Leveraging Protocol-Oblivious Forwarding (POF) to Realize NFV-Assisted Mobility Management

Kai Han, Shengru Li, Junjie Liu, Shaofei Tang, Huibai Huang, Sicheng Zhao, Zuqing Zhu[†]
School of Information Science and Technology,
University of Science and Technology of China, Hefei, China
[†]Email: {zqzhu}@ieee.org

***Abstract*—In the Internet, to enable emerging network services such as e-Health to be delivered to numerous people anytime and anywhere, mobility management plays an important role. In this work, we leverage the protocol-oblivious forwarding (POF) to design a novel network system that can realize highly-efficient mobility management and overcome the drawbacks of existing approaches. Moreover, we explore the forwarding plane programmability provided by POF to realize dynamic virtual network function (vNF) deployment for traffic adaption. We implement our system and conduct experiments to verify that it functions well for NFV-assisted mobility management and achieves reduced handover latency and enhanced QoE.**

***Index Terms*—Software-defined networking (SDN), Protocol-oblivious forwarding (POF), Network function virtualization (NFV), Mobility management.**

I. Introduction

The enormous success of heterogeneous radio access technologies has promoted a revolutionary change on the Internet to architect a mobile and wireless world with exponentially increasing mobile terminals (*e.g.*, smart phones). Hence, emerging network services, such as e-Health and e-Commerce, can potentially be delivered to numerous people anytime and anywhere [1, 2]. To realize this, a powerful mobility management mechanism is essential. For instance, to satisfy the rigid quality-of-service (QoS) and quality-of-experience (QoE) requirements of e-Health applications, the network needs to not only maintain the ongoing sessions without any interruption when the mobile users switch their access points (APs) frequently, but also adjust the application traffics to adapt to the access bandwidth of heterogeneous networks [3].

Previously, the mobility management in traditional IP networks has been investigated intensively. As the earliest and best known protocol to support mobility in IP networks, Mobile IP has already been standardized [4] but it cannot avoid the triangle routing problem when a mobile node (MN) locates far away from its home agent (HoA). Proxy Mobile IPv6 [5] works similarly as Mobile IPv6, introduces a local anchor to work as HoA, and forwards packets to the MN through a tunnel originating from the local anchor. Note that, both Mobile IP and Proxy Mobile IPv6 implement a mobility anchor in the network, which would become the bottleneck of packet transmission when the traffic loads from/to MNs are relatively high. Moreover, once the anchor fails, the whole functionality of mobility management would break down. To address these issues, distributed mobility management (DMM) has been proposed recently, which relies on a flat and flexible network architecture [6]. The major breakthrough of DMM is that instead of using a concrete mobile anchor, it realizes mobility management related functions with distributed network elements. Nevertheless, DMM still utilizes tunnels to forward the packets to MNs, which not only introduce additional overheads but also make packet routing suboptimal.

The major difficulty to achieve powerful mobility management in IP networks is due to its distributed network control and management (NC&M) mechanism. Specifically, when an MN moves, it is very difficult to let all the related network elements know about its new network location quickly and update their operation status accordingly. Fortunately, this issue can be resolved by leveraging the centralized NC&M in software-defined networking (SDN). Since the centralized NC&M facilitates customized routing and switching, the functionality and performance of mobility management can be enhanced significantly with SDN. In [7], the authors proposed an SDN-based mobility management framework and designed an algorithm to place a binding cache in the network. However, updating binding cache would bring extra workload to the SDN controller, which can make the proposed scheme impractical in a relatively busy network. An SDN-based approach has been developed in [8] for adaptive network orchestration.

Nevertheless, the aforementioned SDN-based proposals only considered how to support mobility, but did not fully explore the problem of how to maintain the ongoing sessions of mobile users without any interruption and adjust the application traffics adaptively when they switch APs frequently. For instance, these schemes usually forward packets based on the network address prefix, but when an MN joins a new network, the SDN controller has to assign a new network address prefix, which causes additional handover latency. Also, the MN's gateway would need to perform complicated network address translation to maintain its session. These drawbacks are mainly caused by the limitations of OpenFlow, which is the most popular protocol to implement SDN. More specifically, OpenFlow is protocol-dependent and can only leverage existing network protocols (*e.g.*, IP) to realize mobility management, *i.e.*, the programmability of the forwarding plane is restricted.

