for MIPv4 (F-MIPv4, IETF RFC 4988) and Fast HOs for Mobile IPv6 (F-MIPv6, IETF RFC 4068) are proposed to support fast HOs in MIPv4 and MIPv6, respectively. The basic idea of fast HO is that an MS can determines whether it is moving to a new subnet before the HO happens. The Previous Access Router (PAR) can forward the MS's packets to the New Access Router (NAR). The NAR can buffer the packets until the MS finishes Layer-2 (L2) HO. There are two modes in F-MIP. In the *predictive* mode, the MS sends Fast Binding Update (FBU) and receives Fast Binding Acknowledgement (FBack) on the PAR's link. Packets can be forwarded immediately to the NAR before L2 HO happens. For *reactive* mode, an MS leaves the PAR before sending FBU or receiving the FBack. The FBack message will be received on the NAR's link. After that, the PAR can forward packets as long as the MS transmits/retransmits the FBU to the NAR. Besides, F-MIP over IEEE 802.16e networks can be found in [17] and [18].

3 DESIGN PRINCIPLES

In this paper, we propose a cross-layered solution for intra-CSN mobility for mobile WiMAX networks. The proposed protocol is a generic solution, which can also be used for intradomain mobility in other 802-series networks. Most of the other protocols focus on the HO above link layer. The HO overhead in the link layer is usually neglected. One of the significant contributions of this paper is to propose a protocol to handle both IP and link layer mobilities. Notice that we limit the link layer to those that are common in all IEEE 802-series standards only. The proposed protocol does not intend to replace the original HO mechanisms of the 802.16e standard. Instead, the proposed protocol can cooperate with the L2 HO mechanisms to support seamless data communication for mobile users. The cross-layer design can largely reduce the HO overhead. Moreover, in a nontree network topology, the proposed protocol can easily find a shortest route by conditionally flooding signaling messages. Another contribution of this paper is that the proposed solution is a distributed protocol. It can be applied to any network topology. In the following, we delineate the essential principles of our design:

1. *Fast HO:* Because real-time services are expected to be important services for future wireless services, the proposed protocol should support fast HO for real-time applications. In addition, the proposed protocol should efficiently support fast moving wireless environment.
2. *Cross-layer:* As mentioned above, most intradomain mobility management protocols do not consider link layer HO. Moreover, in IP networks, address translation between IP and link layers is necessary for packet delivery. We intend to propose a cross-layer design that can reduce not only HO delay but also packet delivery overhead.
3. *Scalability:* Many of the previous proposals [9], [10], [11], [13], [19] rely on centralized *gateways* or *mobility agents* for location management and packet routing. When the network size is large, the centralized gateways or mobility agents may cause scalability issues. The proposed protocol should avoid using centralized nodes. The operation of the protocol should be distributed on each network node.

4. *Paging support:* Energy efficiency is essential to mobile terminals. Paging is an effective solution that enables a mobile node to reduce unnecessary location update and enter idle mode when there is no traffic. Therefore, the proposed protocol should support paging.
5. *Timely for deployment:* Mobile WiMAX is currently deployed in some countries. How to achieve fast HO, especially for real-time services, is a timely issue. Although IPv6 might be more efficient than IPv4, it however is not widely deployed. The IPv4 over *Ethernet-like link model* [6] is more likely to be deployed for today's mobile WiMAX networks. To be a practical solution for current deployment, we only consider the IPv4/Ethernet convergence sublayer (IPv4/Ethernet CS).
6. *Flexibility:* Although we consider a solution for mobile WiMAX networks, the proposed protocol should be generic for other IP networks. This is especially beneficial for the integration of heterogeneous wireless IP networks. In addition, unlike some other protocols, the proposed solution should not be limited to tree-based topology only. It should also be applicable to any network topology.

4 PROPOSED FINCH

This section describes the proposed FINCH.

4.1 Mobility Management in WiMAX

Before presenting the proposed FINCH, we first review the HO and mobility management defined in 802.16e [2] and WiMAX [4], [5], [20].

When an MS is moving, the Signal-to-Interference-plus-Noise Ratio (SINR) to the serving BS may be below the sustainable level. Therefore, the MS needs to perform HO. As defined in 802.16e [2], a BS will periodically broadcast information about the network topology. The serving BS also allocates time intervals, which are called *scanning intervals* to the MS. The MS then can seek and monitor suitable neighboring BSs as the target BSs. The HO process includes several stages. The basic idea is similar to the HOs in other systems. The HO initiation in 802.16e can be originated either at the MS or the serving BS. In addition to *hard HO*, 802.16e also supports *Macro Diversity HO (MDHO)* and *Fast BS Switching (FBSS)*. The MDHO essentially is *soft HO*. An MS can communicate with multiple BSs at the same time. The list of active BSs for the MS is maintained in the *Diversity Set*. In FBSS, however, the MS can only communicate with one BS of the diversity set at any given time. The MS uses a fast switching technique to change the serving BS dynamically to improve link quality.

The 802.16e standardizes the MAC layer HO only. We refer to this as L2 HO. To build a complete system, higher layers are still necessary. As mentioned earlier, MIP is chosen by WiMAX Forum to deal with the mobility management in the network layer. We refer to this as Layer-3 (L3) HO. In mobile WiMAX, when an MS leaves its home network, the HA tunnels packets to the MS's current anchor ASN GW. The ASN GW, which is essentially the FA, then further tunnels the packets to the MS. IP in IP Tunneling (IETF RFC 1853) or Generic Routing Encapsulation (GRE, IETF RFC 1701) can be use as the L3 tunneling protocol.

When an MS moves from one BS to another BS, the MS needs to perform L2 HO. If the serving BS and target BS belong to different IP subnets, the MS needs to acquire an NCoA. The MS needs to perform L3 HO as well. That is, the MS needs to register its CoA with the HA. In addition to performing two HO procedures, IP packets in L3 must be encapsulated into MAC frames[2] in L2. Over the air link, IP packets must be encapsulated into 802.16 MAC frames.

4.2 Problem Statement

Many intradomain mobility management protocols have been proposed to take advantage of the hierarchy of tree network topology. However, suboptimal problem [12] may happen in a nontree network topology. Besides, most of the protocols operate on or above the IP layer. They do not address link-layer mobility. Interaction between the mobility management in IP and link layers is not considered either. Especially, IPv4 over the Ethernet-like networks mainly rely on *ARP* (IETF RFC 826) to associate IP address with link-layer address. The broadcast-and-reply nature of ARP wastes bandwidth and causes extra latency. Although cache memory can reduce the use of ARP, it is still inefficient when an MS moves frequently. ARP may seriously waste bandwidth in a highly mobile wireless environment.

An Ethernet-like link model is most likely to be deployed in the backbone of today's WiMAX networks. In WiMAX, same as other 802-series standards, ARP must be used for address resolution in IPv4/Ethernet CS. The IPv4 address in the network layer must be associated with the Ethernet address in the link layer [5], [6], [20]. As one can see, an MS may need to perform two HO procedures in two different layers during each HO. For packet delivery, ARP may be executed in each wired and wireless link, which would significantly increase the packet delivery delay. Although each protocol has been well designed, building a complete system by simply stacking up different protocols together would result in poor performance. Also, if not filtered, ARP messages may wake up the MSs in sleep/idle mode [6]. Therefore, we propose a cross-layer design, which considers the interactions between IP layer and link layer. The proposed paging extension is also compatible with the energy saving mechanism defined in 802.16e.

4.3 Cross-Layer Design

Based on our previous paper [21], we utilize a two-level mobility management technique for fast HO. MIP is used for interdomain (inter-CSN) mobility management. The proposed FINCH is used for intradomain (intra-CSN) mobility management. Besides, FINCH handles the HOs in both IP layer and link layer. As a generic protocol, FINCH deals with *location update* in the link layer and cooperates with the L2 *HO* procedure. That is, an MS performs the HO procedure specified in the L2 standards first, which could be 802.16e or other standards. After that,

---

[2] We use *frame* and *packet* to represent the *Packet Data Unit (PDU)* in layer 2 and layer 3, respectively.

TABLE 1
Example of the FT in BS 1

MS MAC Address	MS IP Address	Forwarding MAC Address	Wireless Port	Time Stamp
08:00:20:03:F6:42	10.1.1.10	NULL	1	2007.02.03. 16.21.32.18
20:EE:18:3A:12:7E	10.1.2.20	82:10:10:04:5A:79	NULL	2007.02.03. 16.21.52.08
…	…	…	…	…

FINCH uses a special table-lookup technique for both link layer and IP layer to update the location. Based on the table, location updates in the link layer and IP layer are coupled together. Consequently, ARP is no longer necessary.

Comparing with those using two different mobility management protocols in the link layer and IP layer, the proposed scheme can reduce the overhead and latency significantly. Because the proposed mobility management protocol is compatible with the IP layer, it can work with any protocols and applications in higher layers. FINCH is particularly suitable for real-time applications such as mobile voice-over-IP (VoIP), which requires fast HO.

Fig. 1 shows a generic network architecture for inter-domain mobility and intradomain mobility, in which a domain (or a CSN) may be configured in many different ways, such as a bus network, a ring network, a star network, a tree network, and so forth. There are different network components including HAs, ASN GWs, BSs, routers, and bridges. The functionality of the network components can be either like an L3 router or an L2 bridge. The mobility management and packet routing within the domain are done by replacing the necessary routing table and bridging table with a *Forwarding Table (FT)*. It is assumed that each terminal (mobile and stationary) in Fig. 1 is an IP terminal and has at least one unique IP address. For an MS, this can be the Home Address (HoA). The MAC address is also unique. Because we focus only on IPv4 networks, the duplication of IP or MAC address can be detected by some techniques, which is outside the scope of this paper. The FT is exemplified in Table 1, which represents the FT in BS 1 in Fig. 1. As shown in the table, there are five fields in each FT:

1. the MAC address of an MS,
2. the IP address (HoA or permanent address) of the MS,
3. the forwarding MAC address to which the IP packets destined to the MS should be forwarded to,
4. the wireless port if the FT is maintained by a BS, which the MS can communicate directly by a wireless port (otherwise, the field is NULL), and
5. the time stamp copied from the original packet sent by the MS.

The detailed algorithms are depicted in the Appendix.

