Mobility Management for Low-Latency Handover in SDN-Based Enterprise Networks

Ce Chen, Yu-Ting Lin, Li-Hsing Yen, Min-Cheng Chan, Chien-Chao Tseng*
Dept. of Computer Science, National Chiao Tung University, Hsinchu, Taiwan 300, R.O.C.
{chen7691, ylin, lhyen, mcjan, cctseng}@cs.nctu.edu.tw

Abstract—**Low-latency handover is crucial to real-time traffic in wireless networks. This paper considers an enterprise network managed by emerging Software Defined Network (SDN) technology. For this networking environment, we propose a mobility management scheme named Mobility SDN (M-SDN) that reduces the traffic pause time caused by a host-initiated layer-2 handover. M-SDN performs handover preparation in parallel with the layer-2 handover that involves N-casting of active flows to every potential handover target. Handover preparation is enabled by efficient address resolution and location tracking. We have implemented a prototype of M-SDN and conducted several experiments to evaluate the performance of M-SDN. Experimental results show that M-SDN effectively reduces the impact of layer-2 handovers without any modification on mobile devices.**

Index Terms—mobility management; software defined network (SDN); seamless handover; service continuity.

I. INTRODUCTION

The popularity of wireless devices and the emergence of Internet of Things (IoT) call for wireless network architecture that provides high bandwidth capacity and ubiquitous coverage. A response is small cell or micro cell technology, which uses extensive low-power access points (APs) to replace relatively fewer high-power APs. By spatial reuse of frequencies, the total bandwidth capacity can be increased. However, reduced coverage of a single AP generally implies frequent handovers. With a high transmission rate, considerable data will get lost during a hard handover. Therefore, it is crucial to reduce the impact of handover on user experience.

This paper considers an environment where mobile devices use IP over IEEE 802.11 to access the network. By default, a native 802.11 device locally detects the need to change its link attachment to the network (i.e., a layer-2 handover) and initiates the handover procedure autonomously. A layer-2 handover may trigger handovers in higher layers. For example, after a layer-2 handover, the host may detect the need to renew its network setting (e.g., IP address) and thus raise a layer-3 handover. Such a host-initiated layer-3 handover is proven time consuming [1] and unacceptable for real-time traffic.

Network-side protocols cannot control the latency of a host-initiated layer-2 handover. We thus focus on layer-3 mobility management schemes that allow mobile hosts to use the same network setting across different subnets. The objective is to minimize the pause of data traffic during handover. The requirement is not to modify behaviors of mobile hosts. Existing mobility approaches can be classified as host-based or network-based. Host-based solutions [2] modify or extend end devices to support the proposed solution, which may not be feasible in many cases. Mobile IP [3] is an example of host-based mobility scheme. Network-based solutions [4] demand no modification on hosts. Proxy Mobile IP [5] is an example of network-based mobility scheme. Most network-based approaches take action only after a layer-2 handover, making it hard to shorten the traffic pause time due to handover.

Many studies improved handover performance by a centralized entity that is aware of the entire network topology [6]. For the same reason, the centralized control characteristic of SDN [7] can facilitate handover procedures and improve overall mobility performance. However, little work has been done until recently. Cui [8] proposed a host-based scheme, which applied SDN technology to enhance Mobile IP, dealing with issues such as triangular routing and ingress filtering. The introduction of SDN improves handover latency. However, this proposal requires host modification. Raza et al. [9] proposed OF-PMIPv6, a network-based mobility scheme that implements PMIPv6 using OpenFlow protocol [10]. OF-PMIPv6 separates the control signaling path from the data communication path. OF-PMIPv6 is transparent to hosts and eliminates re-authentication process and scanning time for hosts by predicting the time and the target of the handover. However, OF-PMIPv6 utilizes MN to send RS message and network to send RA message, which gives rise to more delay time. Moreover, this approach demands IP tunneling instead of OpenFlow forwarding which incurs extra overhead.