The issue with the forwarding plane's programmability can be resolved by introducing the protocol-oblivious forwarding (POF) [9], which is protocol-independent and can create a future-proof SDN environment. POF makes the SDN switch work as a white-box and leverages a generic flow instruction set to realize protocol-independent flow matching [10, 11]. In this work, we leverage POF to design a novel network system that can realize highly-efficient mobility management. We first utilize POF-based source routing (POF-SR) [11] to reduce the path switching latency and simplify the handover procedure. Then, based on the concept of network function virtualization (NFV), we explore the forwarding plane programmability provided by POF to realize dynamic virtual network function (vNF) deployment [12]. Since the vNF can adjust the coding scheme of the video traffic for e-Health applications adaptively according to the access bandwidth of heterogeneous networks, satisfactory QoE can be provided to the mobile users. We implement our system in a practical network testbed, and conduct experiments to demonstrate it. The experimental results indicate that our system for NFV-assisted mobility management functions well and achieves reduced handover latency and enhanced QoE.

The rest of the paper is organized as follows. We design the operation principle and architecture of our network system in Section II. Section III discusses our design of the POF-SR based mobility management scheme. The experimental demonstrations are presented in Section IV. Finally, Section V summarizes the paper.

II. POF-based Mobile Network

In this section, we first introduce POF briefly, and then explain the architecture and working principle of our POF-based source routing (POF-SR) based mobile management.

A. Overview of POF

Acting as a possible replacement of OpenFlow, POF inherits the general network architecture of SDN, *i.e.*, a centralized controller resides in the control plane and manages the switches in the forwarding plane, each of which realizes flow processing by applying the "match-and-act" principle. However, the forwarding procedure of POF was designed to guarantee protocol-independent flow processing. Specifically, POF switches utilize a sequence of generic key assembly and table lookup instructions (POF-FIS) [9, 13] for packet parsing and flow matching. The searching key of each matching field is defined as a tuple <*offset, length*>, where *offset* tells the field's start-location in a packet, *length* stores the length of the field and both of them are in bits. Fig. 1 shows an intuitive example on the packet forwarding procedure of POF. It can be seen that the tuple <*offset* = 208 *bits*, *length* = 32 *bits*> actually refers to the "IPv4 Source Address" field in an IPv4 packet encapsulated with Ethernet header. Hence, a POF switch does not need to know the packet format in advance, which significantly enhances the flexibility and programmability of its packet forwarding procedure. In order to facilitate the programmability, POF-FIS provides POF switches the generic flow instruction set to *parse*, *edit* and *forward* packets in an arbitrary manner [14]. On the contrary, OpenFlow defines each matching field based on an existing network protocol (*e.g.*, TCP port number), and its instructions and actions are also specified in such a way, *i.e.*, operations such as *push*, *pop* and *set* are all associated to a specific field.

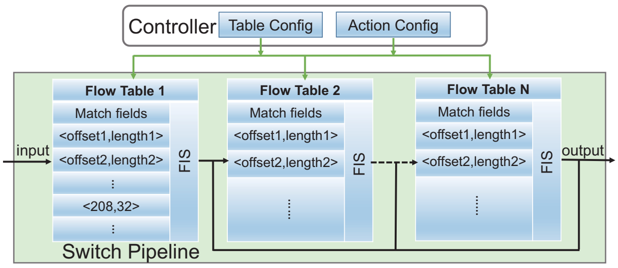
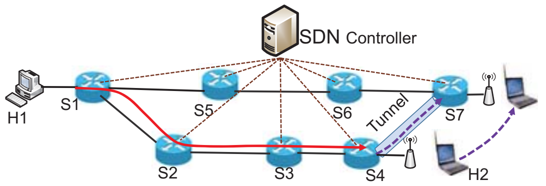


Fig. 1. Packet forwarding procedure in POF.