4.4 Packet Forwarding

We assume that all nodes have L3 functionality, i.e., they are capable of processing IP packets. If a BS does not support IP routing, we assume that there is an Access Router (AR) behind the BS to handle IP packet routing. Remember that Table 1 represents the FT in BS 1 in Fig. 1. Table 1 depicts that any IP packet arriving at BS 1 and bound for MS 1, as identified by its IP address 10.1.1.10,[3] is not forwarded to any other node. The IP packet is transmitted directly over a wireless port (e.g., wireless port 1 as shown in Table 1). Therefore, the first column of *MS MAC address* in the table is the destination MAC address for the MAC frame in the link layer. On the other hand, any IP packet received by BS 1 destined for MS 2, with IP address 10.1.2.20, should be encapsulated into a MAC frame and forwarded to the ASN GW 1 (assuming there is a direct connection between BS 1 and ASN GW 1). By looking up the FT, one can associate the IP address 10.1.2.20 in column two with the MAC address 82:10:10:04:5A:79 in column three, which is the MAC address of the ASN GW 1. Therefore, the MAC frame will be forwarded to the ASN GW 1. When the MAC frame reaches ASN GW 1, same as the process in the standard IP protocol stack, it is decapsulated back to the IP packet. By looking at the FT in ASN GW 1, the IP packet will be encapsulated into a MAC frame and forwarded to next node, which may be a router in CSN 1. By repeating the same process, the IP packet will reach MS 2 eventually. If a node is like an L2 bridge, it simply uses column one instead of column two for address lookup when an L2 frame comes in. In this case, there is no need to use column two in the FT. The forwarding is similar to the aforementioned discussion for L3. In the proposed scheme, an L3 node can also choose not to do the conversion between IP packet and MAC frame if the packet does not need any processing in L3. The algorithm is presented in Algorithm 1 in the Appendix.

4.5 Reducing ARP Messages

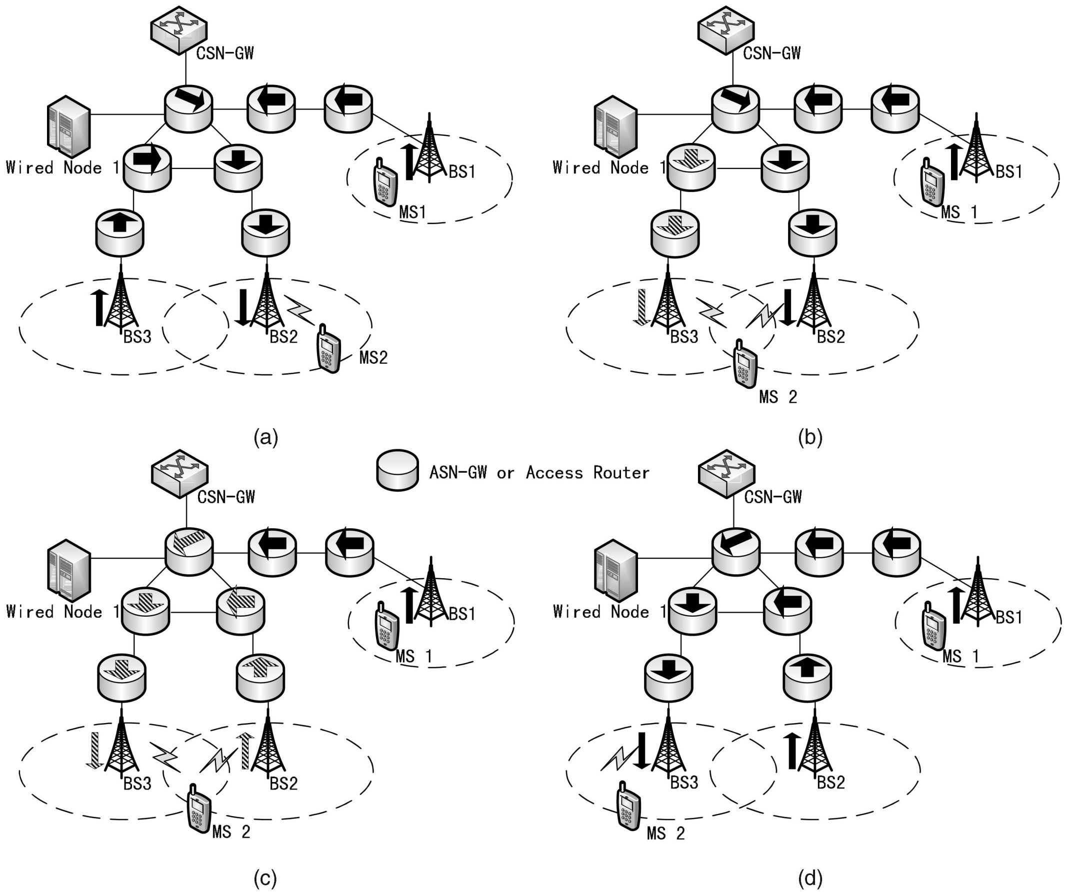
As mentioned earlier, ARP is used in WiMAX to associate the IP address in the network layer with the MAC address in the link layer [5], [20]. ARP may be executed in each wired and wireless link. However, the broadcast-and-reply nature of ARP wastes bandwidth and will cause extra latency. It may also congest the network. Therefore, techniques such as caching the mapping of IP address and MAC address in memory have been deployed. In a highly mobile environment, the broadcast-and-reply ARP may still be necessary because the mapping for an MS may not be in the cache yet, or the mapping in cache may not reflect the correct information. With the proposed FINCH, the ARP is replaced by simply looking up the FT. The IP address field is searched whenever an IP packet comes in. The IP packet is then encapsulated into a MAC frame with either the *Forwarding MAC Address* in column three or the *MS MAC Address* in column one as the destination MAC address. Comparing with ARP, the FT lookup is simple and fast.

4.6 Handover and Location Update

The packet forwarding described in Section 4.4 relies on the correctness of FT. In order to forward packets to an MS successfully, the FT should be properly updated each time

---

[3] The IP address 10.1.1.10 is just an example. One can replace it with a public IP address.



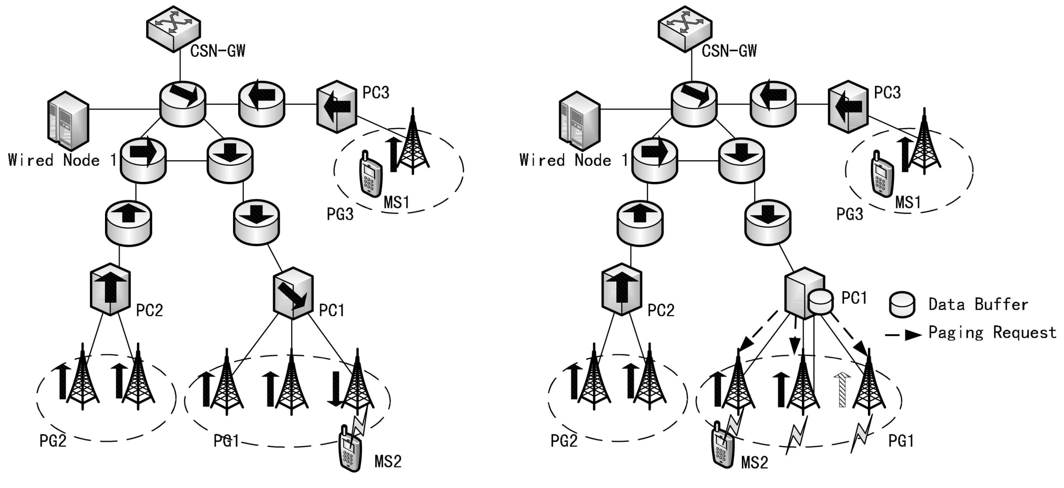
(a) (b) (c) (d)

Fig. 2. Example of address propagation in FINCH when MS 2 hand overs from BS 2 to BS 3.

when an MS moves. This section discusses the HO and location update.

The HO procedure in 802.16e has been presented in Section 4.1. In Fig. 1, assume that initially MS 1 is communicating with BS 1. When MS 1 moves away from BS 1, at a certain point in its movement, MS 1 will hand over to a new BS. Assume that BS 2 in Fig. 1 is the new BS. Once MS 1 receives the permission from BS 2 to associate with, MS 1 will send a MAC frame to BS 2 to update the FT entry of the MS. BS 2 then replicates and forwards the MAC frame to all adjacent nodes including BSs, routers, and other nodes in the same domain. All nodes that received the MAC frame also replicate and forward the MAC frame to their adjacent nodes in the same domain. The payload of the MAC frame carries the frame generation time (time stamp), which indicates what time the original MAC frame was generated in the MS. The MS's MAC address, which is used to search FTs to find the corresponding entry of the MS, is also carried. The header of the MAC frame contains, of course, the source MAC address of the node, which replicates and forwards this frame. When the replicated frames arrive at a node, the node checks the MS's time stamp entry in the FT. If the time stamp field in the FT is different with the one in the frame, it is checked on the forwarding MAC address field. If the forwarding MAC address is different from the source MAC address, the fields of the forwarding MAC address and the time stamp are then updated with the new records. Thus, all frames that arrive at this node have a new route to reach the MS. If the time stamp is the same as the one in the frame or the forwarding MAC address field is same as the source MAC address in the frame header, no update is necessary and no more frame will be replicated from this node. The forwarding of frames is terminated at this point because this node has already been visited by a frame, or it is not necessary to change the forwarding MAC address to reflect a new route to the MS. Thus, the forwarding MAC address in FT accurately reflects the current location of the MS. When the old BS receives the replicated frame, it updates the forwarding MAC address and the wireless port. The termination technique eliminates excessive frame duplication and forwarding but still ensures that all nodes in the network have the current forwarding address of the MS. As the frames propagate, new route(s) through the network can be established to reach the MS. Algorithms 2 and 3 in the Appendix present the location update in the new BS and other devices.

Fig. 2 illustrates an example of how the address propagation technique operates. The network architecture shown in Fig. 2 is merely exemplary and not limiting. By way of notation, the *arrows* in the various nodes show the *forwarding direction* of the node for a data packet destined for MS 2. In this example, the cylinders represent either ASN GWs or ARs. Also, there can be some wired nodes such as an AAA server connecting to the WiMAX network. Fig. 2a depicts the routing directions before MS 2 starts to



(a)

(b)

Fig. 3. Example of FINCH paging extension.

hand over from BS 2 to BS 3. Figs. 2b and 2c present the propagation of the location update message. It can be found that the location update procedure ends at the *crossover* nodes between BS 2 and BS 3, as shown in Fig. 2c. After the HO procedure finishes, Fig. 2d shows the routing directions of the network. Depending on the network propagation speed, network topology, and location of the CH, there might be delay or small amount of packet loss in hard HO. This can be recovered by other techniques. In terms of soft HO, Fig. 2b shows that the MS can receive packets from both BSs before the address propagation is done. After the old BS receives the packet propagated over the network in Fig. 2c, it changes its corresponding entry from the wireless port to the forwarding MAC address in its FT. The old BS then realizes that the MS has handed over to a new BS successfully. Transmissions between the MS and the old BS thus are terminated.