In this paper, we propose Mobility SDN (M-SDN), a mobility management scheme for SDN-based enterprise network. Unlike prior works, M-SDN demands neither host modification nor IP tunneling. It reduces the traffic pause time caused by a host-initiated layer-2 handover by means of handover preparation that projects an imminent handover and performs N-casting to potential handover targets in parallel with the layer-2 handover. M-SDN demands a location server, a mobility application implemented on each controller, and some supports from APs and DHCP servers. M-SDN can support vertical handover and application layer session continuity [11] for service continuity.

The rest of the paper is organized as follows. Section II presents the design rationale behind M-SDN, system overview, and details about the mobility application. Section III describes signaling messages and corresponding procedures for intra-domain and Inter-domain handovers. The evaluation of M-SDN is presented in Section IV while Section V concludes this work.

---

*Responsible for all communications.

II. DESIGN OF MOBILITY SDN (M-SDN)

A. *Design Rationale*

This section explains the idea behind the design of M-SDN. Several approaches like OF-PMIPv6 [8] allow mobile nodes to use the same network setting across different subnets by means of IP tunneling. M-SDN eliminates the need for IP tunneling as it no longer follows the IP routing rule to forward packets. Instead, controllers in M-SDN set up packet forwarding rules (i.e., flow entries) in associated SDN switches for the traffic flows of mobile nodes. However, such a design should address the following issues.

1. Address resolution. Traditionally, IP addresses are used as both host identifier and routing directive. When hosts can use the same network configuration across different subnets, IP address as a host identifier can no longer serve the functionality of routing directive. Therefore, M-SDN creates packet forwarding rules based on MAC address. Because packets are still destined to mobile node's IP address, there should be a scheme that maps IP address to the corresponding MAC address.
2. Location tracking. To find the correct endpoint of a new flow for a mobile node, there should be a scheme that tracks the current location (i.e., serving AP) of each mobile node.
3. Flow redirection. If a mobile node changes its point of attachment to the network, all active flows associated with it should be redirected to the new point of attachment. The redirection process takes time, causing significant impact on handover latency. There should be a scheme that continues all active flows while minimizing the pause of data traffic in these flows.

For address resolution, address resolution protocol (ARP) used in traditional IP networks is inefficient here because ARP request messages would be broadcast to the whole enterprise network. Therefore, M-SDN maintains IP-MAC addressing bindings to find the MAC address corresponding to a particular IP address.

For location tracking, we assume that all APs are under the management of SDN controllers. M-SDN demands a small agent on each AP to notify the controller whenever a new mobile node attaches to the AP.

For low-latency flow redirection, the key is to project a layer-2 handover beforehand and set up necessary flows for the migrating node in parallel with the layer-2 handover. This SDN-based *handover preparation* effectively shrinks overall handover latency and can hopefully complete the flow redirection process before the completion of the accompanying layer-2 handover.

B. *The Systyem*

We consider an enterprise network that is partitioned into multiple SDN domains for efficient and flexible management. We assume that each SDN domain is managed by a single SDN controller. Figure 1 shows the hierarchy of M-SDN. There are four major components in M-SDN: location server, controllers, DHCP servers, and switches. Details about the four components are as follows.

1) Location server

There is one location server in the system, which maintains the current location and IP-MAC address binding for each device in the system. It also helps inter-domain handover operations and performs route optimization for packets flowing through multiple domains.

2) Controllers

A controller has a global view of the network topology in its domain. Each controller is augmented with an application that adds mobility management functionality (i.e., address resolution, location tracking, and handover preparation) to the controller. Details are presented in the next subsection.

3) DHCP server

When a mobile node enters a domain, it configures its IP address and other network-layer setting through a DHCP server deployed in the domain. The DHCP server then sends the IP-MAC address binding of the node to the controller, from which the binding is then reported to the location server.

4) OpenFlow-enabled switches.

We assume that all switches and APs conform to the OpenFlow specification. A layer-2 handover across different Extended Service Sets (ESSs) typically causes a layer-3 handover [10]. To avoid unnecessary layer-3 handovers raised by mobile nodes, we assume that all APs are configured with the same Service Set Identifier (SSID).