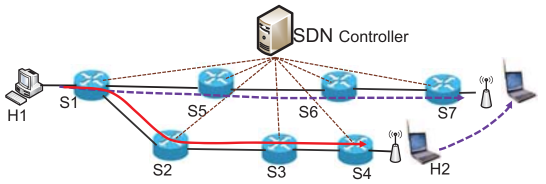
B. POF-based Source Routing for Mobile Networks

It is known that the handover latency in a mobile network directly affects the QoS and QoE of the mobile users, and thus should be minimized. Previous studies on SDN-based mobile networks have designed two types of handover schemes, as shown in Figs. 2(a) and 2(b). In the tunneling scheme in Fig.2(a), after the MN *H2* has moved to the new network location, the controller updates the flow entries in its old and new access switches, *i.e.*, *S4* and *S7*, respectively, and sets up a tunnel between the switches to redirect the packets to *H2*. On the other hand, the re-optimization scheme in Fig. 2(b) takes the clean-slate approach, which means that after the MN has moved, the controller recalculates an optimal path to its new network location and instructs all the related switches to update their forwarding state accordingly. However, both of these two schemes have drawbacks. The tunneling scheme suffers from the tunnel overheads and unnecessary bandwidth usage due to suboptimal routing, while the re-optimization scheme makes the controller update the forwarding state of all the related switches, which might increase the handover latency and make the NC&M more complicated, especially when the MNs move frequently. To address the issues of the two existing SDN-based handover schemes, we leverage POF-SR [9] to design a novel handover scheme, as shown in Fig. 2(c). When the packets from *H1* enter the POF network from the access switch *S1*, they are converted into the POF-SR format in Fig. 2(c). Specifically, the controller calculates the routing path from *H1* to *H2* and encapsulates the path information in a source routing header (SRH) to insert in all the packets between their Ethernet and IP headers. The SRH consists of a *TTL* field and several *Port* fields. The *TTL* field stores the number of switches along the remaining path, and its value will be decreased by 1 in each subsequent switch. The *Port* field identifies the output port of a switch, and after each switch, the corresponding *Port* field is removed from the SRH, as shown in Fig. 2(c).

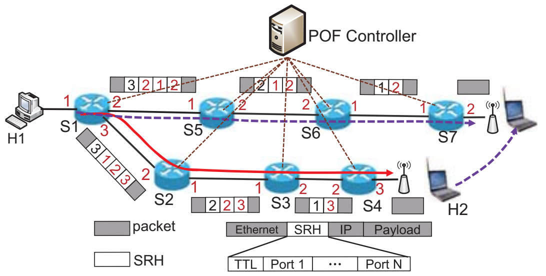
With POF-SR, the ingress switch inserts the SRH that contains the path information to packets, while each subsequent switch only need to use the same procedure to forward the packets to the designated output port stored in *Port* fields. Therefore, for each new flow, new flow entries only needs



(a) Handover with tunneling.



(b) Handover with re-optimization.



(c) Handover with POF-SR.

Fig. 2. Handover schemes in SDN-based mobile networks.

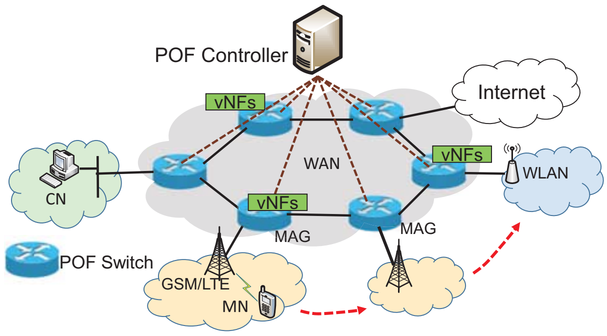


Fig. 3. Architecture of a POF-based mobile network.

to be installed on their ingress switches, while the rest of the related switches just use the pre-installed common flow entries to forward the packets and do not need to interact with the controller during the flow setup. Consequently, the flow setup latency can be reduced significantly, which is essentially important for realizing efficient handover. For example, when *H2* moves to connect to *S7*, the controller only needs to update the flow entries in *S1* to update the SRH of the packets, which are then forwarded on the new path *S1-S5-S6-S7-H2*. Hence, a lot of controller-switch communications are saved in the handover and the handover latency is reduced significantly.