4.7 Characteristics

In the proposed FINCH, there is no need to reroute the individual path for each source node sending packets to the same MS. All source nodes connecting to the network either by fixed or wireless network know the new location of the MS after the location update is complete. The triangular routing in MIP is eliminated, and therefore there is no need for route optimization. In addition, the new forwarding address is updated by the first arrived frame, which implies that the new path may be the one with least congestion or shortest path. The new route pointed to by the forwarding address is the fastest path in the current network condition. Also, the proposed protocol can be used for location update at both link layer and IP layer when MS moves. HOs at both layers can be done by a single protocol. Therefore, it is not necessary to trigger L3 HO after L2 HO is done if it is an across IP subnet HO. In order to attach to a new subnet, the MS needs to get an IP address, such as CoA, for that subnet. This can be done by, for example, using Dynamic Host Configuration Protocol (DHCP, IETF RFC 2131). The proposed scheme deals with HOs and location update and leaves address configuration to other protocols. Although the proposed protocol does not use CoA in FT, CoA is necessary for MIP for interdomain mobility. Unlike other protocols, there is no centralized gateway. The proposed protocol is a distributed protocol and can be used in any network topology. It can also be used in other IP and 802-series networks.

One drawback of the proposed protocol is that it breaks the layer structure by combining the functionality in both link and IP layers. However, it eliminates the redundancy of doing two HOs and location updates in both link and IP layers. The other possible limit of the protocol is that the FT needs to maintain a list of all terminals in the domain. Each domain, however, does not expect to support millions or billions of terminals. By carefully partitioning the domain, the scalability problem should not be a major concern. This has been studied in other intradomain mobility management protocols. The table searching can also be implemented in hardware.

4.8 Paging Extension

In this section, we introduce an optional paging extension, *P-FINCH*, for the proposed FINCH to enhance the energy efficiency and minimize the signaling overhead in location update.

To support paging in mobile WiMAX, each BS is assigned to a *Paging Group (PG)* [5]. The assignment may be based on geographical or load-balancing considerations. Each PG has a unique *PG Identifier (PGI)*. In each PG, there is a *Paging Controller (PC)*. An MS will enter *Idle Mode* if it has no traffic to transmit or receive for a predefined period of time. Therefore, an idle MS does not need to perform location update every time when it moves across the cell boundary. Alternatively, an idle MS performs location update only when it moves to another PG. An idle MS can also perform location update periodically [5].

We use Fig. 3 as an example to illustrate the paging support in FINCH. Assuming that MS 2 is located in PG 1 initially and the original forwarding directions are shown in Fig. 3a. When MS 2 intends to enter *Idle Mode*, it first sends a specific *Deregistration* message to the BS to initiate the *Idle Mode*. The BS then sends an *IM-Entry_MS_State_Change* *request* [5] to *PC 1*. PC 1 then registers MS 2 to a *paging list* and sends out the location update message on behalf of MS 2. After that, all of the packets destined to MS 2 will be forwarded to PC 1 as depicted in Fig. 3b. Once PC 1 receives packets destined to MS 2, PC 1 should buffer the packets. When PC 1 wants to awake MS 2 and deliver the buffered packets to MS 2, it will send *paging request* to all of the BSs within PG 1. The paging request will further be broadcasted by the BSs. After MS 2 receives the paging request, MS 2 wakes up and sends out location update message. When the location update message is forwarded to PC 1, PC 1 will clean up the entry of MS 2 in the paging list and start to forward the buffered packets to MS 2. If MS 2 hand overs to PG 2 when it is still in Idle Mode, MS 2 must send a *location update* message to PC 2 to update its current location. PC 2 then will send out normal location update message on behalf of MS 2. Finally, packets destined to MS 2 will be forwarded to PC 2.

Although we do not describe the signaling messages in details, the paging extension is general enough for most wireless systems. The proposed P-FINCH is also compatible with the mobile WiMAX standards. Because signaling messages are localized within the coverage of the PC if an MS does not move out of the range of the same PG, P-FINCH can significantly reduce signaling overhead and efficiently conserve the energy of MSs. Also, except PCs and BSs, other network nodes operate similarly as the original FINCH nodes without any change. Therefore, the paging extension is easy to implement. Moreover, because paging is initiated by each PC instead of a centralized paging initiator, the proposed P-FINCH is scalable. Furthermore, comparing with the protocols that have only one single node to buffer packets, our paging design is also more robust because packets are buffered by each PC.

5 PERFORMANCE ANALYSIS

In this section, we analyze the performance of the proposed FINCH. Part of the analysis follows the derivation in [15]. For comparison, MIP, F-MIP, HMIP [9], Cellular IP [10], and HAWAII[4] [11] are analyzed as well. As discussed in Section 2, there are two categories of intradomain mobility management protocols. HMIP is categorized as a *tunnel-based* protocol. Cellular IP and HAWAII belong to *host-specific-routing-based* category. Also, MIP is analyzed to compare intradomain mobility management with the original mobility management in mobile WiMAX.

In Section 5.1, we first analyze the HO latency and number of lost packets during HO. Because location update, which is also referred to as registration, is essential for mobility management protocols, Section 5.2 derives the location update cost. Section 5.3 then analyzes the overall cost. The analysis of energy consumption is presented in Section 5.4.

5.1 Analysis of Handover Latency and Packet Loss

The *HO latency* is defined as the time interval during which an MS cannot receive and transmit any packet due to the HO procedure. That is, it is an interval from the time the MS loses the L2 connection with the old BS until the time the MS receives or transmits packets by the new IP address through the new BS.

The *HO latency* consists of *L2 switching delay*, *IP connectivity latency*, and *location update latency* [22], [23]. Therefore, we have

$$D_{HO} = D_{L2} + D_{IP} + D_{LU}, \tag{1}$$

where the parameters are defined as follows:

- D_{HO} is the HO latency.
- D_{L2} is the L2 link switching delay. This delay reflects the 802.16e L2 HO latency.
- D_{IP} is the IP connectivity latency. IP connectivity latency includes the duration of IP layer movement detection, IP address acquisition, and configuration for the MS.
- D_{LU} is the location update latency. It is composed of the latency for binding update and the latency to forward packets to the new IP address.

The L2 HO latency D_{L2} is essential for any IP mobility management protocol. For D_{IP}, although the time required to perform the necessary operations may not be reduced, D_{IP} may be shortened if L3 HO is triggered early, such as that in the *predictive* mode of F-MIP. On the other hand, D_{LU} mainly depends on the length of the path for location update messages. It is observed that micromobility management protocols primarily intend to minimize D_{LU} to reduce the HO latency.

In MIP, D_{LU} includes the time for transmitting the registration message to the HA and then tunneling the packet to the NCoA. When transmitting the binding update for the first time, ARP is essential for address resolution. Therefore, the HO delay of MIP is derived as

$$\begin{aligned} D_{HO,MIP} &= D_{L2} + D_{IP} + D_{LU} \\ &= D_{L2} + D_{IP} + t_{AR} + 2 \times t_{HA-MS} + t_{MIP}, \end{aligned} \tag{2}$$

where t_{AR} is the latency of address resolution, t_{HA-MS} is the one-way delay for packet transmission between HA and MS, and t_{MIP} is the overall computation delay of MIP. Thus, the time required from transmitting a binding update until receiving the first packet at the new address is $2 \times t_{HA-MS}$.

Packets are lost before an MS register its NCoA with the HA. The number of lost packets during HO in MIP is

$$L_{HO,MIP} = \lambda_p \times (D_{HO,MIP} - t_{HA-MS}), \tag{3}$$

where λ_p is the packet arrival rate.

In F-MIP *predictive* mode, an MS can send FBU before L2 HO. The NCoA can be configured before HO. Therefore, D_{IP} can be eliminated. An MS can receive the packets buffered in NAR after sending FBU to the NAR. That is, the MS can start to receive packets without the delay of registration, D_{LU}. The HO latency is

$$D_{HO,F-MIP-Pre} = D_{L2} + t_{AR} + 2 \times t_{NAR-MS}, \tag{4}$$

where t_{NAR-MS} represents the one-way delay for packet transmission between NAR and MS. After L2 HO, the MS transmits FBU to the NAR to access the buffered packets. Therefore, it takes a round-trip latency between NAR and MS to receive the packets.

---

[4] More specifically, we only consider the HAWAII UNF mode [11].

The packet loss during HO in F-MIP is disengaged from the HO latency incurred by L2 and L3. In *predictive* mode, packets are lost if the coordination of fast HO signaling is not correct or packets are failed to be buffered in NAR or PAR. Therefore, if the signaling is correct, the number of lost packets is

$$L_{HO,F-MIP-Pre} = \max\{0, \lambda_p \times (D_{HO,F-MIP-Pre} - t_{NAR-MS}) - Buffer_{NAR}\}, \tag{5}$$

where $Buffer_{NAR}$ denotes the packet buffer size of NAR.

In *reactive* mode of F-MIP, an MS cannot receive the FBack before L2 HO. Therefore, the IP layer movement detection is still needed. However, an MS can transmit FBU with the previous CoA (PCoA) to receive packets. That is, the IP address configuration may be reduced. Therefore, HO latency can be derived as

$$D_{HO,F-MIP-Re} = D_{L2} + t_{IP-mv} + t_{AR} + 3 \times t_{PAR-NAR} + 2 \times t_{NAR-MS}, \tag{6}$$

where t_{IP-mv} is the latency for IP layer movement detection. $t_{PAR-NAR}$ represents the one-way delay for packet transmission between PAR and NAR. Because the tunnel between PAR and NAR must be established before packet forwarding, an MS first sends FBU to the NAR after L2 HO. The delay is t_{NAR-MS}. After that, there is an additional round-trip delay between PAR and NAR to exchange FBU and FBack. The delay is $2 \times t_{PAR-NAR}$. Finally, the PAR forwards packets to the MS through NAR. The latency is $t_{PAR-NAR} + t_{NAR-MS}$.

In *reactive* mode of F-MIP, packets are lost in PAR before setting up the forwarding link between PAR and NAR. The packet loss is derived as

$$L_{HO,F-MIP-Re} = \lambda_p \times (D_{L2} + t_{IP-mv} + t_{AR} + t_{NAR-MS} + t_{PAR-NAR}). \tag{7}$$

Next, we analyze the HO latencies of the micromobility management protocols. FINCH is a cross-layered design for micromobility management. The address resolution latency is eliminated. Therefore, the HO latency is

$$D_{HO,FINCH} = D_{L2} + D_{IP} + 2 \times t_{CR-MS} + t_{Mi-H}, \tag{8}$$

where t_{CR-MS} is the one-way delay for packet transmission between the crossover node and the MS. t_{Mi-H} is the computation delay of the host-specific-routing-based micromobility management protocols.

