A Fast Handover Scheme for SDN Based Vehicular Network

Xing Yin[1,2] and Liangmin Wang[2](✉)

[1] School of Electrical and Information Engineering, Jiangsu University, Zhenjiang, China
staryin@126.com
[2] School of Computer Science and Communication Engineering,
Jiangsu University, Zhenjiang, China
Wanglm@ujs.edu.cn

Abstract. Vehicular network can provide Internet connectivity for mobile vehicle by handover mechanism. However, existing handover schemes still face poor handover performance when they are applied in vehicular network. Software Defined Network (SDN) is a new architecture which can be used to optimize vehicular network by making network devices to be programmable. In this paper, we propose a new fast handover scheme for SDN based vehicular network to improve handover performance. SDN controllers of our scheme predict movement of vehicles by detecting port status of SDN switches, and then they start to perform the proactive handover procedure based on prediction results. Evaluation results show that the handover delay and packet loss of our scheme are lower than the contrast schemes. Simulation results prove that our handover scheme is more fit for delay sensitive vehicular network.

Keywords: Vehicular network · Software Defined Network (SDN) · Handover Predict · Delay

1 Introduction

As an important application area of Internet of Things and mobile Internet technologies, vehicular network is becoming increasingly attractive to researchers in academia and industry during these years. One of the main aims of vehicular network is to allow any vehicle access to the Internet whenever it is traveling on a road, that is, to provide Vehicle-to-Infrastructure (V2I) connectivity [1, 2].

Although Mobile IPv6 (MIPv6) [3] and its derivatives [4] can realize V2I connection, these protocols still have some shortcomings such as high handover delay and packet loss rate [5]. Handover performance is critical to vehicular network since there are many delay sensitive applications which are running on vehicles. Fast mobile IPv6 (FMIPv6) [6] protocol and some extended schemes [7–10] were proposed by using proactive handover mechanism to optimize handover performance of vehicular network to some extend. However, handover delay of these schemes are still high for vehicular communications. Furthermore, the control function and data forwarding are tightly coupled in

© Springer Nature Singapore Pte Ltd. 2018
L. Zhu and S. Zhong (Eds.): MSN 2017, CCIS 747, pp. 293–302, 2018.
https://doi.org/10.1007/978-981-10-8890-2_21

network devices in traditional vehicular network [11], which makes it difficult for these devices to be programmable to improve handover performance.

Software Defined Network (SDN) is a promising network paradigm which can overcome the inflexibility of traditional network by separating the control plane from the data plane [12]. All network devices in data plane are regarded as SDN switches and they are only responsible for data forwarding according to their flow tables. All control functions of these devices are abstracted to a logically centralized SDN controller, which has a global view of the whole network and manages SDN switches by downloading flow entries to their flow tables through the famous south bound interface protocol, i.e. OpenFlow [13]. These characteristics of SDN can make network devices to be more programmable and controllable.

Due to the significant improvements in programmability and flexibility provided by SDN, many researchers try to propose handover schemes based on SDN network to optimize handover performance [14]. Wang et al. [15] proposed a mobility scheme, which we name as SDMA, to provide basic handover function for a mobile nodes in SDN based network. Since SDMA dose not provide a handover optimize method, its handover performance is still not be improved. Yang et al. [16] proposed a handover scheme to provide seamless handover in SDN based satellite network. Their scheme has improved the handover performance for satellite network, but it was not fit for vehicular network.

From the above analysis, we can conclude that existing handover schemes have the following drawbacks:

- Some of them do not make full use of the specially features of vehicular network, such as predictable movement of vehicles, to optimize handover procedure.
- Some of them are based on traditional network, which leads to the inflexibility of their scheme. This drawback will hinder further improvement of handover performance.
- Some of them have complex handover procedures, which leads to high handover delay.

To overcome the above shortcomings of the existing related works, this paper presents a novel fast handover scheme to improve the handover performance of SDN based vehicular network. The main contributions of our proposal is listed as follows:

1. By taking advantage of the predictable characteristic of vehicular network, we propose a proactive handover procedure to optimize handover performance. SDN controllers in our scheme can predict movement direction of vehicles and then perform some handover steps beforehand.
2. By simplifying the handover procedure of our scheme, the handover performance is further improved.

The rest of this paper is organized as follows. In Sect. 2, we give a typical SDN based vehicular network topology. Then we design a new handover procedure based on this topology. In Sect. 3, handover performance such as handover delay and packet loss of our scheme are evaluated and compared with some typical comparatives. In Sect. 4, we conclude our work.