C. Architecture of POF-based Mobile Networks

Before describing the architecture of our proposed POF-based mobile network system, we would like to first clarify the major design objectives of it.

- *Transparent Network-based Mobility Management*: To make the system more user-friendly, we should ensure that the mobility management is accomplished by the network-side without forcing any change on the MNs.
- *Superior Session Continuity during Handover*: The ongoing session of an MN should be maintained without any interruption when it moves and invokes handovers, and a universal ID (UID) should be assigned to each MN to assist the timely handover.
- *NFV-assisted Traffic Adaptation*: To address the situations in which the MN's access bandwidth changes significantly due to its movement, the system should be capable of adjusting the application traffics to adapt to the access bandwidth of heterogeneous networks and guaranteeing the QoE of mobile users.

Fig. 3 illustrates the architecture of our proposed network system. The forwarding plane consists of four types of network elements, *i.e.*, the POF switches, MNs, correspondent nodes (CNs) (*i.e.*, fixed clients and servers), and heterogeneous APs. The POF switches forward packets with POF-SR, and we call the access switches that directly connect to the APs as mobile access gateways (MAGs), which are responsible for detecting the attachment and movement of MNs and reporting them to the controller timely. Specifically, the MAG acts as the anchors for the MNs and lets them access the POF network under the supervision of the controller. Hence, all the mobility management related operations are accomplished by the network-side. Meanwhile, since the network environment (*e.g.*, latency and bandwidth) can change with the MN's network location, we introduce NFV-assisted traffic adaptation to ensure the QoE of clients' services. For instance, the QoE of high-definition (HD) video streaming would be affected significantly when the MN joins a network whose access bandwidth is very limited. This issue can be resolved by leveraging the forwarding plane programmability provided by POF to realize vNFs dynamically for traffic adaptation, *i.e.*, increasing or decreasing the traffic data-rates to MNs on-demand. The controller in the control plane is also programmed to support all the mobility related functionalities.

III. System Design

A. POF Controller for Mobility Management

In our network system, the controller uses the following four functional models, which operate according to the relation in Fig. 4, to realize POF-SR based mobility management.

- *Mobility management entity (MME)*: it provides the core functionalities of mobility management. When an MN joins a network, MME processes the attachment message reported by the corresponding MAG, and updates the information in the user equipment (UE) locator mapping (ULM) module. MME is also in charge of maintaining the ongoing session of an MN and readjusting its packet forwarding path during a handover.
- *User equipment locator mapping (ULM)*: It stores the mapping between the universal ID (UID) of a UE (*i.e.*,

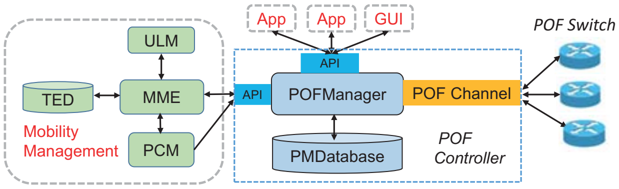


Fig. 4. Mobility management related modules in POF controller.

either a CN or an MN) and its current network location. Therefore, in our network system, a host only needs to know its peer's UID to set up the communication between them, and since its UID would not change when an MN moves, the MN's movement is made transparent to its communication peer.

- *Traffic engineering database (TED)*: It stores the information of in-service flows, including their bandwidth, expiration time, *etc*. Besides, TED also maintains the information about the network status, *e.g.*, active switches and bandwidth usages on links.
- *Path calculation module (PCM)*: It calculates the routing path for a flow based on the network status provided by TED, to fulfill the requests from other modules.