In FINCH, packets are lost when the location update messages are still propagating to the crossover node. The packet loss is

$$L_{HO,FINCH} = \lambda_p \times (D_{HO,FINCH} - t_{CR-MS}). \tag{9}$$

In Cellular IP and HAWAII, location update is only needed between the crossover node and the MS. Therefore, the HO latency is

$$D_{HO,CIP} = D_{HO,HAWAII} = D_{L2} + D_{IP} + t_{AR} + 2 \times t_{CR-MS} + t_{Mi-H}. \tag{10}$$

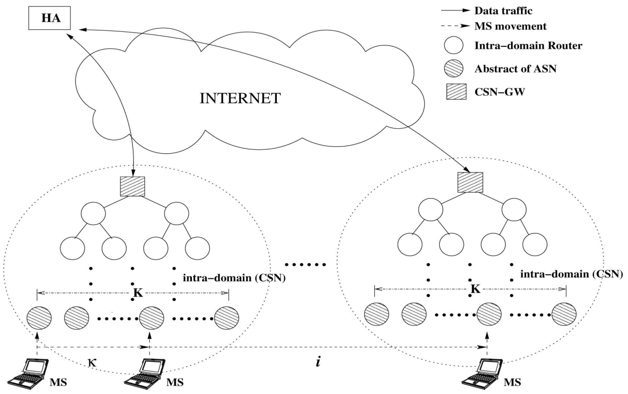


Fig. 4. Network topology for performance analysis.

The packet loss during CIP and HAWAII intradomain HO is

$$L_{HO,CIP} = L_{HO,HAWAII} = \lambda_p \times (D_{HO,CIP} - t_{CR-MS}). \tag{11}$$

In HMIP, packets need to be decapsulated and reencapsulated at the GFA. Therefore, the HO latency is

$$D_{HO,HMIP} = D_{L2} + D_{IP} + t_{AR} + 2 \times t_{CR-MS} + t_{Mi-T}, \tag{12}$$

where t_{Mi-T} is the computation delay of the tunnel-based micromobility management protocols. We include the additional decapsulate and reencapsulate delay in this variable.

The packet loss during the HMIP intradomain HO is

$$L_{HO,HMIP} = \lambda_p \times (D_{HO,HMIP} - t_{CR-MS}). \tag{13}$$

5.2 Analysis of Location Update Cost

In this section, we analyze the signaling cost introduced by location update. We assume that the network topology is configured as that shown in Fig. 4. It is generally agreed that it is difficult to analyze a network mathematically with randomly connected topology. One can observe that intradomain mobility management protocols mostly take advantage of the tree-based hierarchical network topology. In addition, some protocols can only work in tree-based topology. We therefore assume that the network topology is configured as a binary tree. The network architecture is defined with several CSNs that adopt an intradomain mobility management protocol. Each domain (or CSN) is connected as a complete binary tree with K leaf nodes. Because this analysis mainly discusses the costs within the CSNs, we abstract the ASNs as leaf nodes of the tree. Therefore, the K leaf nodes also represent K ASNs. If intradomain mobility is enabled, an MS may use an intradomain mobility management protocol within a CSN domain for intradomain HOs. MIP is adopted for interdomain HOs. Otherwise, if only MIP is used, each leaf node is configured as an MIP FA as that defined in mobile WiMAX for ASN GW.

An MS is assumed to move i steps from the κth node (ASN) of the domain. As in the derivations in [15], the periodic binding update costs that MS refresh HA, FA, and routers in CSN are not considered. The following parameters are defined for the analysis of location update cost:

- ρ is the MS call-to-mobility ratio (CMR). CMR is defined as $\rho = \lambda/\mu$, where λ denotes the call arrival rate. The mean residence time that MS stays in a cell is $1/\mu$ second.
- U is the average cost of location update to its HA in MIP. The cost here can be the delay of the signaling messages, which includes propagation delay and transmission delay.
- S is the cost for setting up a single link when the intradomain mobility management protocol sets up the path in the intradomain.
- A is the cost of address resolution, which includes the delay of ARP operations.
- V is the cost of a unicast signaling message between the PC and the BS. This cost is used for location update from the BS to the PC, and the paging request from the PC to the BS.
- L is the cost for setting up the direct connection between the NAR and PAR in F-MIP.

Let the probability that an MS moves i steps between two consecutive packet arrivals be $\alpha(i)$. In MIP, assuming that all ASN GWs are FAs, the location update cost can be derived as

$$C_{MIP}(\rho) = \sum_{i=0}^{\infty} iU\alpha(i) = \frac{U}{\rho}. \tag{14}$$

F-MIP reduces the HO latency by forwarding the packets from PAR to NAR. However, there is an additional signaling cost to set up the direct connection between PAR and NAR during each HO. The cost for location update is derived as

$$C_{F-MIP}(\rho) = \sum_{i=0}^{\infty} i(U+L)\alpha(i) = \frac{U+L}{\rho}. \tag{15}$$

Based on the network topology depicted in Fig. 4, we first derive $\Phi(n)$, the number of hops for a location update message when an MS moves from the first cell to the $n\text{th}$ cell within the complete binary tree. To derive the number of hops that a location update message traverses when an MS hand overs between two consecutive cells is equivalent to finding the height of the *crossover node* or the *lowest common ancestor problem* [24]. Therefore, the total number of hops for a location update message, $\Phi(n)$, when an MS moves from the first cell to the $n\text{th}$ $(n \geq 2)$ cell, can be derived as

$$\Phi(n) = \sum_{j=2}^{n} \lfloor \log((2j-1) \oplus (2j-3)) \rfloor, \tag{16}$$

where $\oplus$ represents *binary-exclusive-OR* (bit-XOR).

The packet arrivals are assumed to be a Poisson process with rate λ. The cell residence time is assumed to be a random variable with a general density function $f_m(t)$ with the Laplace transform g:

$$g = f_m^*(\lambda) = \int_{t=0}^{\infty} f_m(t)e^{-\lambda t}\, dt = g. \tag{17}$$

Therefore, $\alpha(i)$ can be derived as follows:

$$\alpha(i) = \begin{cases} 1 - \frac{1-g}{\rho}, & \text{if } i = 0, \quad (18) \\ \frac{(1-g)^2 g^{i-1}}{\rho}, & \text{if } i > 0. \quad (19) \end{cases}$$

Because an MS can move across several intradomains, let $i = jK + q$ and $0 \leq q < K$. Then,

$$\alpha(jK+q) = \frac{(1-g)^2}{\rho g}(g^K)^j g^q = yz^j x^q, \tag{20}$$

where $y = \frac{(1-g)^2}{\rho g}$, $z = g^K$, and $x = g$.

We now consider the location update cost of intradomain mobility management protocols. The cost consists of MIP registrations for interdomain HOs and location update messages for intradomain HOs within the CSN. Note that because Cellular IP, HAWAII, and HMIP are IP-based protocols, the ARP cost must be considered in their location update procedures. Moreover, intradomain mobility management protocols mostly combine the inter and intradomain registration procedures to reduce the signaling cost. When receiving the intradomain registration, the gateway node or FA then transforms it into interdomain registration. Therefore, in the analysis, we assume that MS can perform the registration of intradomain mobility by using the same message of MIP registration.

First, we derive the location update cost of the proposed FINCH. When an MS enters a new CSN, it should perform intradomain registration, which is combined with the MIP registration. Therefore, when traversing i cells from the κth cell, the MS will perform $\lfloor \frac{i+\kappa}{K} \rfloor$ registrations. Moreover, the location update of intradomain mobility traverses from the new BS to the original BS. Totally, the location update message will update $2 \times [\Phi(K) \cdot \lfloor \frac{i+\kappa}{K} \rfloor + \Phi(i+\kappa-\lfloor \frac{i+\kappa}{K} \rfloor \cdot K) - \Phi(\kappa)]$ hops. The signaling cost of location update of FINCH, C_{FINCH}, is derived as

$$\begin{aligned} &C_{FINCH}(\kappa, K, \rho) \\ &= \sum_{i=0}^{\infty} \Bigg\{ \left\lfloor \frac{i+\kappa}{K} \right\rfloor U + \left\lfloor \frac{i+\kappa}{K} \right\rfloor \left(\sum_{l=1}^{\log(K)} 2^l - log(K) \right) S \\ &\quad + \left[\Phi(K) \cdot \left\lfloor \frac{i+\kappa}{K} \right\rfloor + \Phi\left(i + \kappa - \left\lfloor \frac{i+\kappa}{K} \right\rfloor \cdot K \right) \right. \\ &\quad \left. - \Phi(\kappa) \right] 2S \Bigg\} \cdot \alpha(i) \\ &= (U + \Phi(K)2S + (2K - 2 - \log(K))S) \\ &\quad \times \frac{(1-g)g^{K-1}}{\rho(1-g^K)g^{\kappa}} + \frac{2S(1-g)^2}{\rho g(1-g^K)} \\ &\quad \cdot \left[\sum_{q=0}^{K-\kappa-1} \Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1} \Phi(q+\kappa-K)g^q \right] - \Phi(\kappa)2S. \end{aligned} \tag{21}$$

The derivation of HAWAII is similar to FINCH except that the cost of ARP should be included. Besides, an MS only needs to register with the *Domain Root Router (DRR)* [11] by using the same message of MIP registration. The cost for flooding intradomain registration is eliminated. The cost is derived as

$$\begin{aligned}
&C_{HAWAII}(\kappa, K, \rho)\\
&= \sum_{i=0}^{\infty} \left\{ \left\lfloor \frac{i+\kappa}{K} \right\rfloor U + \left[\Phi(K) \cdot \left\lfloor \frac{i+\kappa}{K} \right\rfloor + \Phi\left(i+\kappa - \left\lfloor \frac{i+\kappa}{K} \right\rfloor \cdot K \right) \right.\right.\\
&\qquad\qquad \left.\left. - \Phi(\kappa) \right] 2(S+A) \right\} \cdot \alpha(i)\\
&= (U+\Phi(K) \cdot 2(S+A)) \frac{2(1-g)g^{K-1}}{\rho(1-g^K)g^{\kappa}} + \frac{(1-g)^2(S+A)}{\rho g(1-g^K)}\\
&\quad \cdot \left[\sum_{q=0}^{K-\kappa-1} \Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1} \Phi(q+\kappa-K)g^q \right]\\
&\quad - \Phi(\kappa) \cdot 2(S+A).
\end{aligned} \tag{22}$$

Because the registration and location update of both CIP and HMIP act similarly as those in HAWAII, the location update costs are the same as that in HAWAII:

$$C_{HMIP}(\kappa, K, \rho) = C_{CIP}(\kappa, K, \rho) = C_{HAWAII}(\kappa, K, \rho). \tag{23}$$

