

# A hierarchical mobility management scheme based on software defined networking

Xing Yin<sup>1,2,3</sup> · Liangmin Wang<sup>2,3</sup> · Shunrong Jiang<sup>2,3</sup>

Received: 8 July 2017 / Accepted: 23 October 2017 / Published online: 12 January 2018  
© Springer Science+Business Media, LLC 2018

**Abstract** Software Defined Networking (SDN) is a new paradigm which allows the network devices to be more programmable and easily controlled. SDN can be applied to improve the flexibility and performance of mobile networks. As a key problem of SDN-based mobile networks, the mobility management still cannot simultaneously support various different scenarios and has the limitations of high signaling cost and handover delay. In order to overcome these drawbacks of existing related protocols, we propose a SDN-based mobility management scheme named H-SMM by constructing a hierarchical architecture in control plane to support intra-domain and inter-domain handover scenarios at the same time. Then we design handover procedures to

simplify intra-domain handover and optimize inter-domain handover respectively. In addition, the communication initiation procedure is also provided. Qualitative analysis shows H-SMM is an overall mobility management scheme. Calculative analysis indicates that H-SMM has lower signaling cost. Simulation results illustrate that the handover delay and the end to end delay of first packet in communication initiation are lower than contrast schemes. Moreover, our evaluation also shows that the above performance advantages of H-SMM will be more evident with the increasing of domain numbers and distance between domains.

**Keywords** Mobility management · SDN · Hierarchical · Signaling cost · Delay

---

This article is part of the Topical Collection: *Special Issue on Software Defined Networking: Trends, Challenges and Prospective Smart Solutions*

Guest Editors: Ahmed E. Kamal, Liangxiu Han, Sohail Jabbar, and Liu Lu

---

✉ Liangmin Wang  
Wanglm@ujs.edu.cn

Xing Yin  
staryin@126.com

Shunrong Jiang  
jsywow@gmail.com

- <sup>1</sup> School of Electrical and Information Engineering, Jiangsu University, Zhenjiang, China
- <sup>2</sup> School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China
- <sup>3</sup> Jiangsu Key Laboratory of Security Technology for Industrial Cyberspace, Zhenjiang, China

## 1 Introduction

With the rapid development of wireless networks and mobile devices in the last decades, mobile Internet has been widely applied in our daily life. The global mobile data traffic is expected to reach massive 367 exabytes in each year until 2020 [2]. The trend of supporting any one to get any wanted information in any location can bring us much more convenience. Accordingly, it also leads to new challenges for mobility management, which is a critical technique of mobile Internet.

Although mobility management techniques have made some development due to the application and improvement of the widely-accepted mobility protocol, which is Mobile IPv6 (MIPv6) [22]. MIPv6 and its derivatives still have some shortcomings such as non-optimal routing, poor handover performance and high signaling cost [8]. Furthermore, diverse new mobility applications need distinct designs of

mobility management processes, which lead to the requirement for better flexibility in network devices [31]. However, the functions of control and data forwarding are tightly coupled in network devices in traditional network. This makes it difficult for these devices to be programmable [30]. This problem not only partially causes the above drawbacks of existing protocols, but also impedes further improvement and deployment of mobility management techniques [33].

To overcome the problems of traditional network, a promising network paradigm named Software Defined Networking (SDN) [20] separates the control plane from the data plane in network devices, and then makes these devices programmable. All the network devices such as routers and switches in data plane can be regarded as SDN switches, which are only responsible for data forwarding according to their flow table. A logically centralized SDN controller has a global view of the network managed by it, and based on which it can make routing decisions. Controller can determine forwarding strategies of all devices by downloading flow entries through south bound interface (SBI), e.g. OpenFlow [16], to SDN switches to update their flow tables. One controller and all the network devices controlled by it constitute a network, and the coverage of the network is usually named control domain or domain.

Due to these characteristics, SDN provides significant improvements in flexibility, scalability, and performance of mobile networks to keep pace with the development of new requirements and technologies [14]. Moreover, proposing mobility management based on SDN architecture will bring the following benefits.

1. By separating control function from data forwarding devices to a controller, data flows can be forwarded by SDN switches along the optimal route instead of detouring to a location maintaining node. This means route optimization can be solved.
2. Centralized control in SDN architecture can reduce complexity of signaling interaction processes. This means signaling cost and even handover delay can be dropped to some extent.
3. Programmability of SDN switches can enable them to have different kinds of mobility management functions. This means most or entire processes can be handled by network devices instead of end hosts.

Despite the above benefits, introducing SDN into mobile networks has unique issues. As we know, SDN originated in campus/enterprise networks and developed in data center networks (DCN) [19]. Most of these networks are in the form of a single control domain. The researches of SDN in mobile Internet are still in the early stage. Since long distance mobility in SDN-based network will cross

different domains inevitably, mobility networks based on SDN architecture must consider the situation of both single and multiple domains. That is, the handover procedures of mobility management those take place inside one domain and among different domains should be designed simultaneously. The former is defined as an intra-domain handover, and the latter is defined as an inter-domain handover.

Recently, more and more researchers try to provide mobility management schemes in SDN-based network [3, 12, 23, 26–29]. Although some drawbacks such as non-optimal routing caused by traditional network can be overcome by SDN architecture, existing related researches still have some shortcomings. First, intra-domain and inter-domain handover can not be provided or optimized simultaneously. Second, signaling interaction between different controllers in inter-domain mobility causes extra signaling cost. Third, handover delay still needs to be improved for delay sensitive applications.

In this paper, we present a hierarchical SDN-base mobility management (H-SMM) scheme to improve the inadequacies of related researches. The main contributions of our proposal are listed as follows:

1. A hierarchical architecture in control plane was constructed to provide intra-domain and inter-domain mobility simultaneously.
2. With the help of the hierarchical control plane, we designed the handover and communication procedures of H-SMM. First, we proposed the intra-domain handover procedure and constrained the range of signaling interactions inside one domain to optimize the intra-domain handover performance. Second, in order to optimize the performance (e.g. handover delay and signaling cost) of inter-domain handover, we introduced a global controller in the upper sub-layer of control plane to maintain a binding between the identity and the only locator of each mobile node, and then we designed the signaling interaction procedures among different controllers in detail. Last, the communication initiation procedure was also designed to reduce the end to end delay of the first data packet.

The rest of this paper is organized as follows. In Section 2, we review existing mobility management protocols based on both traditional network and SDN. In Section 3, we present a hierarchical architecture in control plane. Based on this architecture, we propose both intra-domain and inter-domain handover procedures. In addition, communication initiation procedure is also described. Then in Section 4, the characteristics and performance of H-SMM are evaluated and compared with some typical comparatives. Finally, we conclude our paper in Section 5.

## 2 Related works

In this section, we review two categories of existing typical mobility management protocols according to their basic network architecture, that is, traditional network based mobility protocols and SDN-based mobility schemes.

### 2.1 Traditional network based mobility protocols

Representatives of mobility protocols based on traditional network are MIPv6 and its derivatives. MIPv6 is the most famous mobility protocol which has been standardized by the Internet Engineering Task Force (IETF). In MIPv6, a mobile node (MN) uses the home address (HoA) assigned in its home network as its identification. Meanwhile, MN uses the care-of address (CoA), which is a new routable IP address assigned by the current network of MN, as its location. When attached to a new network, MN sends a binding update (BU) message, which contains a binding (i.e. mapping) of HoA and CoA, to the home agent (HA) located in its home network to update the binding cache (BC) of HA. BC is a table which maintains the bindings of mobile nodes. After that, a bidirectional tunnel between HA and MN will be established. The communication between MN and the correspondent node (CN) will be performed through this tunnel.

By adopting the basic idea of separating identity and location of MN, MIPv6 can support session continuity of any mobile node when it moves around and changes its point of attachment to the Internet. As can be seen, HA is not only the maintainer of bindings in control plane, but also the forwarder of all packets in data plane. Coupling of both planes in HA leads to the triangle routing problem. All the packets send from CN to MN must detour to HA, which leads to a higher end to end delay. Furthermore, the procedures of address reconfiguration and registration deteriorate handover performance and cause high signaling cost. Although the route optimization mode of MIPv6 and some other relative researches [4, 34] optimize data routing by sending a BU message from MN to CN directly. This so-called correspondent registration mechanism addresses non-optimal routing at the cost of high delay and signaling cost caused by the return routability procedure (RRP), which must be performed to prove the authenticity of CoA [22].

Many extended protocols have been proposed to overcome the above weaknesses of MIPv6. In order to optimize the handover happens inside a small scale network, Hierarchical Mobile IPv6 (HMIPv6) [24] adopts a hierarchical location management mechanism to divides the whole network into several domains and introduces a mobile anchor point (MAP) in each domain to maintain the local binding of MN. Therefore, the signaling exchanging of intra-domain

handover inside one domain is constrained to the current domain, which can optimize intra-domain handover performance. However, inter-domain handover was not considered by HMIPv6.

Fast Mobile IPv6 (FMIPv6) [25] improves the handover performance of MIPv6 by proactive handover mode, which handling layer 3 (L3) handover procedures in advance before layer 2 (L2) handover takes place. The complex handover procedure of FMIPv6 leads to higher signaling cost. Furthermore, intra-domain handover is not considered in FMIPv6.