B. Operation Procedure

Fig. 5 illustrates the operation procedure of our proposed system, which mainly consists of three phases, *i.e.*, initial attachment, flow setup and handover. With the POF-SR based packet forwarding scheme, the system denotes the network location of each UE with a tuple <*switch, port*>, while the UID of an MN is assumed to be its MAC address[1]. Then, when an MN first attaches to an MAG (**Step 1** in Fig. 5(a)), it triggers a logical link control (LLC) message in the MAG to inform the attachment. Since there is no flow entry configured in the MAG for the LLC message, it is forwarded to the POF controller through a *PacketIn* message (**Step 2**). Upon receiving the *PacketIn* message, the controller parses it and raises an attachment event. MME in the controller captures the event and records the information of the MN, *i.e.*, its UID and network location as <*switch, port*>. If the UID can be found in ULM, MME updates the mapping between the UID and its network location, otherwise, it inserts a new entry into ULM to record the UID and its network location.

Next, when a CN needs to communicate with the MN, its packets will first reach its access switch (*i.e.*, *S1* in Fig. 5(a)), as shown in **Step 3**. The access switch sends a *PacketIn* message to the controller since there is no flow entry configured for the flow (**Step 4**). PCM in the controller parses the message and calculates an optimal forwarding path for the flow according to its service type and QoS requirements. Moreover, when PCM finds that traffic adaptation would be needed for the flow, it can direct the flow to go through a vNF (*e.g.*, for video transcoding). Next, the controller sends a *FlowMod* message to the access switch and instructs it to encapsulate the path information in the packets with SRHs (**Step 5**). At this moment, the packets can be sent from the CN to the MN, since with POF-SR, the subsequent switches along the path can forward the packets using the pre-installed common flow entries based on their SRHs (**Steps 6** and **7**). After the flow has been set up, its information, such as expiration time, bandwidth and latency requirements, will be stored in TED.

When the flow between the CN and the MN is active, the MN can move and change its AP. As shown in **Steps 8** and **9**, the re-attachment procedure is almost the same as the initial attachment. Basically, MME will check whether the bandwidth requirement of the MN's service can be satisfied in its new network. If yes, the re-attachment procedure is just the same as the initial attachment. Otherwise, MME asks the PCM to calculate a path that goes through a pre-deployed vNF for traffic adaptation. When the new path has been got, the controller sends a *FlowMod* message to the access switch to update the flow entries for encoding the new SRH on its packets (**Step 10**). Finally, packets from the CN will be forwarded to the MN at the new location (**Steps 11** and **12**).