To demonstrate that the paging extension of FINCH could reduce the overhead of location update, we construct the analytical model for P-FINCH based (21). Although an MS still needs to perform both MIP and intradomain location updates when entering a CSN, the MS only needs to update its location while moving into a new PG. We assume the PC is collocated with the root node of the PG. Each PG is assumed to contain P cells $(P = 2^n, n \geq 1)$. Therefore, the location update cost consists of the normal location update cost originated by the PC and the unicast location update cost from BS to the PC. The location update cost is then derived as follows:

$$\begin{aligned}
&C_{P-FINCH}(\kappa, K, \rho)\\
&= \sum_{i=0}^{\infty} \left\{ \left\lfloor \frac{i+\kappa}{K} \right\rfloor U + \left\lfloor \frac{i+\kappa}{K} \right\rfloor \left(\sum_{l=1}^{\log(K)} 2^l - log(K) \right) S \right.\\
&\qquad + \left[\Phi\left(\frac{K}{P}\right) \cdot \left\lfloor \frac{i+\kappa}{K} \right\rfloor + \Phi\left(\left\lceil \frac{(i+\kappa)\ mod\ K}{P} \right\rceil \right) \right.\\
&\qquad \left. - \Phi\left(\left\lceil \frac{\kappa}{P} \right\rceil \right) \right] \cdot 2S + \left[\left(\frac{K}{P} - 1 \right) \cdot \left\lfloor \frac{i+\kappa}{K} \right\rfloor \right.\\
&\qquad \left.\left. + \left\lceil \frac{(i+\kappa)\ mod\ K}{P} \right\rceil - \left\lceil \frac{\kappa}{P} \right\rceil \right] \cdot V \right\} \cdot \alpha(i)\\
&= \left[U + (2K-2-\log(K)) \cdot S + \Phi\left(\frac{K}{P}\right) \right.\\
&\quad \left. \cdot 2S + \left(\frac{K}{P} - 1 \right) \cdot V \right] \cdot \frac{(1-g)g^{K-1}}{\rho(1-g^K)g^{\kappa}}\\
&\quad + \left[2S \cdot \sum_{q=0}^{K-\kappa-1} \Phi\left\lceil \frac{q+\kappa}{P} \right\rceil g^q + 2S \cdot \sum_{q=K-\kappa}^{K-1} \Phi\left\lceil \frac{q+\kappa-K}{P} \right\rceil g^q \right.\\
&\qquad \left. + V \cdot \sum_{q=0}^{K-\kappa-1} \left\lceil \frac{q+\kappa}{P} \right\rceil g^q + V \cdot \sum_{q=K-\kappa}^{K-1} \left\lceil \frac{q+\kappa-K}{P} \right\rceil g^q \right]\\
&\quad \cdot \frac{(1-g)^2}{\rho g(1-g^K)} - \Phi\left(\left\lceil \frac{\kappa}{P} \right\rceil \right) \cdot 2S - \left\lceil \frac{\kappa}{P} \right\rceil \cdot V.
\end{aligned} \tag{24}$$

To simplify our analytical model, we assume that κ is uniformly distributed with probability $\frac{1}{K}$. That is, $C(K, \rho) = \frac{1}{K}\sum_{\kappa=0}^{K-1} C(\kappa, K, \rho)$. Equation (21) then is rewritten as

$$\begin{aligned}
&C_{FINCH}(K, \rho)\\
&= \frac{U + 2\Phi(K)S + (2K-2-\log(K))S}{\rho K} + \frac{2S(1-g)^2}{\rho g(1-g^K)K}\\
&\quad \times \sum_{\kappa=0}^{K-1} \left[\sum_{q=0}^{K-\kappa-1} \Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1} \Phi(q+\kappa-K)g^q \right]\\
&\quad - \frac{\sum_{\kappa=0}^{K-1} \Phi(\kappa)}{K} \cdot 2S.
\end{aligned} \tag{25}$$

Moreover, (22) is rewritten as

$$\begin{aligned}
&C_{HAWAII}(K, \rho)\\
&= C_{CIP}\\
&= C_{HMIP}\\
&= \frac{U + \Phi(K)2(S+A)}{\rho K} + \frac{2(S+A)(1-g)^2}{\rho g(1-g^K)K}\\
&\quad \times \sum_{\kappa=0}^{K-1} \left[\sum_{q=0}^{K-\kappa-1} \Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1} \Phi(q+\kappa-K)g^q \right]\\
&\quad - \frac{\sum_{\kappa=0}^{K-1} \Phi(\kappa)}{K} \cdot 2(S+A).
\end{aligned} \tag{26}$$

Equation (24) is rewritten as

$$\begin{aligned}
&C_{P-FINCH}(K, \rho)\\
&= \left(U + (2K-2-\log(K))S + \Phi\left(\frac{K}{P}\right) \cdot 2S + \left(\frac{K}{P} - 1\right)V \right)\\
&\quad /(\rho K) + \frac{(1-g)^2}{\rho g(1-g^K)K}\\
&\quad \times \sum_{\kappa=0}^{K-1} \left[2S \cdot \sum_{q=0}^{K-\kappa-1} \Phi\left\lceil \frac{q+\kappa}{P} \right\rceil g^q + 2S \cdot \sum_{q=K-\kappa}^{K-1} \Phi\left\lceil \frac{q+\kappa-K}{P} \right\rceil g^q \right.\\
&\qquad \left. + V \cdot \sum_{q=0}^{K-\kappa-1} \left\lceil \frac{q+\kappa}{P} \right\rceil g^q + V \cdot \sum_{q=K-\kappa}^{K-1} \left\lceil \frac{q+\kappa-K}{P} \right\rceil g^q \right]\\
&\quad - \left(\sum_{\kappa=0}^{K-1} \left[\Phi\left(\left\lceil \frac{\kappa}{P} \right\rceil \right) \cdot 2S + \left\lceil \frac{\kappa}{P} \right\rceil V \right] \right) /K.
\end{aligned} \tag{27}$$

5.3 Analysis of Overall Cost

Based on the analysis of the location update cost, this section analyzes the overall cost. A mobility management protocol generally consists of *location update* and *packet delivery*. Based on the strategies adopted for location update and packet delivery, different protocols exhibit different characteristics. For example, the packet delivery in MIP and HMIP is different with that in HAWAII and FINCH. The encapsulation and decapsulation in MIP and HMIP may cost more than the forwarding in HAWAII and FINCH in terms of processing delay. Moreover, when paging is used, the paging cost must also be considered in the packet delivery cost. Therefore, in this section, we derive the overall cost, which adds up *location update cost* and *packet delivery cost*. The location update cost has been discussed in Section 5.2. The packet delivery cost here mainly considers the delay of packet transmission, routing/forwarding table lookup, and additional tunneling processing. The following additional parameters are defined for the analysis:

- M is the packet delivery cost of MIP.
- F is the packet forwarding/routing cost in the intradomain (or CSN).
- T is the additional reencapsulation and decapsulation cost of MIP, F-MIP, and HMIP.
- B is the cost for buffering packets at NAR in F-MIP.

We consider the same topology in Fig. 4 for the analysis. To analyze the overall cost of MIP, additional tunneling and delivery costs are included in the derivation:

$$T_{MIP}(\rho) = C_{MIP} + M + T = \frac{U}{\rho} + M + T. \tag{28}$$

In F-MIP, packets are delivered to PAR by HA at first. The PAR then forwards the packets to NAR. There are additional forwarding cost, reencapsulation cost, and buffering cost compared with the overall cost in MIP:

$$\begin{aligned} T_{F-MIP}(\rho) &= C_{F-MIP} + M + 2T + F + B \\ &= \frac{U+L}{\rho} + M + 2T + F + B. \end{aligned} \tag{29}$$

For FINCH, CIP, and HAWAII, HA first tunnels data packets to the FA of the serving CSN. The packets then are simply forwarded to the MS hop by hop in the domain. Therefore, the overall costs of FINCH, CIP, and HAWAII are derived as follows:

$$\begin{aligned} &T_{FINCH}(K,\rho) \\ &\quad = C_{FINCH} + M + F \\ &\quad = \frac{U + 2\Phi(K)S + (2K-2-\log(K))S}{\rho K} + \frac{2S(1-g)^2}{\rho g(1-g^K)K} \\ &\qquad \cdot \sum_{\kappa=0}^{K-1}\left[\sum_{q=0}^{K-\kappa-1}\Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1}\Phi(q+\kappa-K)g^q\right] \\ &\qquad - \frac{\sum\limits_{\kappa=0}^{K-1}\Phi(\kappa)}{K}\cdot 2S + M + F, \end{aligned} \tag{30}$$

$$\begin{aligned} &T_{HAWAII}(K,\rho) \\ &\quad = T_{CIP}(K,\rho) \\ &\quad = C_{HAWAII} + M + F \\ &\quad = \frac{U + \Phi(K)2(S+A)}{\rho K} + \frac{2(S+A)(1-g)^2}{\rho g(1-g^K)K} \\ &\qquad \times \sum_{\kappa=0}^{K-1}\left[\sum_{q=0}^{K-\kappa-1}\Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1}\Phi(q+\kappa-K)g^q\right] \\ &\qquad - \frac{\sum\limits_{\kappa=0}^{K-1}\Phi(\kappa)}{K}\cdot 2(S+A) + M + F. \end{aligned} \tag{31}$$

Nevertheless, in addition to forwarding cost, the cost of HMIP consists of additional reencapsulation and decapsulation cost incurred in GFA and the FA at the lowest level of the routing path. According to [9], the other RFAs within the hierarchy may merely change the source and destination IP addresses of the encapsulating IP header without decapsulating it. Therefore, the overall cost is derived as

$$\begin{aligned} &T_{HMIP}(K,\rho) \\ &\quad = C_{HMIP} + M + F + T \\ &\quad = \frac{U + \Phi(K)2(S+A)}{\rho K} + \frac{2(S+A)(1-g)^2}{\rho g(1-g^K)K} \\ &\qquad \times \sum_{\kappa=0}^{K-1}\left[\sum_{q=0}^{K-\kappa-1}\Phi(q+\kappa)g^q + \sum_{q=K-\kappa}^{K-1}\Phi(q+\kappa-K)g^q\right] \\ &\qquad - \frac{\sum\limits_{\kappa=0}^{K-1}\Phi(\kappa)}{K}\cdot 2(S+A) + M + F + T. \end{aligned} \tag{32}$$