Proxy Mobile IPv6 (PMIPv6) [9] is a network-based mobility scheme. The handover procedure is performed by a new network entity named mobile access gateway (MAG) instead of MN. Thus PMIPv6 has better compatibility. By using a local mobility anchor (LMA) to instead HA, PMIPv6 also can provide intra-domain mobility. Therefore, PMIPv6 has the same drawbacks as HMIPv6.

In recent years, there is a growing trend to provide more dynamic and flexible mobility management in a fully distributed way [13]. Another class of mobility solutions distribute the functionality of signal HA to multiple mobile anchors (MA) [35]. Even though these distributed mobility management (DMM) schemes can alleviate or even avoid some problems of the above protocols, such as heavy overhead of HA and the triangle routing problem, the related functions including anchor discovering and switching, BU distribution and synchronization among multiple mobile anchors will lead to higher protocol complexity, larger handover delay and heavier signaling cost in large scale networks.

### 2.2 SDN-based mobility management schemes

With the arising of novel SDN architecture and its application in different kinds of networks, many researchers apply SDN in mobile and wireless networks. Most of the early related researches are concentrated on how to utilize SDN to optimize the existing wireless access networks or core networks. For example, OpenRoads [32] aims to improve robustness of Wireless Local Area Networks (WLANs) and Worldwide Interoperability for Microwave Access (WiMAX) networks during handover by using multicast in Openflow networks. Mobileflow [21] tries to apply SDN in mobile carrier networks. Elgendi et al. [6] proposed a three-tiered SDN-based DMM architecture to improve the efficiency of core network in Long Term Evolution (LTE) networks. Some researchers even began to study the application of SDN-based mobility schemes for 5G networks [5, 18]. Mobility management functions including binding management and handover procedure are not considered in these literatures.

More recently, a few researchers began to propose different types of mobility management schemes based on SDN. Al Sabbagh et al. [3] proposed an algorithm based on OpenRoads to optimize controllers for providing efficient packet processing in mobility scenario. Then all sessions can be maintained when users roam among different kinds of access networks. However, binding update mechanism during handover has not been concerned.

As an early mobility management scheme based on OpenFlow, Papatwibul et al. [23] only focused on the intra-domain mobility. MN has to be involved in handover processes and inter-domain mobility was not considered in this scheme.

According to the idea of DMM, Li, et al. [12] proposed a fully distributed and flat architecture in control plane for supporting inter-domain mobility. Due to this architecture and introducing several kinds of controllers such as default controllers and border controls, the network structure and the handover procedure of this scheme are rather complex. In addition, intra-domain handover has not been mentioned.

Wang et al. [26, 27] proposed a software-defined mobility architecture to solve mobility problems based on SDN. We name their work as SDMA. Although SDMA can support intra-domain mobility, especially the binding placement problem and flow table downloading strategy inside one domain are studied in detail. The scenario of multiple domains was still not considered. The late work of SDMA [28] achieves the improvement to support inter-domain handover by broadcast mechanism. However, broadcasting signaling messages among multiple SDN controllers will lead to higher signaling cost, especially when the number of domain is large or mobility frequency is high.

UbiFlow [29] still adopts a fully distributed architecture in control plane to provide inter-domain mobility in urban-scale software defined networks. UbiFlow constructs a distributed hashing structure (i.e. a chord ring of controllers of different domains) and introduces a supervisory controller to maintain the binding of the controllers of MN before and after handover. With the high efficiency of chord ring lookup algorithm, UbiFlow can reduce the signaling cost during searching and updating binding information. In UbiFlow, the request message for querying the binding of MN needs through  $(\log N_D + 1)$  hops among controllers on average. Note that  $N_D$  is the number of domains. If  $N_D$  is a large number, the above query procedure leads to high end to end delay. Furthermore, UbiFlow still does not provide intra-domain handover mechanism.

### 3 Proposed H-SMM scheme

In this section, we propose a hierarchical architecture in control plane of SDN. Then we represent the intra-domain

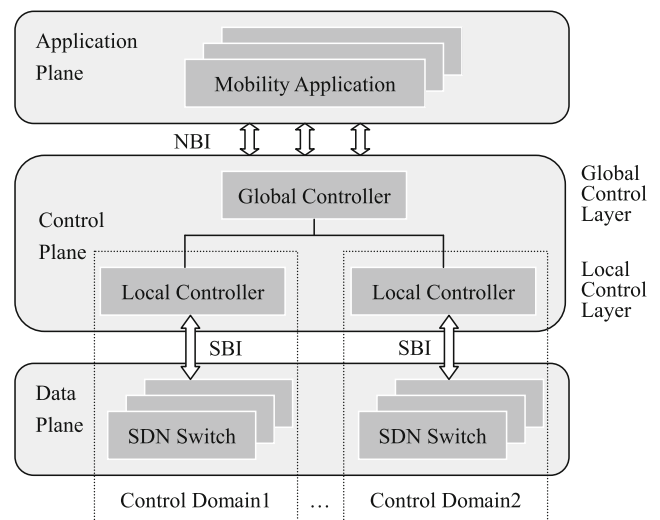
and inter-domain handover procedure, and the communication initiation procedure of our scheme.

#### 3.1 Architecture of H-SMM

Since SDN is mainly deployed in DCN, most existing SDN-based networks are composed of single control domain with one or more SDN controllers which are only responsible for data transmission. However, it is more complex in mobile networks because controllers are not only responsible for making routing decision of their own control domain as they are in DCN, but also responsible for maintaining the binding information of MN. Moreover, if MN moves across different domains, the selection of binding maintainer and binding information interaction between different controllers must be considered carefully.

Similar to the classic SDN architecture defined in the white paper [20] given by the Open Networking Foundation (ONF), all the control functions of H-SMM are realized in the control plane, which lies in the core of SDN architecture. Controllers in the control plane connect to SDN switches in the data plane with SBI, and interact with mobility applications in the application plane with northbound interfaces (NBI). SDN switches still retain the basic data forwarding function.

In order to support inter-domain handover and optimize intra-domain handover, we use the main ideas of the HMIPv6 [24], i.e. hierarchical location management mechanism, to propose a hierarchical architecture in control plane to realize binding maintenance and signaling exchange between different domains. But unlike HMIPv6, we further optimize inter-domain handover in the following procedure design to ensure that our scheme can support and optimize



**Fig. 1** Architecture of H-SMM

both types of handovers at the same time. The architecture of H-SMM is illustrated in Fig. 1.

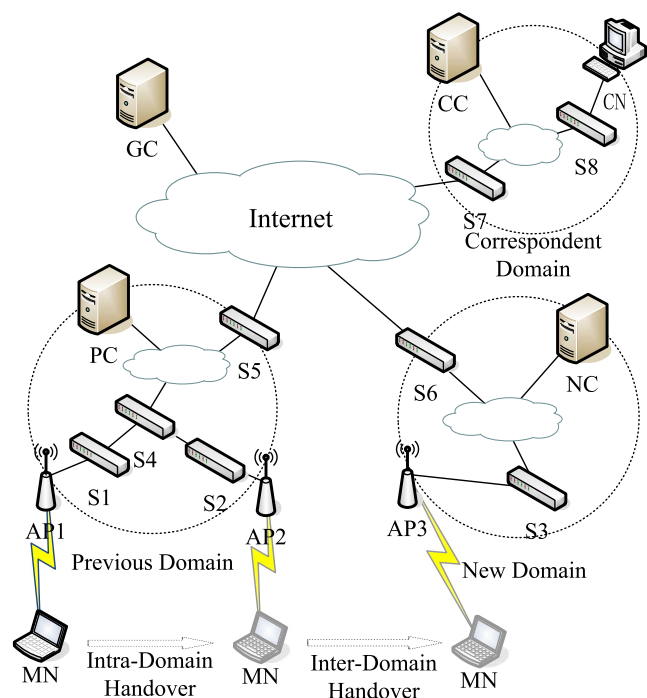
As shown in Fig. 1, the control plane of H-SMM is divided into two sub-layers, that is, the up sub-layer named global control layer which contains one global controller (GC) and the below sub-layer named local control layer which contains multiple local controllers (LC). LC is an ordinary controllers of each domain which has a local view of network status of their respective domain. Besides managing flow tables of SDN switches, each LC is also responsible for both generating a location identifier for each mobile node which has moved into its domain and maintaining bindings of these mobile nodes. Besides, we introduce GC into control plane to support and optimize intra-domain and inter-domain mobility. GC has a global view of the whole network. Then GC will responsible for routing decision of the network among domains if this network is still based on SDN architecture. Furthermore, GC always maintains a binding of each MN for controllers of other domains to query if some correspondent nodes of these domains want to communicate with MN.

### 3.2 Network topology

If only single domain is considered for mobility management, the procedures of handover and communication are relatively simple since MN and CN are always in the same domain. The related descriptions can be seen in literatures [26, 27]. We will only consider the situation that MN and CN are located in different domains in our design.