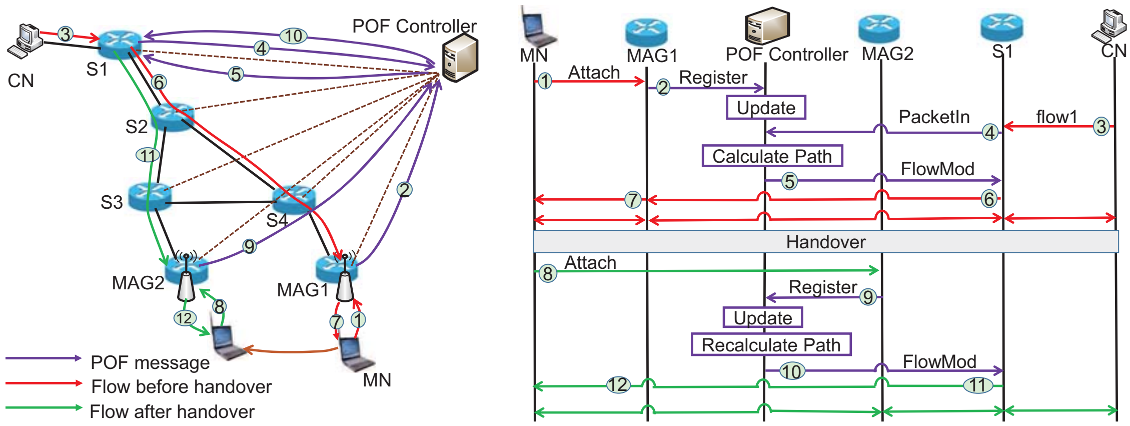
IV. Experimental Demonstration

In this section, we discuss the experimental demonstrations of our proposal and analyze the results. We implement the proposed scheme in a real network testbed whose topology is shown in Fig. 5(a). The testbed consists of 6 POF switches, a POF controller, an MN and a CN. The controller is developed based on the POX platform and runs on a Linux server. Specifically, we extend POX to support POF and implement all the mobility-related functional models in Fig. 4 in it. There are two types of POF switches in the testbed. *S1*-*S4* are wired ones, which are programmed based on a software-based POF switch and run on independent high-performance Linux servers, while *MAG1* and *MAG2* are wireless ones, each of which is realized by reprogramming and updating the image of a common WiFi router that supports IEEE 802.11n. MN and CN are a laptop and a desktop computer, respectively.

A. Evaluations on Handover Latency

Note that, the most important feature of mobility management is to realize timely and smooth handover when an MN has moved and changed its AP. Therefore, the handover latency, *i.e.*, the time duration from when an MN leaves the old AP to when its network services have been resumed by the new AP, is an important metric to evaluate the performance of a mobility management system. To this end, our experimental demonstrations will first compare our proposed system with two existing benchmarks (*i.e.*, the OpenFlow-based tunneling and re-optimization in Figs. 2(a) and 2(b), respectively) in terms of the handover latency. Specifically, for the re-optimization and tunneling schemes, an MN executes the Dynamic Host Configuration Protocol (DHCP) to get a network address from the controller when it attaches to an

[1]The MAC address is used here just for proof-of-concept demonstration. Actually, with the forwarding plane programmability of POF, UID can take other formats that are independent of Ethernet too.



(a) Testbed setup.

(b) Operation procedure.

Fig. 5. Experimental demonstration of POF-SR based mobile management.

AP. Then, the controller knows the mapping between the MN's current location and network address.

But before that, we should analyze the delays that can contribute to the handover latency. Specifically, the handover latency in an SDN-based mobility management system usually consists of three parts. Firstly, it is the delay due to Layer-2 switch (D-L2S), which is the time from when an MN is removed by its old AP to when it has established a Layer-2 connection to its new AP. Since all the three systems use WiFi as in Layer-2, the values of D-L2S from them would have no difference. Secondly, it is the delay due to Layer-3 handover (D-L3H), which is the time from when the MN's Layer-2 connection is reestablished by its new AP to when the Layer-3 reconfiguration for it has been done by the controller. More specifically, the Layer-3 reconfiguration generally involves assigning a new network prefix to the MN. Note that, in our proposed system, the packets to the MN are forwarded according to their SRHs, and thus there is no need to assign a new network prefix to the MN. Consequently, our POF-SR based system can avoid D-L3H completely. Finally, it is the delay due to Layer-3 routing path reconfiguration (D-L3PR), which is the time from when the MN's Layer-3 reconfiguration is done to when it receives the first packet at its new network location. During D-L3PR, the controller updates the mapping between the MN's UID and its network location, recalculates the routing path, and instructs the related switches in the forwarding plane to update the routing path for the packets to the MN. Since our POF-SR does not need to update the flow entries on all the related switches as OpenFlow-based tunneling and re-optimization do, it reduces D-L3PR too. Note that, D-L3PR can also be affected by the transmission delay of network links, especially that of the wireless link between the new AP and the MN. Table I summarizes the analysis above.

The experiment's scenario is as follows. After the MN first attaching to *MAG1*, we run a program on the CN to send packets to the MN continuously. The controller determines the packets' forwarding path as *S1-S2-S4-MAG1*. Then, the MN moves to attach to *MAG2*. After the move, the re-optimization