Now, we calculate the overall cost of P-FINCH. When packets arrive at the PC, it must page the P cells of the whole PG to find the precise location of the MS. After the MS receives the paging message, it then replies a location update message, which needs to traverse $\log(P)$ hops to the PC before the PC begins to forward the buffered packets. Finally, the packets will be forwarded to the MS. Consequently, the overall cost of P-FINCH is derived as

$$\begin{aligned} &T_{P-FINCH}(K,\rho) \\ &\quad = C_{P\_FINCH} + P\cdot V + S\log(P) + M + F \\ &\quad = \frac{U + (2K-2-\log(K))S + \Phi\left(\frac{K}{P}\right)\cdot 2S + \left(\frac{K}{P}-1\right)V}{\rho K} \\ &\qquad + \frac{(1-g)^2}{\rho g(1-g^K)K} \\ &\qquad \times \sum_{\kappa=0}^{K-1}\left[2S\cdot\sum_{q=0}^{K-\kappa-1}\Phi\left\lceil\frac{q+\kappa}{P}\right\rceil g^q + 2S\cdot\sum_{q=K-\kappa}^{K-1}\Phi\left\lceil\frac{q+\kappa-K}{P}\right\rceil g^q\right. \\ &\qquad\qquad \left. + V\cdot\sum_{q=0}^{K-\kappa-1}\left\lceil\frac{q+\kappa}{P}\right\rceil g^q + V\cdot\sum_{q=K-\kappa}^{K-1}\left\lceil\frac{q+\kappa-K}{P}\right\rceil g^q\right] \\ &\qquad - \frac{\sum\limits_{\kappa=0}^{K-1}\left[\Phi\left(\left\lceil\frac{\kappa}{P}\right\rceil\right)\cdot 2S + \left\lceil\frac{\kappa}{P}\right\rceil V\right]}{K} + P\cdot V + S\log(P) + M + F. \end{aligned} \tag{33}$$

5.4 Analysis of Energy Conservation

Energy consumption is crucial to mobile devices. This section analyzes the energy consumption for P-FINCH. The analysis aims to quantify the energy consumption of *an MS*. From an MS's aspect, no matter which mobility management protocol is used, the energy consumption for *packet delivery* is similar. Although packets may traverse different routes in the *backbone network* by using different protocols, the energy for an MS is spent primarily on the transmission and reception of the packets over the air. Packet delivery in the backbone has nothing to do with the energy consumption of an MS. Besides, the amount of packets is mainly decided by the applications. Comparing with the total amount of packets, packet loss due to HO is minimal. From an MS's point of view, therefore, the difference in energy consumption is mainly determined by *location update*. In addition, the signaling behavior of location update over the

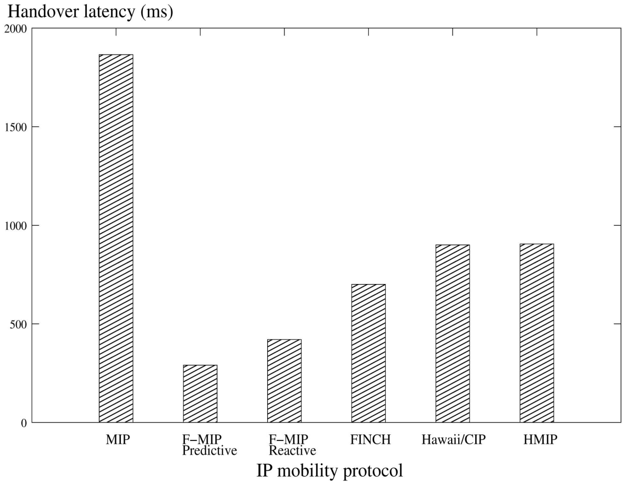


Fig. 5. Comparison of HO latency.

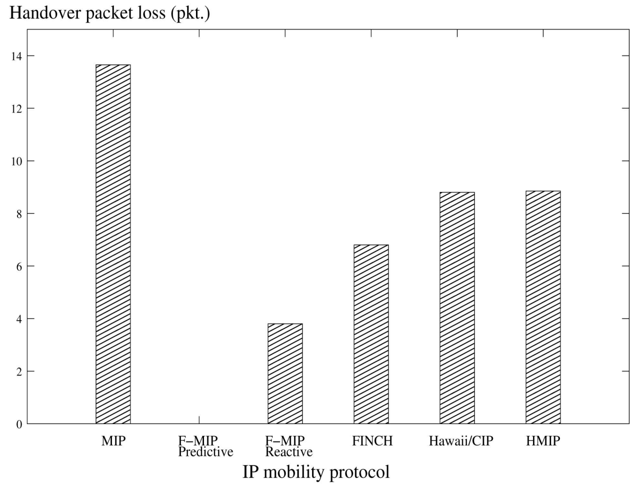


Fig. 6. Comparison of packet loss during HO.

backbone network does not affect the energy consumption of an MS. If paging is not initiated for an idle MS, the MS always transmits a location update message while crossing the cell boundary no matter the MS adopts MIP, HMIP, HAWAII, or FINCH. Therefore, we only need to compare P-FINCH with one of the protocols without paging. We compare the energy consumption of P-FINCH and FINCH with different configurations to demonstrate that P-FINCH can significantly reduce the energy consumption of an MS.

We assume that each uplink transmission consumes u units of energy. In active mode, an MS also consumes r units of energy per time unit while its receiver is turned on. On the other hand, when an MS is in idle mode, it only consumes b units of energy per time unit. As the derivations above, the periodical location update is not considered in the analysis. Therefore, the energy consumption of a FINCH node without paging extension is

$$E_{FINCH}(\rho) = \sum_{i=0}^{\infty} i \cdot \left(u + \frac{r}{\mu}\right) \cdot \alpha(i) = \frac{u}{\rho} + \frac{r}{\lambda}. \tag{34}$$

In addition, like the derivation in Section 5.3, we assume that K is uniformly distributed. The energy consumption of a P-FINCH node is

$$\begin{aligned} &E_{P\_FINCH}(\kappa, K, \rho) \\ &= \sum_{i=0}^{\infty} \left\{ \left[\left(\frac{K}{P}\right) \cdot \left\lfloor \frac{i+\kappa}{K} \right\rfloor + \left\lceil \frac{(i+\kappa)\ mod\ K}{P} \right\rceil \right.\right. \\ &\quad \left.\left. - \left\lceil \frac{\kappa}{P} \right\rceil \right] \cdot u + i \cdot \frac{b}{\mu} \right\} \cdot \alpha(i) \\ &= \frac{u}{\rho P} + \frac{u(1-g)^2}{\rho g (1-g^K) K} \\ &\quad \times \sum_{\kappa=0}^{K-1} \left[\sum_{q=0}^{K-\kappa-1} \left\lceil \frac{q+\kappa}{P} \right\rceil g^q + \sum_{q=K-\kappa}^{K-1} \left\lceil \frac{q+\kappa-K}{P} \right\rceil g^q \right] \\ &\quad - \frac{\sum_{\kappa=0}^{K-1} \lceil \frac{\kappa}{P} \rceil u}{K} + \frac{b}{\lambda}. \end{aligned} \tag{35}$$

6 NUMERICAL RESULTS

This section presents the numerical results of the analysis presented in Section 5.

6.1 Handover Latency and Packet Loss

The HO latency and packet loss of the protocols analyzed in Section 5.1 are illustrated in Figs. 5 and 6. We refer to [25] and the testbed experiments we conducted to set the parameters listed in Table 2.

Fig. 5 compares the HO latencies of various mobility management protocols. The figure shows that MIP has the highest HO latency due to the long registration delay. Because F-MIP disengages the HO latency from IP connectivity and MIP registration, the HO latencies of both *predictive* and *reactive* modes are less than the other protocols. FINCH adopts cross-layered design to reduce the address resolution latency. Therefore, among other micromobility management protocols (HAWAII, CIP, and HMIP), FINCH has the smallest HO latency. Moreover, there is an additional computation delay for encapsulation and decapsulation in HMIP. The HO latency of HMIP is slightly higher than HAWAII and CIP.

Fig. 6 compares the number of lost packets during HO. Because of long HO latency, MIP suffers the largest amount of packet loss. For F-MIP, there is no packet loss in *predictive* mode because we assume that the buffer in NAR is not overflowed. The *reactive* mode still suffers less packet loss because packets can be forwarded immediately after the FBU message arrives at the PAR. For micromobility management protocols, packet loss is directly affected by HO latency. Therefore, FINCH still has the least packet loss among HAWAII, CIP, and HMIP.

Although the analysis shows that F-MIP experiences the smallest HO latency and the least packet loss, F-MIP requires packet reordering and coordination in signaling messages. Also, additional costs are incurred in F-MIP, which will be shown in the next section.

6.2 Location Update and Overall Cost

Figs. 7, 8, 9, 10, and 11 show the numerical results based on the analysis in Sections 5.2, 5.3, and 5.4. The analysis is

TABLE 2
Parameters for HO Latency Analysis

Parameter	Symbol	Value
L2 handover delay	D_{L2}	50 ms
IP connectivity latency	D_{IP}	600 ms
IP movement detection latency	t_{IP-mv}	100 ms
Address resolution latency	t_{AR}	200 ms
One-way delay between HA and MS	t_{HA-MS}	500 ms
MIP computation delay	t_{MIP}	15 ms
One-way delay between NAR and MS	t_{NAR-MS}	20 ms
One-way delay between PAR and NAR	$t_{PAR-NAR}$	10 ms
One-way delay between cross-over node and MS	t_{CR-MS}	20 ms
Computation delay for host-specific-routing-based micro-mobility management protocols	t_{Mi-H}	10ms
Computation delay for tunnel-based micro-mobility management protocols	t_{Mi-T}	15 ms
Packet arrival rate	λ_p	10 packets/s
NAR packet buffer size	$Buffer_{NAR}$	50 packets

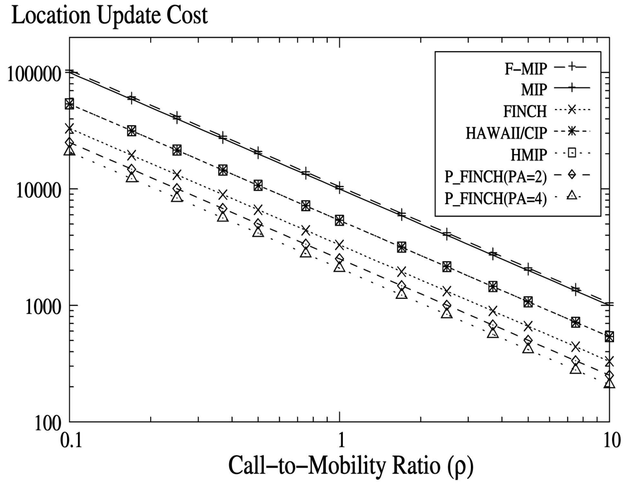


Fig. 7. Location update cost versus CMR.

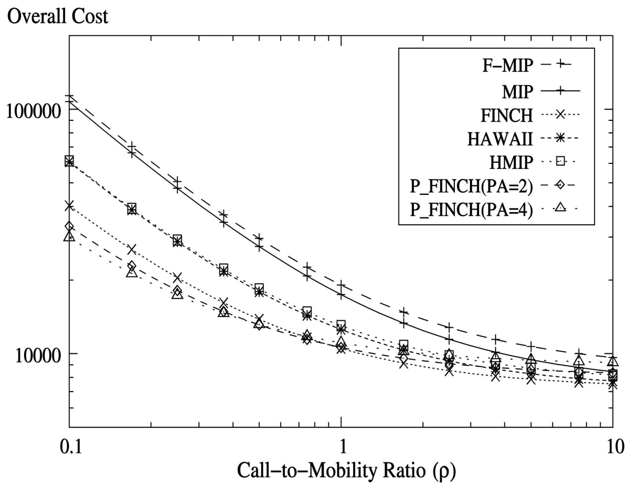


Fig. 8. Overall cost versus CMR.