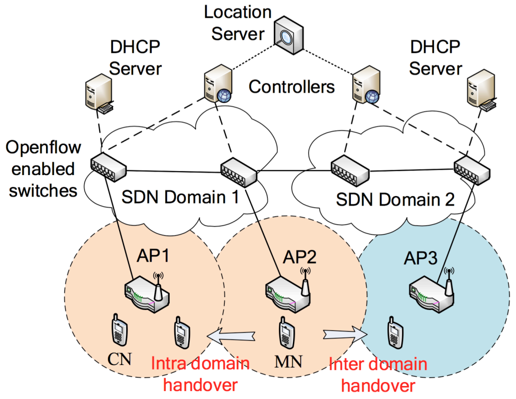

Fig. 1. Hierarchy of M-SDN

IP-MAC addressing binding and location information of mobile nodes are kept by the location server and by controllers. However, those kept by the location server are complete while those kept by a controller are only specific to the controller's domain. This design allows a controller to handle an intra-domain handover locally without the help from the location server. Similar designs can be found in Cellular IP [12], HAWAII [13] and Hierarchy Mobile IP [14].

C. *The Mobility Application*

The mobility application on controller is composed of Core module and REpresentational State Transfer (REST) Interfaces. Refer to Fig. 2.

1) Core Module

The core module works closely with the controller to perform mobility-related operations such as flow redirections. It takes care of mobility-related signaling and operations (handover preparation, location tracking, post-handover house keeping). It also keeps AP neighborhood information. AP neighborhood information could be statically configured, abstracted from historical handover profiles, or judged from the geographic location of each AP [6] [15] [16].

2) REST Interface

The core module communicates with other components such as the controller through REST interface. REST interface is supported by many popular SDN controllers (Floodlight, Open Daylight, Ryu, etc.) REST interface serves as both client and server. It serves as a server when receiving signaling or commands from other components in M-SDN. On the other hand, REST interface acts as client when sending controlling messages to the location server or performing actions to controller.

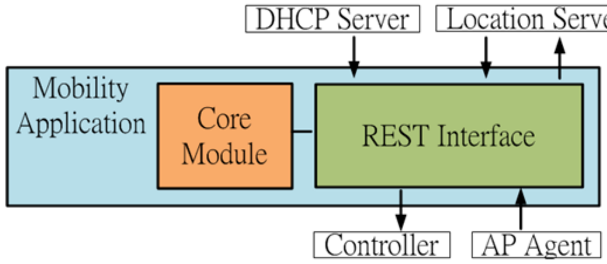

Fig. 2. Modules in mobility application

III. SIGNALING MESSAGE IN HANDOVER OF M-SDN

A. Messages and Procedures

M-SDN defines the following signaling messages:

a) Binding Update: When a mobile node configures or renews its IP address through a DHCP server, the server sends Binding Update to the controller. The controller then forwards the update to the location server.

b) Signal Low Event: When an AP detects that the signal strength with some mobile node drops below a threshold, it sends Signal Low Event to the controller.

c) Pre-handover Request: If a core module detects a need for an inter-domain handover preparation, it sends Pre-handover Request via the location server to the controller of the domain where some handover candidate target APs reside.

d) Report MAC (Device Attachment): When an AP detects a device's attachment, it sends Report MAC message to the controller.

e) Location Update: When a controller receives Report MAC from an AP, it sends Location Update to the location server if the reported MAC address is new to it. The reply of this message from the location server indicates whether the newly-attached mobile node is previously unseen or just handed over from another domain.

f) Handover Notification: When the location server detects an inter-domain handover after receiving Location Update, it sends Handover Notification to the controller in the mobile node's previous domain.

Handover procedure in M-SDN is divided into two phases: pre-handover phase and commit (or post-handover) phase. Handover preparation is done in the pre-handover phase, which consists of the following two steps:

1) Handover projection and candidate APs identification

Whenever an AP projects an imminent handover, the AP sends a signal to inform the controller in its domain. The mobility application on the controller then identifies the set of candidate target APs for the imminent handover.