2 Proposed Scheme

In this section, we construct a SDN based vehicular network topology. Then based on the topology, we propose a handover procedure by using prediction mechanism to optimize the handover performance.

2.1 Network Topology

A typical SDN based network topology for V2I communication in vehicular network is shown in Fig. 1. We assume that access points (AP) are distributed evenly along one side of the road. Wireless signal of these APs can cover the entire road to ensure that every vehicle can connect to one or more APs. Each AP can connect to the Internet through SDN switches. These network devices and their controller compose a district, which we call it as a domain. Figure 1 shows that AP11, AP12, AP13 and S11, S12 of SDN switches, are all controlled by SDN controller C1. The left domain consists of these devices. The right domain consists of AP21, AP22, AP23, and S11, S12 of SDN switches, which are all controlled by C2.

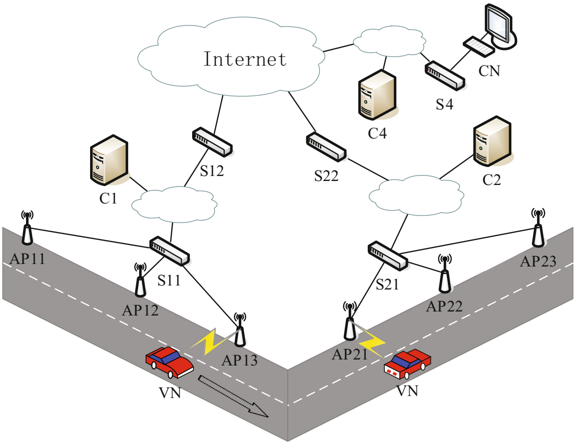


Fig. 1. Typical topology of SDN based vehicular network

In Fig. 1, each vehicle is defined as a vehicular node (VN). Just like a mobile node defined in MIPv6, VN can change its point of attachment to different access points through wireless link when it drives along a road. VN can keep its session with a correspondent node (CN) by applying some mobility management protocols (e.g. MIPv6). However, VN in vehicular network has some special mobility characteristics which are different from that of mobile nodes in broadly defined mobile network. This is because the trajectory of VN is always along a certain road and the direction of movement can be predicted in most cases.

In order for the controllers to have the functionality of predicting the handover of VN, each SDN controller maintains a port status table which is shown as Table 1. This table lists the information of the ports of switches directly connected to access points. Every SDN switch is responsible for reporting the attachment and detachment events to their controller to update the port status table.

Table 1. Port status table

Port ID	Switch	AP ID	Location	List of VN
1001	S11	AP11	Edge	-
1002	S11	AP12	Intermediate	-
1003	S11	AP13	Edge	PIP
2001	S21	AP21	Edge	-
2002	S21	AP22	Intermediate	-
2003	S21	AP23	Edge	-

Items of Table 1 are explained as follows:

- Port ID: Identification number of different port of each SDN switch. Port ID is allocated by the controller of current domain, and the value is unduplicated.
- Switch: The identification of the SDN switch, which the current port belongs to.
- AP ID: The identification of the access point which is connecting to the current port.
- Location: The location of the AP current port connecting to. This attribute has two types of values. The value Edge indicates that the AP is located in the end of current domain (i.e. current road) and the value Intermediate indicates that the AP is not at the two end sides of the current road.
- List of VN: List of up-to-data IP addresses of VNs which are currently connect to this port.

In order to facilitate our analysis, we use the original IP address IP_VN, which is obtained when VN starts up in its home network, as its identity identifier. Furthermore, we use the routable IP address assigned by the controller of the current domain as the location identifier of VN. We assume that the location identifiers of VN when it is in the left domain and the right domain in Fig. 1 are defined as PIP and NIP respectively. The IP address of CN is defined as CN_IP. The following designation and analysis are based on the network topology illustrated in Fig. 1.

2.2 Handover Procedure

Based on the above network topology, we design a handover procedure when a vehicle travels along a road. Since AP11, AP12 and AP13 are connected to the same SDN switch, i.e. S11, the handover procedure that VN moves from AP12 to AP13 is easy to be handled by controller C1. In this situation, C1 makes routing decision according to the status report message from AP13 and then downloads a flow entry to S11 to redirect data flow from one port to another. As for the handover when VN moves from AP13 to AP21, this procedure is a cross domain movement and it is much more complex than the former handover.

In our proposal, we use a predict mechanism to design and optimize handover procedure for vehicular network. The handover procedure of our scheme, which is shown in Fig. 2, is depicted as follows.