TABLE I
COMPARISON OF SDN-BASED MOBILITY MANAGEMENT SYSTEMS

	POF-SR	Re-optimization	Tunneling
New routing paths	Optimal	Optimal	Suboptimal
D-L2S	Has	Has	Has
D-L3H	0	Has	Has
D-L3PR	Short	Long	Long

and POF-SR schemes recalculate the optimal path (*i.e.*, *S1-S2-S3-MAG2*) to forward the packets and keep the ongoing session alive, while the tunneling scheme uses a tunnel (*i.e.*, *MAG1-S4-S3-MAG2*) to redirect the packets from *MAG1* to *MAG2*. For each scheme, we perform 100 handovers for the MN and collect the experimental results as shown in Table II. It can be seen clearly that our POF-SR scheme can avoid D-L3H completely and reduce D-L3PR significantly, when compared with the OpenFlow-based benchmarks. This actually verifies our analysis above. Specifically, for the shortest D-L3PR, our POF-SR scheme achieves it by only updating the flow entries in *S1* when the handover happens. On the contrary, for the re-optimization scheme, the controller has to update the flow entries in all the related switches on both the old and new routing paths, while the controller of the tunneling scheme need to establish a new tunnel between *MAG1* and *MAG2*, which makes its D-L3PR the longest among the three schemes. Meanwhile, we also notice that the results on D-L2S for the three schemes are very similar and D-L2S contributes the most to the handover latency.

TABLE II
EXPERIMENTAL RESULTS ON AVERAGE LATENCIES (MSEC)

	D-L2S	D-L3H	D-L3PR	Handover Latency
POF-SR	192.7	0	27.6	220.3
Re-optimization	192.5	54.2	32.6	279.3
Tunneling	192.2	54.4	35.8	282.4

B. *Demonstrations of NFV-assisted Traffic Adaption*

We then conduct experiments to demonstrate NFV-assisted traffic adaption. This time, the MN receives live HD video

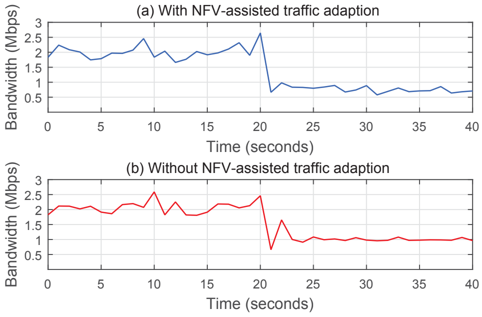


Fig. 6. Received bandwidth of video streaming on MN.

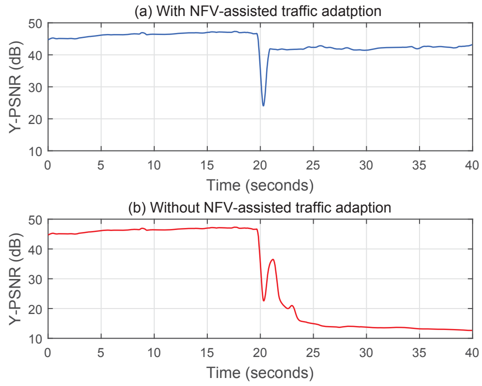


Fig. 7. Y-PSNR of video playback on MN.

streaming from the CN when it attaches to *MAG1*. Since the wireless network condition of *MAG1* is reasonably good (*e.g.*, only few other MNs attaches to *MAG1*), the MN can get sufficient access bandwidth to ensure the QoE of its video streaming service. Then, the MN moves to attach to *MAG2* whose network condition is not that good. Hence, in order to maintain the QoE of the MN's video streaming service, the controller redirects the video traffic to go through a pre-deployed vNF on *S3* for traffic adaption, *i.e.*, decreasing the video's coding rate to adapt to the limited access bandwidth. The initial data-rate of the video traffic is around 2 Mbps, the MN's access bandwidth at *MAG2* is 1 Mbps, and the vNF on *S3* helps to decrease the data-rate to 0.8 Mbps.

The received bandwidth of the MN is shown in Fig. 6, and we compare the schemes with and without the NFV-assisted traffic adaption. It can be seen that the MN first have an access bandwidth of 2 Mbps from *MAG1* and it moves to attach to *MAG2* at $t = 20$ seconds. For the scheme without traffic adaption, the access bandwidth becomes 1 Mbps, while for that with traffic adaption, the bandwidth decreases to 0.8 Mbps due to transcoding. Fig. 7 illustrates the results on the luminance component's peak signal-to-noise ratio (Y-PSNR) of the video playback on the MN. We observe that when there is no NFV-assisted traffic adaption, the Y-PSNR decreases sharply after the handover due to the severe packet loss caused by the limited access bandwidth. On the other hand, when the traffic adaption is in place, the Y-PSNR only declines around 5 dB because the transcoding decreases the traffic data-rate and avoid packet loss. Hence, even though the received bandwidth of the scheme with traffic adaption is lower, the MN's QoE of the video streaming is actually significantly higher. This confirms that our proposed POF-SR based mobility management works effectively to realize timely handover and provide satisfactory QoE to MNs.