2) N-casting

After identifying candidate target APs, the mobility application calculates a route from the current serving AP to each candidate AP and modifies flow tables on all switches along the route for N-casting accordingly. N-casting duplicates packets destined to the node undergoing a handover, one to the present serving AP and the others to all possible handover targets. N-casting allows the node to continuously receive packets destined to it right before and after a layer-2 handover.

The commit phase begins after a layer-2 handover. It is to clean up unneeded flows created in the pre-handover phase.

B. Intra-domain Handover

For intra-domain handover, Figure 3 shows the scenario of pre-handover phase. The message sequence and associated reactions are as follows. (1) The present serving AP (*pAP*) of mobile node (MN) sends Signal Low Event to its controller. (2) After receiving the event, the core module checks the AP neighborhood information to identify all handover candidates. In case of intra-domain handover, all the handover candidates are in the same domain as *pAP*. The core module then queries all active flows of MN from the controller and calculates routes within the domain for N-casting. (3) Based on the result of the route calculations, the core module asks the controller to send OpenFlow flow modification messages to set up flow table rules on associated switches.

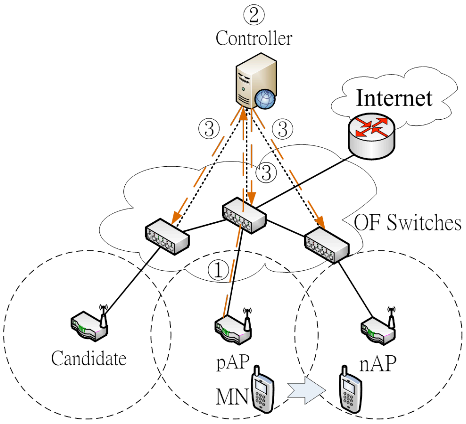

Fig. 3. Scenario of Intra-domain handover (pre-HO)

Figure 4 shows the scenario of the commit phase for intra-domain handover. The message sequence and associated reactions are as follows. (4) When the new serving AP (*nAP*) detects MN's attachment, it sends Report MAC to its controller.

⑸ After receiving the message, the core module realizes that MN has committed a layer-2 handover, and thus sends flow modification messages to affected switches to remove all unneeded flows created for N-casting in the pre-handover phase.

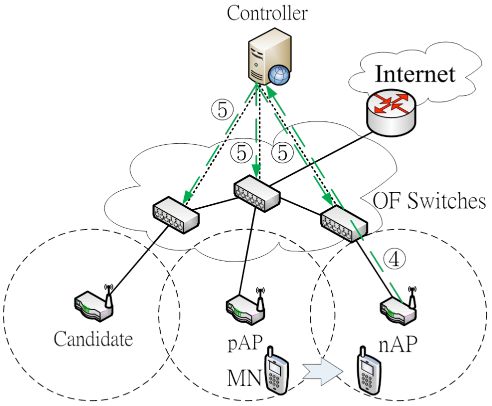

Fig. 4. Scenario of Intra-domain handover (commit)

C. *Inter-domain Handover*

Inter-domain handover differs from intra-domain handover in that at least one handover candidate is outside the current domain. Consequently, N-casting needs to create flows in several domains. This is assisted by the location server.

For inter-domain handover, Figure 5 shows the signaling message flow of the pre-handover phase. We refer to the controllers before and after an MN's handover as *pController* and *nController*, respectively. AP first sends signal-low event to *pController* Here ⑴, ⑵, and ⑶a are the same as ⑴, ⑵, and ⑶ in Fig. 3. ⑶b *pController* sends Pre-handover Request that carries the identifiers of all candidate APs outside its domain to the location server. ⑷ The location server finds out the domains of these APs and forwards the request to *nController*. ⑸ When the mobility application of *nController* receives the request, it computes pre-handover route and install flow table rules on those switches from the domain boundary to the handover target candidate.