1. When VN moves along the road and it will move from the left domain to the right one, VN will detach from AP13 and attach to AP21. S11 detects the change of wireless signal strength. Then S11 sends an extended Port Status (Ext-PS) message containing PIP and the port ID to its controller C1 to report this movement event.
2. After receiving the Ext-PS message, C1 looks up its port status table with the port ID and PIP to predict where VN will move to and which kind of handover will take place. For example, the look up result shows that the previous and the current access point of VN are AP12 and AP 13 respectively and AP13 is an Edge access point of the current road. Therefore, C1 can predict that VN will move from AP13 to AP 21 in the left domain and perform a handover. Then C1 sends a Notify message including IP_VN to C2 to notify the incoming of VN proactively.
3. When receiving the Notify message, C2 generates a new routable IP address (i.e. NIP) for VN as its new location identifier in advance. C2 creates a binding which contains IP_VN and NIP, and then stores this binding in its binding cache. After that, C2 sends binding update (BU) messages to C1 and C4 at the same time. If VN has a home network, C2 also sends a BU message to the controller of VN's home network. Furthermore, C2 sends a Flow-Mod message to S22 to inform it to cache all data packets which take NIP as the destination address.
4. After receiving the BU message form C2, C1 downloads flow entry to S12. After that, S12 will cache all the packets which destination address is PIP and rewrite the destination address to NIP, and then redirects these packets to S22.
5. C4 updates its binding cache (BC) and sends a Flow-Mod message to S4 to add a flow entry about VN in the flow table of S4 when C4 receives the BU message from C2. After that, S4 rewrites the destination address of data packets which send from CN to VN. These packets are routed to the new IP address of VN.
6. Owing to the transparent character of our handover scheme, VN can perform link layer (i.e. Layer 2, L2) handover at any time when it detects the signal of AP21. Note that the step 3, 4, 5, or even 6 can be performed at the same time.
7. After L2 handover is finished and VN attaches to AP21, VN sends a route solicitation (RS) message to S21 to notify the attachment event.
8. When detecting the attachment of VN, S22 sends a Ext-PS message to it controller C2 to report VN's attachment.
9. C2 updates its port status table and send a Flow-Mode message to S21 and S22 respectively for updating their flow tables.
10. When S22 receiving the Flow-Mode message, it updates its flow table and forwards all the packets which destined to NIP to S12. At the same time, S12 updates it flow table according to the Flow-Mod message it has received. Finally, S21 rewrites the destination IP address of these packets to IP_VN and then delivers them to VN.

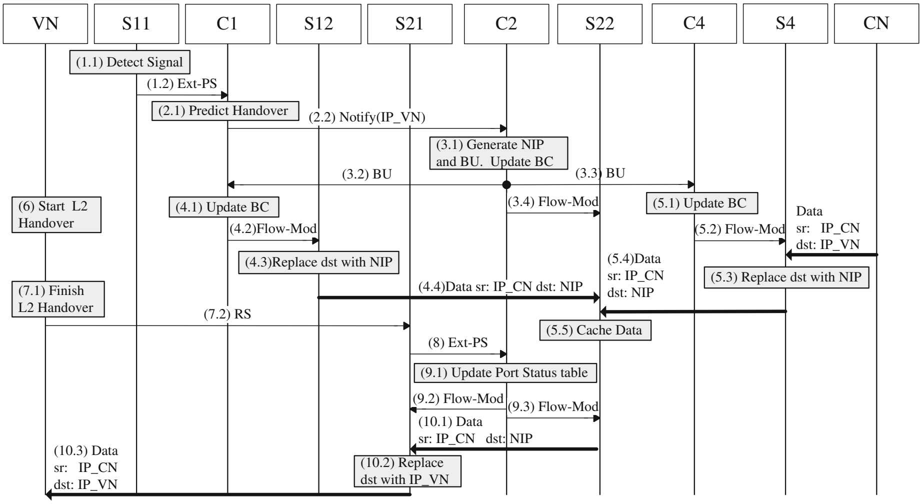


Fig. 2. Handover procedure of our scheme

After the above handover procedure is completed, the communications between CN and VN can be recovered. It should be explained that the above procedures are based on the assumption that CN is stationary. If CN is a mobile node, it can be treated as VN too, and the handover procedure of a mobile CN is the same to that of VN mentioned above.

3 Performance Evaluation

In this section, we evaluated and compared the performance of our proposed scheme, SDMA and FLBH. SDMA is a typical mobility management scheme based on SDN architecture, and FLBH is a representative of handover schemes based on traditional vehicular network.