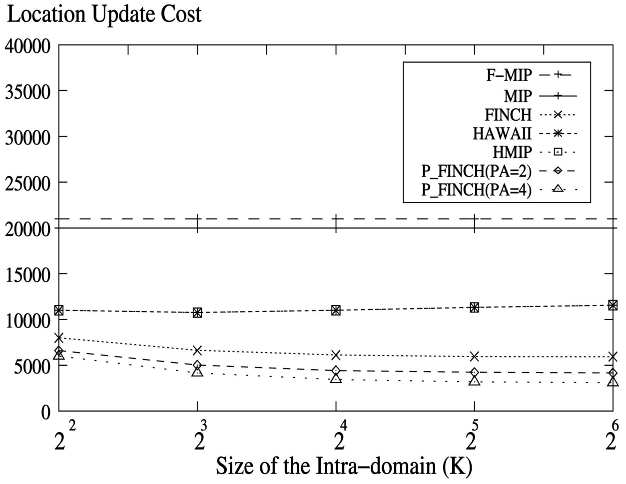


Fig. 9. Location update cost versus size of the intradomain.

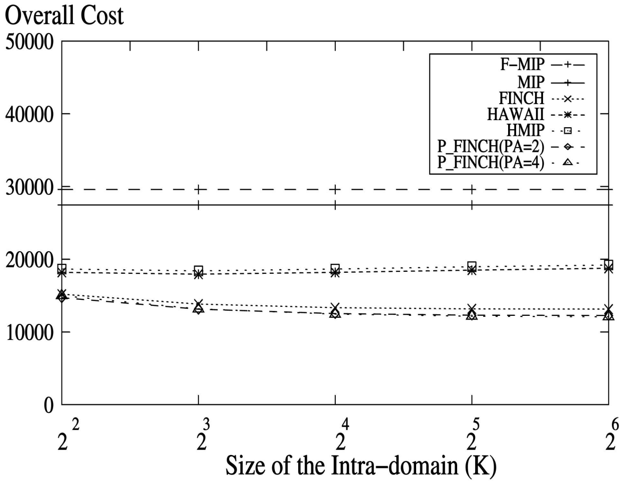


Fig. 10. Overall cost versus size of the intradomain.

validated by simulation using *Network Simulator version 2 (ns-2)* [26]. In the figures, the lines present the analytical results while the simulation results are marked with points. For demonstration purposes, the cell residence time is assumed exponentially distributed. Thus, $g = \frac{1}{1+\rho}$. Other parameters are listed in Table 3, which are obtained from [25] and a testbed we built.

Figs. 7 and 8 depict the location update cost and overall cost versus CMR. For ease of illustration, the y-axes of the figures are set to logarithmic scale. Note that the results are not linear. The size of each domain is set to 8. The increase in CMR implies that the movements of MSs become slower. Therefore, the costs decrease. If ARP is not performed, intuitively the location update cost of FINCH should be higher than that of HAWAII, CIP, and HMIP because FINCH needs to flood the location information to the whole network domain while entering the CSN. However, because FINCH employs a cross-layered design and eliminates ARP, it outperforms other protocols as shown in Fig. 7.

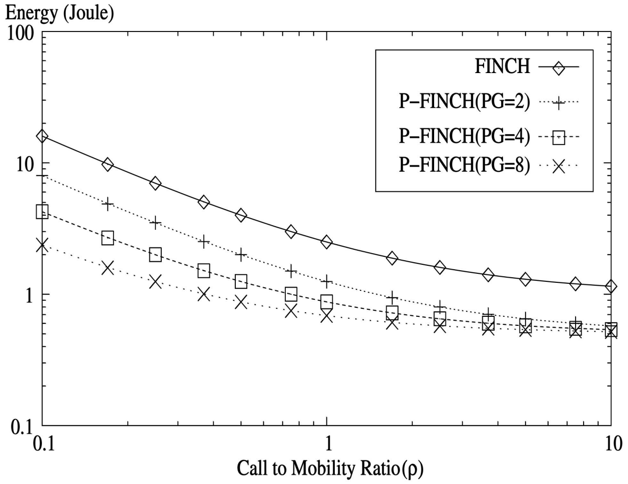


Fig. 11. Comparison of energy consumption in FINCH and P-FINCH.

Most of the literatures do not consider the interaction between L2 and L3. The performance was obtained by merely considering L3 only. As discussed earlier, L2 HO commences after L3 HO. Based on our study, the ARP cost dominates the location update cost, which is usually neglected in most studies. Fig. 7 also indicates that $C_{P-FINCH}$, C_{FINCH}, C_{HAWAII}, C_{CIP}, and C_{HMIP} are all much smaller than C_{MIP} and C_{F-MIP}. This mathematically verifies that employing intradomain mobility management can reduce the location update cost in interdomain mobility management. Because F-MIP introduces additional signaling cost, the cost is even larger than MIP. Besides, Fig. 7 shows that when P-FINCH is applied, the cost is reduced significantly. In addition, when the size of PG increases, the cost decreases. This is because the number of location updates is reduced when the size of PG increases. As shown in Fig. 7, the location update cost can reduce to 17 percent at $CMR = 0.1$ when the size of PG increases from two to four cells.

Fig. 8 illustrates that packet delivery also plays an important role. In Fig. 8, T_{FINCH}, T_{HAWAII}, and T_{HMIP} are still much smaller than T_{MIP} and T_{F-MIP} except when the mobility rate is quite low (CMR is high). Because F-MIP incurs buffer cost in packet delivery, the overall cost is the largest even when the mobility rate is quite low. The drawback of HMIP can be observed in this figure. When the mobility rate is low, the cost of HMIP is larger than that of HAWAII and MIP due to the hierarchical tunneling cost. On the other hand, HAWAII, which is a host-specific-routing-based intradomain mobility management protocol, still performs well when the mobility rate is low. This is because the forwarding cost is quite small. The figure also shows P-FINCH with different sizes of paging area. When the mobility rate is high (CMR is low), P-FINCH can significantly reduce the cost of FINCH. However, P-FINCH incurs more cost when the mobility rate is low (CMR is high). This is because when an MS does not move often, there is no need to page the MS. Paging the whole PG will cause unnecessary cost. From this figure, one can observe the tradeoff between the unnecessary paging cost when mobility rate is low and the excessive location update cost when mobility rate is high. To optimize the overall performance, this figure suggests that the paging extension should be turned on when CMR is less than 1.

TABLE 3
Parameters for Cost Analysis

Variable	Assumption
U	10000
S	500
A	1000
V	200
L	500
M	7000
F	200
T	450
B	500

Figs. 9 and 10 illustrate the relationship between the costs and the size of the intradomain. In the figures, the CMR (ρ) is set as a constant value of 0.5. The other parameters are the same as those in Table 3. In the figures, the costs of MIP remain constant. Because MIP is used for interdomain mobility, changing the size of the intradomain does not affect the interdomain mobility management. Similar to Figs. 7 and 8, Figs. 9 and 10 show that the costs of $C_{P-FINCH}$, C_{FINCH}, C_{HAWAII}, C_{CIP}, and C_{HMIP} are all much smaller than C_{MIP}. In Fig. 9, when the size of the intradomain increases, the cost decreases at first. This is because the possibility of performing location update is reduced when the size of the intradomain increases. However, when the size is extremely large, the costs of C_{HAWAII}, C_{CIP}, and C_{HMIP} start to increase. This is because ARP dominates most of the location update cost when the hierarchy is large. On the other hand, $C_{P-FINCH}$ and C_{FINCH} keep decreasing while the size of the intradomain increases. Fig. 10 depicts that the overall costs of all protocols decrease at first and increase when the size of the intradomain increases. This is because the paging, forwarding, and tunneling costs within the intradomain all increase when the domain size increases. Consequently, when the size is extremely large, the effect of the cost for forwarding or tunneling will surpass the cost for location update. This suggests that the size of the intradomain should be carefully chosen. In this experiment, it is recommended that the size of the intradomain should be less than 4 for HAWAII and HMIP and 16 for FINCH and its paging extension.

6.3 Energy Conservation

Fig. 11 compares the energy consumption of P-FINCH with FINCH. As in the discussion in Section 4.8, the analysis focuses on the energy consumption of an MS for location update. Therefore, only P-FINCH and FINCH are compared. As suggested in the last experiment, the size of each intradomain is set to 16. The ratio of the energy consumption $u : r : b$ is set to $1.5 : 1.0 : 0.5$. As shown in the figure, the paging extension can significantly reduce the energy consumption of the MS. Moreover, the energy consumption decreases as the size of PG increases. This is because increasing the size of PG will reduce the frequency of location updates.

7 CONCLUSION

Mobile WiMAX has been designed to support mobile users moving at vehicular speeds. MIP is adopted as the mobility management protocol by WiMAX Forum. However, it is generally realized that MIP cannot support HOs well when mobile nodes move frequently and/or when the coverage area of a subnet is small. The problem is even exaggerated for real-time services, which require very fast HOs in mobile WiMAX networks. In this paper, we propose to use MIP in mobile WiMAX for interdomain (inter-CSN) mobility management only. We propose a fast HO protocol, FINCH, for intradomain (intra-CSN) mobility management. The protocol is discussed with examples. The analytical models and extensive simulations show that the proposed FINCH can support fast and efficient link layer and intradomain HOs. Because of the cross-layered design, comparing with other intradomain mobility management protocols, the proposed FINCH reduces location update cost and overall cost. The numerical results can also be used to select proper network configurations. Besides, a scalable paging extension for FINCH, P-FINCH, is proposed. The performance analysis also shows that P-FINCH can significantly reduce the signaling overhead and energy consumption if the size of the paging area is well configured.

The proposed protocol is a complement to MIP in which MIP deals with interdomain mobility management in mobile WiMAX. Comparing with MIP, the proposed FINCH does not need IP encapsulation and does not have triangular routing problem. It also reduces the overhead caused by registering CoA with the HA. By unifying the mobility management in layer 2 and layer 3, the overhead and latency in interfacing conventional mobility management protocols in the two layers are eliminated as well.

APPENDIX

DETAILED ALGORITHMS OF FINCH

Algorithms 1, 2, and 3 present the detailed operations of the proposed FINCH protocol.