V. Conclusion

We leveraged POF to design a novel network system that can realize highly-efficient mobility management, which uses POF-SR to reduce the path switching latency and simplify the handover procedure. The proposed system also realized dynamic vNF deployment for traffic adaption. We implemented our system and conducted experiments to verify that it functions well for NFV-assisted mobility management and achieves reduced handover latency and enhanced QoE.

Acknowledgments

This work was supported in part by the NSFC Project 61371117, NSR Project for Universities in Anhui (KJ2014ZD38), the SPR Program of the CAS (XDA06011202), and the NGBWMCN Key Project under Grant No. 2017ZX03001019-004.

References

[1] A. Osseiran *et al.*, "Scenarios for 5G mobile and wireless communications: the vision of the METIS project," *IEEE Commun. Mag.*, vol. 52, pp. 26–35, May 2014.
[2] P. Lu *et al.*, "Highly-efficient data migration and backup for big data applications in elastic optical inter-datacenter networks," *IEEE Netw.*, vol. 29, pp. 36–42, Sept./Oct. 2015.
[3] Z. Zhu, S. Li, and X. Chen, "Design QoS-aware multi-path provisioning strategies for efficient cloud-assisted SVC video streaming to heterogeneous clients," *IEEE Trans. Multimedia*, vol. 15, pp. 758–768, Jun. 2013.
[4] C. Perkins, "IP mobility support for IPv4," *RFC 5944*, 2010. [Online]. Available: https://tools.ietf.org/html/rfc5944
[5] S. Gundavelli, "Proxy mobile IPv6," *RFC 5213*, 2008. [Online]. Available: https://tools.ietf.org/html/rfc5213
[6] F. Giust *et al.*, "Distributed mobility management for future 5G networks: overview and analysis of existing approaches," *IEEE Commun. Mag.*, vol. 53, pp. 142–149, Jan. 2015.
[7] Y. Wang *et al.*, "A solution for IP mobility support in software defined networks," in *Proc. of ICCCN 2014*, pp. 1–8, Aug. 2014.
[8] N. Xue *et al.*, "Demonstration of OpenFlow-controlled network orchestration for adaptive SVC video manycast," *IEEE Trans. Multimedia*, vol. 17, pp. 1617–1629, Sept. 2015.
[9] S. Li *et al.*, "Protocol oblivious forwarding (POF): Software-defined networking with enhanced programmability," *IEEE Netw.*, vol. 31, pp. 12–20, Mar./Apr. 2017.
[10] D. Hu *et al.*, "Design and demonstration of SDN-based flexible flow converging with protocol-oblivious forwarding (POF)," in *Proc. of GLOBECOM 2015*, pp. 1–6, Dec. 2015.
[11] S. Li *et al.*, "Source routing with protocol-oblivious forwarding (pof) to enable efficient e-health data transfers," in *Proc. of ICC 2016*, pp. 1–6, May 2016.
[12] M. Zeng, W. Fang, and Z. Zhu, "Orchestrating tree-type VNF forwarding graphs in inter-DC elastic optical networks," *J. Lightw. Technol.*, vol. 34, pp. 3330–3341, Jul. 2016.
[13] S. Li *et al.*, "SR-PVX: A source routing based network virtualization hypervisor to enable POF-FIS programmability in vSDNs," *IEEE Access*, vol. 5, pp. 7659–7666, 2017.
[14] D. Hu *et al.*, "Flexible flow converging: A systematic case study on forwarding plane programmability of protocol-oblivious forwarding (POF)," *IEEE Access*, vol. 4, pp. 4707–4719, 2016.