Handover delay and packet loss number are important metrics to measure the performance of a handover scheme. Handover delay can directly reflect the length of communication interruption during handover procedure. This metric is critical factor for delay sensitive applications. Packet loss is critical for important data transmission. We analyze these two metrics of different handover schemes through the following simulation.

We carry out our simulation based on the famous SDN simulator, i.e. Mininet 2.2.1 [17]. We use Virtual Box, Pox and Open vSwitch (OvS) as virtual machine, SDN controller and virtual SDN switch respectively in our experiment. We have implemented handover procedures of our scheme and the above two schemes in controllers.

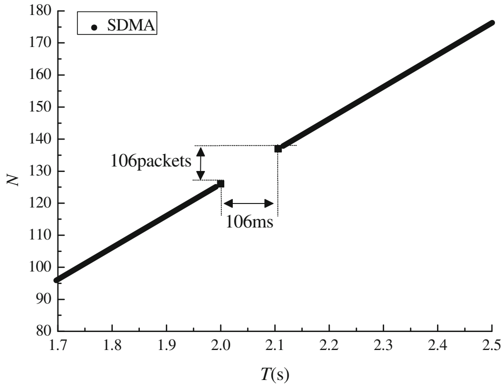
For convenient of experiment, we assume that CN sends data packets to VN continuously during the whole simulation time, and all packets will not be lost in wired and wireless links. Length of data packet is 56 bytes and packet sending rate is 100 packets per second. We ignore the processing delay of data and signaling message in our experiment. Parameters required for simulation are listed in Table 2. The values of these parameters are referred to literature [18, 19].

Table 2. Simulation parameters

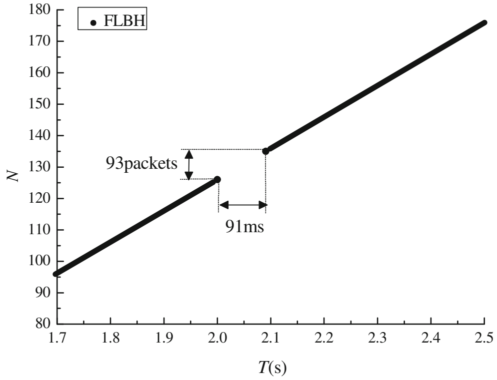
Perimeter	Value
Average delay of one hop wire link (ms)	2
Average delay of one hop wireless link (ms)	10
Average delay of Layer 2 handover (ms)	50
Average bandwidth of wire link (Mbps)	100
Average bandwidth of wireless link (Mbps)	10

The changes of packet sequence numbers with the simulation time are shown in Fig. 3. The abscissa represents the simulation time T, and the ordinate represents the sequence number N. Each point in Fig. 3 represents the sequence number of the data packet received by the VN at time T. The length of the abscissa section corresponding to the blank area in Fig. 3 indicates the handover delay of once handover from AP13 to AP21, and the length of the corresponding ordinate section indicates the total number of packets which have lost during the handover.

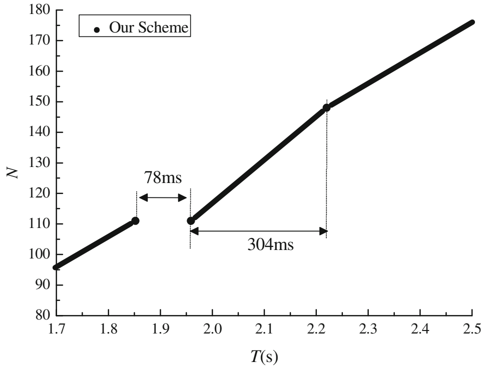
As we can see in Fig. 3(a), the handover delay of SDMA is largest, and all the data packets transmitted during the handover time are lost. This is because SDMA does not provide any proactive handover mechanism to optimize handover performance, even if SDMA can support a simple handover function in SDN based network. Figure 3(b) shows that the handover delay of FLBH is lower than SDMA, this is because FLBH proposes a fast location-based handover to drop the handover delay. However, data loss still takes place in FLBH, which is owing to the luck of data caching and forwarding mechanism. Figure 3(c) indicates that our scheme has the lowest handover delay, and data loss problem does not appear in our simulation. This is because we utilize the prediction method to design a optimized handover procedure based on SDN architecture.



(a) SDMA



(b) FLBH



(c) Our scheme

Fig. 3. Sequence number (*N*) versus simulation time (*T*)

4 Conclusion

In this paper, we proposed a new fast handover scheme based on SDN paradigm by using the characteristic of predictable movement in vehicular network. Simulation result demonstrates that our proposed handover scheme has lower handover delay than the contrastive schemes. Besides, packet loss problem does not occurred in our simulation also. The experiment illustrates that our handover scheme is more suitable for vehicular network environment.