In order to facilitate the designation and analysis of our proposal, a typical topology of mobile networks based on SDN architecture is presented in Fig. 2. As shown in this figure, the whole network is divided into several control domains. Each domain is assumed to be circular in shape. There are at most three domains can be involved in handover procedures, i.e. the previous domain, the new domain and the correspondent domain. The local controllers of the above three domains are named PC, NC and CC respectively. For realizing the hierarchical architecture shown in Fig. 1, we introduce a global controller named GC into the network. GC can be placed in any location designated by system. S1, S2, S3 and S8 are access switches which are the first hop SDN switched of end nodes. S5, S6 and S7 are border switches which can connect their respective domain and the Internet. MN is connected to its access switch via a wireless link and the other parts in the network are connected with wired links. The CN is expected to be stationary in the correspondent domain. MN is movable and will perform an intra-domain handover from S1 to S2, and perform an inter-domain handover from S2 to S3.

Instead of two kinds of location identifier defined in HMIPv6 [24], we use a new routable IP address assigned by



**Fig. 2** Typical Topology of Mobile Networks based on SDN

the local controller of MNs current domain as the location identifier of MN. For example, we assume that the location identifiers of MN when MN is in the previous domain and the new domain are PIP\_MN and NIP\_MN respectively. Defining single locator of MN can simplify handover procedures of our scheme. We still use the original IP address of MN (e.g. the IP address obtained in MNs home network) as its identity identification, which can be represented by IP\_MN.

All the following analysis including scheme designation and performance evaluation is presented in Fig. 2.

### 3.3 Handover procedure

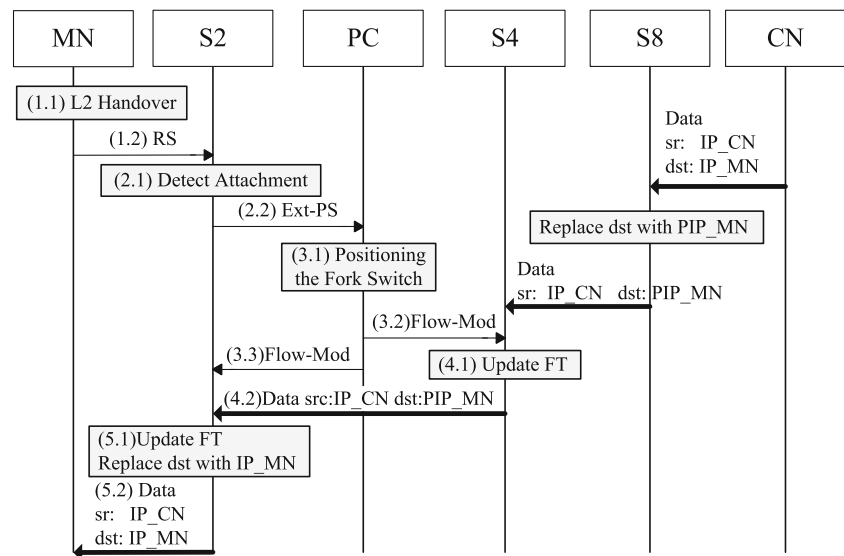
In this section, we design the handover procedures of both intra-domain and inter-domain handover, and the communication initiation procedure after MN has finished handover.

#### 3.3.1 Intra-domain handover

As shown in Fig. 2, when MN moves from AP1 to AP2, a typical intra-domain handover is taken place since the access switches of MN before and after this handover (i.e. S1 and S2) are both inside the previous domain. The intra-domain handover procedure of our scheme, which is shown in Fig. 3, is depicted as follows.

1. MN leaves AP1 and performs Layer 2 (L2) handover. After attaching to AP2, MN sends a Router Solicitation (RS) message defined in the Neighbor Discovery



**Fig. 3** Intra-domain handover procedure of H-SMM

Protocol [17] to its new access switch, i.e. S2. Note that before this handover, the location identifier of MN is PIP\_MN. The controller of the current domain (i.e. PC) maintains a binding cache entry (in form of tuple  $\langle \text{IP\_MN}, \text{PIP\_MN} \rangle$ ) of MN in its binding cache.

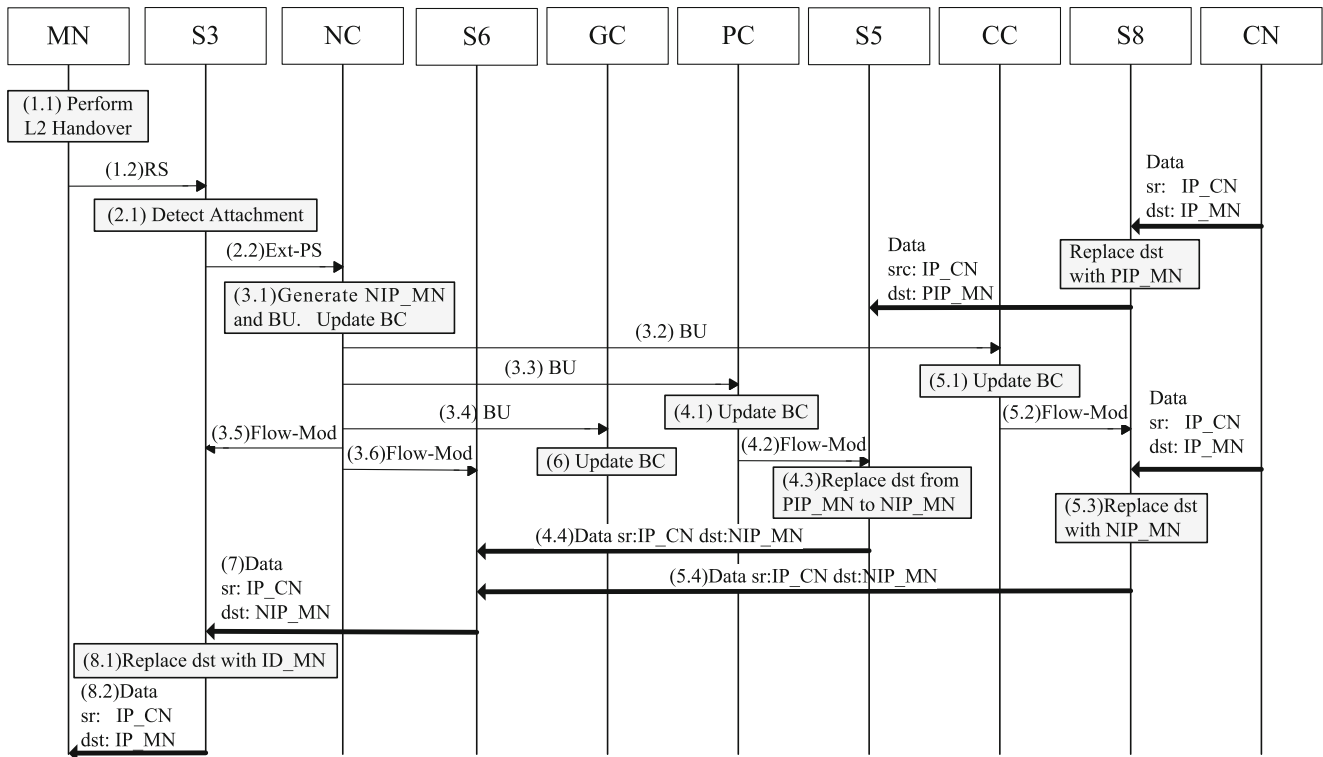
2. S2 detects the attachment of MN when it has received the RS message. Then S2 generates an extended Port-Status (Ext-PS) message which contains IP\_MN and sends the Ext-PS message to the PC to report the attachment event.
3. After receiving Ext-PS, PC looks up its binding cache with IP\_MN extracted from Ext-PS. If there is a binding cache entry related to MN was found, PC can determine this handover is an intra-domain one. After that, PC will calculate the new optimal routing from S5 to S2. PC can determine the fork switch is S4, which lies on both the previous and the new optimal routing, and locates nearest to the current location of MN. Note that the algorithm for locating fork switch can be seen in literature [26]. Meanwhile, PC sends Flow Modification (Flow-Mod) messages to all the switches from the fork switch to the access switch (i.e. from S4 to S2 in Fig. 2) to update their flow tables.
4. After receiving the Flow-Mod message, S4 updates its flow table (FT). All the packets sent from CN to MN are replaced the destination address with PIP\_MN by S8 and delivered to S4. This procedure is the same as that happened before handover took place. Since S4 is the fork switch, S4 redirects these packets to MNs new access switch according to its updated flow table.
5. After receiving the Flow-Mod message from PC, S2 will update its flow table. When data packets have arrived S2, it looks up flow table with PIP\_MN and

replacing the destination address with IP\_MN. Then these data packets return to the original packets send by CN and delivered to MN by S2 ultimately.

### 3.3.2 Inter-domain handover

Assume that the MN leaves AP2 in the previous domain and attaches to AP3 in the new domain. Therefore, a typical inter-domain handover takes place since the access switches before and after this handover (i.e. S2 and S3) are in different domains. The procedure of the inter-domain handover of H-SMM, which is shown in Fig. 4, is listed as follows.