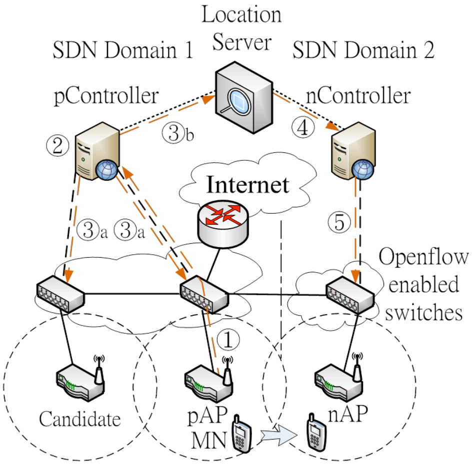

Fig. 5. Scenario of Inter-domain handover (pre-HO)

The signaling message flow of the commit phase for inter-domain handover is shown in Figure 6. Here ⑹ and ⑺ correspond to ⑷ and ⑸ in Fig. 4. ⑻ *nController* sends Location Update to the location server. ⑼ The location server updates the MN's location and sends Handover Notification message to *pController*. ⑽ After receiving the notification, the core module of *pController* removes information of the handover device and deletes flows that are no longer needed. This concludes the inter-domain handover.

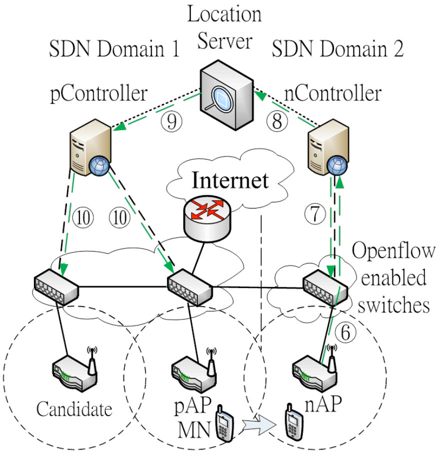

Fig. 6. Scenario of Inter-domain handover (commit)

IV. EVALUATION

We performed some experiments to evaluate the performance of M-SDN in both intra-domain and inter-domain scenarios. The results are compared with those of traditional mobility management scheme. We also tested the effectiveness of M-SDN in real environment.

The experiment topology is the same as Figure 1. Table I lists hardware and software used in the experiments. We installed OpenSwitch in OVS-PCs and APs to let these devices act as OpenFlow switches that forward packets in the enterprise network. We also installed Iperf tool in MN and corresponding node (CN) for throughput and delay measurements.

TABLE I. PHYSICAL ENVIRONMENT OF EXPERIMENT

Node Type	Hardware	Software
Controller	PC	Floodlight
OVS-PC	PC with 4 ports NIC	OpenvSwitch on Ubuntu
AP	D-link DIR-835	OpenvSwitch on OpenWRT
CN	Laptop with Atheros AR9485	Iperf client
MN	PC	Iperf server

A. *Latency of Handover Preparation*

To study the effectiveness of handover preparation, we measure the latency of signaling and internal action taken by mobility application in both intra-domain and inter-domain handovers. Table II shows the results.