Acknowledgements. This research is supported by the National Natural Science Foundation of China Grants (61472001, 61272074, U1736216, U1405255) and the Key research and development plan project of Jiangsu Province (BE2015136).

References

1. Jarupan, B., Ekici, E.: Prompt: a cross-layer position-based communication protocol for delay-aware vehicular access networks. Ad Hoc Netw. **8**, 489–505 (2010). https://doi.org/10.1016/j.adhoc.2009.12.006
2. Ma, S., Jiang, H., Han, M., Chen, L.: Survey of information security research for vehicle electronic control system in vehicle internet environment. Jiangsu Daxue Xuebao **35**, 635–643 (2014). https://doi.org/10.3969/j.issn.1671-7775.2014.06.003
3. Perkins, C., Johnson, D., Arkko, J.: Mobility Support in IPv6. RFC 6275, IETF (2011)
4. Bernardos, C.J.: Proxy Mobile IPv6 Extensions to Support Flow Mobility. RFC 7864, IETF (2016)
5. Gladisch, A., Daher, R., Tavangarian, D.: Survey on mobility and multihoming in future internet. Wirel. Pers. Commun. **74**, 45–81 (2014). https://doi.org/10.1007/s11277-012-0898-6
6. Koodli, R.: Mobile IPv6 fast handovers. RFC 5568, IETF (2009)
7. Schmidt, T., Waehlisch, M., Koodli, R., Fairhurst, G., Liu, D.: Multicast Listener Extensions for Mobile IPv6 (MIPv6) and Proxy Mobile IPv6 (PMIPv6) Fast Handovers. RFC 7411, IETF (2014)
8. Kim, M.S., Lee, S.K., Golmie, N.: Enhanced fast handover for proxy mobile IPv6 in vehicular networks. Wirel. Netw. **18**, 401–411 (2012). https://doi.org/10.1007/s11276-011-0407-y
9. Almulla, M., Wang, Y., Boukerche, A., Zhang, Z.: Design of a fast location-based handoff scheme for IEEE 802.11 vehicular networks. IEEE Trans. Veh. Technol. **63**, 3853–3866 (2014). https://doi.org/10.1109/TVT.2014.2309677
10. Yin, X., Wu, G., Dong, Y.: A proactive handover scheme for high-speed network mobility. J. Southeast Univ. (Nat. Sci. Ed.) **45**, 1038–1045 (2015). https://doi.org/10.3969/j.issn.1001-0505.2015.06.003
11. Xia, W., Wen, Y., Foh, C.H., Niyato, D.: A survey on software-defined networking. IEEE Commun. Surv. Tutorials **17**, 27–51 (2015). https://doi.org/10.1109/COMST.2014.2330903
12. Open Networking Fundation: Software-Defined Networking: The New Norm for Networks (2012). http://www.opennetworking.org/component/content/article/46-sdn-resources/sdn-library/whitepapers/816-software-defined-networking-the-new-norm-for-networks
13. Mckeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J.: OpenFlow: enabling innovation in campus networks. ACM Sigcomm Comput. Commun. Rev. **38**, 69–74 (2008). https://doi.org/10.1145/1355734.1355746

14. Liyanage, M., Gurtov, A., Ylianttila, M.: Software Defined Mobile Networks - SDMN: Beyond LTE Network Architecture, pp. 9–10. Wiley Publishing (2015)
15. Wang, Y., Bi, J., Zhang, K.: Design and implementation of a software-defined mobility architecture for IP networks. Mob. Netw. Appl. **20**, 40–52 (2015). https://doi.org/10.1007/s11036-015-0579-2
16. Yang, B., Wu, Y., Chu, X., Song, G.: Seamless handover in software-defined satellite networking. IEEE Commun. Lett. **20**, 1768–1771 (2016). https://doi.org/10.1109/LCOMM.2016.2585482
17. Mininet: An Instant Virtual Network on your Laptop (or other PC). http://mininet.org
18. Makaya, C., Pierre, S.: An analytical framework for performance evaluation of IPv6-based mobility management protocols. IEEE Trans. Wirel. Commun. **7**, 972–983 (2008). https://doi.org/10.1109/TWC.2008.060725
19. Lee, C.W., Chuang, M.C., Chen, M.C., Sun, Y.S.: Seamless handover for high-speed trains using femtocell-based multiple egress network interfaces. IEEE Trans. Wirel. Commun. **13**, 6619–6628 (2014). https://doi.org/10.1109/TWC.2014.2364179