1. When MN has finished L2 handover and attaches to the AP3, it sends a RS message to its new access switch, i.e. S3.
2. After receiving the RS message, S3 detects the attachment of MN and then sends an Ext-PS message containing IP\_MN to NC, which is the controller of the new domain, to report this attachment event.
3. NC generates a new IP address (i.e. NIP\_MN) for MN as its location identifier in the new domain. Then NC generates a BU message which contains IP\_MN and NIP\_MN. After that, NC updates its BC according to this BU. Furthermore, NC sends a BU message to GC, CC and PC respectively. Simultaneously, NC calculates the optimal route from the border switch (i.e. S6) to S3, and then sends Flow-Mod messages to all the nodes in the optimal route to update their flow tables.
4. PC updates its BC after receiving the BU message. Then PC downloads flow entry to S5 for redirecting the packets destined for MNs previous location to the new location, that is, S5 rewrites the packets, which destination address is PIP\_MN, to NIP\_MN and forwards them



**Fig. 4** Inter-domain handover procedure of H-SMM

to the new domain. PC will delete the binding about MN in BC after a certain life-time.

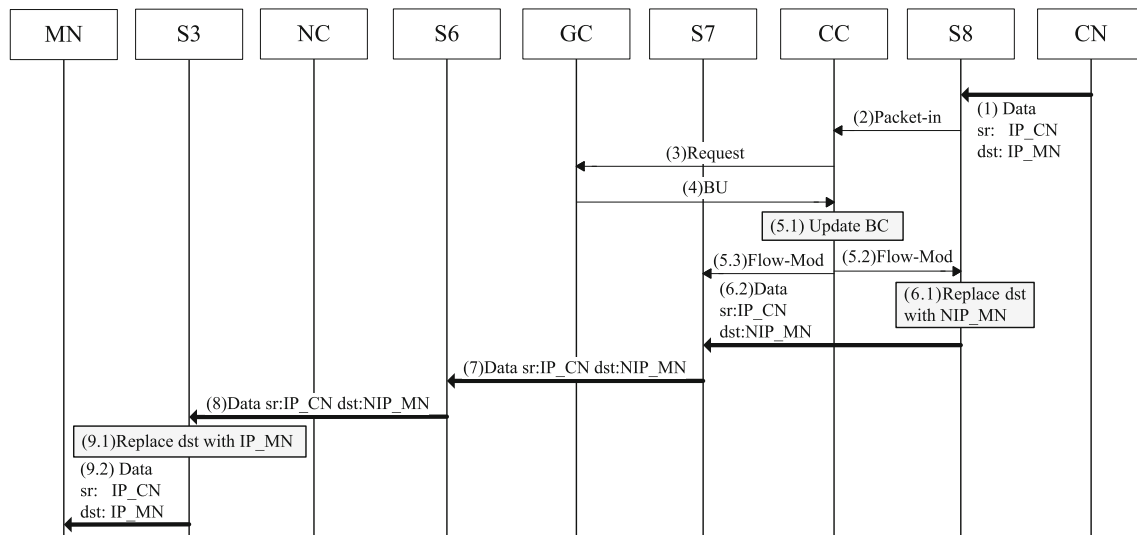
- After receiving the BU message from NC, CC updates its BC and downloads a flow entry to the access switch of CN (i.e. S8) using Flow-Mod message. Then S8 will rewrite the destination address of data packet send by CN to NIP\_MN. These packets will be routed to the border switch (i.e. S7) along the optimal route and then continue to be forwarded to S6.
- GC updates its BC after receiving the BU message form NC. Note that the step 4, 5, 6 can be performed at the same time.
- When receiving the packets which destination is NIP\_MN, S6 and the intermediate switches along the optimal route will forward them to S3 according to their flow tables.
- Finally, S3 rewrites the destination address of these packets to IP\_MN and then deliver them to MN.

After the above inter-domain handover procedure is implemented, the communications between MN and CN can be recovered. It should be explained that the above procedures are under the assumption of the stationary CN. If CN is mobile too, CN can also be treated as MN, and the handover procedures of CN are the same to that of MN mentioned above.

### 3.4 Communication initiation procedure

After the illustration of handover procedures, we will present the communication initiation procedure when MN has finished handover and in a stationary state. Take the communication initiated by CN for example. If the controller of the correspondent domain (i.e. CC) has maintained a binding cache entry for MN, then the packets send from CN to MN will be routed along the optimal route determined by the controller of respective domain. This procedure is similar to that of single control domain defined in the white paper of ONF [20]. If CC has no binding information about MN, for example CC is just start up, the communication initiation procedure is much more complex than the above condition. The procedure of communication initiation shown by Fig. 5 is presented as follows.

- Similar to the communication of traditional wire network, the destination IP address and the source IP address (represented by dst and src respectively in Fig. 5) of the original packets send from CN to MN is IP\_MN and IP\_CN. These packets will be delivered to the access switch of CN (i.e. S8).
- Since there is no flow entry about MN in S8’s flow table, S8 have no information about the current location of S8. S8 will send a Packet-in message, which contains



**Fig. 5** Communication Initiation Procedure of H-SMM

- the data packet, to its controller CC for querying flow entry to forward these packets.
- In the proposed scheme, MN can move among different domains, which makes CC do not know the exact location of MN. Hence, CC sends a Request message to the global controller (i.e. GC) to get binding information of MN.
  - GC queries its binding cache with IP\_MN and gets the binding cache entry of MN. Then GC generates a reply message containing the BU of MN and returns it to CC.
  - After receiving the BU of MN from GC, CC updates its binding cache. Then CC calculates the optimal routing form S8 to the border switch (i.e. S7) and downloads flow entry into all related SDN switches with Flow-Mod messages.
  - S8 updates its flow table according to the Flow-Mod message send by CC. After that, S8 replaces destination address of the original data packets send by CN with the current location identifier of MN, i.e. replaces the destination address from IP\_MN to NIP\_MN and forward the packet according to the action field in its flow table. Since CC has updated related SDN switches in the correspondent domain, all data packets will be routed to S7 alone the optimal route.
  - Data forwarding methods during the inter-domain route from S7 to S6 depends on the architecture of the network between two different SDN control domains. If the network between different domains is still based on SDN, the GC will make routing decision according its global view of the whole network and downloads flow entry to related switches in the optimal route form S7 to S6. If the network between different domains is a traditional network, GC can set up a tunnel between

- S7 and S5 in advance to forward data packet in an IP-in-IP way. We assume that the latter is more accordance with reality. The communication solutions of SDN and traditional mixed network are not the key of our proposal.
- After arriving the border switch of the new domain (i.e. S6), all the packets still can be forward to S3 along the optimal route since NC has made routing decision and downloaded flow entry to the related switches between S6 and S3 after MN attached to S3.
  - Ultimately, S3 rewrites the destination address of these packets from NIP\_MN to ID\_MN and then delivers packets to MN.

Note that data forwarding procedure from MN to CN is similar to the above, and we won't describe it in detail any more.

## 4 Performance evaluation

In this chapter, the characteristic and performances of H-SMM are evaluated and compared. In qualitative analysis, we compared H-SMM with some classic mobility protocols based on traditional network, such as MIPv6 and PMIPv6. The former has been an extended protocol of the IPv6; the latter is the only network-based mobility protocol that has been support by LET. For fair comparison and comprehensive analysis, two typical SDN-based mobility management schemes are also selected as contrasts in both qualitative and quantitative analysis. SDMA [28] is a representative of intra-domain mobility schemes, which gives a simple inter-domain handover procedure with a broadcast way. UbiFlow



**Table 1** Qualitative analysis and comparison

	MIPv6	PMIPv6	SDMA	UbiFlow	H-SMM
Intra-domain	No	Support	Support	No	Support
Inter-domain	Support	No	Support	Support	Support
Modify stack	Yes	No	No	No	No
Architecture	Traditional	Traditional	SDN	SDN	SDN
Data routing	Detour	Detour	Optimal	Optimal	Optimal
Communication manner	Tunnel	Tunnel	Rewrite destination	Redirect data flow	Rewrite destination
Method of query BU	–	–	Broadcast	Chord lookup	Request -Reply

[29] is a representative of inter-domain mobility scheme which does not involved in intra-domain scenario.

#### 4.1 Qualitative analysis

First we present a qualitative analysis and comparison of the characteristics of H-SMM and the other four comparative schemes in Table 1.

As illustrated in Section 3, H-SMM can support both intra-domain handover and inter-domain handover, which makes it an overall and completed mobility scheme. There is no need to modify end hosts protocol stacks in H-SMM. This is because SDN controllers and switches perform all the mobility management operations.

Like other SDN-based schemes, H-SMM separates control function from data plane. All data packets can be transferred by SDN switches along the optimal route, instead of being detoured to some kind of anchor point.

Unlike two other SDN-based mobility schemes, H-SMM adopts a hierarchical control plane and improved procedures to optimize both inter-domain and intra-domain handover. The intra-domain handover of H-SMM was simplified because MN does not have to reconfigure a location identifier. The inter-domain handover was also optimized by using a request-reply method for BU exchanging. Furthermore, the packet routing of H-SMM needs rewriting of destination address by the access switches of both MN and CN.