Algorithm 1 Packet forwarding algorithm

```
search column one/two of the FT for a matching
address;
if (port != NULL) then
  generate a MAC packet in which (destination
  address = MS MAC Address);
  send this MAC packet through the wireless port;
else
  generate a MAC packet in which (destination
  address = Forwarding MAC Address);
  send this MAC packet through the wired/backbone
  network;
end if
```

Algorithm 2 Location update algorithm in the new BS

```
search column one of the FT for a matching address;
update the corresponding entry in the FT;
  update forwarding MAC address to NULL;
  update Wireless Port to the port number;
  update Time Stamp;
send a packet to all directly connected device(s);
```

Algorithm 3 Location update algorithm in other devices

```
search column one of the FT for a matching address;
if (same Time Stamp) or (forwarding MAC address is
the same as the source MAC address) then
  destroy this packet;
else
  update the Time Stamp;
  /* comparing must be done before updating */
  if (forwarding MAC address equals NULL) then
    /* for old BS */
    update the forwarding MAC address with the
    source MAC address;
    update Wireless Port to NULL;
    send a notice to mobile node
  else
    /* for device(s) except the old BS */
    update the Forwarding MAC address with the
    source MAC address;
  end if
  send a packet to all directly connected device(s)
  except the source;
end if
```

ACKNOWLEDGMENTS

This work was supported in part by the National Science Council (NSC) under Grant 96-2752-E-007-003-PAE, Grant 97-2219-E-007-002, and Grant 96-2213-E-007-016. The associate editor coordinating the review of this paper and approving it for publication was A. Valko. Jui-Hung Yeh was with the Department of Computer Science, National Tsing Hua University, Hsinchu 30013, Taiwan, R.O.C, when this work was done.

REFERENCES

[1] *IEEE Std. 802.16-2004, Air Interface for Fixed Broadband Wireless Access Systems,* IEEE, Oct. 2004.

[2] *IEEE Std. 802.16e-2005, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems—Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands,* IEEE, Feb. 2006.

[3] *WiMAX Forum,* http://www.wimax.org/, 2008.

[4] *Stage 2: Architecture Tenets, Reference Model and Reference Points. Part 1: Network Reference Model,* WiMAX Forum Network Architecture, Jan. 2008.

[5] *Stage 2: Architecture Tenets, Reference Model and Reference Points. Part 2,* WiMAX Forum Network Architecture, Jan. 2008.

[6] J. Jee, S. Madanapalli, and J. Mandin, *IP over IEEE 802.16 Problem Statement and Goals,* IETF RFC 5154, Apr. 2008.

[7] A. Dutta, F. Vakil, J.-C. Chen, M. Tauil, S. Baba, N. Nakajima, and H. Schulzrinne, "Application Layer Mobility Management Scheme for Wireless Internet," *Proc. IEEE Int'l Conf. Third Generation Wireless and Beyond (3G Wireless '01),* pp. 379-385, May 2001.

[8] J.-C. Chen and T. Zhang, *IP-Based Next-Generation Wireless Networks.* Wiley, http://www.cs.nthu.edu.tw/~jcchen/book.html, Jan. 2004.

[9] E. Gustafsson, A. Jonsson, and C. Perkins, "Mobile IPv4 Regional Registration," IETF Internet draft, work in progress, Oct. 2006.

[10] A. Campbell, J. Gomez, S. Kim, A.G. Valkó, C.-Y. Wang, and Z.R. Turányi, "Design, Implementation and Evaluation of Cellular IP," *IEEE Personal Comm. Magazine,* Aug. 2000.

[11] R. Ramjee, K. Varadhan, L. Salgarelli, S.R. Thuel, S.-Y. Wang, and T.L. Porta, "HAWAII: A Domain-Based Approach for Supporting Mobility in Wide-Area Wireless Networks," *IEEE/ACM Trans. Networking,* vol. 10, no. 3, pp. 396-410, June 2002.

[12] A.T. Campbell, J. Gomez, S. Kim, and C.-Y. Wan, "Comparison of IP Micro Mobility Protocols," *IEEE Wireless Comm. Magazine,* vol. 9, no. 1, pp. 2-12, Feb. 2002.

[13] S. Das, A. Mcauley, A. Dutta, A. Misra, K. Chakraborty, and S. Das, "IDMP: An Intra-Domain Mobility Management Protocol for Next Generation Wireless Networks," *IEEE Wireless Comm.,* vol. 9, no. 3, pp. 38-45, June 2002.

[14] A.A.-G. Helmy, M. Jaseemuddin, and G. Bhaskara, "Multicast-Based Mobility: A Novel Architecture for Efficient Micromobility," *IEEE J. Selected Areas in Comm.,* vol. 22, no. 4, pp. 677-690, May 2004.

[15] W. Ma and Y. Fang, "Dynamic Hierarchical Mobility Management Strategy for Mobile IP Networks," *IEEE J. Selected Areas in Comm.,* vol. 22, no. 4, pp. 664-676, May 2004.

[16] H. Kim, K.-S.D. Wong, W. Chen, and C.L. Lau, "Mobility-Aware MPLS in IP-Based Wireless Access Networks," *Proc. IEEE Global Telecomm. Conf. (GLOBECOM '01),* pp. 3444-3448, 2001.

[17] H.-J. Jang, J. Jee, Y.-H. Han, S.D. Park, and J. Cha, "Mobile IPv6 Fast Handovers over IEEE 802.16e Networks," IETF Internet draft, work in progress, Mar. 2008.

[18] J. Jee, R. Koodli, S. Park, J. Cha, and H. Jang, "Mobile IPv4 Fast Handovers for 802.16e Networks," IETF Internet draft, work in progress, Oct. 2005.

[19] S. Das, A. Misra, P. Agrawal, and S.K. Das, "TeleMIP: Telecommunications-Enhanced Mobile IP Architecture for Fast Intradomain Mobility," *IEEE Personal Comm. Magazine,* vol. 7, no. 4, pp. 50-58, Aug. 2000.

[20] *Stage 2: Architecture Tenets, Reference Model and Reference Points. Part 3: Informative Annex,* WiMAX Forum Network Architecture, Jan. 2008.

[21] J.-C. Chen and P. Agrawal, "Fast Link Layer and Intra-Domain Handoffs for Mobile Internet," *Proc. IEEE Computer Software and Applications Conf. (COMPSAC '00),* pp. 325-330, Oct. 2000.

[22] R. Koodli and C.E. Perkins, "Fast Handovers and Context Transfers in Mobile Networks," *ACM Mobile Computing and Comm. Rev.,* vol. 31, no. 5, pp. 37-47, Oct. 2001.

[23] C. Makaya and S. Pierre, "An Analytical Framework for Performance Evaluation of IPv6-Based Mobility Management Protocols," *IEEE Trans. Wireless Comm.,* vol. 7, no. 3, pp. 972-983, Mar. 2008.

[24] G. Navarro, "A Guided Tour to Approximate String Matching," *ACM Computing Surveys,* vol. 33, no. 1, pp. 31-88, Mar. 2001.

[25] V. Chikarmane, C. Williamson, and R. Bunt, "Implementing Mobile IP Routing in 4.3 BSD Unix: A Case Study," *Internetworking: Research and Experience,* vol. 6, no. 4, pp. 209-227, Dec. 1995.

[26] *The Network Simulator—ns-2,* http://www.isi.edu/nsnam/ns, 2008.

Jui-Hung Yeh received the PhD degree from the National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2008. He is currently a postdoctoral researcher with Realtek-NCTU Joint Research Center, National Chiao Tung University (NCTU), Hsinchu. His research interests include mesh network, QoS, mobility management, integration of wireless networks, and performance evaluation. He is a member of the IEEE.

Jyh-Cheng Chen received the PhD degree from the State University of New York at Buffalo in 1998. He joined the Department of Computer Science and the Institute of Communications Engineering, National Tsing Hua University (NTHU), Hsinchu, Taiwan, as an assistant professor in 2001. He was promoted to an associate professor and a professor in 2002 and 2008, respectively. Prior to joining NTHU, he was a research scientist at Bellcore/Telcordia Technologies, Morristown, New Jersey, from August 1998 to August 2001. He is a coauthor of the book *IP-Based Next-Generation Wireless Networks* (Wiley, 2004). He has published more than 70 papers. He is the holder of 17 US patents, five ROC patents, and four PRC patents. He received the 2000 Telcordia CEO Award, the 2001 SAIC Executive Science and Technology Council (ESTC) Publication Award, the 2004 NTHU New Faculty Research Award, the 2006 NTHU Outstanding Teaching Award, and the 2007 Best Paper Award for Young Scholars from the IEEE Communications Society Taipei and Tainan Chapters and IEEE Information Theory Society Taipei Chapter. He is a technical editor of the *IEEE Wireless Communications*. He was a guest editor of the *IEEE Journal on Selected Areas in Communications*, special issue on all-IP wireless networks, May 2004. He was the technical program cochair of the Ninth IFIP/IEEE International Conference on Mobile and Wireless Communications Networks (MWCN '07) held in Ireland in September 2007. He was the technical program cochair of the Third IEEE International Conference on Information Technology: Research and Education (ITRE '05). He has been on the technical program committee of numerous international conferences, including IEEE INFOCOM 2005-2006, the IEEE Global Telecommunications Conference (GLOBECOM) 2005-2008, and the IEEE International Conference on Communications (ICC) 2007-2009. He was a tutorial speaker at IEEE GLOBECOM 2002, 2003, and 2006 on the subject of next-generation wireless networks. He leads the development of WIRE1x, which is one of the most important implementations of IEEE 802.1x supplicant. He is a senior member of the IEEE and the ACM.

Prathima Agrawal received the BE and ME degrees in electrical communication engineering from the Indian Institute of Science, Bangalore, India, and the PhD degree in electrical engineering from the University of Southern California in 1977. She is the Samuel Ginn distinguished professor of electrical and computer engineering in the Department of Electrical and Computer Engineering and the director of the Wireless Engineering Research and Education Center, Auburn University, Auburn, Alabama. Prior to her present position, she was an assistant vice president of the Network Systems Research Laboratory and an executive director of the Mobile Networking Research Department, Telcordia Technologies (formerly Bellcore), Morristown, New Jersey, where she worked from 1998 to 2003. From 1978 to 1998, she worked in various capacities at AT&T/Lucent Bell Laboratories, including creating and serving as head of the Networked Computing Research Department, Murray Hill, New Jersey. Her research interests include computer networks, mobile and wireless computing, and communication systems. She has published more than 150 papers in IEEE and other journals and conference proceedings in areas related to networks and VLSI. She holds 50 US patents. She received the Distinguished Alumnus Award from the Indian Institute of Science in 2006, the Telcordia CEO Award in 2000, and the Distinguished Member of the Technical Staff Award from AT&T Bell Laboratories in 1985. She is the recipient of the IEEE Computer Society's Distinguished Service Award in 1990 and the IEEE Third Millennium Medal in 2000. She is a fellow of the IEEE and the Institution of Electronics and Telecommunications Engineers (IETE), India, and a member of the ACM.