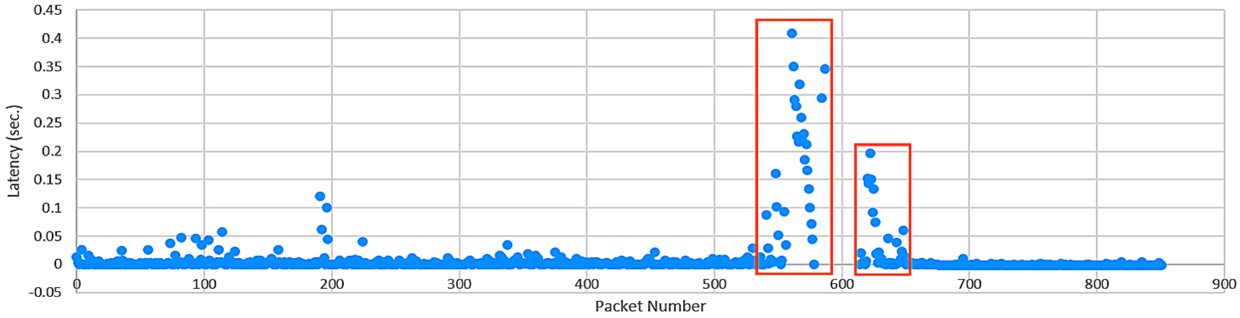

Fig. 8. Experiment with M-SDN: packet latency

The location update latencies are 9 ms and 11 ms in intra-domain and inter-domain handovers, respectively. In case of intra-domain handover, the latency of MAC Report signaling accounts for the result. In case of inter-domain handover, additional 2 ms was added by Location Update. The result indicates that the location update delay introduced by the hierarchical mobility management in M-SDN is negligible.

The mobility application performs three major tasks to set up N-casting: querying active flows of the mobile node undergoing a handover, calculating route, and setting up flow tables. The measured delays of these tasks are 1 second, 5 ms, and 1~2 ms/flow, respectively.

The minimum time required for handover preparation is the sum of the location update latency and the delay of setting up N-casting. The result is about 1.03 sec, which is dominated by flow query time. This amount of query time can be further reduced or eliminated if the mobility application records all flows as ONOS does [17]. If the time between the occurrence of a Signal Low Event and the completion of the corresponding layer-2 handover is longer than the minimum time required for handover preparation, the handover preparation latency can be totally transparent to the MN, and thus achieving a seamless network-layer handover.

TABLE II. SIGNALING TIME EVALUATION

Delay factors		**Delay time**
Location update (Signaling)	Intra-domain (AP to Controller)	9 ms
	Inter-domain (AP to Controller to Location Server)	11 ms
N-casting	Query MN Related Flows	1 sec.
	Route Calculation	5 ms
	Flow Table Setup	1~2 ms per flow

B. *Improvement by M-SDN*

Besides measuring the latency of handover preparation, we also conducted experiments to demonstrate the improvement by M-SDN on handover performance.

To study how M-SDN improves handover performance, we tested two handover scenarios: one with SDN-enabled handover preparation and the other without. In both scenarios, Iperf was used to generate one-way 200-Kbps UDP traffic from CN to MN. Handover preparation for the first test scenario was performed 12 seconds after the flow started. Handover occurred at 35 seconds after the flow started in both tests. Two APs installed OpenWRT were used for the experiments. A layer-2 handover in this environment took about 1.5 second.

Native SDN network without M-SDN does not support mobility. Therefore, the controller did not see anything unusual after the change of the MN's attachment point. Consequently, MN did not receive any packet after a layer-2 handover since all packets destined to it were still forwarded to its original attachment point. With M-SDN, total 26 packets were lost during the handover. Nevertheless, the flow for the UDP traffic continued after a layer-2 handover.

Two figures help us take a close look at the result. Figure 7 shows the relationship between time and packet numbers, while Figure 8 shows the latency of each packet in M-SDN. As can be seen in Figure 8, the packet latency rises when MN gradually left its serving AP. The latency gradually drops back to normal after the layer-2 handover.

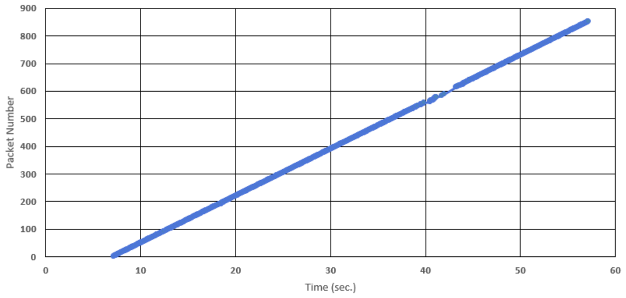

Fig. 7. Experiment with M-SDN: Packet Number vs. Time

Our experiment result indicates that with the help of M-SDN, the pause time of data stream after a layer-2 handover can be reduced significantly. MN continued receiving packets from the new serving AP after a layer-2 handover and only 26 packets were lost during the whole handover period. We conclude that M-SDN effectively reduces the impact of handover without any modification on mobile devices.