#### 4.2 Calculative analysis of signaling cost

We will present a theoretical evaluation and comparison of H-SMM and the other two SDN-based mobility schemes, SDMA and UbiFlow, according to their signaling cost. Note that intra-domain handover procedure has not been considered in UbiFlow. Accordingly, only H-SMM and SDMA

**Table 2** Parameters of calculative analysis

Perimeter	Explanation	Value
$H_w$	One hop of wire link. This is a unit value.	1
$H_{wl}$	One hop of wireless link.	5
$H_{Inter}$	Average distance between two domains.	$\beta H_w$
$H_{Ctl-S}$	Average hops between a switch and its controller.	$H_w$
$H_{Ctl-Ctl}$	Average hops between two SDN controllers.	$(\beta + 2)H_w$
$H_{MN-AS}$	Average hops between MN and its access switch.	$H_w + H_{wl}$
$H_{CN-AS}$	Average hops between CN and its access switch.	$H_w$
$H_{AS-BS}$	Average hops between an access switch and the border switch of current domain.	$2H_w$
$L_{RS}$	Length of RS message (bytes)	52
$L_{BU}$	Length of BU message (bytes)	84
$L_{Packet-In}$	Length of Packet-In message (bytes)	80
$L_{Flow-Mod}$	Length of Flow-Mod message (bytes)	116
$L_{Ext-PS}$	Length of Ext-PS message (bytes)	80
$L_{Request}$	Length of Request message (bytes)	64
$L_{Notify}$	Length of Notify message (bytes)	68

are considered in the performance analysis of inter-domain handover.

Our performance evaluation is based on the network topology shown in Fig. 2. The whole network is composed of several control domains, and the domain number  $N_D$  is variable. In each domain, there will be one SDN controller and several SDN switches. The average hop counts between two border switches of different domains is defined as a variable  $\beta$ . We assume that unit transmission cost in a wireless link is 5 times of that in a wire link, and every SDN switch connects directly to the controller in one domain. The explanations and values of the other perimeters of our calculative analysis are listed in Table 2. Some of these values are referred to literature [10].

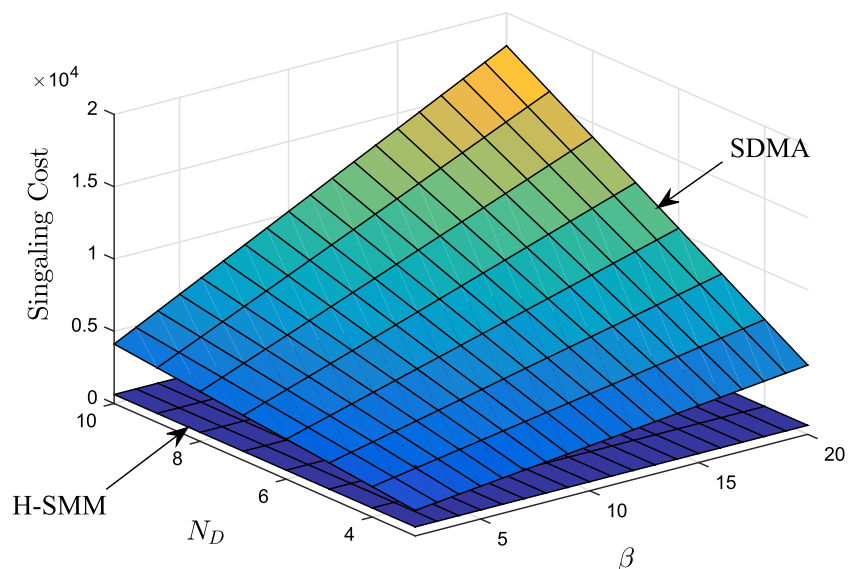
Signaling exchanges between different nodes will result in an additional waste of bandwidth and a reduction of bandwidth utilization [7], especially when handover happens frequently or the number of mobile nodes is numerous. This kind of cost caused by signaling is called signaling cost, which is an important performance metric to measure the cost and effectiveness of a mobility scheme.

The signaling cost of messages  $x$  sent from the source node  $A$  to the source node  $B$  can be calculated as the product of the length of messages  $x$  (defined as  $L_x$ ) and the hop distance from  $A$  to  $B$  (defined as  $H_{A-B}$ ) [11]. The signaling cost of messages  $x$  is represented as  $C_{A-B}^x$ , which can be calculated as follows:

$$C_{A-B}^x = L_x \times H_{A-B} \quad (1)$$

Overall signaling cost of a handover procedure can be calculated by summing up all cost of each signaling message. We will analyze and compare the overall signaling cost during once handover or once communication initiation procedure of the above mentioned SDN-based mobility schemes.

**Fig. 6** Signaling cost of single intra-domain handover versus  $N_D$  and  $\beta$



#### 4.2.1 Signaling cost of intra-domain handover

As can be seen in Fig. 3, the location identifier of MN is generated by the controller of the current domain, which will not regenerate a new one after MN moves inside of this domain. Then registration of new binding among different controllers still can be avoided. The overall signaling cost during once intra-domain handover in H-SMM is calculated as follows:

$$\begin{aligned} C_{Intra-HO}^{H-SMM} &= C_{MN-S2}^{RS} + C_{S2-PC}^{Ext-PS} + C_{PC-S2}^{Flow-Mod} + C_{PC-S4}^{Flow-Mod} \\ &= L_{RS}(H_w + H_{wl}) + L_{Ext-PS}H_w + 2L_{Flow-Mod}H_w \end{aligned} \quad (2)$$

As for the SDMA, location identifier of MN is generated by the access switch, which leads to reconfiguring address by MNs new access switch and broadcasting BU messages by the controller even if MN moves inside of the same domain. The overall signaling cost of intra-domain handover in SDMA is calculated as follows:

$$\begin{aligned} C_{Intra-HO}^{SDMA} &= C_{MN-S2}^{RS} + C_{S2-PC}^{BU} + 3C_{PC-S}^{Flow-Mod} \\ &\quad + (N_D - 1)C_{PC-CC}^{BU} + 3C_{CC-S}^{Flow-Mod} \\ &= L_{RS}(H_w + H_{wl}) + 6L_{Flow-Mod}H_w \\ &\quad + (N_D - 1)L_{BU}(H_{Inter} + 2H_w) + L_{BU}H_w \end{aligned} \quad (3)$$

After substituting the parameters in Table 2 into formulas (2) and (3), the comparison of signaling costs of intra-domain handover in H-SMM and SDMA is shown in Fig. 6. We can see that the signaling cost of H-SMM is a constant. This means the value will not increase with the rise of either  $N_D$  or  $\beta$ . In contrast to H-SMM, SDMA has higher signaling cost during intra-domain handover. This trend will become more pronounced with the increase of  $N_D$  and  $\beta$ . The cost value of SDMA rises drastically when  $N_D$  and  $\beta$  increasing at the same time. We can conclude that H-SMM

can keep lower signaling cost, which will not change with the varying of  $N_D$  and  $\beta$ .

#### 4.2.2 Signaling cost of inter-domain handover

According to the procedure designed in Fig. 4, BU message will be send from NC to other three controllers, i.e. GC, PC and CC. The overall signaling cost of H-SMM during inter-domain handover is calculated as follows:

$$\begin{aligned}
 C_{Inter-HO}^{H-SMM} &= C_{MN-S3}^{RS} + C_{S3-NC}^{Ext-PS} + C_{NC-PC}^{BU} + C_{NC-S3}^{Flow-Mod} \\
 &\quad + C_{NC-S6}^{Flow-Mod} + C_{PC-S5}^{Flow-Mod} + C_{CC-S8}^{Flow-Mod} \\
 &= L_{RS}(H_w + H_{wl}) + L_{Ext-PS}H_w + 4L_{Flow-Mod}H_w \\
 &\quad + 3L_{BU}(H_{Inter} + 2H_w)
 \end{aligned} \tag{4}$$

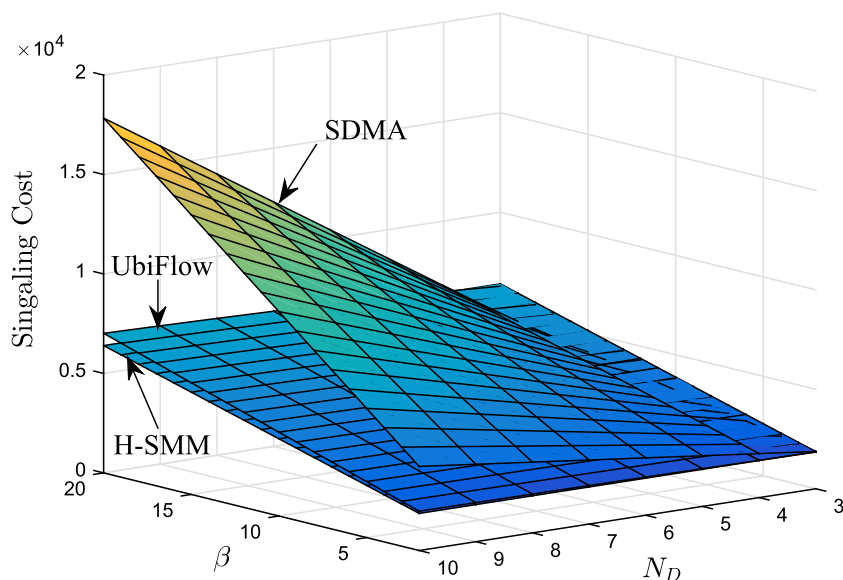
During inter-domain handover of SDMA, the controller in the current domain of MN will broadcast BU messages to all the other controllers, which will leads to large number of signaling transmitted among different domains. The overall signaling cost of inter-domain handover in SDMA can be calculated as follows:

$$\begin{aligned}
 C_{Inter-HO}^{SDMA} &= C_{MN-S3}^{RS} + C_{S3-NC}^{Ext-PS} + (N_D - 1)C_{Cil-Cil}^{BU} \\
 &\quad + 3C_{NC-S}^{Flow-Mod} + 3C_{CC-S}^{Flow-Mod} + C_{PC-S5}^{Flow-Mod} \\
 &= L_{RS}(H_{wl} + H_w) + L_{Ext-PS}H_w + 7L_{Flow-Mod}H_w \\
 &\quad + (N_D - 1)L_{BU}(H_{Inter} + 2H_w)
 \end{aligned} \tag{5}$$

In UbiFlow, it is necessary to obtain the address of previous access switch by querying a chord ring, which is constructed by controllers, for redirecting the data flow. The overall signaling cost of inter-domain handover in UbiFlow is calculated as follows:

$$\begin{aligned}
 C_{Inter-HO}^{UbiFlow} &= C_{MN-PC}^{RS} + (\log N_D + 1) \times C_{Cil-Cil}^{Request} + C_{Cil-Cil}^{BU} \\
 &\quad + C_{NC-PC}^{Notify} + C_{PC-S5}^{Flow-Mod} + C_{S6-NC}^{Packet-In} + 3C_{NC-S}^{Flow-Mod} \\
 &= ((\log N_D + 1)L_{Request} + L_{BU} + L_{Notify})(T_{Inter} + 2H_w) \\
 &\quad + L_{RS}(H_w + H_{wl}) + L_{Packet-In}H_w + 4L_{Flow-Mod}H_w
 \end{aligned} \tag{6}$$

**Fig. 7** Signaling cost of single inter-domain handover versus  $N_D$  and  $\beta$



By substituting related parameters, the comparison of signaling costs during inter-domain handover is shown as Fig. 7.

In Fig. 7, the signaling cost of SDMA is significant higher than the other two schemes and the value rises with  $N_D$  and  $\beta$  obviously. This is caused by broadcasting binding information to all the other controllers in SDMA. UbiFlow has lower signaling cost by using a more efficient chord ring lookup algorithm. The signaling cost of H-SMM is the lowest one among three schemes. This is because there are only three BU messages send between domains during inter-domain handover in H-SMM.

#### 4.2.3 Signaling cost of inter-domain handover

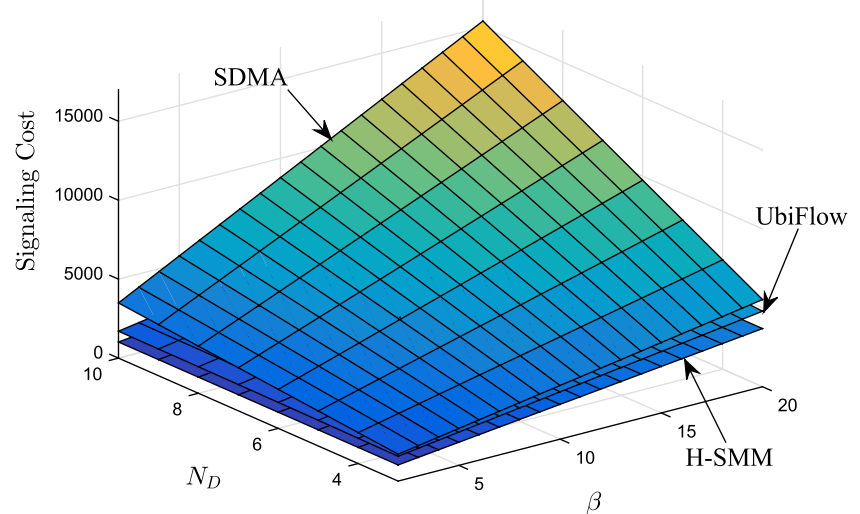
In H-SMM, CC requests BU message from GC and then downloads flow entries to related switch for forwarding data packet. The overall signaling cost of H-SMM during communication initiation is calculated as follows:

$$\begin{aligned}
 C_{Commu}^{H-SMM} &= C_{S8-CC}^{Packet-In} + C_{CC-GC}^{Request} + C_{GC-CC}^{BU} + 3 \times C_{CC-S}^{Flow-Mod} \\
 &= L_{Packet-In}H_w + 3L_{Flow-Mod}H_w \\
 &\quad + (L_{Request} + L_{BU})(H_{Inter} + 2H_w)
 \end{aligned} \tag{7}$$

In SDMA, BU message only contains the IP address of the controller of the domain where MN resides. Then CC has to query the location of MN from NC. The overall signaling cost of H-SMM during communication initiation is calculated as follows:

$$\begin{aligned}
 C_{Commu}^{SDMA} &= C_{S8-CC}^{Packet-In} + (N_D - 1)C_{CC-PC}^{Request} + C_{PC-CC}^{BU} \\
 &\quad + 3C_{CC-S}^{Flow-Mod} + C_{S6-NC}^{Packet-In} + 3C_{NC-S}^{Flow-Mod} \\
 &= ((N_D - 1)L_{Request} + L_{BU})(\beta + 2) \\
 &\quad + 2L_{Packet-In} + 6L_{Flow-Mod}
 \end{aligned} \tag{8}$$

**Fig. 8** Signaling cost of communication initiation versus  $N_D$  and  $\beta$



UbiFlow still has to perform chord ring lookup algorithm for relaying data flow. The overall signaling cost of H-SMM during communication initiation is calculated as follows:

$$\begin{aligned}
 C_{Commu}^{UbiFlow} &= C_{S8-CC}^{Packet-In} + (\log N_D + 1) \times C_{Ctl-Ctl}^{Request} + C_{Ctl-Ctl}^{BU} \\
 &+ 3C_{CC-S}^{Flow-Mod} + C_{S6-NC}^{Packet-In} + 3C_{NC-S}^{Flow-Mod} \quad (9) \\
 &= ((\log N_D + 1)L_{Request} + L_{BU})(T_{inter} + 2H_w) \\
 &+ (2L_{Packet-In} + 6L_{Flow-Mod})H_w
 \end{aligned}$$

The comparison of signaling costs during communication initiation stage is shown as Fig. 8.

In Fig. 8, SDMA has the highest signaling cost, which is raised obviously with  $N_D$  and  $\beta$ . This is because CC still needs to perform a broadcast for querying the binding of MN. The signaling cost of UbiFlow is lower and raises more gently. This is owing to the high efficiency of the chord ring lookup algorithm of UbiFlow. H-SMM still has the lowest signaling cost. The reason is that the signaling interaction between CC and GC can be realized by a request and a reply message.

### 4.3 Simulation and result

We evaluated the performance advantages of our proposed H-SMM and compared with SDMA and UbiFlow by implementing the above schemes based on the most famous SDN simulator, i.e. Mininet [1]. In order to configure

our simulation environment, we installed virtual machine (VM) image of Mininet version 2.2.1, which has integrated Ubuntu 14.04, in Virtual Box hypervisor. All the virtual SDN switches have been built by Open vSwitch (OvS) and all the SDN controllers are based on POX. We used Iperf as the performance testing tool to obtain the time related metrics of each message.

In simulations, the network topology was still constructed as shown in Fig. 2. Handover procedures of MN were simulated by detaching MN from one SDN switch and then attaching it to another. We have implemented handover and communication related functions in different controllers according to the above mentioned designation.

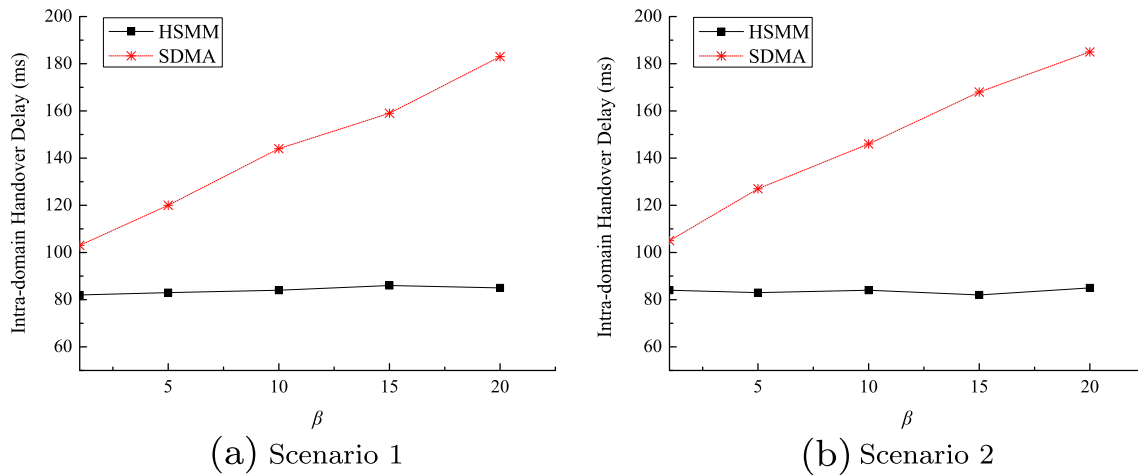
In order to verify the performances of the three schemes in different scale of SDN-based networks, we set two scenarios according to the different value of  $N_D$ .