V. CONCLUSION

We have proposed M-SDN to reduce traffic pause time caused by a host-initiated layer-2 handover in SDN-based enterprise network. SDN demands neither host modification nor IP tunneling. M-SDN designates a location server together with mobility application on controller to keep IP-MAC addressing binding, topology information, and device location information for the realization of handover preparation, i.e., handover target projection and N-casting. Experimental results show that the time for handover preparation is around one second and thus could be done in parallel with layer-2 handover. The results also show that the flow for UDP datagrams continued after a layer-2 handover, though some packets were dropped.

Future work includes an accurate prediction of handover target among all potential candidates. With that prediction, we need only bi-casting rather than N-casting. Another possible improvement is on flow query time. Finally, we will conduct more experiments to justify the effectiveness of M-SDN.

VI. ACKNOWLEDGEMENT

This work is sponsored in part by Ministry of Science and Technology, Taiwan, under grant numbers 104-2622-8-009-001 and 104-2221-E-009-021-MY3, and by D-Link under grant number 104C557.

VII. REFERENCES

[1] L.-H. Yen et al., “Experimental study of mismatching ESS-subnet handoffs on IP over IEEE 802.11 WLANs,” *Proc. 8th Int’l Conf. on Wireless and Optical Communications Networks* (WOCN), Paris, France, May 2011.

[2] J. Lee et al., “Host-based distributed mobility management support protocol for IPv6 mobile networks,” in *IEEE WiMob*, 2012.

[3] C. Perkins, “IP mobility support for IPv4, revised,” *IETF RFC 5944*, Nov. 2010.

[4] K.-S. Kong et al., “Handover latency analysis of a network-based localized mobility management protocol”, *Proc. IEEE ICC*, 2008.

[5] S. Gundavelli et al., “Proxy Mobile IPv6,” *IETF RFC 5213*, Aug. 2008.

[6] C.-C. Tseng et al., “Topology-aided cross-layer fast handoff designs for IEEE 802.11/Mobile IP environments,” *IEEE Communications,* vol. 43, no. 12, pp. 156-163, Dec. 2005.

[7] Open Networking Foundation, “Software-Defined Networking: the new norm for networks”, *ONF White Paper*, 2012.

[8] D. Cui, “Mobile handoff research based on SDN/OpenFlow,” *Master Thesis, University of Chinese Academy of Sciences*, May 2013.

[9] S. M. Raza et al., “Leveraging PMIPv6 with SDN,” *Proc. 8th Int’l Conf. on Ubiquitous Information Management and Communication* (ICUIMC), 2014.

[10] N. McKeown et al., “OpenFlow: enabling innovation in campus networks,” *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp.69-74, Apr. 2008.

[11] M.-C. Chan et al., "A cross-layer architecture for service continuity and multipath transmission in heterogeneous wireless networks," *Proc. IEEE WCNC,* pp. 4853-4858, Shanghai, China, Apr. 2013.

[12] A. G. Valkó, “Cellular IP: a new approach to Internet host mobility,” *ACM SIGCOMM Computer Communication Review*, vol. 29, no. 1, pp. 50-65, Jan. 1999.

[13] R. Ramjee et al., “HAWAII: a domain-based approach for supporting mobility in wide-area wireless networks”, *IEEE/ACM Trans. on Networking*, vol. 10, no. 3, pp. 396-410, Jun. 2002.

[14] H. Soliman et al., “Hierarchical Mobile IPv6 (HMIPv6) Mobility Management”, *IETF RFC 5380*, October 2008.

[15] A. Mishra et al., “Context caching using neighbor graphs for fast handoffs in a wireless network,” *Proc. INFOCOM*, pp. 351-361, Mar. 2004.

[16] C.-C. Tseng et al., “Location-based fast handoff for 802.11 networks,” *IEEE Commun. Lett.*, vol. 9, no. 4, pp. 304–306, Apr. 2005.

[17] P. Berde et al., “ONOS: towards an open, distributed SDN OS,” in *HotSDN*, Aug. 2014.