- Scenario 1:  $N_D = 10$ . This scenario has a characteristic of smaller number of domains, which means the scale of the whole network is also small.
- Scenario 2:  $N_D = 100$ . This scenario has a characteristic of larger number of domains, which means the scale of the whole network is also large.

For convenient of experiment, we assume that CN sends data packets to MN continuously during the whole simulation time, and all packets will not be lost in wired and

**Table 3** Parameters of simulations

Perimeter	Explanation	Value
$T_w$	Average delay of one hop wire link (ms)	2
$T_{wl}$	Average delay of one hop wireless link (ms)	10
$T_{L2}$	Average delay of Layer 2 handover (ms)	50
$B_w$	Average bandwidth of wire link (Mbps).	100
$B_{wl}$	Average bandwidth of wireless link (Mbps).	10
$L_{Data}$	Length of data packet (bytes).	56
$R_{Data}$	Packet sending rate (Packets per second).	10



**Fig. 9** Average delay of single intra-domain handover versus  $\beta$

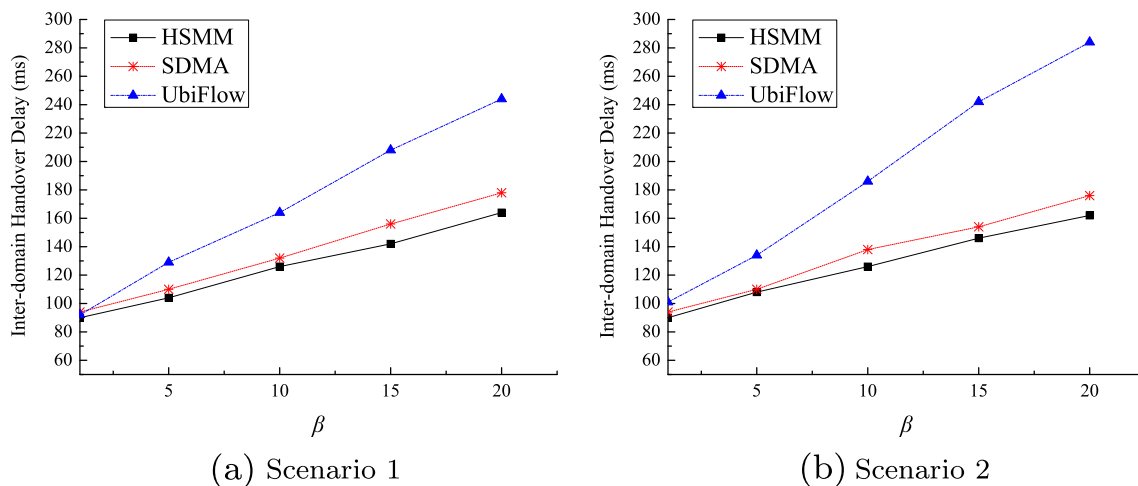
wireless links and will reach the destination without failure. Note that we ignore the processing delay of data and signaling message in our experiment. The parameters required for simulations are shown in Table 3. The settings of these parameters are referred to literature [15].

Handover delay is one of the most important metrics to measure the performance of a mobility management scheme, especially for delay sensitive applications. Handover delay can directly reflect the length of communication interruption during handover procedure. In order to obtain handover delay in our experiment, we recorded the simulation time  $T_1$  of the last packet received by MN before the interruption, and the simulation time  $T_2$  of the first packet received by MN after the interruption. The difference between  $T_2$  and  $T_1$  is the delay of single handover. In order to reduce the error, we ran each scheme three times and counted the average value of handover delay.

#### 4.3.1 Intra-domain handover delay

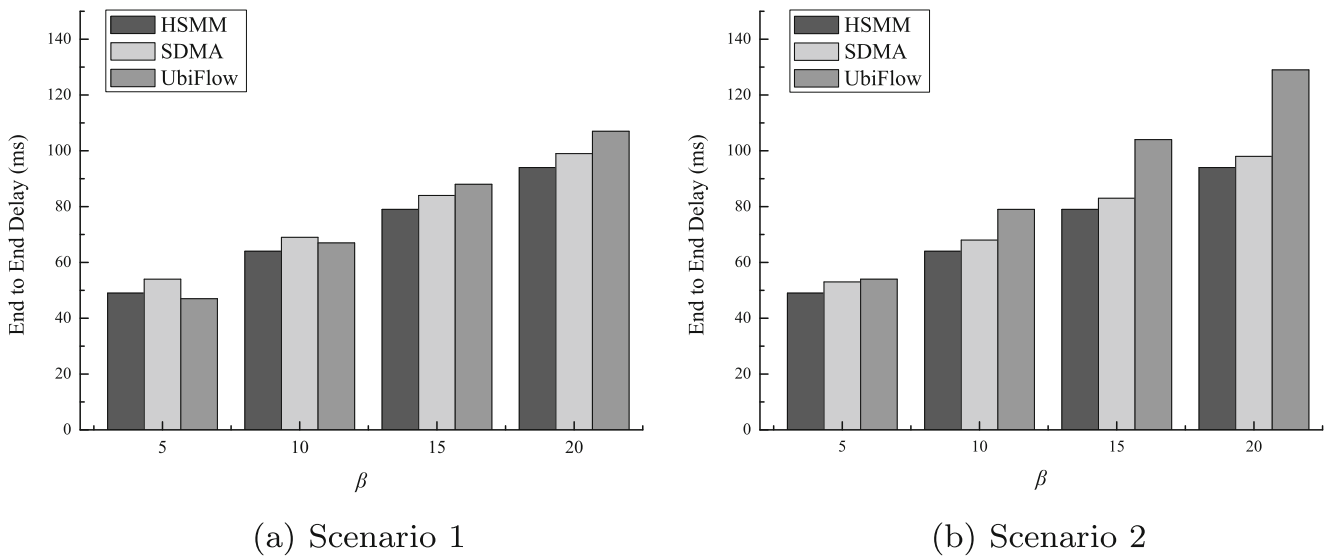
First we analyze the handover delay during an intra-domain handover in SDMA and H-SMM. Variety of handover delay with the change of  $\beta$  in the two scenarios is shown in Fig. 9.

It can be seen from Fig. 9 that intra-domain handover delay of H-SMM is lower and the values will not increase with the rise of  $\beta$ . The phenomenon and reason are both similar to signaling cost shown in Fig. 6. On the contrary, SDMA will change location identifier and perform binding update with broadcast by the current controller of MN. This leads to higher intra-domain handover delay, and this disadvantage will be more evident with the rise of  $\beta$ . By comparing Figs. 9a and b, we can see that SDMA has similar handover delay in both scenarios. This is because the average one way delay of broadcast messages has nothing to do with  $N_D$ .



**Fig. 10** Average delay of single inter-domain handover versus  $\beta$





**Fig. 11** Average end to end delay in communication initiation versus  $\beta$

#### 4.3.2 Inter-domain handover delay

Then we analyze the handover delay of a single inter-domain handover of different schemes which were shown in Fig. 10.

As shown in Fig. 10, inter-domain handover delay of all three SDN-based schemes will increase with the rise of  $\beta$ , the reason is that inter-domain interaction for propagate binding information is inevitable. UbiFlow has to perform  $(\log N_D + 1)$  steps lookup on the chord ring constructed by controllers to get the binding information before handover, which leads to obviously rising of handover delay with the raise of  $N_D$  and  $\beta$ . H-SMM and SDMA gain slower rising trend of handover delay by using a request and response mechanism to get related binding information. By downloading flow entry to the related switches beforehand during the handover procedure, H-SMM has lowest handover delay. Even though this advantage is not obvious when the value of  $\beta$  is small, it will be more pronounced with the increase of  $\beta$ .

#### 4.3.3 End to end delay of communication initiation

Last, we evaluate the end to end delay of the first data packet in communication initiation. This performance parameter is critical to time-sensitive applications. Most of existing SDN-based mobility schemes can obtain optimal route. But during the routing of the first data packet from end to end in SDN-based mobile network, the procedures of flow table updating and binding query procedure are necessary for routing decision. These processes will cause extra delay and then reduce the real-time performance of proposed schemes.

We tested and compared the end to end delay from CN to MN during communication initiation of different mobility schemes. As we can see the results shown in Fig. 11, the end to end delay of the first packet of all these schemes will raise with increase of  $\beta$ . This is due to the binding query procedure among different domains of these schemes. By optimizing the handover procedure, H-SMM still has the advantage of having lowest delay, which rises gently with the extending of  $\beta$ . SDMA lacks of updating flow table of related switches in advance, which leads to weak inferiority. By comparing Fig. 11a and b, we can see that the end to end delay of UbiFlow rises quickly with both  $\beta$  and  $N_D$ . This disadvantage is due to the chord ring lookup process in each communication initiation.

## 5 Conclusion

In this paper, we proposed an H-SMM mobility management scheme based on SDN paradigm. Characteristic analysis demonstrates that H-SMM is an overall mobility scheme which can support intra-domain and inter-domain handover. Hosts involvement in mobility management can be avoided. Calculative analysis validates that H-SMM has lower signaling cost in handover and communication procedures. Simulation results illustrated that both intra-domain and inter-domain handover delay of H-SMM is smaller. End to end delay of communication initiation is still lower than the contrast schemes. The advantages of both the above delay make H-SMM more appropriate to delay sensitive applications. Evaluation also indicated that all the above performance advantages of H-SMM are more evident with

rising of domain numbers and average hops between different domains. Moreover, we can conclude from the single handover experiment that H-SMM will be more dominant when mobility frequency is higher.

**Acknowledgements** This research is supported by the National Natural Science Foundation of China (U1736216, 61472001, 61702231, U1405255), the National Key Research and Development Program of China (2017YFB1400703), the Key Research and Development Plan Project of Jiangsu Province (BE2015136) and the Natural Science Foundation of Jiangsu Province (BK20170556).

We gratefully acknowledge Prof. Liangmin Wang, who pointed out the research direction, planned the progress and organized the paper. Under his guidance, Xing Yin designed the scheme, wrote the paper and performed the experiments. We also acknowledge Heng Pan for her innovation idea of constructing a hierarchical structure in the control plane. Furthermore, we acknowledge Shunrong Jiang for his modification of our paper.

## References

1. Mininet: An Instant Virtual Network on your Laptop (or other PC). <http://mininet.org/> (2015). Accessed 23 Sept (2016)
2. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast 2015–2020 Q&A. <http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-forecast-qa.html> (2017). Accessed 10 June (2017)
3. Al Sabbagh A, Papatwibul P, Banjar A, Braun R (2013) Optimization of the OpenFlow controller in wireless environments for enhancing mobility. In: IEEE conference on local computer networks, pp 930–935
4. Calderon M, Bernardos CJ, Bagnulo M, Soto I, Oliva ADL (2006) Design and experimental evaluation of a route optimization solution for NEMO. *IEEE J Select Areas Commun* 24(9):1702–1716
5. Contreras LM, Cominardi L, Qian H, Bernardos CJ (2016) Software-defined mobility management: Architecture proposal and future directions. *Mob Netw Appl* 21(2):226–236
6. Elgendi I, Munasinghe KS, Jamalipour A (2016) A three-tier SDN based distributed mobility management architecture for DenseNets. In: IEEE international conference on communications, pp 1–6
7. Giust F, Bernardos CJ, Oliva ADL (2014) Analytic evaluation and experimental validation of a network-based IPv6 distributed mobility management solution. *IEEE Trans Mob Comput* 13(11):2484–2497
8. Gladisch A, Daher R, Tavangarian D (2014) Survey on mobility and multihoming in future internet. *Wirel Pers Commun* 74(1):45–81
9. Gundavelli S, Leung K, Devarapalli V, Chowdhury K, Patil B (2008) Proxy Mobile IPv6. Tech. rep., IETF, RFC 5213
10. Kim SM, Choi HY, Han YH, Min SG (2015) An adaptation of proxy mobile IPv6 to openflow architecture over software defined networking. *IEICE Trans Commun* 98(4):596–606
11. Lee JH, Ernst T, Chung TM (2010) Cost analysis of IP mobility management protocols for consumer mobile devices. *IEEE Trans Consum Electron* 56(2):1010–1017
12. Li Y, Wang H, Liu M, Zhang B, Mao H (2014) Software defined networking for distributed mobility management. In: GLOBE-COM workshops, pp 885–889
13. Liu D, Seite P (2015) Distributed mobility management: Current practices and gap analysis. Tech. rep., IETF, RFC 7429
14. Liyanage M, Gurtov A, Ylianttila M (2015) Software defined mobile networks (SDMN): Beyond LTE network architecture. Wiley Publishing, pp 9–10
15. Makaya C, Pierre S (2008) An analytical framework for performance evaluation of IPv6-based mobility management protocols. *IEEE Trans Wirel Commun* 7(3):972–983
16. Mckeown N, Anderson T, Balakrishnan H, Parulkar G, Peterson L, Rexford J, Shenker S, Turner J (2008) OpenFlow: enabling innovation in campus networks. *Acm Sigcomm Comput Commun Rev* 38(2):69–74
17. Narten T, Simpson WA, Nordmark E, Soliman H (2007) Neighbor Discovery for IP version 6 (IPv6). Tech. rep., IETF, RFC 4861
18. Nguyen TT, Bonnet C, Harri J (2016) SDN-based distributed mobility management for 5G networks. In: Wireless communications and networking conference
19. Nunes BAA, Mendonca M, Nguyen XN, Obraczka K, Turletti T (2014) A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Commun Surv Tutor* 16(3):1617–1634
20. Open Networking Foundation: Software-Defined Networking: The New Norm for Networks. <http://www.opennetworking.org/component/content/article/46-sdn-resources/sdn-library/whitepapers/816-software-defined-networking-the-new-norm-for-networks>(2012). Accessed 6 June (2017)
21. Pentikousis K, Wang Y, Hu W (2013) Mobileflow: Toward software-defined mobile networks. *IEEE Commun Mag* 51(7):44–53
22. Perkins C, Johnson D, Arkko J (2011) Mobility support in IPv6. Tech. rep., IETF, RFC 6275
23. Papatwibul P, Banjar A, Sabbagh AAL, Braun R (2014) Developing an application based on OpenFlow to enhance mobile IP networks. In: Local computer networks workshops, pp 936–940
24. Soliman H, Castelluccia C, Elmalki K, Bellier L (2008) Hierarchical mobile IPv6 (HMIPv6) mobility management. Tech. rep., IETF, RFC 5380
25. Schmidt TE, Waehlich M, Koodli R, Fairhurst G, Liu D (2014) Multicast Listener Extensions for Mobile IPv6 (MIPv6) and Proxy Mobile IPv6 (PMIPv6) Fast Handovers. Tech. rep., IETF, RFC 4711
26. Wang Y, Bi J (2014) A solution for IP mobility support in software defined networks. In: International conference on computer communication and networks, pp 1–8
27. Wang Y, Bi J (2015) Software-defined mobility support in IP networks. *Comput J* 59(2):159–177
28. Wang Y, Bi J, Zhang K (2015) Design and implementation of a software-defined mobility architecture for IP networks. *Mob Netw Appl* 20(1):40–52
29. Wu D, Arkhipov DI, Asmare E, Qin Z, Mccann JA (2015) Ubi-Flow: Mobility management in urban-scale software defined IoT. In: IEEE conference on computer communications (INFOCOM), pp 208–216
30. Xia W, Wen Y, Foh CH, Niyato D, Xie H (2015) A survey on software-defined networking. *IEEE Commun Surv Tutor* 17(1):27–51
31. Yang M, Li Y, Jin D, Zeng L, Wu X, Vasilakos AV (2014) Software-defined and virtualized future mobile and wireless networks: A survey. *Mob Netw Appl* 20(1):4–18
32. Yap KK, Huang TY, Kobayashi M, Chan M, Sherwood R, Parulkar G, McKeown N (2009) Lossless handover with n-casting between WiFi-WiMAX on OpenRoads. *ACM Mobicom (Demo)* 12(3):40–52
33. Yap KK, Sherwood R, Kobayashi M, Huang TY, Chan M, Handigol N, Mckeown N, Parulkar G (2010) Blueprint for introducing innovation into wireless mobile networks. In: ACM SIGCOMM

workshop on virtualized infrastructure systems and architectures, pp 25–32

34. Yin X, Wu GX, Dong YQ, Zhang SF (2015) Route optimization scheme for nested mobile networks based on extending neighbor discovery protocol. *J Commun* 36(4)
35. Zuniga JC, Bernardos CJ, Oliva ADL, Melia T, Rui C, Reznik A (2013) Distributed mobility management: A standards landscape. *IEEE Commun Mag* 51(3):80–87



**Xing Yin** received the B.E. degree in computer application technology from Jiangsu University, Zhenjiang, China, in 2005. He received the Ph.D. degree in computer system architecture with the School of Computer Science and Engineering, Southeast University, Nanjing, China, in 2016. He is currently an assistant professor with the School of Electrical and Information Engineering, Jiangsu University, Zhenjiang, China. He is a post-doc who is supervised by profes-

sor Liangmin Wang. His research interests include mobile Internet, software defined networking, and network virtualization.



**Liangmin Wang** received the B.S. degree in computational mathematics from Jilin University, Changchun, China, in 1999 and the Ph.D. degree in cryptology from Xidian University, Xi'an, China, in 2007. He is currently a Full Professor with the School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His current research interests include security protocols and Internet of Things.



**Shunrong Jiang** received the B.E. degree in information engineering from Chongqing University, Chongqing, China, in 2008. He received the Ph.D. degree in communication and information systems with the School of Telecommunications Engineering, Xidian University, Xi'an, China, in 2016. He joined the Department of Internet of Things, Jiangsu University as an assistant professor in August 2016. His research interests include security and privacy for wireless networks, cloud computing, etc.